DESIGN OF TRAINING SYSTEMS
COMPUTERIZATION OF THE EDUCATIONAL TECHNOLOGY ASSESSMENT MODEL (ETAM) – VOLUME 2

FOCUS ON THE TRAINED MAN

MAY 1977

TRAINER ANALYSIS AND EVALUATION GROUP

ORLANDO, FLORIDA 32813
DESIGN OF TRAINING SYSTEMS
COMPUTERIZATION OF THE EDUCATIONAL TECHNOLOGY ASSESSMENT MODEL (ETAM)
VOLUMES 1 AND 2

ABSTRACT

This two volume final report summarizes the analysis, design and development activities associated with the Educational Technology Assessment Model (ETAM). It contains relevant background information and results of prior studies leading to the finalized ETAM procedures and computerized routines. A comparison of the manual versus the automated approach is included. Data base structures and the ETAM program flow are described and related to each other. The appendix includes the results of a study on the indexing of innovations and the assignment of taxonomic descriptors to courses, job/tasks, and instructional vehicles. The appendix also includes program documentation on the ETAM Range-of-Effect, bibliographic references, and additional information supporting the ETAM design. The detailed ETAM procedures supporting this report are included in TAEG Report No. 12-3, Phase II-A Report. TAEG Report No. 32, The Development of Scaling Procedures, supports the computerized approach taken in scaling variables. Other important background and reference information can also be found in a report written by Drs. R. B. Miller and A. F. Smode titled "Major Innovations in Training Technology."

Phases I, II, II-A, III and IV were accomplished by the IBM Federal Systems Division with the Training Analysis and Evaluation Group, Orlando, Florida, providing technical guidance and support. The overall DOTS objective is to provide Naval Education and Training Command (NAVEDTRACOM) management with additional tools in the form of computerized mathematical models to assist in predicting the quantitative impact of training resource decisions. The planning process will be enhanced by providing decision makers with the capability to economically and rapidly consider a wider range of alternatives.

Phase I was a study and definition effort resulting in a complete functional description of the NAVEDTRACOM; a strategic definition of the social, political, economic and technological environments pertinent to the naval education and training system in the 1980's; a list of existing and potential models amenable to computerization and to improving the decision-making process.

Phase II was devoted to the selection and development of three mathematical models from the Phase I list of candidates. The three were the System Capabilities/Requirements and Resources (SCRR), the Individualized Training Simulation System (ITSS), and the Training Process Flow (TPF) models.

Phase III centered on evaluating the selected models at the Fleet Training Center, Norfolk, VA. An important recommendation from the Test and Evaluation conducted during Phase III was that DOTS should investigate model applications at higher command levels.

Phase IV responded to the recommendations of Phase III by (1) operating and testing at TRAPAC, San Diego, CA, the models developed in Phase II, and (2) developing a new Training Requirements Analysis Model (TRAM) and field testing it at CNTECHTRA, Memphis, Tenn.

The major effort supporting Phase II-B was the ETAM procedural development during Phase II-A. In addition to the procedural development, computer applications were described, scenarios using the ETAM procedures were prepared, and ETAM validity and logical structure were confirmed.
**REPORT DOCUMENTATION PAGE**

<table>
<thead>
<tr>
<th>1. REPORT NUMBER</th>
<th>2. GOVT ACCESSION NO.</th>
<th>3. RECIPIENT'S CATALOG NUMBER</th>
</tr>
</thead>
<tbody>
<tr>
<td>TAEG Report No. 40</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>4. TITLE (and Subtitle)</th>
<th>5. TYPE OF REPORT &amp; PERIOD COVERED</th>
</tr>
</thead>
<tbody>
<tr>
<td>DESIGN OF TRAINING SYSTEMS</td>
<td>Final 8/76 – 5/77</td>
</tr>
<tr>
<td>Computerization of the Educational Technology Assessment Model (ETAM) - Volume 2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>7. AUTHOR(s)</th>
<th>8. CONTRACT OR GRANT NUMBER(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Larry R. Duffy</td>
<td>N61339-73-C-0097</td>
</tr>
<tr>
<td>Robert B. Miller, Ph.D.</td>
<td></td>
</tr>
<tr>
<td>James D. Staley</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>9. PERFORMING ORGANIZATION NAME AND ADDRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>International Business Machines</td>
</tr>
<tr>
<td>7900 N. Astronaut Blvd.</td>
</tr>
<tr>
<td>Cape Canaveral, FL 32920</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>11. CONTROLLING OFFICE NAME AND ADDRESS</th>
<th>12. REPORT DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naval Training Equipment Center</td>
<td>May 1977</td>
</tr>
<tr>
<td>Training Analysis and Evaluation Group</td>
<td></td>
</tr>
<tr>
<td>Orlando, FL</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>15a. SECURITY CLASS. (of this report)</th>
<th>Distribution is Unlimited.</th>
</tr>
</thead>
</table>

**DISTRIBUTION STATEMENT (of this Report)**

Approved for Public Release: Distribution is Unlimited.

**DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)**

**SUPPLEMENTARY NOTES**

**KEY WORDS (Continue on reverse side if necessary and identify by block number)**

- Training Management Systems
- Computer Based Models
- Training Resources Evaluation Modeling
- Educational Technology
- Training Technology
- Interactive Training Data Base
- Decision Theory

**ABSTRACT (Continue on reverse side if necessary and identify by block number)**

This two volume final report summarizes the analysis, design and development activities associated with the Educational Technology Assessment Model (ETAM). It contains relevant background information and results of prior studies leading to the finalized ETAM procedures and computerized routines. A comparison of the manual versus the automated approach is included. Data base structures and the ETAM program flow are described and related to each other. The appendix includes the results of a study on the indexing of innovations and the assignment of taxonomic descriptors to courses, job/tasks, and instructional vehicles.
20. ABSTRACT (Cont'd)

The appendix also includes program documentation on the ETAM Range-of-Effect, bibliographic references, and additional information supporting the ETAM design. The detailed ETAM procedures supporting this report are included in TAEG Report No. 12-3, Phase II-A Report. TAEG Report No. 32, The Development of Scaling Procedures, supports the computerized approach taken in scaling variables. Other important background and reference information can also be found in a report written by Drs. R. B. Miller and A. F. Smode titled "Major Innovations in Training Technology."

Phases I, II, II-A, III and IV were accomplished by the IBM Federal Systems Division with the Training Analysis and Evaluation Group, Orlando, Florida, providing technical guidance and support. The overall DOTS objective is to provide Naval Education and Training Command (NAVEDTRACOM) management with additional tools in the form of computerized mathematical models to assist in predicting the quantitative impact of training resource decisions. The planning process will be enhanced by providing decision makers with the capability to economically and rapidly consider a wider range of alternatives.

Phase I was a study and definition effort resulting in a complete functional description of the NAVEDTRACOM; a strategic definition of the social, political, economic and technological environments pertinent to the naval education and training system in the 1980's; a list of existing and potential models amenable to computerization and to improving the decision-making process.

Phase II was devoted to the selection and development of three mathematical models from the Phase I list of candidates. The three were the System Capabilities/Requirements and Resources (SCRR), the Individualized Training Simulation System (ITSS), and the Training Process Flow (TPF) models.

Phase III centered on evaluating the selected models at the Fleet Training Center, Norfolk, VA. An important recommendation from the Test and Evaluation conducted during Phase III was that DOTS should investigate model applications at higher command levels.

Phase IV responded to the recommendations of Phase III by (1) operating and testing at TRAPAC, San Diego, CA, the models developed in Phase II, and (2) developing a new Training Requirements Analysis Model (TRAM) and field testing it at CNTECHTRA, Memphis, Tenn.

The major effort supporting Phase II-B was the ETAM procedural development during Phase II-A. In addition to the procedural development, computer applications were described, scenarios using the ETAM procedures were prepared, and ETAM validity and logical structure were confirmed.
TAEG REPORT NO. 40

DESIGN OF TRAINING SYSTEMS

COMPUTERIZATION OF THE EDUCATIONAL TECHNOLOGY ASSESSMENT MODEL (ETAM)

VOLUME 2

Robert B. Miller
Larry R. Duffy
James D. Staley

This Study Was Performed by

International Business Machines Corporation

for the

Training Analysis and Evaluation Group

May 1977

GOVERNMENT RIGHTS IN DATA STATEMENT

Reproduction of this publication in whole or in part is permitted for any purpose of the United States Government.

ALFRED F. SMEDE, Ph.D., Director
Training Analysis and Evaluation Group
The Design of Training Systems (DOTS) project objectives are in conso-
ance with the requirements of Advanced Development Objective ZPN07
(formerly ADO 43-03X), Education and Training Development. ZPN07 in-
cludes a number of projects concerned with demonstrating and evaluating
the technical, operational and financial feasibility of applying advanced
 technological applications to improving the training process.

The Bureau of Naval Personnel initiated the original ADO in 1966 to make
naval training more responsive to the changing times. As one project
under this effort, DOTS was designed to improve the process of managing
training resources through application of the techniques of system analysis
and system simulation as accomplished through mathematical modeling. The
end objective is a family of computerized mathematical models enabling
training management to more rapidly predict the impact of changes in train-
ing resource availability or requirements.

The majority of education and training was reorganized in 1971 under one
command, Chief of Naval Education and Training (CNET). Because of this
change, DOTS responsibility was transferred to CNET in March of 1972;
more specifically, to the Training Analysis and Evaluation Group (TAEG),
Orlando, Florida. The new CNET organization greatly increased the potential
benefits to be gained from the increased application of new management tech-
niques and, therefore, from the DOTS' R&D effort. DOTS began in February
of 1973, with the majority of tasking being contracted to the International
Business Machines Corporation, Federal Systems Division, Cape Kennedy
Facility, located at Cape Canaveral, Florida.

The Training Analysis and Evaluation Group, Dr. A. Smode, Director, project
team members Messrs. M. Middleto and W. Lindahl, complemented the con-
tracted effort by providing direction and guidance and in establishing
organizational interfaces.

The model was developed by Mr. L. R. Duffy, Dr. R. B. Miller, and Mr. J. D.
Staley. Mr. R. E. Hallman and Mr. L. R. Duffy provided management for the
project.
## TABLE OF CONTENTS

### APPENDIX

<table>
<thead>
<tr>
<th>A</th>
<th>STUDY OF DESCRIPTOR AND INDEXING TECHNIQUES FOR ETAM RANGE-OF-EFFECT ANALYSIS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>INTRODUCTION .......... A-1</td>
</tr>
<tr>
<td></td>
<td>PROCEDURES FOR INDEXING THE INNOVATION .......... A-2</td>
</tr>
<tr>
<td></td>
<td>PROCEDURES FOR INDEXING INSTRUCTIONAL VEHICLES, TRAINING COURSES AND JOB-TASKS .......... A-11</td>
</tr>
<tr>
<td></td>
<td>INDEXING OBJECTIVES .......... A-12</td>
</tr>
<tr>
<td></td>
<td>SEARCH RATIONALES AND STRATEGIES .......... A-17</td>
</tr>
<tr>
<td>B</td>
<td>KEY FIGURES AND ILLUSTRATIONS REFERENCED IN THIS ETAM REPORT</td>
</tr>
<tr>
<td>C</td>
<td>TRAINING COST MODEL PROGRAM</td>
</tr>
<tr>
<td>D</td>
<td>ETAM RANGE-OF-EFFECT PROGRAM DOCUMENTATION</td>
</tr>
<tr>
<td></td>
<td>SECTION D.1 INTRODUCTION .......... D.1-1</td>
</tr>
<tr>
<td></td>
<td>ORGANIZATION .......... D.1-1</td>
</tr>
<tr>
<td></td>
<td>EQUIPMENT ENVIRONMENT .......... D.1-1</td>
</tr>
<tr>
<td></td>
<td>PROGRAM ENVIRONMENT .......... D.1-2</td>
</tr>
<tr>
<td></td>
<td>ETAM SYSTEM INITIALIZATION .......... D.1-3</td>
</tr>
<tr>
<td></td>
<td>LOAD OF THE ABBREVIATED COURSE DATA BASE .......... D.1-3</td>
</tr>
<tr>
<td></td>
<td>ABBREVIATED VEHICLE DATA BASE LOAD .......... D.1-6</td>
</tr>
<tr>
<td></td>
<td>ABBREVIATED TASKS DATA BASE LOAD .......... D.1-6</td>
</tr>
<tr>
<td></td>
<td>ESTABLISHING A NEW ETAM PROJECT .......... D.1-9</td>
</tr>
<tr>
<td></td>
<td>EXECUTION OF THE RANGE OF EFFECT SEARCH .......... D.1-9</td>
</tr>
<tr>
<td></td>
<td>PROCESSING OF SEARCH RESULTS .......... D.1-11</td>
</tr>
</tbody>
</table>
## TABLE OF CONTENTS

### APPENDIX

<table>
<thead>
<tr>
<th>SECTION D.2</th>
<th>PROGRAM DOCUMENTATION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROGRAM P1</td>
<td></td>
<td>D.2-2</td>
</tr>
<tr>
<td>PROGRAM P2</td>
<td></td>
<td>D.2-7</td>
</tr>
<tr>
<td>PROGRAM P3</td>
<td></td>
<td>D.2-13</td>
</tr>
<tr>
<td>PROGRAM P5A</td>
<td></td>
<td>D.2-17</td>
</tr>
<tr>
<td>PROGRAM P5B</td>
<td></td>
<td>D.2-22</td>
</tr>
<tr>
<td>PROGRAM P5C</td>
<td></td>
<td>D.2-27</td>
</tr>
<tr>
<td>PROGRAM P7</td>
<td></td>
<td>D.2-32</td>
</tr>
<tr>
<td>PROGRAM P7A</td>
<td></td>
<td>D.2-38</td>
</tr>
<tr>
<td>PROGRAM P7B</td>
<td></td>
<td>D.2-41</td>
</tr>
<tr>
<td>PROGRAM P8</td>
<td></td>
<td>D.2-47</td>
</tr>
<tr>
<td>PROGRAM P8A</td>
<td></td>
<td>D.2-52</td>
</tr>
<tr>
<td>PROGRAM P9</td>
<td></td>
<td>D.2-58</td>
</tr>
<tr>
<td>PROGRAM P9A</td>
<td></td>
<td>D.2-62</td>
</tr>
<tr>
<td>PROGRAM P14</td>
<td></td>
<td>D.2-68</td>
</tr>
<tr>
<td>PROGRAM P16</td>
<td></td>
<td>D.2-72</td>
</tr>
<tr>
<td>PROGRAM P17</td>
<td></td>
<td>D.2-79</td>
</tr>
<tr>
<td>PROGRAM P18</td>
<td></td>
<td>D.2-84</td>
</tr>
<tr>
<td>PROGRAM P20</td>
<td></td>
<td>D.2-87</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SECTION D.3</th>
<th>NCSS EXECUTIVE SEQUENCES ASSOCIATED WITH MULTIPLE PROGRAMS</th>
<th>PAGE</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SECTION D.4</th>
<th>ETAM FILE FORMATS, AS IMPLEMENTED</th>
<th>PAGE</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SECTION D.5</th>
<th>MISCELLANEOUS CONTROL CARD SOURCE LISTINGS</th>
<th>PAGE</th>
</tr>
</thead>
</table>
TAEG REPORT NO. 40

TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>APPENDIX</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SECTION D.6  SOURCE LISTINGS OF PL/1 PROGRAMS</td>
<td>D.6-1</td>
</tr>
<tr>
<td>E  AN INTERACTIVE COMPUTER INTERVIEW</td>
<td></td>
</tr>
<tr>
<td>F  BIBLIOGRAPHY</td>
<td></td>
</tr>
</tbody>
</table>
THIS PAGE INTENTIONALLY LEFT BLANK.
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE NO.</th>
<th>DESCRIPTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-1</td>
<td>FORMAT OF DESCRIPTORS FOR INDEXING AN INNOVATION OR DATA BASE ENTITIES: TRAINING COURSES</td>
<td>A-4</td>
</tr>
<tr>
<td>A-2</td>
<td>FORMAT OF DESCRIPTORS FOR INDEXING AN INNOVATION OR DATA BASE ENTITIES: JOB TASKS</td>
<td>A-5</td>
</tr>
<tr>
<td>A-3</td>
<td>FORMAT OF DESCRIPTORS FOR INDEXING AN INNOVATION OR DATA BASE ENTITIES: INSTRUCTIONAL VEHICLES</td>
<td>A-6</td>
</tr>
<tr>
<td>A-4</td>
<td>EXAMPLES OF ENTITY SELECTION BASED UPON INNOVATION DESCRIPTORS (TRAINING COURSE DESCRIPTORS)</td>
<td>A-7</td>
</tr>
<tr>
<td>B-1</td>
<td>SUMMARIZED PRELIMINARY FEASIBILITY PROFILE</td>
<td>B-3</td>
</tr>
<tr>
<td>B-2</td>
<td>COST/SAVINGS DATA SHEET</td>
<td>B-4</td>
</tr>
<tr>
<td>B-3</td>
<td>DECISION TREE FOR ASSESSMENT OF TRAINING INNOVATION (EXAMPLE OF FORM COMPLETED WITH INITIAL OUTCOME VALUES AND PROBABILITIES)</td>
<td>B-5</td>
</tr>
<tr>
<td>B-4</td>
<td>DECISION TREE FOR ASSESSMENT OF TRAINING INNOVATION (EXAMPLE OF FOLDING BACK PROCESS)</td>
<td>B-6</td>
</tr>
<tr>
<td>B-5</td>
<td>FORMAT FOR DESCRIBING A SUPPLEMENTARY BENEFIT OR LIABILITY</td>
<td>B-7</td>
</tr>
<tr>
<td>D.1-1</td>
<td>INITIALIZATION (STARTUP) OF ETAM SYSTEM DATA SETS PRIOR TO CONSTRUCTION OF THE ABBREVIATED DATA BASES</td>
<td>D.1-4</td>
</tr>
<tr>
<td>D.1-2</td>
<td>DETAIL SYSTEM FLOW ABBREVIATED COURSE DATA BASE LOAD</td>
<td>D.1-5</td>
</tr>
<tr>
<td>D.1-3</td>
<td>DETAIL SYSTEM FLOW ABBREVIATED VEHICLE DATA BASE LOAD</td>
<td>D.1-7</td>
</tr>
<tr>
<td>D.1-4</td>
<td>DETAIL SYSTEM FLOW ABBREVIATED TASK DATA BASE LOAD</td>
<td>D.1-8</td>
</tr>
<tr>
<td>D.1-5</td>
<td>DETAIL SYSTEM FLOW PROJECT FILE GENERATION AND RANGE-OF-EFFECT SEARCH</td>
<td>D.1-10</td>
</tr>
<tr>
<td>D.1-6</td>
<td>DETAIL SYSTEM FLOW RANGE-OF-EFFECT SEARCH RESULTS PRINT/EDIT</td>
<td>D.1-12</td>
</tr>
</tbody>
</table>
THIS PAGE INTENTIONALLY LEFT BLANK.
INTRODUCTION

A major topic in the Educational Technology Assessment Model (ETAM) is a body of knowledge and a set of procedures for determining the range of courses, job-tasks, and instructional vehicles in the Navy that can be affected by a proposed innovation. This range establishes the multipliers of benefits, liabilities and costs that are potential in the innovation, and therefore central to decisions about adoption or rejection.

The initial phase of ETAM specified a taxonomic structure within and between major classes of training entities—courses of instruction, job-tasks, instructional vehicles. Elements in the classification system were each identified and clearly defined. Taken all together, the terms in this classification system were intended to structure the essential functions in training technology. The terms, as sets of functional descriptors, were to be used for the practical indexing of entities such as courses, job-tasks and instructional vehicles in the Navy's inventory. The same set of terms (and their defining concepts) would also be used in "describing" the innovation, or rather in "indexing" the innovation.

The descriptor set used in indexing the innovation could then be applied to the data base inventories of Navy training courses (for example) in making a first pass at selecting those courses that were relevant to the innovation—perhaps 50 out of over 4,000 courses, or 200 out of over 4,000 courses.

This study reports the moving of this background effort towards implementation. A procedure is described for indexing an innovation so as to create the content of a search query to be applied by automatic search on a computerized data base. Techniques are described for indexing data base entities such as training courses, job-tasks and instructional vehicles. The same descriptor set is used that is applied to the innovation. Data about each of the entities are contained in Navy listings with, usually, brief descriptions.

Because the application of subject matter descriptors to innovations and to data base entities with the intent of using one to search on the other requires a good deal of human judgment, the strategies for indexing searches and indexing data base entities are discussed. "Indexing" means the selection of a group of descriptors to apply to the data base entity or to the search specification. Also, a rationale for the search procedures specified for ETAM and for the indexing procedure applied to data base entities is offered. Simplicity for the user is a prime consideration.

The final judgment of relevance or irrelevance of an entity to an innovation is not based on automatic matching of descriptors in the search specification and data base entities. Rather, it is a series of operations in which the human
assessor of the value of the innovation examines progressively more contextual information about each candidate entity. The candidate entities are those that survive each next step in winnowing out those that are clearly irrelevant.

Under some specified search conditions, the ETAM level of search is bypassed for direct access to the applicable Navy data base. The identification of entities relevant to the proposed innovation enables specific judgments of magnitude of benefit, liability and cost factors to be taken into account for each entity. There are specific references that can be objectively cited for assessing overall benefits, liabilities and costs. Thus, the validity of the ensuing decision to accept or reject or modify the innovation is given strength; and, if the innovation is adopted, the targets for its application have already been specifically, rather than vaguely, identified.

Recommendations are included that, before widespread adoption, the search and indexing procedures should be tried out and, where necessary, fine-tuned within the defined objectives and theoretical models underlying the present taxonomic structure of ETAM.

PROCEDURES FOR INDEXING THE INNOVATION

It is assumed that the indexer of the innovation or of the data base entities (training courses, instructional vehicles, job-tasks) has training and expertise in the index content and definitions, and has acquired some proficiency in search strategies such as are discussed later in this appendix.

INDEXING THE INNOVATION. The following procedures assume that the indexer is familiar with the definitions of taxonomic terms in the ETAM Phase II-A Final Report, specifically with the procedures described in Task 5 for Determining Range-of-Effect.

The source data which the assessor converts into an indexed description of the innovation may consist of various prose descriptions of the innovation and its intended uses and applications. The assessor may, through his general background in training and training operations in the Navy, amplify or modify this description in his thinking.

The assessor will know that the hits revealed by the indexed search will be only candidates for further examination in determining relevance. The further examination will be made by the assessor looking at the titles of the entities and further contextual descriptions of them.

His indexing policy will tend, therefore, to be generous and lean toward inclusion rather than exclusion of possibilities.

1. Determine Type of Innovation

Follow the procedures described in the ETAM Phase II-A Final Report, Task 5, in determining the type of innovation—primarily applicable to training courses, job-tasks, or to instructional vehicles. This step may be merely a starting point. It is possible that the course of inquiry may lead to all three classes of entities being explored for potential benefits, liabilities, costs and other effects.
2. Determine Use of ETAM Or Other Data Base

If the innovation applies with certainty only to concretely named entities, bypass ETAM search and go directly to the applicable Navy data base such as CANTRAC on courses, Naval Directory of Training Devices or Occupational Standards on job-tasks. Use of the name of the entity, for example radar ASQ-7, as the key for identifying the relevant training courses or job-tasks or instructional equipment. This is the fastest and most efficient search method—using the most specific and uniquely identifying characteristics of the object search in the search query.

Assume, however, that a subject matter search rather than a named entity search is indicated by the characteristics of the innovation. Proceed to Step 3.

3. Describing the Innovation for Search Query

For reasons already stated, the description of the search query may be somewhat broader in scope than would be made by a "tight" characterization of the innovation for purposes other than determining potential range-of-effect.

Select the applicable format type—training courses, job-tasks, or instructional vehicles. (See Figures A-1, A-2 and A-3.) It is advisable to prepare the descriptor selection with a paper format. Put a check mark or circle around a numbered item to indicate selection of a descriptor. If the descriptor is to indicate a NOT characteristic for excluding known non-relevant entities, put a large negative sign in front of the numbered item.

Scan the entire set of descriptors on the format before making selections. The following rules govern the manner in which query descriptors should be formulated.

- Only an item with a numerical prefix code can be selected. "Categories" of descriptions are underlined (or in capital letters) and merely title a meaningful group or subset of the description.

- Query descriptors between categories form a logical AND relationship. All must be present in the index descriptor for the entity if selection is to take place. However, if the indexed entity in the ETAM data base does not have an indexed entry under a given category, but the search query does have a selected descriptor under that category, the search operation will ignore that category in selecting or rejecting that entity in the data base. (See examples 1a, 1b and 1c in Figure A-4).

- Query descriptors within a category form a logical OR relationship. Any one that exists in the search query that is present in the index descriptor for an entity will cause the entity to be selected. (See examples 6a and 6b in Figure A-4.) This assumes, of course, that an AND relationship does not exist due to a query descriptor in another category. (See example 5e in Figure A-4.)
ETAM TRAINING COURSE DESCRIPTORS

Reference knowledge
(01) System purposes
(02) Organizational roles
(03) Contexts of operation
(04) Organizational rules
(05) Other

Enabling knowledge
(10) Operational goal criteria
(11) Nomenclature, identif., location
(12) Procedural descriptions
(13) Job relevant facts, rules
(14) Other

Task formats
(20) Procedure formats
(21) Decision formats
(22) Construction formats
(23) Other

Gross job categories
(30) Operations
(31) Maintenance
(32) Service and administration
(33) Command
(34) Other

Objective task variables as manifest in the training

Equipment and objects used:
(40) Real (41) Simulated (42) Symbolic

Environments in which task is trained:
(43) Real (44) Simulated (45) Symbolic

Tools/instruments used in training:
(46) Real (47) Simulated (48) Symbolic

Reference/enabling information in doing task:
(49) Applied (50) Not Applied

Criteria of task performance:
(51) Real (52) Simulated (53) Symbolic

Task functions dominant in training
(60) Goal projection
(61) Scan-detect
(62) Identify
(63) Interpret
(64) Procedure following
(65) Decide
(66) Construct, plan
(67) Track
(68) Motor performance
(69) Interpersonal interaction
(70) Recall task-cycle information
(71) Recall enabling information
(72) Adapt improvisationally/impromptu
(73) Other

Stage of learning
(80) Orientation, familiarization
(81) Task nomenclature, identifications, locations, facts, rules
(82) Task formats at conceptual level
(83) Procedures at verbal level only
(84) Task components with guidance
(85) Entire job-task procedurally at barely acceptable mastery
(86) Highly proficient in work context
(87) Unusual task conditions
(88) Performance at key man level
(89) Refresher learning

FIGURE A-1. FORMAT OF DESCRIPTORS FOR INDEXING AN INNOVATION OR DATA BASE ENTITIES: TRAINING COURSES
ETAM JOB-TASK DESCRIPTORS

ADMINISTRATIVE

Routine paperwork
(01) Forms filling
(02) Document file-management
(03) Decide-encode
(04) Screen-filter distribute
(05) Other

Non-routine paperwork
(11) Construct messages-reports
(12) Analyze-interpret-evaluate
(13) Construct recommendation-proposal
(14) Construct-plan
(15) Other

Office equipment operation
(21) Typewriter
(22) Telephone, etc.
(23) Reproducer
(24) Computer terminal
(25) Teletype
(26) Other

INTERPERSONAL

Inform-instruct-manage
(31) Brief-debrief
(32) Instruct-train
(33) Assign-monitor-coordinate
(34) Evaluate
(35) Advise-inform-negotiate
(36) Other

TECHNICAL

Procedures
(41) Sequential
(42) Strategic-adaptive
(43) Interpersonal-team member
(44) Other

Type of procedure (main emphasis)
(51) Scan-detect: symbolic (including maps, transduced signals, radar)
(52) Scan-detect: natural events, things
(53) Identify: symbolic (including transduced signals)
(54) Identify: natural events, things

TECHNICAL (Cont'd)

(55) Interpret: symbolic
(56) Interpret: natural
(57) Perceptual-motor
(58) Cognitive operations
(59) Manual
(60) Communicate

With or without equipment
(61) With equipment or applied to equipment (paper is "equipment")
(62) Without equipment

Decide
(71) Diagnose-analyze
(72) Select-choose
(73) Under stress or load
(74) Other

Construct-repair-plan
(81) Manual construct or repair
(82) Cognitive construct-plan

Track-aim-steer
(91) Applicable
(92) Time stress
(93) Information-load stress

FIGURE A-2. FORMAT OF DESCRIPTORS FOR INDEXING AN INNOVATION OR DATA BASE ENTITIES: JOB-TASKS
### INSTRUCTIONAL VEHICLES: DESCRIPTOR LIST

<table>
<thead>
<tr>
<th>Vehicle type</th>
<th>Vehicle properties</th>
<th>Type of external control operated by student</th>
<th>Feedback presentation logic</th>
<th>Response evaluation logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>(01) Instructor</td>
<td>(21) Visual</td>
<td>(61) Not applicable directly</td>
<td>(71) Not applicable</td>
<td>(81) Not internal—depends on instructor or student evaluation</td>
</tr>
<tr>
<td>(02) Static graphics</td>
<td>(22) Auditory</td>
<td>(62) Artificial or symbolic response</td>
<td>(72) Selects next stimulus item or</td>
<td>(82) Evaluation limited to student's immediate response</td>
</tr>
<tr>
<td>(03) Animated graphics</td>
<td>(23) Kinesthetic/vestibular</td>
<td>(63) Representational response by symbolic selection</td>
<td>(73) Gives evaluation of preceding</td>
<td>(83) Evaluation extended to a set of student responses</td>
</tr>
<tr>
<td>(04) Audio</td>
<td>(24) Tactile</td>
<td>(64) Representational response by dummy control activation</td>
<td>(74) Selects and presents guidance information</td>
<td>(84) Tolerance limits on acceptable student responses: FIXED</td>
</tr>
<tr>
<td>(05) Physical models, demonstration</td>
<td></td>
<td>(65) Task-manipulative response, non-dynamic in time and force</td>
<td></td>
<td>(85) Tolerance limits on acceptable student responses: VARIABLE</td>
</tr>
<tr>
<td>(06) Procedural trainers: symbolic</td>
<td></td>
<td>(66) Task-manipulative, dynamic and interactive</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(07) Procedural trainers: physical but not functional</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(08) Procedural trainers: functional</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(09) Task and system simulators</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(10) Real equipment, itself</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class of training objective</th>
<th>Type of content displayed</th>
<th>Selection source for sequencing</th>
</tr>
</thead>
<tbody>
<tr>
<td>(11) Reference knowledge</td>
<td>(31) Text-verbal</td>
<td>(51) Internal program</td>
</tr>
<tr>
<td>(12) Knowledge-task specific/enabling</td>
<td>(32) Diagrammatic</td>
<td>(52) Instructor</td>
</tr>
<tr>
<td>(13) Task-skill formats</td>
<td>(33) Abstracted pictorial representation</td>
<td>(53) Student choice</td>
</tr>
<tr>
<td>(14) Skill training</td>
<td>(34) Pictorial representations</td>
<td>(54) Student performance</td>
</tr>
<tr>
<td></td>
<td>(35) Physical representations</td>
<td>(55) Combinations of the above</td>
</tr>
<tr>
<td></td>
<td>(36) Other</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type of presentation sequence</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>(41) Library of frames or items</td>
<td></td>
<td>(51) Internal program</td>
</tr>
<tr>
<td>(42) Presentation sequence not applicable</td>
<td></td>
<td>(52) Instructor</td>
</tr>
<tr>
<td>(43) Fixed sequential frames or items</td>
<td></td>
<td>(53) Student choice</td>
</tr>
<tr>
<td>(44) Random selection of frame sequences</td>
<td></td>
<td>(54) Student performance</td>
</tr>
<tr>
<td>(45) Dynamic change of content within frame</td>
<td></td>
<td>(55) Combinations of the above</td>
</tr>
</tbody>
</table>

**FIGURE A-3. FORMAT OF DESCRIPTORS FOR INDEXING AN INNOVATION OR DATA BASE ENTITIES: INSTRUCTIONAL VEHICLES**
<table>
<thead>
<tr>
<th>EXAMPLE NO.</th>
<th>INNOVATION QUERY DESCRIPTOR</th>
<th>SAMPLE ENTITY INDEX DESCRIPTOR*</th>
<th>SELECTED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1a</td>
<td>03 21</td>
<td>03 21 30</td>
<td>YES</td>
</tr>
<tr>
<td>b</td>
<td></td>
<td>03 22 30</td>
<td>NO</td>
</tr>
<tr>
<td>c</td>
<td></td>
<td>03 30</td>
<td>YES</td>
</tr>
<tr>
<td>2a</td>
<td>03</td>
<td>03 21 30</td>
<td>YES</td>
</tr>
<tr>
<td>b</td>
<td></td>
<td>21 22 30</td>
<td>YES**</td>
</tr>
<tr>
<td>3a</td>
<td>21</td>
<td>03 21 30</td>
<td>YES</td>
</tr>
<tr>
<td>b</td>
<td></td>
<td>21 22 30</td>
<td>YES</td>
</tr>
<tr>
<td>4a</td>
<td>03 -21</td>
<td>03 21 30</td>
<td>NO</td>
</tr>
<tr>
<td>b</td>
<td></td>
<td>03 22 30</td>
<td>YES</td>
</tr>
<tr>
<td>5a</td>
<td>03 -21 22</td>
<td>03 21 30</td>
<td>NO</td>
</tr>
<tr>
<td>b</td>
<td></td>
<td>03 22 30</td>
<td>YES</td>
</tr>
<tr>
<td>c</td>
<td></td>
<td>03 21 22 30</td>
<td>YES</td>
</tr>
<tr>
<td>d</td>
<td></td>
<td>03 30</td>
<td>YES</td>
</tr>
<tr>
<td>e</td>
<td></td>
<td>02 21 22</td>
<td>NO</td>
</tr>
<tr>
<td>6a</td>
<td>21 22</td>
<td>03 21 30</td>
<td>YES</td>
</tr>
<tr>
<td>b</td>
<td></td>
<td>03 22 30</td>
<td>YES</td>
</tr>
</tbody>
</table>

* Descriptors underlined are primary basis for selection/non-selection.
**No 01-05 index descriptors - therefore entry is selected.

FIGURE A-4. EXAMPLES OF ENTITY SELECTION BASED UPON INNOVATION DESCRIPTORS (TRAINING COURSE DESCRIPTORS)
A query descriptor will select any entity which does not have any index descriptor in that category, provided any AND relationship is satisfied. (See example 2b in Figure A-4.)

Select one or more descriptors within a category. It is not necessary to select a descriptor in every category—a search query could conceivably consist of only one descriptor in one category. If a descriptor within a category is selected, it should best fit the innovation. If the assessor is uncertain about which descriptors best fit the innovation and the intent of this stage in search, he may select several descriptors within one category and thereby play it safe. Descriptors selected within the category are logically OR'd to each other. This means that (with respect to this category) the entity will be selected as a hit if either one or the other descriptor (or both) match the index descriptors in the data base entity. This OR relationship is automatically created when the assessor selects more than one descriptor within a single category. He may select as many descriptors within a category as he chooses. The OR relationship among index terms serves to expand the range of entities in a data base that will be selected.

The assessor may be aware that if some index descriptor appears in a data base entity, that entity is thereby irrelevant to the search objective. He then enters the number of the descriptor along with a NOT function that is applied to it. This means that if an entity in the data base contains that descriptor in its index, the entity will be rejected no matter how well it matches other search descriptors. Notice that it makes sense to use a NOT descriptor only if no positive descriptors are selected within the same category that contains the NOT descriptor. However, it should be recognized that what may be logically equivalent in the formulation of search arguments may not be psychologically equivalent. In any event, if the assessor inadvertently includes a NOT descriptor in a category which also contains positive valued descriptors, the search logic ignores the NOT item as a redundant and irrelevant specification. (See example 5c in Figure A-4.)

The descriptor OTHER: Ordinarily in constructing a search specification this descriptor is ignored. But on some occasions, none of the descriptors in a category fits, but the searcher feels strongly that there ought to be a descriptor in this category for the innovation. He may ask for OTHER and be prepared to scan through the miscellany of entities that the indexers of the data base have labeled with this descriptor because they too could not find the right descriptor for it. It is likely to be a long chance. The descriptor OTHER may be used along with other descriptors in the category in order to be sure that all the possibilities have been examined.
The search specification is readyed for entry. The assessor may request—or automatically be presented with—a display or printout of the entire search specification. This display is identical in appearance to the format for selecting descriptors, except that only the selected descriptors—under their category names—are presented. Any NOT descriptors are so identified in the display with a preceding minus sign. The specific search operation is also identified (in human readable symbols) with the innovation's project file number and a search entry code identification. These data and the particular search specification will be retained for retrieval. It should also be carried along with any output listings that result from the data base search.

4. The Search Specification Is Entered

The assessor interactively selects a function that enters the specification into data base search.

5. Preliminary Search Output: Count of Hits

Because the search specification may produce a very large list of hits that would be time-consuming to display, the system begins by presenting an overall count of items that match the search specification. The assessor may recognize at once that something is wrong with his search specification. He may then review and revise it, and resubmit it for search.

6. Inspection of a Sample of Hits

The assessor may request display of a sample of the listing. Perhaps the first twenty-five items are displayed to him. The content of listings in the ETAM data base are limited to the title of the entity and the Navy data base code identification of the entity. In the case of a training course, the CIN/CDP codes and title of the course are presented. In job-tasks, the rating, rate, job-task code number and title according to the Navy's Occupational Standards directory is given. In instructional vehicles, the Directory of Naval Training Devices code identification, Federal Stock Number, and name of the device are presented.

7. Revision of Search Specification

The listing of the previous search results to the assessor may be sufficient for him to revise his search specification. This is likely to make it more constraining. He may resubmit the search.

8. Rejection of Unwanted Entities

In many cases, the brief listing of an entity may provide sufficient context information to enable the assessor to reject the entity from further consideration. An interactive program will be provided to assist the user in the editing of the results of a search. This manual editing will allow deletion or addition of entities as required.
9. Saving the Results of a Search

When the user has made his selection of wanted entities in the results of a search, the results will be automatically saved as a component of the project data base. The results are subject to further perusal or editing as required.

10. Combining the Results of Successive Searches

For the purpose of procedural simplicity, the user is not given the opportunity to formulate complex Boolean search arguments. A complicated pattern of AND and OR relationships requires the formulation of a number of individually simpler search specifications. The user may use the interactive search results editing program to combine search outcomes as required.

11. Getting Further Context from Navy Data Bases

Ordinarily the information gleaned from entity titles is not likely to be sufficient for final decisions of relevance to the innovation. (Ordinarily it requires less information to make the decision of "irrelevance" than the decision of "relevance." The decision alternatives here are "relevant," "irrelevant," and "undecided." ) The coding format in ETAM of Navy data base entities may be equivalent to that in the respective Navy data base. If so, a machine readable output from ETAM may serve as direct input to the Navy data base. If not, some manual translations may be required for accessing the computer files. The paper equivalent of the same files may be accessed by the code designators. The latter makes for more nuisance and time in paper shuffling, but is not impossible.

In the final stages of decision about relevance, the assessor will be scanning candidate course descriptions at the level of CANTRAC summaries or more detailed course outlines and schedules, NOTAP (Naval Occupational Task Analysis Program) job-task descriptions lying behind the job-task titles in the Occupational Standards directory, or the descriptions in the Naval Training Devices listings. In the case of instructional vehicles, the TECEP* directory can give still greater assistance in focusing on equipment and media types of relevance to the inquiry.

The fact that the listing of relevant entities will be coded consistent to Navy data base codes for the same entities will enable access to cost, usage and other data essential for the processing of costs and benefits in later ETAM tasks.

12. Entering Results Into the Project Data Base

Search results, edited as required, are saved as a specific file of the project data base for that innovation. The source search arguments that were used as a basis of the search are also preserved as separate files of the same project data base. The contents of either the search arguments or the search results could be applied to other innovations that are equivalent or overlapping in objectives.

COMMENT. This description has not included a variety of user convenience functions at the terminal. These deal with such factors as user ID, file ID, correction of unintended entries, tutorial services dealing both with the procedure itself and with reference to the meanings of each of the descriptors in the descriptor sets. These are important considerations, but outside the present scope of effort to implement.

PROCEDURES FOR INDEXING INSTRUCTIONAL VEHICLES, TRAINING COURSES AND JOB-TASKS

Indexing the subject matter in a data base is a discipline. It requires a working conceptual knowledge of the taxonomic nomenclature, but no nomenclature of subject matter is free of substantial ambiguity. The ETAM indexing nomenclature is no exception.

In ETAM, the objects to be indexed have two orders of ambiguity. Although the intent of ETAM is to index, say, a given instructional vehicle, the indexer must use as his reference not his direct knowledge about that vehicle, but a summary description of the vehicle. The same applies to training courses and to job-tasks. The indexer must be able to interpret the often cryptic and always incomplete descriptions of the objects being referenced. His task, thus, is more difficult than the librarian kind of indexer of documents who can work with the document content at its face value. Assigning a descriptor to an object is therefore an act of interpretive judgment. The judgment centers on the tradeoff between what is likely to be gained for retrieval objectives by adding, or by not adding, a given index term to the title or name of the object.

The ultimate validity of an indexed data base can be tested not by truth but by practicality. Assume the following conditions: A sample of users--in this case, of ETAM assessors--is given unrestricted time in which to search exhaustively through all of the data base content of course titles, course descriptions and other references. The objective of the searches is to match, say, training courses relevant to the knowledge of the specific training innovation. This process results, presumably, in a number of hits on "relevant" objects. Now let equivalent searchers use the subject matter index and the indexed data base for equivalent searches. To the extent that results in hits from the second procedure match those obtained by the first procedure, the index and indexing operations (taken together as they must be) are valid or effective.

A comparison of the time effort and other costs associated with each of the two procedures would be measures of relative efficiency. In the practical world of cost-to-benefit ratios, even substantial reductions from a measure of Utopian validity can be offset by large cost reductions. In the design of search procedures, there tends to be a tradeoff between the average effort spent in putting the needle into the haystack and the average effort that will be spent in finding it, or the cost of not finding it at all.
This argument is intended to motivate the would-be indexer who correctly realizes that the notion of "perfect indexing" is meaningless. In technical jargon, subject matter indexing is in the class of operations that are "heuristic."

INDEXING OBJECTIVES

Policy 1. The primary objective of indexing the data base is to help the ETAM analyst determine the range-of-effect of a proposed innovation or change to existing (or projected) Navy practices or facilities for training.

The indexer should strive continuously to bear in mind his goal object--to give the ETAM user a practical tool for determining the range-of-effect of a proposed innovation in training technology (whatever its nature). The indexer is creating a job aid. The indexed content is not an end in itself.

A different user and different purposes would certainly generate a different indexing content and structure, and different indexing policies.

Policy 2. In assigning descriptors, err on the side of including the irrelevant rather than excluding the relevant.

This policy is based on the importance of getting the fullest reasonable range of applicability of the innovation or change on which the data base query is based. Rejects at this search level will not again be candidates for inquiry. The indexer must also remember that his index will generate only a preliminary pass at selecting candidates, and that final decisions to accept or reject objects yielded by the automatic search will be made by the ETAM assessor while examining the full context of course titles, course descriptions and so on.

Obviously this advice must be followed temperately. It applies where the indexer has reason to be ambivalent about including or excluding a given descriptor while indexing an object. He will have to take into account the entire context of the description of the object plus his interpretation of what lies behind the description presented to him. The significance of the descriptor to that context should also weight his decision to accept or reject. If the descriptor, or an approximation of the descriptor, seems to have a central importance to the context (of the course, of the vehicle, of the job-task or job) then he should weight in favor of inclusion and let the assessor later decide on relevance. But if the descriptor seems to apply to a relatively unimportant factor in that particular training, or job or vehicle, there is little practical loss if when he is uncertain he rejects using the descriptor. By so doing, he increases search efficiency for the ultimate user.

Indexing Procedure. The following steps include the preparation by the indexer for indexing the ETAM data base content. It is assumed that the indexer will become a skilled specialist in performing this function as contrasted with a host of persons performing this function only occasionally.

1. Acquire a clear picture of the task objectives of the users of the indexed materials.

A continuing awareness of the purpose to be served by the indexed content should help the indexer to do what is sufficient and necessary for the ETAM user. The indexer's strategies
and decisions can remain clearer and more relevant if purpose is restricted to specific needs. The indexing of content for retrieval is a reciprocal operation to querying and searching. Ideally, the indexer acquires some experience in simulated search problems and thus becomes sensitized to the searcher's difficulties.

2. Learn the nomenclature and concepts of the indexes.

The meaning of each index term is given in the ETAM reference material. In many cases, these meanings are somewhat specific to the ETAM context. For example, the term "decision format" cannot be sensibly used without the specific ETAM definition. The user who queries the data base will use the same reference material for selecting index terms in his queries. It is therefore essential that the person who indexes the data base objects, and the person who indexes queries to that data base, share in a single reference that defines the index terminology. Notice that most index terms are clustered into categories or groups. These groupings have a contextual purpose for both the indexer and the searcher. After a number of hours that combine study and application practice, the indexer will begin to find that the appropriate index terms will often emerge almost automatically when he looks at the reference content that describes the object to be indexed. This automaticity has value in efficient classification; it has a liability in that the indexer may become stereotyped in assigning descriptors when he should be exercising contextual judgment.

3. Learn the search rules and strategies that will be applied to the indexed materials.

The indexing of the data base content is a critical step in determining a range-of-effect. But it is only the first in a series of steps taken by the assessor in seeking to match the description of an innovation with entities identified in the data base and ultimately with real entities--jobs, courses and instructional vehicles. The indexed search operations could result in a collection of "probable" relevant items and the rejection of "improbably" relevant items. The final decision of relevance occurs when the assessor examines the contextual descriptions of the items that have been screened as probably relevant.

The indexer should be intimately familiar with these operations and strategies, and the kinds of judgment the assessor will have to make as he proceeds. The indexer can stipulate that the assessor uses the same definitions and meanings of index terms that he uses and which are contained in the shared glossary reference.
By being aware of the judgmental operations of the assessor and query-maker, the indexer can realistically control criteria for precision in indexing. This awareness can be crucial to the indexer's speed and comfort in assigning descriptor nomenclature to object descriptions. He should, of course, have a realistic knowledge of what happens when a data base search is screened by descriptors in various logical relations--AND, OR and NOT, according to ETAM query construction rules. (See preceding material in this appendix for those rules.)

4. Practice assigning descriptors to several scores of entities in each entity class.

The results of this practice are thrown away. The intent is two-fold. One is to master the classificatory format so that the indexer can think with it more or less spontaneously. After a first pass at indexing several dozen entity descriptions (at whatever the level of description is to be used), perform a second and independent pass on the same sample of materials in a day or two--after some degree of forgetting of the first pass. Then compare the results of the first and second passes for reliability. There will be variations, and this would be expected even among highly skilled indexers. But the novice indexer should attempt to think through and account for the differences between his first and his second indexing of the same item. It is not necessary to think of one or the other as being "wrong" when two or more passes on the same material do not match precisely. This thinking through of the variations can lead to a relatively consistent decision policy in assigning descriptors.

5. Index on the basis of entity context as well as specific elements of description.

Assume one is indexing a course title and description. Before assigning any descriptor terms, one should read all of the information that is going to be used (or is available) for the indexing of the course as an entity. One should get the entire context in mind. This enables judgment in selecting index terms that will be useful in a practical way. For example, an operator's course on radar equipment mentions that "routine maintenance operations and diagnosis" is an item of study. The entire course is limited to 10 days. It is a pretty safe bet that the "maintenance diagnostics" will be limited to a few external, procedural tests in order to determine whether the set is operationally usable or not. Almost certainly there will not be circuit and module levels of troubleshooting taught and learned, nor complex inferential strategies applied by the operator. Therefore, the expressions "maintenance" and "diagnosis" in this context would probably not warrant applying to this course, or course segment, nor the descriptors "maintenance" and "decide" (the latter including the meaning of "diagnose") for that course, at least on the basis described here.
The indexer will, of necessity, have to make a number of inferences of this kind. Sometimes his guesses will be wrong. But if he has a general background knowledge of the subject matter (training courses in this example) and uses the contextual information presented to him, his batting average must be better than if he relies only on words and phrases taken out of context.

The descriptor "Other" appears in some categories. This is an escape hatch and should be used with reluctance. On occasion, however, a category may connote concepts that are important for identifying something important about the entity to be indexed, but none of the descriptors in the category make even an approximate fit.

Use the term "Other" in the expectation that a searcher will sometime have a reciprocal difficulty in finding the right descriptor—or several descriptors—and specify "Other." The "Other" file will be a miscellany of items that the searcher will have to look through in terms of their contexts in the hope of finding a hit.

If several indexers agree that the "Other" term is being used too frequently in a given category, the definitions of the other descriptors should be restudied. If this is still unsatisfactory, the definitions of the descriptors may have to be somewhat modified. If this is still unsatisfactory, another descriptor may have to be added to the category.

But changes to the present index, especially after any widespread adoption, should be made with great reluctance. That is because changes will invalidate to some degree the existing indexes of data base entities. A test of motivation to change is willingness to reindex existing entries. This is less a threat if an index term is added to the present list. But as index terms are added, the human difficulties in using the index, either for search or for entities, escalate.

6. Use descriptors to identify key factors in the entity.

Apply a descriptor to an entity (course, job or vehicle) only if the descriptor identifies something important in a practical sense. Some examples will clarify.

Although there can be exceptions to the following principle, it is generally a good one to follow—a factor, feature or element which is less than 5% of the cost, or time or function of the total entity does not justify identification by a descriptor. Thus, a fifteen minute discussion on "system purposes" in a two-week course for operator or maintenance personnel would not justify that the index to these courses include the descriptor "system purposes." But, if a half day or a full day in the two-week course were spent on this topic, it would justify the inclusion of the descriptor.
The 5% principle does not apply if the reference is highly important. The solo pilot may spend less than 5% of total mission time on navigation, but it is a key function to mission effectiveness. This would therefore be an appropriate activity to account for in a job-task characterization—in our indexing system, navigation would be indexed as "interpret" and "cognitive operations."

The data for making these quantitative and qualitative judgments rarely appear in the entity descriptions upon which the indexer generally depends. In consequence, the indexer should have general knowledge and practical acquaintance with the subject matter he is indexing.

Assume that the indexer has general knowledge background of the subject matter being referenced by the description from which he is making the index; assume also that the indexer has thorough familiarity with the meanings of all of the descriptor terms; finally assume that he has had a few dozen hours of experience in applying the descriptor set to these entity descriptions. It should take him between two and five minutes, with a median value of less than three minutes, to index whatever entity is being indexed—the job-task, the course of instruction, or the instructional vehicle. Some entities—such as homogeneous units of instruction—could be indexed far more rapidly—less than a minute. Entities in each of these kinds of materials tend to group themselves so that the indexer learns to recognize what pattern of descriptors applies to members of the subsets. Additional time is required when a standard pattern is not the case.

If the indexer labors for many minutes on a large proportion of individual entities or items, agonizing over the choice of a descriptor, something is wrong. He has not been properly oriented to the task—he may be unduly compulsive about resolving ambiguities, or his information background for the task may be inadequate. On the other hand, if the indexing always proceeds as rapidly as his hand can check off descriptors, he may not be giving enough critical examination and thought to the subject matter. Properly done, indexing is hard mental work. It requires holding a context of information in mind and exercising degrees of judgment not only in what descriptors to select, but also which ones to exclude.

7. Use a standardized format for indexing.

One indexer of samples of job-tasks, courses and instructional vehicles used copies of the format shown in Figures A-1, A-2 and A-3. A special heading was added that specified Navy data base identification code, title of the entity and, as a special factor in course descriptions, course length in days. These data were written by hand and, in many cases, the careful writing (lettering for legibility) often took more time than the indexing of the entity.
A circle drawn around the code number of the descriptor meant the selection of that descriptor. One sheet of paper was used for each entity. It was easy to erase in making corrections or changing a choice. A separate manual step was required for entering these data into the computer.

This indexer felt more comfortable with the entire layout of the descriptor set before him than he would if he had to select or reject descriptors, one at a time, by the prompting method. In fact, this is probably an undesirable procedure. Unless a graphic display can present the entire descriptor pattern all at one time, as well as obviate the need to copy entity titles and codes by hand, the paper format may continue to be preferred.

SEARCH RATIONALES AND STRATEGIES

Consider a data base conceptually like the two parts of a telephone book. On the white pages there appear unique identifiers associated with a given telephone--subscriber name, subscriber address, subscriber telephone number. Assume, like social security numbers, that there are no duplicate subscriber names. Given the subscriber's name, his telephone number can be unequivocally identified. There is no ambiguity in reference. If the directory is up-to-date the telephone number linked to the subscriber's uniquely identifying name or code is evoked without uncertainty.

Think now of using the yellow pages. Assume you wish to find the telephone number of someone who sells mousetraps. There is no subject matter entry called "mousetraps....sellers of--." You must now make inductive and deductive inferences about the class of sellers who sell mousetraps. Perhaps you think of "rodent control" or "department store" or "hardware store." You are making inferences that this class term includes a seller of mousetraps. You may be right in any of these cases, or wrong in some or all of them. Subject matter descriptors are ambiguous even when carefully defined, unless they specifically point out exhaustively all the members of the set in the class name. This latter is an example of a "closed set." Most class names identify open sets. In ETAM indexing we will be talking mainly about class names that apply to open sets of references.

SUBJECT MATTER SEARCH IS HEURISTIC. Subject matter names, as contrasted with unique entity names, are to some degree ambiguous. The assignment of a subject matter term to an entity in a data base requires some judgment, and the selection of a subject matter term for a search query also requires some judgment. Matching a subject matter intent with a subject matter description therefore is not the result of rigorous processes; consequently, success is only probabilistic and may be increased by the use of a different kind of probe. Success probability also increases to the extent that the indexer of the subject matter shares his definition of subject matter terms with the searcher who selects terms for the query.
There may be cases where the searcher's concept of what he is searching for may be appropriately modified either by inspection of the list of permissible index terms (in a closed indexing system) or by examining the content of an early query and consequently changing the terms he has used in previous queries.

That a success is probabilistic in some degree, and may be based on iterative, converging operations, justifies treating subject matter searching as a heuristic rather than as a rigorous operation. This fact should make a difference in how the search procedure, and its information support, is designed.

SUBJECT MATTER SEARCH IS STRATEGIC. This means that some policies in procedure will be more efficient or more effective, or both, than other policies. Strategies are formulated on the basis of tradeoff variables which may be optimized formally (as by equations) or informally (as by human judgment).

The purpose of search is to find in some universe of entities (such as all Navy jobtask in a data base of job-tasks) only those entries that fit the search objective and all of those entries that fit the search objective. The search objective is not necessarily the equivalent of a search specification. The search objective is what the searcher wants or needs to find in the context of a specific innovation; the search specification is his translation of this need or requirement into descriptors and logical connectors among search descriptors that come from the index lists.

We must assume that the searcher can recognize an entity appropriate to his objective if he can inspect all of the information context that is associated with that entity such as, in this case, a full description of the task--its setting, tools and equipment and objects worked on or with, the work functions performed, environment performed in, and explanatory text.

Assume that the data base has 30,000 job-task titles (entities) and associated with each job-task title is a set of descriptors, an abstract, and a full textual description of the job-task entity. An inefficient, but highly effective way of searching, would be reading each and all of the full text descriptions for all 30,000 job-tasks. The next more efficient method, but less certain of complete effectiveness, would consist of reading all abstracts and, only if the abstract sounded promising, to read the full text description about the entity. This procedure entails some risk of missing desired entities as well as selecting entities that turn out to be irrelevant, because the abstract is an incomplete context and susceptible to misrepresentation.

The next more efficient method, but correspondingly more risky, is to make the first winnowing on the basis of index terms assigned to entities--descriptors. (It is possible that an intermediate method would consist of examining merely the titles of the 30,000 entities by a human interpreter of the probable meaning of each task title.)

We can see that there is a tradeoff between the degree of search efficiency--the labor spent in the search--and the degree of search effectiveness (getting all of the relevant entities in the universe of entities). This tradeoff extends to the size of the index list and the complexity of descriptor relationships that are assigned to entities in the data base.
There is another kind of tradeoff—this is between the precision and extent of effort in indexing the entities, and the relative simplicity and effectiveness of search. The greater the expertise and effort spent in indexing the data base content, the less the effort (other things equal) in search, assuming equivalent criteria for search effectiveness.

In the search process there is a tradeoff between making a search mesh so broad that there will be many false drops (items that turn out to be irrelevant) but does capture a large proportion of all relevant entities, as contrasted with a search mesh that is too fine and, although producing few false drops, has also filtered out a substantial proportion of relative items.

It should be evident that the design of an indexing and search technique is a bundle of compromises that hopefully are based on enlightened practicality.

In principle, search strategy produces a progressive winnowing out of the irrelevant. It is a convergence towards the desired subset through a process of exclusion. It is much like the skilled playing of the game called Twenty Questions where you find what you want by progressively excluding, logically, what you do not want. Unfortunately, this assertion cannot be taken too literally else we are back at reading every document in the library.

SUBJECT MATTER DESCRIPTORS. The following considerations were deliberately taken into account in developing the descriptor list. The descriptor list must serve both the indexer of the content of the data base and the searcher who will query the data base.

Closed Set of Descriptors for a Topic Domain. Recall that topic domain consist of courses and course characteristics—instructional vehicles and their characteristics, and job-tasks and their characteristics. Each domain has about 50 descriptors. A closed and relatively small set simplifies the computer operations of setting up records and fields and search algorithms that are both simple and efficient. All terms can be (and are) carefully defined for both indexer and searcher, thus their usage is relatively standardized. The potential loss of specificity and flexibility in entity description are offset by the ease with which the meanings of the descriptors in this limited set can be learned and their meanings understood and remembered, and their application standardized. Application of the present set of descriptors may show empirically the desirability of discarding, modifying or supplementing members in the list. These changes can be made while still retaining a closed set. By definition, a closed set is under control. We should remember that the topical subject matter level of search is not generally intended to be sufficient in itself to identify or reject target hits as relevant to the search objective.

Few Descriptors per Category. With some exceptions, a category of description contains fewer than eight descriptors. Several sets go to ten or a dozen. A set of this small size is readily learned and its members discriminated from each other. Because more than one descriptor in a category may be used in indexing a reference entity or in creating a query, there is little need to avoid redundancy or seek mutual exclusivity in the descriptor set. By avoiding such a requirement, indexing and querying are simplified perhaps by an order of magnitude. This simplification offsets the loss of precision for a given search.
iteration. Topical exhaustiveness for a collection of subject matter descriptors is perhaps more important than precision in this iteration of search. But even the exhaustiveness criterion is a hope that, in most searches, the category "Other" will get few hits. If not (and the searcher has AND'd the descriptor "Other" with a number of additional descriptors, thus limiting the target area), the searcher may have to use a different strategy.

If the descriptor "Other" in a set of descriptors is used too frequently, a revision of the descriptor list is indicated. A benchmark for "frequent" might be 10% of queries in which "Other" is the only descriptor selected in a category of descriptors.

Few Descriptors in an Entire Topical Domain. Fifty descriptors, subset into at least a half-dozen topical groups, are fairly readily learned, especially if they make sense as individual descriptors and in the way they are organized. The entire fifty fit rather readily on a single printed page or display, and can be shown all at once to the indexer or the searcher. Thus, the data entry format for composing a query may begin as a printed page with blanks, or as a display on a graphic terminal.

Avoidance of Descriptor Hierarchies. When they are used to compose a query, the descriptors are all at one logical level. It is well known that attempts to make and use formal hierarchical relationships among functional or topical concepts create procedural difficulties for the user, and put some serious constraints on computer search for at least some kinds of search content. In any event, the small number of descriptors per search domain makes hierarchic structure unnecessary either for the human (except as noted) and for the computer system. All of the descriptors can be treated as a single level list.

TAXONOMIC REFERENCE OF DESCRIPTORS. The source of the present sets of descriptors was not merely convenience for indexing. The source is an applied theory of learning--the essential processes, products and supports for the learning of useful tasks. Each term in each set has a somewhat different meaning for a learning process or outcome than other terms in the set. Each term also derives some of its meaning from its contextual association with the other terms in the clusters and sets. The indexer's task will be somewhat simplified if he recognizes the structure inherent in each set, rather than treating them as an arbitrary list created merely for indexing convenience. Their primary purpose, individually and collectively, is to apply standard names for concepts that generalize an instance of a set of properties, such as found in the innovation, to a some universe of entities that share those properties in a practical as well as theoretical sense.

For a more complete exposition of these ideas, see Task 5, Section III in the ETAM Phase II-A Final Report, as well as Appendix A to that report.

Any standardized terminology is bound to have examples with which it cannot readily cope. If the exceptions are rare--perhaps less than 5%--the standards are likely to be useful. The penalty is some proportion of extra human judgment for coping with these exceptions and perhaps putting them into a "miscellaneous"
file. If, however, the exceptions approach the frequency of items that readily fit the standards, something is wrong with the standards or with the way in which they are applied.

These comments are intended to apply to some who may feel compulsively driven to find a pre-established niche for 100% of the cases in some real world universe. There will be temptations to add terms to the descriptor set in order to fit perhaps small subsets of entities. The value of a closed set of descriptors can be lost if the set increases in size beyond those that can be kept readily in mind after a few hours of practice.

If changes or additions are contemplated—and indeed they should be from time-to-time—they should be reviewed in terms of benefits and liabilities to the entire logic of the classification structure. It is easy to anticipate many sources of pressure to make changes. Very possibly the entire taxonomic logic underlying the descriptor set justifies technical challenge. It would be far better from economic and other reasons to make sufficient tryouts so that the entire descriptor set could be strongly stabilized at the outset before general adoption in practice.

SEARCH ALTERNATIVES. The searcher will be advised to examine his search objectives and the information about the innovation he has as starting point, and choose his search method accordingly. He should be able to bypass any operations that interfere with his progress by demanding that he go to a higher level of description than he already has available for bounding his search region. (Search strategy will be discussed in a later topic.) In any event, the searcher should not be required to enter any level of inquiry that obfuscates his available definition of his search problem. Furthermore, search procedures should not demand his going through any process that does not make a contribution either in range or scope of potential relevance, or in narrowing the search universe towards his objective. He should be able to bypass non-contributory steps in search procedure. The ETAM search procedures enable this flexibility.

Thus, the searcher who wants to get the broadest topical definition of his inquiry should be guided by the ETAM search descriptors in doing so. The ETAM search will yield a broad set of candidate job-tasks, course titles, instructional devices. This first set of candidates, if reasonable in number, may be examined individually at the level of job-task names, or course titles, or titles of instructional vehicles. On the other hand, the searcher may wish to—and be able to—use descriptors in the respective Navy data bases for further subsetting the first list of candidates into a second derived list of still more probably relevant candidates. He may therefore use descriptor names appropriate to automatic search in those data bases, e.g., the Navy's Occupational Standards data base, NOTAP, CANTRAC, etc., permit, in varying degrees, this kind of search.

The searcher may elect to go directly to the Navy data bases for inquiry based on open descriptor sets applied to these respective data bases. Entity titles and terms within titles can be used to construct search arguments. Thus, many task titles in NOTAP include the name of the equipment, tool or instrument on which or with which the task is performed. After the searcher makes a culling pass through the titles of entities, printed out or displayed as a consequence of the query, the searcher may make another pass through the remaining candidates with the supplemental context for each entity presented to him. This would be
the final search pass for accepting or rejecting candidates that had survived
the progression of excluding filters imposed by successive search queries.

If the automatic search system is interactive and on-line, a number of successive
search passes can be made in perhaps a few minutes, so that the process is
neither as complicated nor as lengthy as the description of it.

It should be clear that the searcher is not forced to take any one search route
from start to finish. It should be noted that this kind of user flexibility is
somewhat unusual in query systems.

SEARCH PROCEDURES. The major factors that make formulating a query easy or
difficult are selecting descriptors, selecting logical connectives between
descriptors, and composing the content of the query for entry into the computer.
The descriptor issue has already been discussed. Because the total descriptor
lists are so relatively small for each domain there is no justification for
introducing the complexities of hierarchic structure. What is called the "coordi-
nate descriptor" method is being used for creating a query. The user merely
enters those descriptors that have relevance to the identification or positive
rejection of candidate entities in the data base. If a descriptor is not selected,
it is a "don't care" variable in the computer's search logic. Descriptors are
organized into meaningful clusters called "categories." The cluster title
communicates the context meaning to the user, thus simplifying his interpretation
of the meaning of the descriptors within the cluster. In itself, the category
title is merely a part of the format display; it does not enter into the search
argument unless it is also a numbered descriptor, because there are no descriptors
logically subordinate to it in the descriptor set.

The query format itself allows the procedural equivalent of a check list operation
in selecting descriptors. This format is desirable whether the user is a novice
or expert to the system. Because it is a standardized presentation to the
searcher, the descriptor content can rather quickly be learned by the user in
repeated usage.

Research (some of it conducted in IBM) has shown that, at least in programming,
the use of logical connectives such as AND, AND NOT, OR and EXCLUSIVE OR, is
fraught with human error, even among experts. The logical OR, especially in
combination with bracketed expressions, is particularly inviting to human mis-
derstanding and error. These experiences have been taken seriously in formu-
lating what orders of connective logic are permitted the user, and not offered
to the user, in making up a single query.

The AND logic is ordinarily simple for the user to get accustomed to, especially
if no compound statements are to be strung together. (The procedures do not
allow compound statements.) Thus, the searcher may want to get a listing of all
courses that provide Enabling Knowledge of "Procedural descriptions" AND in the
Gross Job category of "Maintenance." The user quickly learns that only those
items which are identified both as teaching procedural descriptions and that
teach in the context of maintenance jobs will be selected by the query. (In
preliminary training of the user, it would not be taken for granted that even
this concept is self-evident.) The AND operation does not require terms in an
expression to be logically grouped and sequentially ordered in any fashion.
The OR relationship has been arbitrarily allowed to apply to single descriptors, taken one at a time, within the same category. Descriptors within a category cannot logically be grouped by AND connectors and OR'd to another group of logically connected descriptors. The OR relationship can apply only between individual descriptors within a category. The selected content of one category cannot be OR'd with the content of another category. This would raise the potential logical complexity in a single query beyond what is felt to be a generally acceptable level.

The OR relationship within a category can be useful for several purposes. Assume that the searcher is not quite sure whether the task function he is concerned with is "Interpret" or "Decide." That is, he is not quite sure which of these terms was used to index the target material for which he is searching. To be safe, he can select both Interpret and Decide, which means that he will get entities that bear either one or the other term, or both. He can thus defer his decision on precise relevance until he sees further contextual information about the candidate items selected by his query. Note that this capability copes with some degree of ambiguity in descriptors, both as applied by indexers and by searchers.

The distinction between Inclusive OR and Exclusive OR was not considered operationally relevant to the ETAM application. The OR in ETAM procedures is an Inclusive OR—it does not care whether either one or the other, or both, the OR'd items describe the target entity.

If the searcher wants to widen his search universe more broadly than is permitted by the logical limitations in a single query format, he may use several queries, each of which is the equivalent to a complex OR to the other queries, and combine the results of the several queries. This avoids the convoluted thinking that is required for making a single, multilevel, search statement. If the cost of making even occasional logical errors is prorated across the additional time for making multiple queries, it is doubtful that any efficiency advantage would appear for the more complex procedures. "Logical elegance" in statement is not a relevant human factors criterion.

Finally, some comments are deserved about the ability to frame any query into the content of a single displayed page that is uncluttered with computerese jargon, and symbols that have no bearing on the user's objective—to formulate a search statement on the variables in which he is interested. It should be simple for the user to check over what he has entered, to know what he has done, and if he can recognize errors of omission or commission to correct them directly and at once; and if the response to the searcher's query appears to him like a tray full of nonsense, he can readily check back over the total content of the query he entered as a search argument and perhaps identify what was wrong with it. Thus, the data entry format and its contents is a useful documentation of the query both for diagnosing limitations in the search argument, or for purposefully modifying it.

The test of these rationales will come, of course, when samples of novice users are actually confronted with formats, terminals, and search problems. Assume familiarity with the meaning of the descriptors and with the purpose of the query operations. Within 15 to 60 minutes, the novice should be entering meaningful queries without assistance, and on the average make fewer than perhaps 5% errors in procedure that are not identified and corrected at once.
SEARCH STRATEGIES. The design of query structures, search systems and data bases should facilitate flexibility in the searcher's method of attack. That method may vary depending on the values in the search objective and on the information the searcher has at the beginning of the search. Both of these issues are joined in principles that embody search strategy. The definition of strategy here is to "optimize" an outcome for some given level of risk and effort, or conversely to minimize risk and effort in achieving some level of search outcome. These concepts should become clarified when some examples are examined of efficient search technique under various kinds of starting information.

The general strategic principle is to formulate queries at each stage of inquiry that eliminate the largest proportions of irrelevant entities from those entities that still remain to be searched. In practice, this means using descriptors that appear most rarely (as compared with other descriptors) in data base descriptions. In general, a unique identifier such as the "name" of a given entity is comparatively rare as an index term in a data base. An example in a job-task inventory would be the name of a piece of operational equipment such as "radar ABC-10." Note also that unlike entity attributes where meanings are inevitably ambiguous, a unique identifier is low in ambiguity. If the searcher knows that all of his ultimate target set of relevant entities will include "radar ABC-10" in their indexed descriptions, his first search query should include "radar ABC-10" as a search term. In this case, or one like it, the searcher would bypass the ETAM level of subject matter search and put his inquiry directly into the job-task inventory data base. To repeat, this assumes that entities in the job-task inventory are indexed by "equipment worked on" and that "radar ABC-10" is the descriptor, and the only descriptor, for that equipment, applied to relevant job-task entities in that data base. (It is assumed that if a descriptor is assigned to an entity in a data base, that automatic search can be made on that descriptor or category of descriptors.) Call a descriptor that is a specific name for an entity, or for a set of equivalent entities, by the term "nominal descriptor." A query can include a number of nominal entities that are logically "OR'd" to each other if the query format permits, else a sequence of queries may achieve the same end.

The use of nominal descriptors does not entirely rule out the missing of relevant items. It is frequently possible that in the operational setting, a given object is treated as a component in a larger object, and only the latter is named as the descriptor. Thus, "radar ABC-10" may sometimes be a component in "radar system XYZ," and, if so, only the latter is identified. This is why the rules used by the indexer should be explicit and communicated to the searcher. Under conditions of uncertainty, and where it is important to determine maximum range of relevant entities, the only recourse is by proceeding from the more inclusive categories into contextual search. Contextual search is made when the searcher manually examines the full description of each entity after automatic search on descriptors has been carried as far as it is allowed safely to go.

When on the other hand the searcher is not sure of what he wants, or lacks nominal descriptors for narrowing the search quickly, he is advised to go broad at the start. This is because if relevant items are winnowed out at any stage of progressive search, they will not be examined again during that particular search cycle. Again, the tradeoff is between the risk of missing relevant items versus amount of effort spent in the total search and identify operation. The selection of an improperly limiting attribute in an early query can lock out
a large proportion of eligible entities. In bibliographic search the searcher may not care about seeing the "complete" set of potentially relevant documents, only some subset which is representative and falling within the limits of what he can assimilate. But in range-of-effect analysis, the searcher is properly concerned with the fullest possible range of applicability of the innovation. He is concerned with the application "universe," not a sample from it.

But even with this objective the maximum exclusion principle operates. When the searcher adds descriptors linked by AND, he is superimposing filters on the search. A carefully chosen AND NOT descriptor can filter out large subsets of data base entities that are irrelevant to the inquiry.

An ideal inquiry system would enable the searcher to examine an output listing (whatever its level or form) and randomly sample from its contents at the level of contextual detail. By this means he could often determine what further restrictions to put into his next queries. Note that here again there is a difference between bibliographic searches and range-of-effect search. In the former, the searcher may ask for the number of titles identified by the request applied to the data base. If the number exceeds some arbitrary limit set by the searcher, he may tighten his inquiry in an attempt to restrict the number of hits to a number he feels able to manage. The ETAM assessor, on the other hand, is obliged to attend to all relevant entities.

Even when the searcher is quite confident at the start that he can subset the data base with a definitive set of descriptors, he may be well advised to begin with a broader net. The starting assumptions of the relevant characteristics and boundaries of target entities may be revealed as too narrow. Inspecting even a few samples of context about entities not thought to be applicable could change preconceptions and enlarge the field. This might be called "strategic serendipity." A handy--and pertinent--example is "diagnostic technique" or troubleshooting. An innovation may, at the start, be identified as a training method for strategic diagnosis of equipment ailments. The method may be equally applicable to the diagnosis of human ailments. It is possible that some context in one or both may reveal that "search strategy" of the kind we are discussing in the present topic is equally applicable when appropriately defined--the strategic selection of a series of tests (queries) where each test in the series logically excludes from further suspicion (or relevance for further inquiry) some major segment of the equipment (data base). The potential applicability, hence value, of the innovation becomes greatly expanded.

In conclusion, search for information can combine efficiency and effectiveness, but depend on a combination of expertise and enabling knowledge. The knowledge applies to the defining concepts of the descriptor set, the rules used for indexing the target subject matter, and something about the contexts of those subject matters. The expertise applies to strategies used in searching large data banks as well as in overcoming information deficiencies that may exist in the searcher himself. As in other tasks that use strategies, the expert may achieve several times the results of a novice at a fraction of the time.
GENERAL CONCLUSIONS. These rationales are offered as justification for the decisions made in the section and formatting of ETAM descriptors and ETAM query procedures where simplicity for the user was traded off against semantic precision and logical universality. The user is offered flexibility in search method by not being ruled from proceeding directly to data base content if the knowledge of target objectives enables this to be done. The basis for choosing an alternative was described in the context of search strategies.

The functions of the human component in the search operation have been made explicit so that even in the default of automatic modes for aiding search, the searcher could do the job manually with file cards. In any event, the final judgment of the relevance or irrelevance of target entities must depend on human judgment operating on contextual information about those entities. That context may be in a printout or in the searcher's head, or a combination of both.

The combinations of all of these factors makes up an information search "system." The structure proposed here is sufficiently modular and simple in its interfaces and content to be readily changed when actual usage suggests modifications. It is realistic to expect demand for changes, but it is important to determine whether the change should be in elements of method, information content, or the rationales behind them.
APPENDIX B
KEY FIGURES AND ILLUSTRATIONS REFERENCED IN THIS ETAM REPORT

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>TITLE</th>
<th>REFERENCE</th>
</tr>
</thead>
<tbody>
<tr>
<td>B-1</td>
<td>Summarized Preliminary Feasibility Profile</td>
<td>III-48</td>
</tr>
<tr>
<td>B-2</td>
<td>Cost/Savings Data Sheet</td>
<td>III-56</td>
</tr>
<tr>
<td>B-3</td>
<td>Decision Tree for Assessment of Training Innovation</td>
<td>III-60</td>
</tr>
<tr>
<td></td>
<td>(Example of Form Completed with Initial Outcome Values and Probabilities)</td>
<td></td>
</tr>
<tr>
<td>B-4</td>
<td>Decision Tree for Assessment of Training Innovation</td>
<td>III-61</td>
</tr>
<tr>
<td></td>
<td>(Example of Folding Back Process)</td>
<td></td>
</tr>
<tr>
<td>B-5</td>
<td>Format for Describing a Supplementary Benefit or Liability</td>
<td>III-138</td>
</tr>
</tbody>
</table>

* The pages referenced in this Appendix are from:
<table>
<thead>
<tr>
<th>SUBTASK</th>
<th>RISK DEGREE</th>
<th>UNACCEPTABLY HIGH RISK OR OF TRIVIAL IMPORTANCE</th>
<th>MODERATE RISK OR OF MODERATE IMPORTANCE</th>
<th>NO RISK OR OF SUBSTANTIAL IMPORTANCE</th>
<th>SUPPORTIVE OR OF HIGH OR CRUCIAL IMPORTANCE</th>
<th>COMMENTS, NATURE OF RISK, ETC.</th>
<th>NUMBER OF RISK REDUCTION PROJECTS REQ'D AND IDENTIFIED</th>
<th>ESTIMATED PROJECT COSTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>IMPORTANCE TO PROJECTED NAVY MISSION</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>02</td>
<td>ORGANIZATIONAL COMPATIBILITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>03</td>
<td>GOALS/POLICY COMPATIBILITY</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>04</td>
<td>STATE-OF-THE-ART</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>05</td>
<td>R&amp;D FUNDING REQUIREMENTS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>06</td>
<td>TECHNICAL SUPPORT REQUIREMENT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>07</td>
<td>ATTITUDBINAL ACCEPTANCE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>08</td>
<td>OVERALL PROFILE AND TOTAL PROJECTS AND DOLLARS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**FIGURE B-1. SUMMARIZED PRELIMINARY FEASIBILITY PROFILE**
<table>
<thead>
<tr>
<th>PROJECT</th>
<th>PROJECT NAME</th>
<th>PROJECT NO.</th>
<th>DATE</th>
</tr>
</thead>
</table>

**PERSONNEL**

<table>
<thead>
<tr>
<th>JOB</th>
<th>MAN MONTHS</th>
<th>COST/MAN MONTH</th>
<th>JOB COST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**EQUIPMENT**

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>QUANTITY</th>
<th>UNIT COST</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**SPACE**

<table>
<thead>
<tr>
<th>REASON</th>
<th>SQUARE FEET</th>
<th>COST/SQ. FT.</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TRAVEL (NON-STUDENT)**

<table>
<thead>
<tr>
<th>FROM - TO - REASON</th>
<th>NO. OF TRIPS</th>
<th>COST/TRIP</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**STUDENTS**

<table>
<thead>
<tr>
<th>NO. OF STUDENTS - REASON</th>
<th>NO. OF DAYS</th>
<th>COST/STUDENT DAY</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**OTHER**

<table>
<thead>
<tr>
<th>DESCRIPTION</th>
<th>MAN MONTHS/QT.</th>
<th>UNIT COST</th>
<th>COST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**TOTAL PROJECT COSTS**

$
FIGURE B–3. DECISION TREE FOR ASSESSMENT OF TRAINING INNOVATION
(EXAMPLE OF FORM COMPLETED WITH INITIAL OUTCOME VALUES AND PROBABILITIES)
FIGURE B-4. DECISION TREE FOR ASSESSMENT OF TRAINING INNOVATION (EXAMPLE OF FOLDING BACK PROCESS)
TASK: DETERMINE RANGE-OF-EFFECT

1. Identify the name of the variable or factor which is a supplemental benefit or a liability.

2. Estimate the magnitude of effect of the benefit/liability.
   a. If possible, quantify the unit of magnitude, and the multiplier factor, to apply to the unit of magnitude appropriate to the intended scope of the application of the innovation.
   b. If numerical quantification is impractical, express magnitudes in relative terms.

3. What is the number of entities or instances (jobs, courses, students, etc.) to which this magnitude of benefit (or liability) applies?

4. Estimate the relative importance of the added benefit (or liability) to the overall effectiveness of the innovation.
   Use a scale from 1 to 10.

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>trivial importance</td>
<td>moderate importance</td>
<td>crucial</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>identifiable but minor</td>
<td>very substantial</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Any additional statements that may add or qualify information for the decision maker may be added informally.

5. Estimate the range across which the magnitude of benefit (or liability) may occur.
   a. The estimate of magnitude made in item 2 presumably assumed conditions moderately favorable to the benefit occurring (or moderately unfavorable for the liability to occur) at that magnitude.
   b. Estimate the magnitude of the benefit under conditions highly favorable for the innovation. (The converse for a liability.)
   c. Estimate the magnitude of the benefit under conditions highly adverse to the innovation.

6. State reasons and justifications for these expectations.

FIGURE B-5. FORMAT FOR DESCRIBING A SUPPLEMENTARY BENEFIT OR LIABILITY
APPENDIX C

TRAINING COST MODEL PROGRAM
**PROGRAM** - TRAINING COST MODEL

**TECEP COST MODEL ADAPTATIONS FOR ETAM BY L. DUFFY**

- THIS MODULE MUST BE MODIFIED WITH NEW READ AND FORMAT STATEMENTS TO RECEIVE DATA FROM THE ORIGINAL AND MODIFIED EXC FILE FROM THE ETAM PROJECT DATA BASE.
- THE MODULE PRESENTLY CYCLES AND DEVELOPS A MATRIX OF INVESTMENT AND RECURRING COST/SAVINGS DATA FOR EACH COURSE RUN UNDER BASELINE CONDITIONS AND THEN WITH THE BENEFIT PATTERN APPLIED.

**DATE** - 3/4/77

**PROGRAM SETUP**

```plaintext
REAL INTSPO, IMODEV, MSCSPU, MSCSTD, IMDMT, INSTSL, INFLAT, INRATE
REAL NSTUD, NTRAV, NINSTR, NADMIN, NFACIL, NSUPPL, NMISC, NEQUIP, NIMD
REAL NSTUD, NTRAV, NINSTR, NADMIN, NFACIL, ISUPPL, IMSIC, IEQUIP, IIMD
INTEGER GRAD, PURCHF, AEOUP, DEPF
INTEGER DIGIT(20) /1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20/

DIMENSION RPT1, RPT2, RPT3, RPT4, RPT5, RPT6, RPT7, LAST
DIMENSION STUD(20), GRA(20), STUDMW(20), SMWRC(20), ADB(20), PSP(20)
DIMENSION RINSTR(20), RADMIN(20), REQUIP(20), RFACIL(20), IMDMT(20)
DIMENSION RSUPPL(20), RMISC(20), CSTUD(20), CTRAV(20), CINSTR(20)
DIMENSION CADMIN(20), CEOUIP(20), PEQUIP(20), PDEP(20), AEQUIP(20)
DIMENSION COPMT(20), CFACIL(20), CIMDRT(20), CSUPPL(20)
DIMENSION CMISC(20), DISCNT(20), INFLAT(20), CGRAD(20), CSTIN(20)
DIMENSION CSP(20), RATA(100), RABA(100), COPTOT(20), MGRA(20)
DIMENSION ADBX(20)
DIMENSION ETA0(20), ETAREC(20), ETATRAC(20), ETAMIN(20), ETAMIN(20), ETA0(20)
DIMENSION ETA0(20), ETA0(20), ETA0(20), ETA0(20), ETA0(20), ETA0(20), ETA0(20)
DIMENSION ETA0(20), ETA0(20), ETA0(20), ETA0(20), ETA0(20), ETA0(20), ETA0(20)
DIMENSION ETA0(20), ETA0(20), ETA0(20), ETA0(20), ETA0(20), ETA0(20), ETA0(20)
DIMENSION ETAMIN(20), ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20)
DIMENSION ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20)
DIMENSION ETAMN(20), ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20), ETAM0(20)
DIMENSION CMN0, NOSOF, NOGOF, NOSOF, NOSOF, NOSOF, NOSOF

**ETAM DESIGN MOD - REMOVE 'NAMELIST' STATEMENTS AND SUBSTITUTE 'FORMAT' STATEMENTS TO READ FROM EXC FILE, ONCE UNDER BASELINE CONDITIONS, ANOTHER WITH THE BENEFIT PATTERN APPLIED TO INPUT VARIABLES.**

**NAMELIST/TECHF/ COURSE, N, RATE, TRENCH, RCRATE, ACKYT, MWSCHP, TSPO, INTSPO, AMTSPO, EQTSPO, PURCHF, DEPF, LOFES, SOFTI, SOFTI, SOFTI, SOFTI, IMODEV
2, PUIMD, UPDTE, SUPBPO, SUPSTD, MSCSPU, MSCSTD, PESP, TLENGT, LOFFA, WSM01 NAMELI0/COST/STDSS, STCSLT, STCSTZ, INSTSL, ADMSL, EQUTC, PCTPCH, CPS 1QF, CLOM, EVM, CUSUP, CMISC, OETR, INRATE, FASC
NAMELI0/DIMF/GRAD, COPMT**

**PROGRAM INITIALIZATION**

**ETAM DESIGN MOD - ETAM ENTRY POINT FOR START OF LOOP ONE.**

**INITIALIZE LOOP INDICATOR (IND1).**

IND1 = 0

**DATA COURSE /"NONE"/
TPEQUP = 0.0
AAEQUP = 0.0
PUCBCH = 99999
DEPH = 99999
HPVD = 0.0
HVPFA = 0.0
NSTUD = 0.0
 PSTUD = 0.0
FSTUD = 0.0
MTRAV = 0.0
DTRAV = 0.0
FTRAV = 0.0
NINSTR = 0.0
VINSTR = 0.0
FINSTR = 0.0
NADMIN = 0.0
DADMIN = 0.0
FADMIN = 0.0
NFACIL = 0.0
DFACIL = 0.0
FFACIL = 0.0
NSUPPL = 0.0
DSUPPL = 0.0
FSUPPL = 0.0
NMISC = 0.0
DMISC = 0.0
FMISC = 0.0
NEQUP = 0.0
OEQUP = 0.0
TEQUP = 0.0
VIMO = 0.0
OIMO = 0.0
IMO = 0.0
GIMO = 0.0
DAOB = 0.0
SAUB = 0.0
NDGO = 0.0
JPS = 0.0
JAOB = 0.0
ABATE = 99999
ACRATE = 99999
ASCYTM = 99999
WSCHP = 99999
NSCHP = 99999
TSPSOD = 99999
PESP = 99999
INTSPO = 99999
AMTSP = 99999
EMTSP = 99999
JFEQ = 99999
SQFTST = 99999
SQFTIN = 99999
SQFITM = 99999
IMODEV = 99999
PUIMD = 99999
UPDATE = 99999
SJPSP = 99999
SUPSTD = 99999
MSCPD = 99999
MSCSTD = 99999
STUDSL = 99999
STCSTI = 99999
STCST2 = 99999
INSTSL = 99999
ADMSL = 99999
EQUITC = 99999
PCTPCH = 99999
CPQFT = 99999
CIMD = 99999.
READ INPUT DATA
*** ETAM DESIGN MOD - CHANGE 'READ' STATEMENTS TO READ FROM ORIGINAL
AND MODIFIED (WITH BENEFIT PATTERN) EXC FILE.
*** ETAM DESIGN MOD - A LOGICAL '1' IN ITS RESPECTIVE POSITION
SELECTS A REPORT. THESE SHOULD BE SELECTABLE
INTERACTIVELY. THE REPORT NAMES ARE:
RPT1 - INPUT TECHNICAL FACTORS
RPT2 - INPUT COST FACTORS
RPT3 - OUTPUT TECHNICAL RESULTS (TOTAL)
RPT4 - OUTPUT TECHNICAL RESULTS (ANNUAL)
RPT5 - OUTPUT COST RESULTS (TOTAL)
RPT6 - OUTPUT COST RESULTS (ANNUAL)
RPT7 - MISSING DATA ANALYSIS
LAST - INDICATES LAST INPUT STREAM DATA CASE
READ(5,6000) RPT1,RPT2,RPT3,RPT4,RPT5,RPT6,RPT7,LAST
READ MISSING DATA ANALYSIS AND DEFAULT ROUTINES

ETAM DESIGN MOD - EACH INPUT VARIABLE IS CHECKED FOR PRESENCE AND REASONABILITY. MISSING OR OUT-OF-RANGE DATA ARE RECORDED BY STATEMENT NUMBER BY THE 'MISSING SUBROUTINE'. VARIABLES WITH REASONABLE DEFAULT SUBSTITUTES HAVE A CRITICALITY CODE=1, VARIABLES WITH NO DEFAULT DATA HAVE CRIT CODE=0. FOLLOWING IS A LIST OF VARIABLES BY STATEMENT NUMBER INCLUDING CRIT CODE AND PRESENT DEFAULT VALUE.

<table>
<thead>
<tr>
<th>NO.</th>
<th>VAR NAME</th>
<th>DESCRIPTION</th>
<th>DEFAULT</th>
<th>CRIT</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>N</td>
<td>PLANNING PERIOD</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>40</td>
<td>ARATE</td>
<td>ATTENTION RATE</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>50</td>
<td>TLENGTH</td>
<td>TRAINING LENGTH (WKS)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>55</td>
<td>TLENGTH</td>
<td>HRS/WK IN MEDIUM</td>
<td>40.0</td>
<td>1</td>
</tr>
<tr>
<td>60</td>
<td>RCRATE</td>
<td>RECYCLE RATE</td>
<td>5.0ARATE</td>
<td>1</td>
</tr>
<tr>
<td>90</td>
<td>WSCHOP</td>
<td>WKS/YR SCHOOL OPERATIONALISTER</td>
<td>50.0</td>
<td>1</td>
</tr>
<tr>
<td>95</td>
<td>WSCHOP1</td>
<td>WKS/YR</td>
<td>50.0</td>
<td>1</td>
</tr>
<tr>
<td>100</td>
<td>TSPSD</td>
<td>TIME STUD POS DOWN</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>110</td>
<td>PESP</td>
<td>EXCESS STUD POSITIONS</td>
<td>0.10</td>
<td>1</td>
</tr>
<tr>
<td>120</td>
<td>INTSPO</td>
<td>INST/STUD POS RATIO</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>130</td>
<td>AMTSPO</td>
<td>ADMIN/STUD</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>140</td>
<td>EQTSPD</td>
<td>EQUIP/PURCHASE POLICY</td>
<td>1.0</td>
<td>1</td>
</tr>
<tr>
<td>150</td>
<td>DEPF</td>
<td>EQUIP DEPRECIATION</td>
<td>1.0</td>
<td>1</td>
</tr>
<tr>
<td>175</td>
<td>LDDFA</td>
<td>LIFE OF FACILITY (YRS)</td>
<td>N</td>
<td>1</td>
</tr>
<tr>
<td>180</td>
<td>LDFED</td>
<td>LIFE OF EQUIPMENT (CONDITIONAL) 0/1</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>200</td>
<td>SQFTST</td>
<td>SQ FT/STUDENT</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>210</td>
<td>SQFTIN</td>
<td>SQ FT/INSTRUCTOR</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>220</td>
<td>SQFTAM</td>
<td>SQ FT/ADMINISTRATIVE</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>230</td>
<td>IMDEV</td>
<td>INSTR MAT DEV RATIO</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>235</td>
<td>IMDEV1</td>
<td>INSTR MAT MAINT RATIO</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>290</td>
<td>SUPSPD</td>
<td>SUPPLIES/STUD POS</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>300</td>
<td>SUPSPD</td>
<td>SUPPLIES/STUDIAN</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>310</td>
<td>MSCSPD</td>
<td>MISC/STUDENT</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>320</td>
<td>MISCSTD</td>
<td>MISC/STUDENT</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>330</td>
<td>STUDSL</td>
<td>STUDENT SALARY</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>340</td>
<td>STCST1</td>
<td>TRAVEL TO/FROM SCHOOL</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>350</td>
<td>STCST2</td>
<td>TRAVEL DURING SCHOOL</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>360</td>
<td>INSTSL</td>
<td>INSTRUCTOR SALARY</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>370</td>
<td>ADMSTL</td>
<td>ADMINISTRATIVE SALARY</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>380</td>
<td>EQUNTC</td>
<td>EQUIP UNIT COST</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>400</td>
<td>PCTPCH</td>
<td>PCT MAX EQUIP PURCH</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>410</td>
<td>CPSQFT</td>
<td>COST SQ FT/INSTR MAT DEC</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>420</td>
<td>CIMD</td>
<td>COST/HR INSTR MAT DEC</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>430</td>
<td>EVIM</td>
<td>ANNUAL MAINTENANCE PCT</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>440</td>
<td>CSUPP</td>
<td>COST OF SUPPLIES</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>450</td>
<td>CHIS</td>
<td>COST OF MISCELLANEOUS</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>460</td>
<td>DRATE</td>
<td>DISCOUNT RATE</td>
<td>0.10</td>
<td>1</td>
</tr>
<tr>
<td>470</td>
<td>INRATE</td>
<td>INFLATION RATE</td>
<td>0.08</td>
<td>1</td>
</tr>
<tr>
<td>475</td>
<td>FACST</td>
<td>FACIL ACQ/REFURB COST</td>
<td>0.0</td>
<td>1</td>
</tr>
<tr>
<td>480</td>
<td>GRAD</td>
<td>GRADUATES EACH YEAR</td>
<td>-0</td>
<td>1</td>
</tr>
<tr>
<td>490</td>
<td>COPTM</td>
<td>O&amp;M COSTS EACH YEAR</td>
<td>0.0</td>
<td>1</td>
</tr>
</tbody>
</table>

30 IF (N.GT.0.AND.N.LE.20) GO TO 40
40 IF (ARATE.GE.0.0.AND.ARATE.LT.1.) GO TO 50
50 IF (TLENGTH.GT.0.0.AND.TLENGTH.LE.200.) GO TO 55
55 IF (TLENGTH.GT.0.0.AND.TLENGTH.LE.121.) GO TO 60
60 IF (RCRATE.GE.0.0.AND.RCRATE.LE.1.) GO TO 70
70 IF (ARCYTMGE.0.0.AND.ARCYTM.LE.200.) GO TO 90
   IF (RCREQ.EQ.0.0.) GO TO 80
90 IF (TLENGTH.LT.2.) GO TO 80
   CALL MISSNG (70,0)  
   GO TO 90
ARCTYMT = 0.0
CALL MISSNG (70, 1)
90 IF (WSCHOP. GT. 0.0 .AND. WSCHOP. LE. 52.0) GO TO 95
WHUPL = 50.0
CALL MISSNG (90, 1)
95 IF (WSHUP1. GT. 0.0 .AND. WSHUP1. LE. 150.0) GO TO 100
WHUPL = 50.0
CALL MISSNG (95, 1)
100 IF (TSPOSU. GE. 0.0 .AND. TSPOSU. LT. 1.0) GO TO 110
TPSU = 0.0
CALL MISSNG (100, 1)
110 IF (PESP. GE. 0.0 .AND. PESP. LE. 1.0) GO TO 120
PESP = 0.1
CALL MISSNG (110, 1)
120 IF (INTSPO. GE. 0.0 .AND. INTSPO. LE. 1.0) GO TO 130
INTSPO = 0.0
CALL MISSNG (120, 1)
130 IF (AMTSPO. GE. 0.0 .AND. AMTSPO. LE. 1.0) GO TO 140
AMTSPO = 0.0
CALL MISSNG (130, 1)
140 IF (ETQTSPO. GE. 0.0 .AND. ETQTSPO. LE. 1.0) GO TO 160
IF (EQUNTC. NE. 99999.0 .AND. EQUNTC. NE. 0.0) GO TO 150
ETQTSPO = 0.0
CALL MISSNG (140, 1)
GO TO 160
150 CALL MISSNG (140, 0)
160 IF (PURCHF. GE. 1.0 .AND. PURCHF. LT. 1.0) GO TO 170
PURCHF = 1
CALL MISSNG (160, 0)
170 IF (DEPF. GE. 1.0 .AND. DEPF. LT. 1.0) GO TO 175
DEPF = 1
CALL MISSNG (170, 1)
175 IF (LOFFA. GE. 1.0 .AND. LOFFA. LE. 50.0) GO TO 180
LOFFA = N
CALL MISSNG (175, 1)
180 IF (LOFEQ. GE. 1.0 .AND. LOFEQ. LE. 50.0) GO TO 200
IF (ETQTSPO. GE. 0.0 .AND. ETQTSPO. LT. 1.0) GO TO 190
LOFEQ = N
CALL MISSNG (180, 1)
GO TO 200
190 CALL MISSNG (180, 0)
200 IF (SQFTST. GE. 0.0 .AND. SQFTST. LT. 1000.0) GO TO 210
SQFTST = 0.0
CALL MISSNG (200, 1)
210 IF (SQFTIN. GE. 0.0 .AND. SQFTIN. LT. 1000.0) GO TO 220
SQFTIN = 0.0
CALL MISSNG (210, 1)
220 IF (SQFTAM. GE. 0.0 .AND. SQFTAM. LT. 1000.0) GO TO 230
SQFTAM = 0.0
CALL MISSNG (220, 1)
230 IF (IMDDEV. GE. 0.0 .AND. IMDDEV. LT. 1000.0) GO TO 250
IF (PUIMD. GT. 0.0 .AND. PUIMD. LE. 1.0) GO TO 240
IF (UPDATE. GT. 0.0 .AND. UPDATE. LE. 1.0) GO TO 240
IMDDEV = 0.0
CALL MISSNG (230, 1)
GO TO 250
240 CALL MISSNG (230, 0)
250 IF (PUIMD. GT. 0.0 .AND. PUIMD. LE. 1.0) GO TO 270
IF (UPDATE. GT. 0.0 .AND. UPDATE. LE. 1.0) GO TO 260
PUIMD = 1.0
CALL MISSNG (250, 1)
GO TO 270
260 CALL MISSNG (250, 0)
270 IF (UPDATE. GT. 0.0 .AND. UPDATE. LE. 1.0) GO TO 290
IF (IMDDEV. NE. 0.0) GO TO 280
IF (PUIMD. GT. 0.0 .AND. PUIMD. LT. 1.0) GO TO 280
UPDATE = 0.0
CALL MISSNG (270, 1)
GO TO 290
280 CALL MISSNG (270, 0)
290 IF (SUPSPD. GE. 0.0 .AND. SUPSPD. LT. 10000.0) GO TO 300
SUPSPD = 0.0
CALL MISSNG (290, 1)
300 IF (SUPSTD. GE. 0.0 .AND. SUPSTD. LT. 1000.0) GO TO 310
SUPSTD = 0.0
CALL MISSNG (300,1)
310 IF (MSCSPQ.GE.0. AND. MSCSPQ.LT.1000.) GO TO 320
MSCSPQ = 0.0
CALL MISSNG (310,1)
320 IF (MSCSTD.GE.0. AND. MSCSTD.LT.1000.) GO TO 330
MSCSTD = 0.0
CALL MISSNG (320,1)
330 IF (STUDSL.GE.0. AND. STUDSL.LT.50000.) GO TO 340
STUDSL = 0.0
CALL MISSNG (330,1)
340 IF (STCST1.GE.0. AND. STCST1.LT.10000.) GO TO 350
STCST1 = 0.0
CALL MISSNG (340,1)
350 IF (STCST2.GE.0. AND. STCST2.LT.10000.) GO TO 360
STCST2 = 0.0
CALL MISSNG (350,1)
360 IF (INSTSL.GE.0. AND. INSTSL.LT.50000.) GO TO 370
INSTSL = 0.0
CALL MISSNG (360,1)
370 IF (ADMSL.GE.0. AND. ADMSL.LT.50000.) GO TO 380
ADMSL = 0.0
CALL MISSNG (370,1)
380 IF (EQUNTC.GE.0. AND. EQUNTC.NE.99999.) GO TO 400
IF (EQTSPO.NE.0.) GO TO 390
EQUNTC = 0.0
CALL MISSNG (380,1)
390 CALL MISSNG (380,1)
GOTO 400
400 IF (PCTPCH.GE.0. AND. PCTPCH.LT.1.) GO TO 410
PCTPCH = 0.0
CALL MISSNG (400,1)
410 IF (CPSQFT.GE.0. AND. CPSQFT.LT.99999.) GO TO 420
CPSQFT = 0.0
CALL MISSNG (410,1)
420 IF (CIMD.GE.0. AND. CIMD.LT.50000.) GO TO 430
CIMD = 0.0
CALL MISSNG (420,1)
430 IF (EVIM.GE.0. AND. EVIM.LT.1.) GO TO 440
EVIM = 0.0
CALL MISSNG (430,1)
440 IF (CSUPP.GE.0. AND. CSUPP.LT.99999.) GO TO 450
CSUPP = 0.0
CALL MISSNG (440,1)
450 IF (CMIS.GE.0. AND. CMIS.LT.99999.) GO TO 460
CMIS = 0.0
CALL MISSNG (450,1)
460 IF (ORATE.GE.0. AND. ORATE.LT.1.) GO TO 470
ORATE = 1.0
CALL MISSNG (460,1)
470 IF (INRATE.GE.0. AND. INRATE.LT.1.) GO TO 475
INRATE = 1.08
CALL MISSNG (470,1)
475 IF (FACST.GE.0. AND. FACST.LT.99999.) GO TO 480
FACST = 0.0
CALL MISSNG (475,1)
480 IF (GRAD(N).GE.0. AND. GRAD(N).LT.99999.) GO TO 490
CALL MISSNG (480,0)
490 IF (COPMT(N).GE.0. AND. COPMT(N).LT.99999.) GO TO 510
DO 500 I=1,N
500 COPMT(I) = 0.0
CALL MISSNG (490,1)
C ***
510 IF (NOGO.EQ.1) GO TO 960
IF (N.GE.20) GO TO 518
INDEX5 = N+1
DO 515 I=INDEX5,20
COPMT(I) = 0.0
515 CONTINUE
C ***
MODEL EQUATIONS - TECHNICAL FACTORS
UIMDTM = TLENGH*IMDDEV*PUIMD*TLEGTH
GUIMD = UIMDTM*CIMD
RVIM = GUIMD*EVIM
IF (LOFFA.LE.N) GO TO 519
RFVA = (LOFFA-N)*(FACST/LOFFA)
519 DO 520 I=1,N
C-8
COURSE TECHNICAL FACTORS TO ACCUMULATE FOR ETAM CMR FILE

ETAGRD(I) = GRAD(I)/(1.0-ARATE)
ETASTU(I) = STUD(I)
ETAAOB(I) = AOB(I)
ETAINS(I) = INSTR(I)
ETAADM(I) = ADMIN(I)
ETAEQUP(I) =EQUP(I)
ETAFAC(I) = FACIL(I)
ETASUP(I) = SUPPL(I)

520 ETAMIS(I) = RMISC(I)
IF (IAOB.LT.0.) IAOB=IAOB*(-1.)
IAOB = (IAOB/SAOB)+.5)*100
IF (IAOB,GT,5.AND.,JAOB,NE,0) CALL MISSNG (520,IAOB)

THIS ROUTINE CALCULATES EQUIPMENT PURCHASE AND DEPRECIATION SCHEDULES FROM USER SELECTED POLICY OPTIONS

- PURCHASE POLICIES ANALYSIS

GO TO (530,540,550,560,570,580,590,600,610).PURCHF

POLICY 1 - PURCHASE TOTAL AVERAGE RQMTS IN FIRST YEAR OF EQUIP LIFE

530 CONTINUE
DO 534 I=1,N,LOFEQ
TEMPI = 0.0
INDEX1 = LOFEQ+I-1
DO 532 J=INDEX1,INDEX1+1
TEMPJ = REQUIPJ+TEMPJ
534 CONTINUE
GO TO 620

POLICY 2 - PURCHASE TOTAL PERCENT OF MAXIMUM RQMTS IN FIRST YEAR OF EQUIP LIFE

540 CONTINUE
DO 554 I=1,N,LOFEQ
TEMPI = 0.0
INDEX1 = LOFEQ+I-1

DO 542 J=1,INDEX1
IF (TEMP1.LT.REQUIP(J)) TEMP1=REQUIP(J)
542 CONTINUE
PEQUIP(I)= TEMP1*PCTPCH
544 CONTINUE
GO TO 620

POLLICY 3 - PURCHASE UP TO AVERAGE RQMTS BUT NOT EXCEEDING RQMTS IN YR OF PURCH

POLLICY 4 - PURCHASE UP TO PCT OF MAX RQMTS BUT NOT EXCEEDING RQMTS IN YR OF PURCH

- DEPRECIATION POLICIES ANALYSIS

POLICY 1 - DEPRECIATE LINEARLY OVER LIFE OF EQUIPMENT

POLICY 2 - DEPRECIATE BY SUM-OF-DIGITS OVER LIFE OF EQUIPMENT

C-10
GO TO 730

POLICY 3 - SET DEPRECIATION SCHEDULE TO FULLY DEPRECIATED ALL YEARS

CONTINUE
GO TO 730
CONTINUE
GO TO 630
CONTINUE
GO TO 630
CONTINUE
GO TO 630
GO TO 630
GO TO 630
GO TO 730
GO TO 730
GO TO 630
GO TO 630

- COST FACTORS

730 L = N + 1
IF (L.GT.20) GO TO 750
GO TO 740
L = 20
GO TO 740
GRADE = 0
GO TO 740
CONTINUE
TOTAEO = 0.0
GO TO 740
L = 1, N
CONTINUE
TOTAEO = TOTAEO + PEQUIP(I)
GO TO 740
CONTINUE
TAEQUIP = TAEQUIP + AEQUIP(I)
CONTINUE
LOBSLT = 1 - LOFEQ
IF (LOBSLT.LT.0.0) GO TO 770
CONTINUE
TOTAEO = TOTAEO - PEQUIP(LOBSLT)
CONTINUE
AEQUIP(I) = TOTAEO
CONTINUE
CSUPPL(I) = RSUPPL(I)*CSUPP
CONTINUE
CMISC(I) = RMISC(I)*CMIS
CONTINUE
CINVI = CEQUIP(I)
CONTINUE
CREC1 = AEQUIP(I)*COPMT(I)
CONTINUE
CREC2 = RFACIL(I)*CPSTOF
CONTINUE
CREC3 = IMDMT(I)*CMMD
CONTINUE
CREC = CREC1 + CREC2 + CREC3 + CSTUD(I) + CTRAV(I) + CINSTR(I) +
CONTINUE
CADM(I) + CSUPPL(I) + CMISC(I)
CONTINUE
COURSE COST FACTORS TO ACCUMULATE FOR ETAM CMR FILE

- FIRST YEAR

IF (I.NE.1) GO TO 771
CONTINUE
ETAINV(I) = CINVI + FACST + CMMD
CONTINUE
CFACIL(I) = CREC2 + FACST
CONTINUE
CIMDNT(I) = CREC3 + CMMD
CONTINUE
771 IF (I.EQ.1) GO TO 772
CONTINUE
ETAINV(I) = CINVI
CONTINUE
CFACIL(I) = CREC3
CONTINUE
CIMDNT(I) = CREC3
CONTINUE
772 ETAREC(I) = CREC3
CONTINUE
CEQUIP(I) = CINVI + CREC1
CONTINUE
IF (I.NE.N) GO TO 773
CONTINUE
ETAINV(I) = ETAINV(N) - RVEQ - RFVA - RVIM
CONTINUE
DISCNT(I) = (2.0*DRATE)/(2.0*(1.0+DRATE)**I)

C-11
TAEG REPORT NO. 40

INFLAT(I) = (2.0*(1.0+INRATE)**I)/(2.0+INRATE)
NSTUD = NSTUD+CSTUD(I)

DISCNT = DSTUD+CSTUD(I)*DISCNT(I)

ISTUD = ISTUD+CSTUD(I)*INFLAT(I)

FSTUD = FSTUD+CSTUD(I)*INFLAT(I)*DISCNT(I)

NTRAV = NTRAV+CTRAV(I)

DTRAV = DTRAV+CTRAV(I)*DISCNT(I)

ITRAV = ITRAV+CTRAV(I)*INFLAT(I)

FTRA = FTRA+CTRAV(I)*INFLAT(I)*DISCNT(I)

NINSTR = NINSTR+CINSTR(I)

FINSTR = FINSTR+CINSTR(I)*INFLAT(I)

FINSTR = FINSTR+CINSTR(I)*INFLAT(I)*DISCNT(I)

FADMIN = FADMIN+CADMIN(I)

FAADMIN = FADMIN+CADMIN(I)*INFLAT(I)*DISCNT(I)

NSUPPL = NSUPPL+CSUPPL(I)

USUPPL = USUPPL+CSUPPL(I)*INFLAT(I)

FSUPPL = FSUPPL+CSUPPL(I)*INFLAT(I)*DISCNT(I)

FMISC = FMISC+CMISC(I)

DMISC = DMISC+CMISC(I)*DISCNT(I)

NMISC = NMISC+CMISC(I)*INFLAT(I)

IF (1,NE,N) GO TO 780
CEQUIP(I) = CEQUIP(I)-REVO
CIM0MT = CIM0MT-I-RRIM

IF (1,NE,N) GO TO 780

CFACIL(I) = CFACIL(I)-RFVA

DEQUIP = DEQUIP-CEQUIP(I)*DISCNT(I)

IEQUIP = IEQUIP-CEQUIP(I)*INFLAT(I)

FEOQUIP = FEOQUIP-CEQUIP(I)*INFLAT(I)*DISCNT(I)

NIM0 = NIM0+CIM0MT(I)

DIM0 = DIM0+CIM0MT(I)*DISCNT(I)

DIM0 = DIM0+CIM0MT(I)*INFLAT(I)

CFACIL = CFACIL+CFACIL(I)

DFACIL = DFACIL+CFACIL(I)*DISCNT(I)

IFACIL = IFACIL+CFACIL(I)*INFLAT(I)

FFACIL = FFACIL+CFACIL(I)*INFLAT(I)*DISCNT(I)

COPTOT(I) = CSTUD(I)+CTRAV(I)*CINSTR(I)+CADMIN(I)+CFACIL(I)

1

CSGRAD(I) = COPTOT(I)-GRAD(I)

CSGRAD(I) = CSGRAD(I)-STUD(I)

CSGRAD(I) = CSGRAD(I)-TFACIL(I)

APEQUIP = APEQUIP-TPEQUIP/N

AAEQUIP = AAEQUIP-TAEQUIP/N

SNAOAC = NSTUD+NINSTR+NADMIN+NFACIL+NSUPPL+NMISC+NEQUIP+NIM0

SDOAC = DSTUD+DTRAV+DINSTR+DADMIN+DFACIL+DSSUPPL+DMISC+DEQUIP+DIM0

SIM = SIM+ITRAV+ITRIM+ISADMIN+IFACIL+ISUPPL+IMISC+IEQUIP+IM0

SFACIL = FSUPPL+FTRA+FINSTR+FADMIN+FFACIL+FSUPPL+FMISC+FEQUIP+FIM0

GNAOAC = SNAOAC/GRAD

G2OAC = SDOAC/GRAD

G1AOAC = SIAOAC/GRAD

GFAOAC = SFAOAC/GRAD

VUAOAC = SNAOAC/AASIN*N)

UOAOAC = SDOAC/AASIN*N)

UIAOAC = SIAOAC/AASIN*N)

UFADOAC = SFAOAC/AASIN*N)

PN2OAC = SNAOAC/IPSP

PD2OAC = SDOAC/IPSP

P1AOAC = SIAOAC/IPSP

PA2OAC = SFAOAC/IPSP

DRATE = DRATE*100

INRATE = INRATE*100.

WRITE INPUT TECHNICAL FACTORS

900 IF (.NOT.RPT) GO TO 910
WRITE (6,1000) COURSE,DRATE,N,INRATE,(GRAD(I),I=1,10),(GRAD(I),I=1,10),ARATE,TSPOS,TLENGTH,PESG,RCRATE,
1
1000 FORMAT (*1.28X,COURSE COST ANALYSIS'/*:IX,'INPUT':/':3X,'COURSE NUMBER':1,X,'A4.25X,'DISCOUNT RATE':1,'F4.1,' 'PERCENT':1,'X,'PLANNING PERIOD':1,'I2.27X,' 'INFLATION RATE':1,'F4.1,' 'PERCENT':1,'X,' 'INPUT TECHNICAL FACTORS':/':3X,'PLNG YR':',1',2,'C-12
TAEG REPORT NO. 40

1003 FORMAT

1. 010, F3.9X, RATE

2. C-13

WRITE (6,1010) ARCTYM, TLEGMT, WSCOMP, IMODEV, INTSPO, PUIMD,

AASUP, UPDAT, SQFTST, EIVM, SQFTAM, SQFTAM, SUPSTD, PURCHP, MSCSPO, DEPF, MSCSTD, WSHOPI

1010 FORMAT (3X,

'AVG. RECYCLE TIME (WKS)' - 'F4.1, 9X,

'TIME IN MEDIUM' - 'F5.1, 3X,

'WEEKS SCHOOL OPERATES' - 'F5.1, 3X,

'CURS. DEV HRS/COURSE HR.' - 'F5.1, 3X,

'INSTR./STUD. POS. RATIO' - 'F5.2, 8X,

'PCT COURSE REQUIRING DEV' - 'F4.2, 3X,

'ADMIN./STUD. POS. RATIO' - 'F4.3, 7X,

'PCT INSTR. MAINTAINED' - 'F3.2, 3X,

'SQ FEET/STUDENT POS.' - 'F6.2, 8X,

'PCT INSTR. MAINTAINED.' - 'F3.2, 3X,

'SUPPLIES/STUDENT POSITION' - 'F5.1, 3X,

'SUPPLIES/STUDENT' - 'F5.1, 3X,

'PURCHASE POLICY' - 'F5.1, 3X,

'MISC/STUDENT POSITION' - 'F5.1, 3X,

'DEPRICATION POLICY' - 'F1L1X,

'MISC/STUDENT' - 'F5.1, 3X,

'WEEKS SCHOOL AVAILABLE' - 'F4.1

WRITE INPUT COST FACTORS

910 IF (.NOT.RPT2) GO TO 920

WRITE (6,1001) COURSE, DRATE, N, INRATE, (COPMT(1)).I=1,10,

(COPMT(1)).I=1,10, STUDSL, CIDAD, STCDST, CSUPP, STCDST2,

CMIS, INSTSL, PCTPCH, ADMSL, CPSQFT

1001 FORMAT (11, 28X, 'COURSE COST ANALYSIS:', /, 'X, ' INPUT:', /, '3X,

'PERCENT', /, 'X, 'PLANNING PERIOD' = 'F4.1, 3X,

'INFLATION RATE' = 'F4.1, 3X,

'INPUT COST FACTORS' = 'F4.1, 3X,

'AVG. RECYCLE TIME (WKS)' = 'F4.1, 3X,

'RATE' = 'F4.1, 3X,

'QECN' - 'F6.2, 3X,

'OLMN' - 'F6.0, 3X,

'PLNLG YR.' - 'F6.2, 3X,

'INSTR. SALARY (ANNUAL)' = 'F6.2, 3X,

'STUDY TRAVEL TO/FROM' = 'F6.2, 3X,

'SUPPLIES COST' - 'F6.2, 3X,

'STUDY TRAVEL IN COURSE' - 'F6.2, 3X,

'MISCELLANEOUS COST' = 'F6.2, 3X,

'INSTR. SALARY (ANNUAL)' = 'F6.2, 3X,

'PCT MAX EQUIP PURCHASE' = 'F6.2, 3X,

'FACIL. SALARY (ANNUAL)' = 'F6.2, 3X,

'FACIL. COST/UNIT' = 'F6.2, 3X,

'FACIL. INIT/REFURB' = 'F6.2, 3X,

'FACIL. LIFE' = 'F6.2, 3X,

WRITE (6,1007) EQUATC, LOFEP, FACST, LOFFA

1007 FORMAT (3X,

'OUTD. EQUIPMENT UNIT COST' = 'F9.2, 6X,

'FACIL. INIT/REFURB COST' = 'F9.2, 6X,

'FACIL. LIFE' = 'F12, 3X,

'FACIL. INIT/REFURB COST' = 'F9.2, 6X,

'FACIL. LIFE' = 'F12, 3X,

WRITE SUMMARY OF ANNUAL TECHNICAL RESULTS

920 IF (.NOT.RPT3) GO TO 930

WRITE (6,1003) COURSE, DRATE, N, INRATE, AGRAD, AASIN, AADB, APSA,

AIDN, CADMIN, AEEQUP, AEEQUP, AEEQUP, AFACIL, AIMT,

ASUPPL, AMSC

1003 FORMAT (11, 28X, 'COURSE COST ANALYSIS:', /, 'X, ' INPUT:', /, '3X,

'PERCENT', /, 'X, 'PLANNING PERIOD' = 'F4.1, 3X,

'INFLATION RATE' = 'F4.1, 3X,

'OUTPUT TECHNICAL RESULTS' = 'F4.1, 3X,

'AVG. NO. OF GRADUATES RQD' = 'F10.1, 3X,

C-13
TAEG REPORT NO. 40

WRITE ANNUAL TECHNICAL RESULTS

930 IF (.NOT.RPT4) GO TO 940
WRITE (6,1005) COURSE,DRATE,N,INRATE,(DIGIT(I),I=1,5),
( (GRAD(I),I=1,5),(STUD(I),I=1,5),
AOB(I),I=1,5),(PSPI(I),I=1,5),
(RINSTRI,I=1,5), (PEQUIP(I),I=1,5),
(REQUIP(I),I=1,5),(AEQUIP(I),I=1,5),
(ADMIN(I),I=1,5),(FACIL(I),I=1,5),
(RMISC(I),I=1,5),
IF (N.LE.5) GO TO 940
WRITE (6,1005) COURSE,DRATE,N,INRATE,(DIGIT(I),I=6,10),
(GRAD(I),I=6,10),(STUD(I),I=6,10),
AOB(I),I=6,10),(PSPI(I),I=6,10),
(RINSTRI,I=6,10),(ADMIN(I),I=6,10),
(REQUIP(I),I=6,10),(AEQUIP(I),I=6,10),
(FACIL(I),I=6,10),(RMISC(I),I=6,10),
IF (N.LE.10) GO TO 940
WRITE (6,1005) COURSE,DRATE,N,INRATE,(DIGIT(I),I=11,15),
(GRAD(I),I=11,15),(STUD(I),I=11,15),
AOB(I),I=11,15),(PSPI(I),I=11,15),
(RINSTRI,I=11,15),(ADMIN(I),I=11,15),
(REQUIP(I),I=11,15),(AEQUIP(I),I=11,15),
(FACIL(I),I=11,15),(RMISC(I),I=11,15),
IF (N.LE.15) GO TO 940
WRITE (6,1005) COURSE,DRATE,N,INRATE,(DIGIT(I),I=16,20),
(GRAD(I),I=16,20),(STUD(I),I=16,20),
AOB(I),I=16,20),(PSPI(I),I=16,20),
(RINSTRI,I=16,20),(ADMIN(I),I=16,20),
(REQUIP(I),I=16,20),(AEQUIP(I),I=16,20),
(FACIL(I),I=16,20),(RMISC(I),I=16,20),
1005 FORMAT (11,28X,'COURSE COST ANALYSIS',/1X,'INPUT',/13X,
'STATE',/4X,'25X','DISCOUNT RATE',/4X,1,
'PERCENT',/3X,'PLANNING PERIOD',/12,27X,
'INFLATION RATE',/F4.1,'PERCENT',/11X,
'OUTPUT TECHNICAL RESULTS',/32X,
'YEAR',/12,'YEAR',/12,'YEAR',/12,'YEAR',/12,
'YEAR',/12,'YEAR',/12,'YEAR',/12,
'NO. OF GRADUATES RQD',/5(4X,16)/3X,
'NO. OF STUDENTS IN',/5(4X,16)/3X,
'AVG. IN-BOARD',/5(4X,16)/3X,
'AVG. STATION POSITIONS',/5(4X,16)/3X,
'NO. OF INSTRUCTORS RQD',/5(4X,16)/3X,
'NO. OF ADMIN. PERS RQD',/5(4X,16)/3X,
'NO. OF EQUIPMENTS RQD',/5(4X,16)/3X,
'ANNUAL EQUIP PURCHASES',/5(4X,16)/3X,
'ANNUAL EQUIP AVAILABLE',/5(4X,16)/3X,
'NO. SQUARE FEET RQD',/5(4X,16)/3X,
'NO. SUPPLIES RQD',/5(4X,16)/3X,
'NO. MISCELLANEOUS RQD',/5(4X,16)/3X,
WRITE SUMMARY OF ANNUAL COST RESULTS

940 IF (.NOT.RPT5) GO TO 950
WRITE (6,1002) COURSE,DRATE,N,INRATE,NSTUD,DSTUD,ISTUD,FSTUD,
VTRAV,DTRAV,TTRAV,FTRAV,MINSTR,DINSTR,IINSTR,
FINSTR,ADMIN,DAADM,IAADM,FADM,NEQUIP,DEQUIP,
WRITE ANNUAL COST RESULTS

950 IF (.NOT.RPT6) GO TO 960
WRITE (6,1004) COURSE,DRATE,N,INRATE,(DIGIT(1),I=1,5),
(CSTUD(I),I=1,5),(CTRAV(I),I=1,5),
(CINSTR(I),I=1,5),(CADMIN(I),I=1,5),
(CEQUIP(I),I=1,5),(CFACIL(I),I=1,5),
(CMISC(I),I=1,5),(CSUPPL(I),I=1,5),
(CGRAD(I),I=1,5),(CSTIN(I),I=1,5),
(CSP(1),I=1,5)
IF (N.LE.5) GO TO 960
WRITE (6,1004) COURSE,DRATE,N,INRATE,(DIGIT(1),I=6,10),
(CSTUD(I),I=6,10),(CTRAV(I),I=6,10),
(CINSTR(I),I=6,10),(CADMIN(I),I=6,10),
(CEQUIP(I),I=6,10),(CFACIL(I),I=6,10),
(CMISC(I),I=6,10),(CSUPPL(I),I=6,10),
(CGRAD(I),I=6,10),(CSTIN(I),I=6,10),
(CSP(1),I=6,10)
IF (N.LE.10) GO TO 960
WRITE (6,1004) COURSE,DRATE,N,INRATE,(DIGIT(1),I=11,15),
(CSTUD(I),I=11,15),(CTRAV(I),I=11,15),
(CINSTR(I),I=11,15),(CADMIN(I),I=11,15),
(CEQUIP(I),I=11,15),(CFACIL(I),I=11,15),
(CMISC(I),I=11,15),(CSUPPL(I),I=11,15),
(CGRAD(I),I=11,15),(CSTIN(I),I=11,15),
(CSP(1),I=11,15)
IF (N.LE.15) GO TO 960
WRITE (6,1004) COURSE,DRATE,N,INRATE,(DIGIT(1),I=16,20),
(CSTUD(I),I=16,20),(CTRAV(I),I=16,20),
(CINSTR(I),I=16,20),(CADMIN(I),I=16,20),
(CEQUIP(I),I=16,20),(CFACIL(I),I=16,20),
(CMISC(I),I=16,20),(CSUPPL(I),I=16,20),
(CGRAD(I),I=16,20),(CSTIN(I),I=16,20),
(CSP(1),I=16,20)
1004 FORMAT ('*12,28X,'COURSE COST ANALYSIS' /,'(1X,'INPUT::'/,'13X,'COU
TAEG REPORT NO. 40

SUPPLIES  $5(IX,F9.0),/3X,
MISCELLANEOUS $5(IX,F9.0),/3X,
TOTAL OPERATIONAL $5(IX,F9.0),/3X,
COST PER STUDENT INPUT $5(IX,F9.0),/3X,
COST PER STUDENT POSITION $5(IX,F9.0)

WRITE MISSING STATEMENT NUMBERS

960 IF (.NOT.RPT7) GO TO 990
   IF (K.LE.1) GO TO 990
   WRITE (6,5000)
   K = K-1
   WRITE (6,5001) ((MATA(I),MATB(I)),I=1,K)
5000 FORMAT (11,'MISSING DATA ANALYSIS')
5001 FORMAT (IX,'MISSING DATA CODE = ',I6,'CRIT CODE = ',I6)

ETAM DESIGN MOD - THIS ROUTINE TEMPORARILY STORES THE RESULTS OF LOOP ONE - BASELINE RUN.

990 IF (IND1.NE.0) GO TO 992
   DO 991 M=1,20
      ETAT01(M) = ETAGRD(M)
      ETAT02(M) = ETASTU(M)
      ETAT03(M) = ETA08(M)
      ETAT04(M) = ETA09(M)
      ETAT05(M) = ETA10(M)
      ETAT06(M) = ETAT08(M)
      ETAT07(M) = ETAT09(M)
      ETAT11(M) = ETAREC(M)
991 IND1 = 1
   GO TO 10

ETAM DESIGN MOD - THIS ROUTINE CALCULATES THE DIFFERENCE BETWEEN LOOP ONE - BASELINE RUN, AND LOOP TWO - BENEFIT

992 DO 993 M=1,20
   ETAGRD(M) = ETAT01(M) - ETAGRD(M)
   ETASTU(M) = ETAT02(M) - ETASTU(M)
   ETA08(M) = ETAT03(M) - ETA08(M)
   ETA09(M) = ETAT04(M) - ETA09(M)
   ETA10(M) = ETAT05(M) - ETA10(M)
   ETA08(M) = ETAT06(M) - ETA08(M)
   ETA09(M) = ETAT07(M) - ETA09(M)
   ETA09(M) = ETAT08(M) - ETA09(M)
   ETA11(M) = ETAT11(M) - ETA11(M)
993 IND = 0
   GO TO 995

ETAM DESIGN MOD - THE FOLLOWING OUTPUT REPRESENTS THE COST MODEL RESULTS FOR COURSES WHICH WILL BE SUMMARIZED BY COURSE AND OUTPUTED TO THE CMR FILE FOR USE IN THE DECISION AND FINANCIAL ANALYSIS MODULES.
IT REPRESENTS THE DIFFERENCE BETWEEN LOOP ONE WHICH CALCULATED BASELINE AND LOOP TWO WHICH HAD THE BENEFIT PATTERN APPLIED.

WRITE ETAM OUTPUT RESULTS

1510 FORMAT ('11',28X,'ETAM OUTPUT FACTORS',/32X,
   'YEAR ',I2,' YEAR ',I2,' YEAR ',I2,' YEAR ',I2,' YEAR ',I2,
   'YEAR ',I2,' YEAR ',I2,' YEAR ',I2,' YEAR ',I2,
   'NO. OF GRADUATES RQD ',S1X,'NO. OF STUDENTS RQD ',S1X,
   'AVERAGE ON BOARD ',S1X,'NO. OF ADMIN PERS RQD ',S1X,
   'NO. OF EQUIPMENTS RQD ',S1X,'NO. SQUARE FEET RQD ',S1X,
C-16
TAEG REPORT NO. 40

**NO. SUPPLIES ROD**
*5(1X,F9.0),//,3X*,

**NO. MISCELLANEOUS ROD**
*5(1X,F9.0),//,3X*,

**ANNUAL INVESTMENT DOLLARS**
*5(1X,F9.0),//,3X*,

**ANNUAL RECURRING CST/SAV**
*5(1X,F9.0)*

```
WRITE (6,1510) (DIGIT(I),I=1,5),
(ETAAD(I)),I=1,5), (ETATST(I)),I=1,5),
(ETAADM(I)),I=1,5), (ETAEQ(I)),I=1,5),
(ETAPF(I)),I=1,5), (ETAINV(I)),I=1,5),
(ETAREC(I)),I=1,5)

WRITE (6,1510) (DIGIT(I),I=6,10),
(ETAAD(I)),I=6,10), (ETATST(I)),I=6,10),
(ETAADM(I)),I=6,10), (ETAEQ(I)),I=6,10),
(ETAPF(I)),I=6,10), (ETAINV(I)),I=6,10),
(ETAREC(I)),I=6,10)

WRITE (6,1510) (DIGIT(I),I=11,15),
(ETAAD(I)),I=11,15), (ETATST(I)),I=11,15),
(ETAADM(I)),I=11,15), (ETAEQ(I)),I=11,15),
(ETAPF(I)),I=11,15), (ETAINV(I)),I=11,15),
(ETAREC(I)),I=11,15)

WRITE (6,1510) (DIGIT(I),I=16,20),
(ETAAD(I)),I=16,20), (ETATST(I)),I=16,20),
(ETAADM(I)),I=16,20), (ETAEQ(I)),I=16,20),
(ETAPF(I)),I=16,20), (ETAINV(I)),I=16,20),
(ETAREC(I)),I=16,20)

*** IF (.NOT.LAST) GO TO 10
1511 CONTINUE
1512 STOP
```

C-17
INTRODUCTION

The purpose of this appendix is to provide documentation for those programs that have been implemented to date in the computerization of ETAM. Broadly, this includes programs in two general support areas - those modules used to construct the data bases used by the interactive ETAM system, and the programs which implement the Range-of-Effect search and reporting process.

Neither the ETAM system, nor the environment in which the current system is implemented is of a trivial nature. The key to interpretation of a majority of this appendix presumes at least a minimal working knowledge on the part of the reader with both basic ETAM and the programming requirements of the timesharing vendor.

ORGANIZATION

This appendix is organized into six major sections. Section D.1 illustrates the logic flow of the eighteen programs implemented in the current ETAM system. A separate coverage of each of the programs is made in Section D.2. Section D.3 documents the timesharing system executive routines that are common to more than one program. The details and formats of the online files required to support the ETAM system are covered in Section D.4. Input and control data used in the makeup of the online ETAM system are documented in Section D.5. In Section D.6 appear listings of the source cards for each of the programs in the current system. In summary, the contents of this appendix are:

- Section D.1 Introduction
- Section D.2 Program Descriptions
- Section D.3 System Executive Routines
- Section D.4 Online File Formats
- Section D.5 Input and Control Data
- Section D.6 Program Listings (Service Cards)

EQUIPMENT ENVIRONMENT

All ETAM programs are designed to run in an interactive mode on a remote time-shared computer. Timesharing services and support were obtained on a lease basis from National CSS, Inc., Norwalk, Connecticut. The National CSS VP/CSS* system is not documented within this publication. A basic knowledge of sign-on, sign-off, editing, and program execution are required of the ETAM User.

* Trademark of National CSS, Inc.
The ETAM User requires a standard remote terminal such as an IBM 2741, or a Teleterm 1030*. The terminal is connected to the timesharing vendor's computer via a dial-up telephone line and a modem. Further information on the distinct types of terminal equipments which can be employed in conjunction with the timesharing support should be obtained from National CSS, Inc.

As currently configured, the ETAM system utilizes thirty cylinders of online direct-access storage (disk storage) and all programs will function in a 256K byte partition of storage.

PROGRAM ENVIRONMENT

The implications associated with the following program documentation and VP/CSS executive sequences presupposes a working knowledge above that of the casual User with the basic services provided by the National CSS (NCSS) timesharing vendor. These services are formally documented by NCSS and are not reproduced within this manual. To assist those who are unfamiliar with the NCSS documentation, the following references are mentioned:

<table>
<thead>
<tr>
<th>System Function</th>
<th>NCSS Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>Typing Conventions, Login, Logout</td>
<td>A Guide to the National CSS Computer System for Application Product Users</td>
</tr>
<tr>
<td></td>
<td>(NCSS Form 991) Section 1</td>
</tr>
<tr>
<td>System Interruptions</td>
<td>A Guide to the National CSS Computer System for Application Product Users</td>
</tr>
<tr>
<td></td>
<td>(NCSS Form 991) Section 7</td>
</tr>
<tr>
<td>CSS COMMANDS SUMMARY</td>
<td>A Guide to the National CSS Computer System for Application Product Users</td>
</tr>
<tr>
<td></td>
<td>(NCSS Form 991) Appendix B</td>
</tr>
<tr>
<td>CSS File Creation and Maintenance</td>
<td>A Guide to the National CSS Computer System for Application Product Users</td>
</tr>
<tr>
<td></td>
<td>(NCSS Form 991) Section 3</td>
</tr>
<tr>
<td></td>
<td>VP/CSS EDit Command (NCSS Form 108)</td>
</tr>
<tr>
<td>VP/CSS EXECUTIVE Language</td>
<td>VP/CSS EXECUTIVE Language (EXEC)</td>
</tr>
<tr>
<td></td>
<td>(NCSS Form 109-5)</td>
</tr>
</tbody>
</table>

* Trademark of Computer Devices, Inc.
ETAM SYSTEM INITIALIZATION

A total of eighteen programs are involved in the current ETAM system. In a broad sense, these can be divided into two distinct functional areas. Eleven programs are used in the generation and maintenance of the ETAM abbreviated data bases. The remaining seven programs implement the ETAM Range-of-Effect (ROE) search process.

Figure D.1-1 illustrates the basic initialization process that is required prior to the build of the abbreviated data bases. All files are indicated with a number; this "File Number" will be consistent throughout this appendix. The card files of Figure D.1-1 must be transmitted to the timesharing vendor's computer site and introduced to the online system via the terminal "Offline Read" process. Executive STARTUP has been designed to function with any or all of the card files indicated. This executive can be run multiple times, if required. All of the input data is sorted via executive STARTUP, and the Master Reference file (File Number 1) is set to zero by Program PI8.

As indicated in Figure D.1-1, the Descriptor Master Index is created by running Program PI6. All of the programs of the ETAM system are supported by a unique executive. The executive for each program bears the name "RUN" followed by the unique program name. Thus, Program PI6 is executed by an invocation of Executive RUNPI6.

LOAD OF THE ABBREVIATED COURSE DATA BASE

Figure D.1-2 illustrates the load of the four files of the Abbreviated Course data base. Each of the three data bases of the ETAM system (Course, Vehicle, and Task) is constructed by building a basic data file, a file directory, and a descriptor file. For the Course entity, two directories are required.

Program PI4 uses the CIN (Course Identification Number) of the Course descriptor input card file as a selection criteria for obtaining Course data from the NITRAS tape. In the current implementation, descriptors have been assigned to 121 unique CIN's. The NITRAS tape contains data relative to more than 4000 Courses. The following load counts were extracted from the output of Program PI4:

132 : Total Descriptor Cards Read
121 : Total Unique CIN's on Descriptor Cards
 50 : Total Number CIN's with No Data Match in NITRAS
111 : Total Number Course Records extracted from NITRAS
4,127 : Total Number Course Records on NITRAS Tape
INITIALIZATION (STARTUP) OF ETAM SYSTEM DATA SETS PRIOR TO CONSTRUCTION OF THE ABBREVIATED DATA BASES

All Files shown have filename = ETAM

Names shown above are filetype; e.g., the sorted course descriptor output of Executive STARTUP is called ETAM SCDESC

FIGURE D.1-1. INITIALIZATION (STARTUP) OF ETAM SYSTEM DATA SETS PRIOR TO CONSTRUCTION OF THE ABBREVIATED DATA BASES
DETAIL SYSTEM FLOW
ABBREVIATED COURSE DATA BASE LOAD

NOTES:
- Names in square brackets are program internal Filenames (Job Control Language DDNAME's)
- Numbers indicated on the files are File Reference Numbers as discussed in the text

FIGURE D.1-2. DETAIL SYSTEM FLOW ABBREVIATED COURSE DATA BASE LOAD
The two Course directories are constructed by Program P7. The Abbreviated Course data file is used as the source. A directory ordered by Course CIN and a directory ordered by Course CDP are required.

The Course descriptor card input is used by Program P7A to construct the Course descriptor file (File Number 6). This file will be the reference to be used in the ETAM Range-of-Effect search process.

Each of the programs of the data base build processes introduce counts into the Master Reference file (File Number 1). This file will be used by all programs which reference the abbreviated data bases. These files will be accessed randomly and the counts provided by the Master Reference file will be used as program internal selfchecks to prevent input/output errors.

The final program of the Course data base build process is the Course Print Program, Program P7B. This program offers interactive options to request print of the Course data base contents.

Program P20 is not shown on the diagrams of this appendix. This program's function is to print the entities of a selected data base, followed by all descriptors applicable to each entity. This program provides an interactive option to allow print of Course entities plus descriptor data.

ABBREVIATED VEHICLE DATA BASE LOAD

The load of the three files (data, directory, and descriptor) of the Vehicle data base is illustrated in Figure D.1-3. Both the data file and the descriptor file are loaded from card input as transmitted to the timesharing computer system. At present, data coverage for 63 unique Vehicle stock numbers has been provided. Record counts relative to the Vehicle data files are written into the Master Reference file by Program P9.

Printer dumps of the Abbreviated Vehicle data base contents are the output of Program P8A. Options (interactive) supporting data file print and print of the general Vehicle type descriptors are available. Program P20 (not shown on Figure D.1-3) can be used to produce a listing of all Vehicle entities and the specific descriptors applied to each.

ABBREVIATED TASKS DATA BASE LOAD

The build of the Abbreviated Tasks data base is shown in D.1-4. This is the largest of the three ETAM data bases. The current system supports 670 distinct Tasks. Both the Tasks data file and descriptor file are loaded from card input. Due to the bulk of the output for the Vehicle entity, Program P9 allows selection of the printer output device. The
DETAIL SYSTEM FLOW
ABBREVIATED VEHICLE DATA BASE LOAD

NOTES - Names in square brackets are program internal Filenames (Job Control Language DDNAME's)
- Numbers indicated on the files are File Reference Numbers as discussed in the text

FIGURE D.1-3. DETAIL SYSTEM FLOW: ABBREVIATED VEHICLE DATA BASE LOAD
DETAIL SYSTEM FLOW
ABBREVIATED TASK DATA BASE LOAD

NOTES — Names in square brackets are program internal Filenames (Job Control Language DDNAME's)
- Numbers indicated on the files are File Reference Numbers as discussed in the text

FIGURE D.1-4. DETAIL SYSTEM FLOW ABBREVIATED TASK DATA BASE LOAD
online terminal, offline printer, or both may be chosen interactively by the User. Program P9 inserts record counts from the Vehicle data base files into the Master Reference file (File Number 1).

Functioning in a manner parallel to that of Program P8A for Vehicle data, Program P9A will produce printer dumps of the contents of the Tasks data base. For a listing of Task entity and all related descriptors, Program P20 must be used.

ESTABLISHING A NEW ETAM PROJECT

All of the files generated thus far have been ETAM "System" files which are of equal utility to all ETAM Projects. The files covered from this point forward are unique to each unique ETAM project.

The first step in the introduction of a new ETAM Project to the ETAM System is accomplished by running Program P17. This is illustrated on Figure D.1-5. The four Project Data Base files of interest in the Range-of-Effect search are the ID, REC, REV, and REJ files (File Numbers 26, 27, 28, and 29). These files contain an identification of the specific project and indicate those descriptors that are to be used in the forthcoming ROE search. The other Project Data Base files that can be created via Program P17 are of textual interest only at this point.

Program P1 can be used to print any/all of the Project Data Base file contents for a specific project.

EXECUTION OF THE RANGE-OF-EFFECT SEARCH

The Search Argument Edit Program P2 accomplishes the first step of the Range-of-Effect search process. The search arguments for the entities Courses, Vehicles, and Tasks are located in files REC, REV, and REJ, respectively. The User may peruse those descriptors already in the system and delete or add as is necessary. The final action of Program P2 is to copy the edited results into the search argument file (File Number 30).

File SARG (File Number 30) is generated as a result of running Program P2. As shown in Figure D.1-5, this is the only method of generating the SARG file. Since the SARG file is the prime input to the Search Program P3, it is required that Program P2 be executed prior to executing Program P3.

Program P3 accomplishes the actual search. No extensive output is directed to the terminal by Program P3. The search results are accumulated into the RESU file (File Number 31). The User is informed by terminal output as to the size (number of records selected as a result of the search) of the results file RESU. If, due to the number of records selected in the search,
FIGURE D.1-5. DETAIL SYSTEM FLOW: PROJECT FILE GENERATION AND RANGE-OF-EFFECTS SEARCH

NOTES — Names in square brackets are program internal Filenames (Job Control Language DDNAME's)
- Numbers indicated on the files are File Reference Name's as discussed in the text
the User determines that an obvious error has been made in the selection of search arguments, the argument set may be edited by re-running Program P2. A subsequent run of the search Program P3 will create a new results file.

PROCESSING OF SEARCH RESULTS

Figure D.1-6 indicates the logic flow of the three programs that are involved in the processing of the results of a Range-of-Effect search. Program P5A is used to print the results of a search; the results file RESU is used as the input to Program P5A. This program also numbers each of the records as they are printed. This is vital since the results edit program (Program P5B) requires that record numbers be entered as a part of the interactive commands.

Program P5B allows the User to add or delete records from the search results. Upon termination, Program P5B copies all original and added results into the extract search results file REE (File Number 32).

Program P5C prints the contents of the extract results file in a finished format. This listing is the conclusion of the Range-of-Effect search process.
DETAIL SYSTEM FLOW
RANGE-OF-EFFECTS SEARCH RESULTS PRINT / EDIT

NOTES — Names in square brackets are program internal Filenames
(Job Control Language DDNAME's)
— Numbers indicated on the files are File Reference Numbers
as discussed in the text

FIGURE D.1-6. DETAIL SYSTEM FLOW RANGE-OF-EFFECT SEARCH RESULTS PRINT/EDIT

D.1-12
In this Section are documentation packages covering each of the eighteen programs of the current ETAM system implementation. Specific information relative to each program is provided in the following general areas:

1. Executive support required
2. Itemization of all data files processed (input and output), their internal/external names and data formats
3. Discussion of program function
4. Listings of any executive routine(s) associated only with the program being covered
5. Specific User instructions as to the use of program features
6. Error messages generated and their meaning
7. Sample output resulting from program execution
PROGRAM NAME: P1
ENTRY POINT: P1
RUN EXECUTIVES(S): RUNP1, NONAME, IFNAME, JCL, JCL2
USER OUTPUT VIA: Terminal, Offline Printer, or Both
USER PROMPTING: Internal to Program: (1) File(s) to be Printed (2) Output Device(s) Selection
FUNCTION: Prints Contents of Any/All Project Files

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSPRINT</td>
<td>OUTPUT</td>
<td>N/A: THIS IS THE OFFLINE PRINTER</td>
<td></td>
<td>FA, 121, 121</td>
<td></td>
</tr>
<tr>
<td>ID</td>
<td>INPUT</td>
<td>26</td>
<td>DASD</td>
<td>Projectname/ID</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>EXD</td>
<td>INPUT</td>
<td>33</td>
<td>DASD</td>
<td>Projectname/EXD</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>BPT</td>
<td>INPUT</td>
<td>34</td>
<td>DASD</td>
<td>Projectname/BPT</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>RKP</td>
<td>INPUT</td>
<td>35</td>
<td>DASD</td>
<td>Projectname/RKP</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>RRPJ</td>
<td>INPUT</td>
<td>36</td>
<td>DASD</td>
<td>Projectname/RRPJ</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>RRPK</td>
<td>INPUT</td>
<td>37</td>
<td>DASD</td>
<td>Projectname/RRPK</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>VQAL</td>
<td>INPUT</td>
<td>38</td>
<td>DASD</td>
<td>Projectname/VQAL</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>VARF</td>
<td>INPUT</td>
<td>39</td>
<td>DASD</td>
<td>Projectname/VARF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>SCEN</td>
<td>INPUT</td>
<td>40</td>
<td>DASD</td>
<td>Projectname/SCEN</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>ALTP</td>
<td>INPUT</td>
<td>41</td>
<td>DASD</td>
<td>Projectname/ALTP</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REC</td>
<td>INPUT</td>
<td>27</td>
<td>DASD</td>
<td>Projectname/REC</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REV</td>
<td>INPUT</td>
<td>28</td>
<td>DASD</td>
<td>Projectname/REV</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REJ</td>
<td>INPUT</td>
<td>29</td>
<td>DASD</td>
<td>Projectname/REJ</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>RESU</td>
<td>INPUT</td>
<td>31</td>
<td>DASD</td>
<td>Projectname/RESU</td>
<td>FB, 16, 800</td>
</tr>
<tr>
<td>REE</td>
<td>INPUT</td>
<td>32</td>
<td>DASD</td>
<td>Projectname/REE</td>
<td>FB, 16, 800</td>
</tr>
<tr>
<td>MREF</td>
<td>INPUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td></td>
</tr>
</tbody>
</table>
### Internal Filename (DDNAME) Usage Refer To File Reference Number Media External NCSS Filename/Filetype Format

<table>
<thead>
<tr>
<th>Filename</th>
<th>Usage</th>
<th>Reference Number</th>
<th>Media</th>
<th>Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIND</td>
<td>DIRECT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td></td>
</tr>
<tr>
<td>COURSE</td>
<td>DIRECT</td>
<td>7</td>
<td>DASD</td>
<td>ETAM/COURSE</td>
<td></td>
</tr>
<tr>
<td>DCIN</td>
<td>INPUT</td>
<td>8</td>
<td>DASD</td>
<td>ETAM/DCIN</td>
<td></td>
</tr>
<tr>
<td>DCDP</td>
<td>INPUT</td>
<td>9</td>
<td>DASD</td>
<td>ETAM/DCDP</td>
<td></td>
</tr>
<tr>
<td>VEHS</td>
<td>DIRECT</td>
<td>15</td>
<td>DASD</td>
<td>ETAM/VEHS</td>
<td></td>
</tr>
<tr>
<td>DVEH</td>
<td>INPUT</td>
<td>16</td>
<td>DASD</td>
<td>ETAM/DVEH</td>
<td></td>
</tr>
<tr>
<td>TASKS</td>
<td>DIRECT</td>
<td>22</td>
<td>DASD</td>
<td>ETAM/TASKS</td>
<td></td>
</tr>
<tr>
<td>DTASK</td>
<td>INPUT</td>
<td>23</td>
<td>DASD</td>
<td>ETAM/DTASK</td>
<td></td>
</tr>
</tbody>
</table>

### Discussion

This program has as its function the printing of all files of the ETAM system called "Project Files". Program P1 has been designed to allow the interactive selection of any or all of the files to be printed.

### Listing of EXECUTIVE =RUNP1

```
P RUNP1 EXEC

&TYPE OFF
&COMMENT INTERACTIVE PROJECT FILE PRINT PROGRAM.
&COMMENT SINGLE ENTRY PARM REQUIRED - PROJECT NAME.
&COMMENT EXEC = DEVICE WILL PROMPT FOR OUTPUT DEVICE.
&IFGINDEX EO 1 &GOTO -STP1
EXEC NONAME RUNP1
-STP1 &ALPHA3 = &1
EXEC IFNAME
EXEC JCL
EXEC JCL2
FILEDEF SYSPRINT PTR RE FA LR 121 BL 121
LOAD P1 (CLEAR LIBE) PLILIB
START (BRIEF)
FILEDEF = CLEAR
&EXIT
```
Program P1 Specific User Instructions

This program processes only those files associated with a unique ETAM project within a single run. Accordingly, the name of the project must be provided by the User when invoking the RUNP1 executive. On the sample runs to follow, this is demonstrated using the ETAM project name "PN123".

Upon entry and after each processing command has been completed, the User will be prompted for entry as follows:

** ENTER FILETYPE FOR OUTPUT, MENU, OR QUIT **

A "MENU" selection will cause the printing of a list of filetypes and titles for output. Following the presentation, a return is made to the prompt message as above. The "QUIT" option is self-explanatory. If a valid filetype is entered, the system will then prompt for the desired output device, as follows:

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT

The response options are self-explanatory to this prompt.

If, instead of a specific filetype, a response of "ALL" is made to the first general prompt, above, all filetypes associated with the current project will be output. Following the "ALL" selection, the device selected for output will be used for all output. After processing an "ALL" option, return is made to the first general prompt as usual.

Program P1 Error Messages

If the response to a specific prompt is not decodeable, the following error message is output:

** LAST ENTRY UNRECOGNIZED - TRY AGAIN **

The above message will be followed by a return to the general prompt message that preceded it.

In response to the prompt for output device, an unrecognized input will result in the following:

UNABLE TO DETERMINE ENTRY TYPE - TRY AGAIN
(followed by a return to allow another device selection)
In the current ETAM implementation, the following filetypes have not been fully specified:

<table>
<thead>
<tr>
<th>Filetype</th>
<th>Project File</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXC</td>
<td>Course Extract Data Base</td>
</tr>
<tr>
<td>EXJ</td>
<td>Tasks (Jobtasks) Extract Data Base</td>
</tr>
<tr>
<td>EXV</td>
<td>Vehicle Extract Data Base</td>
</tr>
<tr>
<td>CMR</td>
<td>Cost Model Results</td>
</tr>
<tr>
<td>TRER</td>
<td>Tree Results</td>
</tr>
<tr>
<td>FINR</td>
<td>Financial Results</td>
</tr>
</tbody>
</table>

An attempt to print one of these filetypes will result in the following message:

** FILETYPE xxxx IS NOT YET SERVICEABLE, TRY AGAIN **

where the name of the filetype of concern replaces "xxxx" in the message.

Program Pl Sample Run Output

The following illustrates a run of Program Pl servicing the Project Files of Project "PN123".

RUNP1 PN123

PROJECT NAMED PN123 ALREADY EXISTS, IS THIS CORRECT? (RESPOND YES OR QUIT)

YES

GENERAL-PURPOSE PROJECT FILE PRINT PROGRAM IS NOW STARTING...

**: ENTER FILETYPE FOR OUTPUT, MENU, OR QUIT **:

PROJECT FILETYPES FOR OUTPUT ARE:

ID - PROJECT DESCRIPTION
EXD - EXTRACT DEFAULTS
BPT - BENEFIT PATTERN
RKP - RISK PROFILE
RRPJ - RISK REDUCTION PROJECTS
RRPK - RISK REDUCTION PACKAGES
VQAL - VARIABLES QUALIFICATION
VARF - VARIABLE REFERENCES
SCEN - SCENARIOS
ALTP - ALTERNATE PROJECTS
REC - COURSES ROE SEARCH ARGUMENTS
REV - VEHICLES ROE SEARCH ARGUMENTS
REJ - TASKS ROE SEARCH ARGUMENTS
RESU - ROE TOTAL SEARCH RESULTS
REE - ROE SEARCH RESULTS
LISTING OF PROJECT FILE CONTENTS
PN123 / REC - COURSES ROE SEARCH ARGUMENTS

30 OPERATIONS
31 MAINTENANCE
40 EQUIPMENT & OBJECTS USED: REAL
64 PROCEDURE FOLLOWING
80 ORIENTATION, FAMILIARIZATION
81 TASK NOMENCLATURE, IDENTs, LOCATIONS, FACTS, RULES
82 TASK FORMATS AT CONCEPTUAL LEVEL
83 PROCEDURES AT VERBAL LEVEL ONLY
84 TASK COMPONENTS WITH GUIDANCE
85 ENTIRE JOB-TASK PROCEDURALLY AT BARELY ACCEPTABLE MASTERY
86 HIGHLY PROFICIENT IN WORK CONTEXT

A TOTAL OF 11 RECORDS ARE PRESENT WITHIN THIS FILE

PROJECT-FILE PRINT PROGRAM IS NOW TERMINATING
**Program Name:** P2  
**Entry Point:** P2  
**Run Executives(s):** RUNP2, NONAME, IFNAME, JCL, JCL2  
**User Output Via:** Terminal Only  
**User Prompting:** Internal Program Only  
**Function:** Interactively Modify/Examine the Contents of the ROE Search Argument Files REC, REV, and REJ (File Reference Numbers 27, 28, and 29)

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MREF</td>
<td>INPUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DIND</td>
<td>DIRECT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>SARG</td>
<td>OUTPUT</td>
<td>30</td>
<td>DASD</td>
<td>Projectname/SARG</td>
<td>F, 240, 240</td>
</tr>
<tr>
<td>ID</td>
<td>INPUT</td>
<td>26</td>
<td>DASD</td>
<td>Projectname/ID</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REC</td>
<td>INOUT/OUTPUT</td>
<td>27</td>
<td>DASD</td>
<td>Projectname/REC</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REV</td>
<td>INOUT/OUTPUT</td>
<td>28</td>
<td>DASD</td>
<td>Projectname/REV</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REJ</td>
<td>INOUT/OUTPUT</td>
<td>29</td>
<td>DASD</td>
<td>Projectname/REJ</td>
<td>F, 80, 80</td>
</tr>
</tbody>
</table>

**Discussion**

This program serves a twofold purpose:

For the User: This program allows interactive printing, adding, and deleting of descriptors to be used as search arguments in future Range-of-Effects (ROE) searches.
For the System: Program P2 creates a search argument file (File Number 30) using the descriptors supplied via Project Files REC, REV, and REJ. The ROE Search Program (Program P3) cannot be run successfully until this file has been created.

NOTE - This implies that Program P2 must be run at least one time prior to running Program P3.

Listing of EXECUTIVE = RUNP2

&TYPE OFF
&COMMENT INTERACTIVE EDIT OF SEARCH ARGUMENTS FROM PROJECT FILES
&COMMENT REC, REV, REJ TO THE SEARCH ARGUMENT PF = SARG.
&COMMENT SINGLE ENTRY PARM REQUIRED IS PROJECT NAME.
&COMMENT OUTPUT ONLY TO TERMINAL IN THIS SEQUENCE.
&IF &INDEX EO 1 &GOTO -STP1
EXEC NONAME RUNP2
-STP1 &ALPHA3 = &1
EXEC IFNAME
EXEC JCL
EXEC JCL2
LOAD P2 (CLEAR LIBE) PLILIB
START (BRIEF)
FILEDEF :: CLEAR
GEXIT

Program P3 Specific User Instructions

Program P2 processes only those files associated with a unique ETAM project during a single run. It is mandatory that the name of the specific project be provided by the User when invoking the RUNP2 executive. On the sample runs to follow, this is illustrated using the ETAM project name "PN123".

This program has been designed to work in an interactive manner; all User input/output is through the online terminal.

After program initialization, the following message will appear at the terminal:

** MAKE INITIAL ENTRY SELECT
The anticipated response to this prompt is the word "SELECT" followed by one of the options "COURSES", "VEHICLES", or "TASKS". This allows the User to select one of the three ETAM search entities for consideration. An illegal entry at this point (usually mispelling) will be flagged by the message:

** SELECTION ERROR - TRY AGAIN

(followed by the initial entry prompt)

A successful "SELECT" entry results in the following output:

** XXXXXXXX SEARCH DESCRIPTORS SELECTED
** YY DESCRIPTORS NOW SELECTED

where "XXXXXXXX" will indicate the entity type selected, and "YY" indicates the total number of descriptor search arguments presently selected for this Project. The above is then followed by:

** MAKE NEXT ACTION REQUEST

This message is referred to in the following as the "general prompt". It follows all User entry to allow further entry. The User may, at this point, make one of the following entries:

SELECT (followed by) COURSES or VEHICLES or TASKS
ADD number
DELETE number
PRINT
FILE
QUIT

The "SELECT" entry functions in the same manner as discussed previously. The "ADD" and "DELETE" entries allow the User to modify the contents of the REC, REV, and REJ files. The "number" fields represent the descriptor numbers to be added or deleted. Once an entity type (Courses, Vehicles, or Tasks) has been selected, the "ADD", "DELETE", and "PRINT" verbs apply only to descriptors of that entity type until a new type is selected.

Selection of the "QUIT" option terminates Program P2 without permanent modification to any of the search designator (argument) files.

The "FILE" option terminates Program P2; all files to be modified are altered permanently in accordance with the interactive entry commands processed during Program P2 operation.
Program P2 Error Messages

During Program P2 initialization, the contents of the REC, REV, and REJ Project Files are input. These files were loaded previously by a run of Program P17. P17 merely loaded the files with no error diagnosis on the contents. Program P2 accomplishes error detection during their load. In the event of an invalid descriptor, the descriptor is removed from the file of its origin and the following message results:

** ILLEGAL DESCRIPTOR - XXXX - DETECTED IN YYYYYYYY FILE **
** DESCRIPTOR WILL BE IGNORED AND REMOVED FROM THE YYYYYYYY FILE **

In the above message, "XXXX" will indicate the invalid descriptor number and "YYYYYYYY" the entity type (Course, Vehicle, or Tasks).

The following general purpose message results for any entry error following the general prompt message:

** UNIDENTIFIED ACTION REQUEST - TRY AGAIN **

An illegal descriptor number will be signified by:

** KEYBOARD ENTRY - XXXX - CONTAINED ILLEGAL NUMERIC CHARACTER**

Selection of a descriptor number that does not apply to the selected entity type results in:

** XXXX IS NOT A LEGAL DESCRIPTOR FOR YYYYYYYY TYPE **

The following self-explanatory message results from an illegal deletion request:

** DESIGNATOR NUMBER XXXX HAS NOT BEEN SELECTED; IT CANNOT BE DELETED **

A general-purpose error message can result during Program P2 operation:

** ERROR NUMBER XX HAS OCCURED **
(followed by a general data-output line)

In the above, the error number "XX" has the following meaning:

01, - Error occured during "PRINT" option processing.
02 Direct access indexing into the Master Descriptor Index file (File Number 3) failed. The second error line presents the indices of concern.
03 - While processing an "ADD" entry, an indexing problem arose into the same file as error number 1, above.

04 - This is an internal program indexing problem encountered while attempting rewrite of the descriptor search argument files ("FILE" option processing).

Program P2 Sample Run Output

The following sample illustrates a complete run of Program P2. Only the "COURSES" type descriptors are referenced in this run.

```
RUNP2 PN123

PROJECT NAMED PN123 ALREADY EXISTS, IS THIS CORRECT? (RESPOND YES OR QUIT)

YES $$$

SEARCH DESCRIPTOR EDITOR RUN FOR PROJECT : PN123

3-D PROCEDURAL TRAINER

*** MAKE INITIAL ENTRY SELECT

_SELECT COURSES

*** COURSES SEARCH DESCRIPTORS SELECTED

*** 11 DESCRIPTORS NOW SELECTED

*** MAKE NEXT ACTION REQUEST

_PRINT

::: GROSS JOB CATEGORIES
3# OPERATIONS
31 MAINTENANCE

*** OBJECTIVE TASK VARIABLES AS MANIFEST IN THE TRAINING
4# EQUIPMENT & OBJECTS USED: REAL

*** TASK FUNCTIONS DOMINANT IN TRAINING
6# PROCEDURE FOLLOWING

*** STAGE OF LEARNING
-8# ORIENTATION, FAMILIARIZATION
-81 TASK NOMENCLATURE, IDENTS, LOCATIONS, FACTS, RULES
83 PROCEDURES AT VERBAL LEVEL ONLY
84 TASK COMPONENTS WITH GUIDANCE
85 ENTIRE JOB-TASK PROCEDURALLY AT BARELY ACCEPTABLE MASTERY
86 HIGHLY PROFICIENT IN WORK CONTEXT
87 UNUSUAL TASK CONDITIONS
```
TAEG REPORT NO. 40

*** MAKE NEXT ACTION REQUEST
DELETE 87
COURSES DESIGNATOR NUMBER 87 HAS BEEN DELETED

*** MAKE NEXT ACTION REQUEST
ADD 82
ADDED DESIGNATOR = 82 TASK FORMATS AT CONCEPTUAL LEVEL
TOTAL DESIGNATORS NOW SELECTED : 11

*** MAKE NEXT ACTION REQUEST
FILE
MODIFIED COURSES DESIGNATORS HAVE BEEN SAVED
SEARCH DESIGNATOR EDITOR NOW TERMINATING

D.2-12
TAEG REPORT NO. 40

PROGRAM NAME: P3
ENTRY POINT: P3

RUN EXECUTIVES(S): RUNP3, NONAME, IFNAME, JCL, JCL2

USER OUTPUT VIA: Terminal Only
USER PROMPTING: Not Required

FUNCTION: Range-Of-Effects Search Module

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MREF</td>
<td>INPUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DIND</td>
<td>DIRECT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>SARG</td>
<td>INPUT</td>
<td>30</td>
<td>DASD</td>
<td>Projectname/SARG</td>
<td>F, 240, 240</td>
</tr>
<tr>
<td>RESU</td>
<td>OUTPUT</td>
<td>31</td>
<td>DASD</td>
<td>Projectname/RESU</td>
<td>FB, 16, 800</td>
</tr>
<tr>
<td>ID</td>
<td>INPUT</td>
<td>26</td>
<td>DASD</td>
<td>Projectname/ID</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>COURSE</td>
<td>DIRECT</td>
<td>7</td>
<td>DASD</td>
<td>ETAM/COURSE</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DCIN</td>
<td>INPUT</td>
<td>8</td>
<td>DASD</td>
<td>ETAM/DCIN</td>
<td>F, 760, 760</td>
</tr>
<tr>
<td>VEHS</td>
<td>DIRECT</td>
<td>15</td>
<td>DASD</td>
<td>ETAM/VEHS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DVEH</td>
<td>INPUT</td>
<td>16</td>
<td>DASD</td>
<td>ETAM/DVEH</td>
<td>F, 750, 750</td>
</tr>
<tr>
<td>TASKS</td>
<td>DIRECT</td>
<td>22</td>
<td>DASD</td>
<td>ETAM/TASKS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DTASK</td>
<td>INPUT</td>
<td>23</td>
<td>DASD</td>
<td>ETAM/DTASK</td>
<td>F, 1500, 1500</td>
</tr>
<tr>
<td>DESC</td>
<td>INPUT</td>
<td>6</td>
<td>DASD</td>
<td>ETAM/DESC</td>
<td>F, 213, 213</td>
</tr>
<tr>
<td>VDESC</td>
<td>INPUT</td>
<td>12</td>
<td>DASD</td>
<td>ETAM/VDESC</td>
<td>F, 213, 213</td>
</tr>
<tr>
<td>TDESC</td>
<td>INPUT</td>
<td>19</td>
<td>DASD</td>
<td>ETAM/TDESC</td>
<td>F, 213, 213</td>
</tr>
</tbody>
</table>
Discussion

The Range-of-Effects search process is accomplished within Program P3. The descriptor search arguments are input via File Number 30. As mentioned in the section covering Program P2, File Number 30 is created only through a successful execution of Program P2.

Only synopsis type output is directed to the online terminal as a result of executing Program P3. The search results output is passed via filetype Program P3. The search results output is passed via filetype RESU (File Number 31) to the Programs P5A, P5B, and P5C for output editing and processing.

Listing of EXECUTIVE = RUNP3

```plaintext
&TYPE OFF
&COMMENT RANGE-OF-EFFECT (ROE) SEARCH EXECUTION.
&COMMENT SINGLE ENTRY PARM REQUIRED IS PROJECT NAME.
&COMMENT OUTPUT ONLY TO TERMINAL IN THIS SEQUENCE.
&IF &INDEX EO 1 &GOTO -STP1
EXEC NONAME RUNP3
-STP1 &ALPHA3 = &1
EXEC IFNAME
EXEC JCL
EXEC JCL2
LOAD P3 (CLEAR LIBE) PLILIB
START (BRIEF)
FILEDEF " CLEAR
&EXIT
```

Program P3 Specific User Instructions

The Project Files to be referenced by Program P3 in a single unique run can only be the files associated with a single ETAM project. The name of the desired project must be supplied by the User when invoking the RUNP3 executive. On the sample terminal outputs shown below, the project being referenced is "PN123".

Only the project name must be supplied by the User when initiating a Program P3 run. As mentioned above, Program P3 cannot be run until Program P2 has been successfully run at least one time.
Program P3 Error Messages

The following general error text can be output from Program P3:

** ERROR NUMBER XX HAS OCCURRED **

(followed by a general data-output line)

where the error number "XX" in the above message has the following significance. All are internal program indexing errors.

01 - Indexing error in handling categories within the Master Descriptor Index file (File Number 3).

02 - Total number of input search descriptors exceeds 99. Note that no descriptor is greater than a two-digit number.

03 - More than 15 categories of descriptors found for a single entity within the Master Descriptor Index.

04 - Attempted to access a record in the Abbreviated Course file (File Number 7) beyond the bounds of the file.

05 - Same error type as type 04, except for the Abbreviated Vehicle file (File Number 15).

06 - Same error type as type 04, except for the Abbreviated Tasks file (File Number 22).

Program P3 Sample Run Output

The following reflects a complete run of Program P3 against the sample project "PN123".

RUNP3 PN123

PROJECT NAMED PN123 ALREADY EXISTS,
IS THIS CORRECT? (RESPOND YES OR QUIT)

YES

$5$5
RANGE-OF-EFFECTS (ROE) SEARCH PROGRAM IS STARTING

RANGE-OF-EFFECTS SEARCH FOR PROJECT : PN123
3-D PROCEDURAL TRAINER

::: NOW PROCESSING COURSES SEARCH - 11 DESCRIPTORS SELECTED

TOTAL NUMBER DESCRIPTOR RECORDS EXAMINED : 122
TOTAL RECORDS SELECTED WITH MATCHING DATA : 37
TOTAL RECORDS SELECTED WITHOUT ABBREVIATED DATA : 24
ACCUMULATED SEARCH OUTPUT RECORDS, THUS FAR : 61

D.2-15
NOW PROCESSING VEHICLES SEARCH - Ø DESCRIPTORS SELECTED

NO SEARCH DESCRIPTORS SPECIFIED FOR VEHICLES TYPE DATA - SEARCH IS BYPASSED

NOW PROCESSING TASKS SEARCH - Ø DESCRIPTORS SELECTED

NO SEARCH DESCRIPTORS SPECIFIED FOR TASKS TYPE DATA - SEARCH IS BYPASSED

DESIGNATOR SEARCH PROGRAM IS TERMINATING

TOTAL ACCUMULATED SEARCH OUTPUT RECORDS : 61
PROGRAM NAME: P5A
ENTRY POINT: IHECMS (Execution Parameter Supplied by Executive Up on Entry)

RUN EXECUTIVES(S): RUNP5A, NONAME, IFNAME, DEVICE, JCL, JCL2

USER OUTPUT VIA: Terminal, Offline Printer, or Both

USER PROMPTING: External Executive - Output Device Select

FUNCTION: Print Results of Range-Of-Effects Search

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>INPUT</td>
<td>26</td>
<td>DASD</td>
<td>Projectname/ID</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>RESU</td>
<td>INPUT</td>
<td>31</td>
<td>DASD</td>
<td>Projectname/RESU</td>
<td>FB, 16, 800</td>
</tr>
<tr>
<td>MREF</td>
<td>INPUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>COURSE</td>
<td>DIRECT</td>
<td>7</td>
<td>DASD</td>
<td>ETAM/COURSE</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DCIN</td>
<td>INPUT</td>
<td>8</td>
<td>DASD</td>
<td>ETAM/DCIN</td>
<td>F, 760, 760</td>
</tr>
<tr>
<td>DCDP</td>
<td>INPUT</td>
<td>9</td>
<td>DASD</td>
<td>ETAM/DCDP</td>
<td>F, 1200, 1200</td>
</tr>
<tr>
<td>VEHDS</td>
<td>DIRECT</td>
<td>15</td>
<td>DASD</td>
<td>ETAM/VEHS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DVEH</td>
<td>INPUT</td>
<td>16</td>
<td>DASD</td>
<td>ETAM/DVEH</td>
<td>F, 750, 750</td>
</tr>
<tr>
<td>TASKS</td>
<td>DIRECT</td>
<td>22</td>
<td>DASD</td>
<td>ETAM/TASKS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DTASK</td>
<td>INPUT</td>
<td>23</td>
<td>DASD</td>
<td>ETAM/DTASK</td>
<td>F, 1500, 1500</td>
</tr>
<tr>
<td>SYSPRINT</td>
<td>OUTPUT</td>
<td>N/A: THIS IS THE OFFLINE PRINTER</td>
<td></td>
<td>FA, 121, 121</td>
<td></td>
</tr>
</tbody>
</table>

Discussion

The direct results of the Range-of-Effects search (Program P3) are printed by this program.
Listing of EXECUTIVE = RUNP5A

@TYPE OFF
@COMMENT ROE PAW SEARCH RESULTS PRINTED HEREIN,
@COMMENT SINGLE ENTRY PARM REQUIRED - PROJECT NAME
@COMMENT EXEC = DEVICE WILL PROMPT FOR OUTPUT DEVICE
@IF @INDEX EQ 1 @GOTO -STP1
EXEC NONAME RUNP5A
-STP1 &ALPHA3 = &1
EXEC IFNAME
&SPACE
EXEC DEVICE
&SPACE
EXEC JCL
EXEC JCL2
FILEDEF SYSPRINT PTR RE FA LR 121 BL 121
LOAD P5A (CLEAR LIB) PLILIB
START IHECMS &ALPHA1 (BRIEF)
FILEDEF ; CLEAR
&EXIT

Program P5A Specific User Instructions

A single run of Program P5A processes only those Project Files associated with a single unique project. The name of the project of concern must be provided by the User when invoking the RUNP5A executive. In the sample output provided below, this is illustrated by reference to project "PN123".

The Range-of-Effects search program (Program P3) provides no direct output, only summary information. This program (Program P5A) produces an itemized listing of the search results using the RESU (File Number 31) as input.

The output records are numbered. These numbers will provide reference for any editing activity that might follow by running Program P5B.

Program P5A Error Messages

None generated.
Program P5A Sample Run OUtput

The following reflects a complete run of Program P5A, using the sample project name "PN123".

RUNP5A PN123

PROJECT NAMED PN123 ALREADY EXISTS,
IS THIS CORRECT? (RESPOND YES OR QUIT)

YES

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT

BOTH

*$EXECUTION:

::: INITIAL SEARCH RESULTS FOR PROJECT -PN123 :::

3-D PROCEDURAL TRAINER

PAGE NUMBER :  1

RANGE-OF-EFFECTS RESULTS FOR : COURSES
-----------------------------------------------
1 A1#2#6#0# NOTE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN:
2 A1#2#9#3#0# NOTE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN:
3 A1#2#9#5#0# NOTE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN:
4 A1#5#5#0#2#2# INERT NAV PRINC
5 A2#4#1#1#5#8# IS A
6 A4#1#1#1#6#78 EA A
7 A4#1#1#1#6#267 EA A
8 A4#3#1#1#1#35 EOD BASIC NAVY
9 A4#3#1#1#1#36 EOD REFR NAVY
10 A4#3#1#1#1#36 EOD REFR NAVY
11 A4#3#1#1#1#36 EOD REFR NAVY
12 A4#9#1#4#2#2# DIVER FIRST
13 A5#5#1#1#1#33 CAR HAND BAS
14 A5#5#1#1#39#31 UNREP MECH/HYU
15 A5#5#1#1#1#34 FORK LIFT OPER
16 A5#5#1#1#6#5#3 CTO A
17 A6#5#1#1#5#39# 12#5# PSI BT
18 A6#5#1#1#5#224 OX GENR 6L16#P
19 A6#5#1#1#5#6#8 463E SOLAR GAS TURBIN

D.2-19
TAEG REPORT NO. 40

21 A67W1111 3078 WATCH REPAIR
22 A79W1111 337 M WELD/HPRES PIPE
23 A72W1113 4662 UT-J
24 B3JW1115 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
25 C WJ3722 2887 TA4JF FAM PILOTS
26 C WJ3722 90 7 TA4JF FAM PILOTS
27 C WJ3722 90 7 TA4JF FAM PILOTS
28 C WJ3722 9738 TA4JF FAM PILOTS
29 C 2C3352 2576 UH1N A/C FAM/P/
30 C 2C3352 7529 AA18 INT MAI
31 C 2C3353 7914 A7AB ATT HEADING
32 C 2C3353 7915 A7AB ATT HEADING
33 C 121W1111 346L AWM23 RADIO FREQ
34 C 121W1111 544L AWM23 RADIO FREQ
35 C 2C3352 344C E2C WEA SYS FAM
36 C 2C3352 341D 54H6077 PROP INT
37 C 2C3352 7365 CVA/CV ALW SUPV
38 D 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
39 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
40 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
41 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
42 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
43 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
44 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
45 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
46 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
47 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
48 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
49 E 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
50 I 2C3352 5360 MULTI-THREAT TNG
51 I 2C3352 2637 ADV EW OP'S CRSE
52 I 2C3352 2181 IPC
53 J 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
54 K 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
55 L 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
56 L 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
57 M 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
58 N 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
59 O 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
60 P 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
61 Q 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN

PAGE NUMBER : 2

RANGE-OF-EFFECTS RESULTS FOR : COURSES

56 L 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
57 M 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
58 N 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
59 O 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
60 P 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN
61 Q 2C3352 NOE - NO MATCHING NITRAS DATA AVAILABLE FOR THIS CIN

D.2-20
RESULTS FILE NOW AT END-OF-FILE

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL NUMBER COURSE RECORDS PROCESSED</td>
<td>61</td>
</tr>
<tr>
<td>TOTAL NUMBER VEHICLE RECORDS PROCESSED</td>
<td>9</td>
</tr>
<tr>
<td>TOTAL NUMBER JOBTASK RECORDS PROCESSED</td>
<td>9</td>
</tr>
<tr>
<td>TOTAL NUMBER ROE RESULT RECORDS READ</td>
<td>61</td>
</tr>
<tr>
<td>RECORDS WITHOUT MATCHING ABBREVIATED DATA</td>
<td>24</td>
</tr>
<tr>
<td>TOTAL NUMBER DIRECTORY SEARCH FAILURES</td>
<td>6</td>
</tr>
</tbody>
</table>
PROGRAM NAME : P5B

ENTRY POINT :

RUN EXECUTIVES(S) : RUP5B, NONAME, IFNAME, JCL, JCL2

USER OUTPUT VIA : Terminal Only

USER PROMPTING : Internal Program Only

FUNCTION : Interactive Edit of Range-Of-Effects (ROE) Results

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>INPUT</td>
<td>26</td>
<td>DASD</td>
<td>Projectname/ID</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REE</td>
<td>OUTPUT</td>
<td>32</td>
<td>DASD</td>
<td>Projectname/REE</td>
<td>FB, 16, 800</td>
</tr>
<tr>
<td>RESU</td>
<td>INPUT/OUTPUT</td>
<td>31</td>
<td>DASD</td>
<td>Projectname/RESU</td>
<td>FB, 16, 800</td>
</tr>
<tr>
<td>MREF</td>
<td>INPUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>COURSE</td>
<td>DIRECT</td>
<td>7</td>
<td>DASD</td>
<td>ETAM/COURSE</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DCIN</td>
<td>INPUT</td>
<td>8</td>
<td>DASD</td>
<td>ETAM/DCIN</td>
<td>F, 760, 760</td>
</tr>
<tr>
<td>DCDP</td>
<td>INPUT</td>
<td>9</td>
<td>DASD</td>
<td>ETAM/DCDP</td>
<td>F, 1200, 1200</td>
</tr>
<tr>
<td>TASKS</td>
<td>DIRECT</td>
<td>22</td>
<td>DASD</td>
<td>ETAM/TASKS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DTASK</td>
<td>INPUT</td>
<td>23</td>
<td>DASD</td>
<td>ETAM/DTASK</td>
<td>F, 1500, 1500</td>
</tr>
<tr>
<td>VEHS</td>
<td>DIRECT</td>
<td>15</td>
<td>DASD</td>
<td>ETAM/VEHS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DVEH</td>
<td>INPUT</td>
<td>16</td>
<td>DASD</td>
<td>ETAM/DVEH</td>
<td>F, 750, 750</td>
</tr>
</tbody>
</table>

Discussion

This program allows the User to peruse and edit the results of a previous Range-of-Effects search. In this regard:

- Program P3 is the search routine. This module creates the un-edited results file RESU (File Number 31).

- Program P5A prints the RESU file and numbers the records therein. These record numbers are required for reference in the operation of Program P5B.

D.2-22
Program P5B modifies the search results RESU as desired by the User. A modified file, containing only original search results and added records is created by this program; this is the REE file (File Number 32).

Listing of EXECUTIVE = RUNP5B

```
&TYPE OFF
&COMMENT INTERACTIVE EDIT OF THE ROE SEARCH RESULTS.
&COMMENT SINGLE ENTRY PARM REQUIRED IS PROJECT NAME.
&COMMENT OUTPUT ONLY TO TERMINAL IN THIS SEQUENCE.
GIF &INDEX EO 1 &GOTO -STP1
EXEC NONAME RUNP5B
-STP1 RALPHA3 = G1
EXEC IFNAME
EXEC JCL
EXEC JCL2
LOAD P5B (CLEAR LIBE) PLILIB
START (BRIEF)
FILEDEF #: CLEAR
GEXIT
```

Program P5B Specific User Instructions

Program P5B processes only those files associated with a unique ETAM project during a single run. It is mandatory that the name of the Specific project be provided by the User when invoking the RUNP5B executive. On the sample runs to follow, this is illustrated using the ETAM project name "PN123".

After initialization, Program P5B returns to the following general prompt message:

```
** MAKE NEXT ACTION REQUEST
```

This invitation for entry can be followed by one of the following entries:

- DELETE n1
- DELETE n1 n2
- PRINT n1
- PRINT n1 n2
- ADDC /cin/cdp/
- ADDV /stocknumber/
- ADDT /rate/rank/jobtask/
- QUIT
The "DELETE" entries allow deletion of one or a range of entities in the search results file. The numbers "n1", "n2" refer to the numbered records in the printer listing of the search results as provided by Program P5A.

The "PRINT" commands provide a printed output of the record(s) indicated.

To add an entity not currently in the search results, the "ADDC", "ADDV", and "ADDT" entries are used for adding a Course, Vehicle, or Task, respectively. The Vehicle "stocknumber" must be present for an "ADDV" entry. For an "ADDC" entry, only the "cin" portion is required. When only "cin" is provided, all Courses bearing that "cin" designation will be selected from the Abbreviated Course data base. For the "ADDT" entry, only a single field is required. To match entries in the Abbreviated Tasks data base, all three must be provided.

The "QUIT" option rewrites the RESU file (File Number 31) with all entries - original, deleted, new (added). When printing the RESU file using Program P1, the status of each entry will be indicated. The Extract Results file (File Number 32) is composed by Program P5B. Only original and added entities are copied into this file.

Program P5B Error Messages

The following error text results from an illegal response to the general prompt message:

** CANNOT DECODE LAST ENTRY - TRY AGAIN

(followed by the general prompt message)

Sufficient internal table space has been reserved within Program P5B to allow a total of 500 new entity additions. Should an attempt be made to add more than this total, the following results:

** A TOTAL OF 500 NEW RECORDS HAVE BEEN ADDED
ANOTHER RUN OF THIS ROUTINE MUST BE MADE TO ENTER FURTHER ADDITIONS.

At this point, if the "QUIT" option is elected, the Program will rewrite the results files and terminate. A second invocation of Program P5B will then allow 500 more additions to be made.

When entering record numbers in conjunction with the "DELETE" and "PRINT" options, two errors can occur. One is the addressing of a record that is beyond the limits of the results file RESU. A second error can occur when using a range-type entry, e.g., DELETE n1 n2. If n1 is greater in value than n2, or either is outside the bounds of the RESU file, the following common message results:

** RANGE ERROR IN RECORD NUMBER
Program P5B Sample Run Output

The following terminal output sample illustrates the common entries in the use of Program P5B.

RUNP5B PN123

PROJECT NAMED PN123 ALREADY EXISTS, IS THIS CORRECT? (RESPOND YES OR QUIT)

YES

::: ROE SEARCH RESULTS EDIT FOR PROJECT PN123
::: 3-D PROCEDURAL TRAINER

::: 61 HAVE BEEN LOADED INTO MEMORY FOR EDITING

::: MAKE NEXT ACTION REQUEST
DELETE 1 3
DELETED RECORD NUMBER : 1 THRU 3

::: MAKE NEXT ACTION REQUEST
DELETE 23
DELETED RECORD NUMBER : 23

::: MAKE NEXT ACTION REQUEST
DELETE 38 49
DELETED RECORD NUMBER : 38 THRU 49

::: MAKE NEXT ACTION REQUEST
DELETE 53
DELETED RECORD NUMBER : 53

::: MAKE NEXT ACTION REQUEST
DELETE 55
DELETED RECORD NUMBER : 55

::: MAKE NEXT ACTION REQUEST
ADD /A1010108/

THE FOLLOWING RECORD(S) HAS/HAVE BEEN ADDED...

RECNO/CIN/CDP/COURSE : 62/A10101087654/ WRA-4 CMB MA
TAEG REPORT NO. 40

*** MAKE NEXT ACTION REQUEST
PRINT 1 61

RECNO/CIN/CDP/TITLE : 1/A1020060/ / (NO MATCHING DATA)
RECNO/CIN/CDP/TITLE : 2/A1020093/ / (NO MATCHING DATA)
RECNO/CIN/CDP/TITLE : 3/A1020095/ / (NO MATCHING DATA)
RECNO/CIN/CDP/TITLE : 4/A1930050/2213/ INERT NAV PRINC
RECNO/CIN/CDP/TITLE : 5/A2420010/6529/ IS A
RECNO/CIN/CDP/TITLE : 6/A4120010/6078/ EA-A
RECNO/CIN/CDP/TITLE : 7/A4120010/6287/ EA A

... Output interrupted at this point ...

RECNO/CIN/CDP/TITLE : 58/N7010320/ / (NO MATCHING DATA)
RECNO/CIN/CDP/TITLE : 59/Q220015/ / (NO MATCHING DATA)
RECNO/CIN/CDP/TITLE : 60/8300010/ / (NO MATCHING DATA)
RECNO/CIN/CDP/TITLE : 61/8300012/ / (NO MATCHING DATA)

*** MAKE NEXT ACTION REQUEST
QUIT

*** ROE SEARCH RESULTS EDITOR IS TERMINATING

TOTAL RECORDS ADDED : 1
TOTAL RECORDS IN PROJECT EXTRACT FILE : 44
Internal Filename (DDNAME) | Usage | Refer To File Reference Number | Media | External NCSS Filename/Filetype | Format
--- | --- | --- | --- | --- | ---
SYSPRINT | OUTPUT | N/A : THIS IS THE OFFLINE PRINTER | | | F, 121, 121
REE | INPUT | 32 | DASD | Projectname/REE | FB, 16, 800
ID | INPUT | 26 | DASD | Projectname/ID | F, 80, 80
MREF | INPUT | 1 | DASD | ETAM/MREF | F, 80, 80
COURSE | DIRECT | 7 | DASD | ETAM/COURSE | F, 80, 80
DCIN | INPUT | 8 | DASD | ETAM/DCIN | F, 760, 760
DCDP | INPUT | 9 | DASD | ETAM/DCDP | F, 1200, 1200
VEHS | DIRECT | 15 | DASD | ETAM/VEHS | F, 80, 80
DVEH | INPUT | 16 | DASD | ETAM/DVEH | F, 750, 750
TASKS | DIRECT | 22 | DASD | ETAM/TASKS | F, 80, 80
DTASK | INPUT | 23 | DASD | ETAM/DTASK | F, 1500, 1500

Discussion

Program P5C is the last step in the Range-of-Effects (ROE) search process. The edited results of a search are sorted and printed. The record sort is executed at the executive level in executive SORT5C.
Listing of EXECUTIVE = RUNP5C

@TYPE OFF
&COMMENT ROE EXTRACT SEARCH RESULTS PRINTED HEREIN.
&COMMENT SINGLE ENTRY PARM REQUIRED - PROJECT NAME
&COMMENT EXEC = DEVICE WILL PROMPT FOR OUTPUT DEVICE
&IF &INDEX EO 1 &GOTO -STP1
EXEC NONAME RUNP5C
-STP1 &ALPHA3 = &1
EXEC IFNAME
&SPACE
EXEC DEVICE
&SPACE
EXEC JCL
EXEC JCL2
EXEC SORT5C
FILEDEF SYSPRINT PTR RE FA LP 121 BL 121
LOADMOD P5C
START IHECMS &ALPHA1 (BRIEF)
FILEDEF := CLEAR
&EXIT

Listing of EXECUTIVE = SORT5C

@TYPE OFF
&COMMENT SORT SUPPORT FOR ROE RESULTS FN/REE PROJECT FILE.
&COMMENT NO EXPLICIT ENTRY PARMS REQUIRED FOR THIS EXEC.
&COMMENT ALPHA3 GLOBAL CONTAINS PROJECT FILENAME ON ENTRY.
&GDSN = . ! ! &ALPHA3
STATE &GDSN REE P
&IF &INDEXX NE & &GOTO -STPS1
ERASE &GDSN REE P
-STPS1 STATE &ALPHA3 REE P
&IF &INDEXX EO & &GOTO -STPS2
&SPACE
&PRINT PROJECT FILE &ALPHA3 / REE DOES NOT EXIST!
-SERP &SPACE
&PRINT EXEC SEQUENCE RUNP5C IS TERMINATING ABNORMALLY...
&SPACE
&QUIT
-STPS2 &STACK 1 1 4 16
&STACK KT
SORT &ALPHA3 PEE (BPIEF)
&GRET = &INDEXX
&PRINT &GRET &INDEXX &ALPHA3
&SPACE
&IF &GRET EO & &GOTO -STPS3

D.2-28
Program P5C Specific User Instructions

Each run of Program P5C is associated with a unique ETAM project results file. Accordingly, the name of the project of concern must be provided by the User when invoking the RUNP5C executive. On the sample run to follow, this has been accomplished using the project name "PN123".

Program P5C Error Messages

The following error message text can result from running Program P5C:

** ERROR NUMBER XX HAS OCCURED **

(followed by a general error data output line)

This message is significant for only the following error number "XX":

01 - The REE input file (File Number 32) cannot contain deleted entities. This error occurs if such an entity is passed into the REE file by Program P5B.

Program P5C Sample Run Output

The following is a complete run of Program P5C for project "PN123".

RUNP5C PN123

PROJECT NAMED PN123 ALREADY EXISTS,
IS THIS CORRECT? (RESPOND YES OR QUIT)

YES
OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT
BOTH

$88 $8 PN123

EXECUTION:

**EXTRACT SEARCH RESULTS FOR PROJECT - PN123**
3-D PROCEDURAL TRAINER

<table>
<thead>
<tr>
<th>PAGE NUMBER</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 ORIGINAL</td>
<td>A19335</td>
</tr>
<tr>
<td>2 ORIGINAL</td>
<td>A24231</td>
</tr>
<tr>
<td>3 ORIGINAL</td>
<td>A41201</td>
</tr>
<tr>
<td>4 ORIGINAL</td>
<td>A41271</td>
</tr>
<tr>
<td>5 ORIGINAL</td>
<td>A43131</td>
</tr>
<tr>
<td>6 ORIGINAL</td>
<td>A431314</td>
</tr>
<tr>
<td>7 ORIGINAL</td>
<td>A433319</td>
</tr>
<tr>
<td>8 ORIGINAL</td>
<td>A433025</td>
</tr>
<tr>
<td>9 ORIGINAL</td>
<td>A491314</td>
</tr>
<tr>
<td>10 ORIGINAL</td>
<td>A551019</td>
</tr>
<tr>
<td>11 ORIGINAL</td>
<td>A551027</td>
</tr>
<tr>
<td>12 ORIGINAL</td>
<td>A551068</td>
</tr>
<tr>
<td>13 ORIGINAL</td>
<td>A580016</td>
</tr>
<tr>
<td>14 ORIGINAL</td>
<td>A651020</td>
</tr>
<tr>
<td>15 ORIGINAL</td>
<td>A652051</td>
</tr>
<tr>
<td>16 ORIGINAL</td>
<td>A652068</td>
</tr>
<tr>
<td>17 ORIGINAL</td>
<td>A670011</td>
</tr>
<tr>
<td>18 ORIGINAL</td>
<td>A70327</td>
</tr>
<tr>
<td>19 ORIGINAL</td>
<td>A72913</td>
</tr>
<tr>
<td>20 ORIGINAL</td>
<td>C153722</td>
</tr>
<tr>
<td>21 ORIGINAL</td>
<td>C153722</td>
</tr>
<tr>
<td>22 ORIGINAL</td>
<td>C153722</td>
</tr>
<tr>
<td>23 ORIGINAL</td>
<td>C153722</td>
</tr>
<tr>
<td>24 ORIGINAL</td>
<td>C153722</td>
</tr>
<tr>
<td>25 ORIGINAL</td>
<td>C153722</td>
</tr>
<tr>
<td>26 ORIGINAL</td>
<td>C153834</td>
</tr>
<tr>
<td>27 ORIGINAL</td>
<td>C1023793</td>
</tr>
<tr>
<td>28 ORIGINAL</td>
<td>C1023793</td>
</tr>
<tr>
<td>29 ORIGINAL</td>
<td>C1213111</td>
</tr>
<tr>
<td>30 ORIGINAL</td>
<td>C1213111</td>
</tr>
<tr>
<td>31 ORIGINAL</td>
<td>C603472</td>
</tr>
<tr>
<td>32 ORIGINAL</td>
<td>C6023536</td>
</tr>
</tbody>
</table>
TAEG REPORT NO. 40

33 ORIGINAL C6463103 7365 CVA/CV ALW SUPV
34 ORIGINAL J2210357 538Q MULTI-THREAT TNG
35 ORIGINAL J2330203 2637 ADV EW OP'S CRSE
36 ORIGINAL J2430974 2181 IPC
37 ORIGINAL K2330666 215Y SUB EW OP-ADV
38 ORIGINAL L6610566 NONE (NO ABBREVIATED FILE DATA AVAILABLE)
39 ORIGINAL M198106E NONE (NO ABBREVIATED FILE DATA AVAILABLE)
40 ORIGINAL N7100209 NONE (NO ABBREVIATED FILE DATA AVAILABLE)
41 ORIGINAL Q 2C0015 NONE (NO ABBREVIATED FILE DATA AVAILABLE)
42 ORIGINAL R300115 NONE (NO ABBREVIATED FILE DATA AVAILABLE)
43 ORIGINAL R300120 NONE (NO ABBREVIATED FILE DATA AVAILABLE)
44 ADDED A101010U 7654 (NO ABBREVIATED FILE DATA AVAILABLE)

""" NOW AT LOF ON SEARCH RESULTS FILE """

TOTAL NUMBER COURSE RECORDS PROCESSED : 44
TOTAL NUMBER VEHICLE RECORDS PROCESSED : 9
TOTAL NUMBER JOBTASK RECORDS PROCESSED : 9
TOTAL NUMBER ROE RESULT RECORDS READ : 44
RECORDS WITHOUT MATCHING ABBREVIATED DATA : 7
TOTAL NUMBER DIRECTORY SEARCH FAILURES : 9
TAEG REPORT NO. 40

PROGRAM NAME : P7
ENTRY POINT : P7
RUN EXECUTIVES(S) : RUNP7
USER OUTPUT VIA : Terminal Only
USER PROMPTING : None
FUNCTION : Makeup Course CIN Directory (File Number 8) and CDP Directory (File Number 9)

NOTE : This program contains an internal SORT invocation

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MREF</td>
<td>INOUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>COURSE</td>
<td>INPUT</td>
<td>7</td>
<td>DASD</td>
<td>ETAM/COURSE</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DCIN</td>
<td>OUTPUT</td>
<td>8</td>
<td>DASD</td>
<td>ETAM/DCIN</td>
<td>F, 760, 760</td>
</tr>
<tr>
<td>DCDP</td>
<td>OUTPUT</td>
<td>9</td>
<td>DASD</td>
<td>ETAM/DCDP</td>
<td>F, 1200, 1200</td>
</tr>
<tr>
<td>SORTWK01</td>
<td>SORT</td>
<td>N/A</td>
<td>DASD</td>
<td>(TEMP)</td>
<td></td>
</tr>
<tr>
<td>SORTWK02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SORTWK03</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Discussion
This module constructs the two directories associated with the Abbreviated Course file. The Course file itself is used as the primary input. The directories will consist of Course CDP numbers and record pointers in one file and Course CIN numbers and pointers in the other.

Each directory is sorted by its key field (CIN or CDP); a PL/1 invocation of the standard PL/1 sort package is included within Program P7. Some of the messages output from this module are the standard Sort/Merge messages and warnings.
Listing of EXECUTIVE = RUNP7

** TYPE OFF**

**COMMENT MAKEUP OF COURSE DIRECTORIES (CIN AND CDP)**

**COMMENT NO ENTRY PARMS**

FILEDEF COURSE DSK ETAM COURSE RE F LR 8@ BL 8@
FILEDEF MREF DSK ETAM MREF RE F LR 8@ BL 8@
FILEDEF DCIN DSK ETAM DCIN RE F LR 76@ BL 76@
FILEDEF DCDP DSK ETAM DCDP RE F LR 12@ BL 12@
ATTACH TEMPS AS 192 BRIEF
FILEDEF SORTWK@1 DSK W1 SORT T
FILEDEF SORTWK@2 DSK W2 SORT T
FILEDEF SORTWK@3 DSK W3 SORT T
LOAD P7 (CLEAR LIBE) PLILIB
START (BRIEF)
FILEDEF " CLEAR
DETACH 192 BRIEF
EXIT

Program P7 Error Messages

In the event of an error detected by the Sort/Merge package:

** SORT RETURN CODE INVALID, CODE WAS 16 **

COURSE DIRECTORY CONSTRUCT PROGRAM IS NOW TERMINATING

The following message strings are caused by overflow of the internal table allotted for directories. The current implementation allots a CIN directory maximum of 304 entries, and a CDP directory max of 1200 entries. One of the message sequences below will result in the event of overflow:

**** CIN TABLE OVERFLOW ****
/or/

**** CDP TABLE OVERFLOW ****

DIRECTORY CONSTRUCT PROGRAM IS TERMINATING DUE TO ERROR
** SORT RETURN CODE INVALID, CODE WAS 8 **

COURSE DIRECTORY CONSTRUCT PROGRAM IS NOW TERMINATING

Program P7 Specific User Instructions

None.
Program P7 Sample Run Output

The following represents an actual run of P7. The output is shown in its entirety. Note that this output is directed only to the online terminal printer device. The output below represents the actual contents of both the Course CIN and CDP directories in the current ETAM implementation.

**RUNP7**

$  

ABBREVIATED COURSE DATABASE CIN/CDP DIRECTORY  
CONSTRUCT PROGRAM  

IER036I - B = 363  
IER037I - G = 543  
IER038I - NMAX = 11970  

**EOF ON COURSE INPUT FILE ; SORT PHASE START**  

IER045I - END SORT PH  
IER049I - SKIP MERGE PH  

**SORT PHASE COMPLETE ; STARTING CIN DIRECTORY MAKEUP**  

<table>
<thead>
<tr>
<th>CIN NUM</th>
<th>CIN</th>
<th>TOTAL CDPS</th>
<th>REC PTR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A 000111</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>A 2E0013</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>A 2G0014</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>A 8C0015</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>A1H10108</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>A1930050</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>A2420019</td>
<td>1</td>
<td>8</td>
</tr>
<tr>
<td>8</td>
<td>A4120019</td>
<td>2</td>
<td>9</td>
</tr>
<tr>
<td>9</td>
<td>A4310011</td>
<td>1</td>
<td>11</td>
</tr>
<tr>
<td>10</td>
<td>A4310014</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>11</td>
<td>A4330019</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>12</td>
<td>A4330025</td>
<td>1</td>
<td>14</td>
</tr>
<tr>
<td>13</td>
<td>A4910014</td>
<td>1</td>
<td>15</td>
</tr>
<tr>
<td>14</td>
<td>A5000021</td>
<td>2</td>
<td>17</td>
</tr>
<tr>
<td>15</td>
<td>A5000028</td>
<td>2</td>
<td>19</td>
</tr>
<tr>
<td>16</td>
<td>A5000032</td>
<td>4</td>
<td>23</td>
</tr>
<tr>
<td>17</td>
<td>A5100012</td>
<td>1</td>
<td>24</td>
</tr>
<tr>
<td>18</td>
<td>A5320015</td>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td>19</td>
<td>A5420014</td>
<td>1</td>
<td>26</td>
</tr>
<tr>
<td>20</td>
<td>A5510019</td>
<td>1</td>
<td>27</td>
</tr>
<tr>
<td>21</td>
<td>A5510027</td>
<td>1</td>
<td>28</td>
</tr>
<tr>
<td>22</td>
<td>A5510068</td>
<td>1</td>
<td>29</td>
</tr>
<tr>
<td>23</td>
<td>A5700010</td>
<td>1</td>
<td>30</td>
</tr>
<tr>
<td>24</td>
<td>A5800016</td>
<td>1</td>
<td>31</td>
</tr>
<tr>
<td>25</td>
<td>A6100010</td>
<td>2</td>
<td>33</td>
</tr>
<tr>
<td>26</td>
<td>A6100027</td>
<td>2</td>
<td>35</td>
</tr>
<tr>
<td>27</td>
<td>A6520050</td>
<td>1</td>
<td>36</td>
</tr>
<tr>
<td>28</td>
<td>A6520068</td>
<td>1</td>
<td>37</td>
</tr>
<tr>
<td>29</td>
<td>A6700011</td>
<td>1</td>
<td>38</td>
</tr>
<tr>
<td>30</td>
<td>A6700025</td>
<td>1</td>
<td>39</td>
</tr>
<tr>
<td>31</td>
<td>A7010027</td>
<td>2</td>
<td>41</td>
</tr>
<tr>
<td>32</td>
<td>A7020024</td>
<td>1</td>
<td>42</td>
</tr>
<tr>
<td>CDP Num</td>
<td>CDP</td>
<td>REC PTR</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>---------</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0133</td>
<td>27</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0134</td>
<td>29</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0143</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0133</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0327</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1034</td>
<td>195</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>1035</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>TAEG REPORT NO. 40</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1Ø36</td>
<td>12</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>2Ø1Ø</td>
<td>2Ø</td>
<td></td>
</tr>
<tr>
<td>1Ø</td>
<td>2Ø2M</td>
<td>21</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>2Ø4E</td>
<td>8Ø</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>2Ø4G</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>2Ø4Ø</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>2Ø5W</td>
<td>97</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>2Ø5Y</td>
<td>11Ø</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>2Ø82</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>2121</td>
<td>39</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>2131</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>2181</td>
<td>1Ø2</td>
<td></td>
</tr>
<tr>
<td>2Ø</td>
<td>2213</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>2374</td>
<td>57</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>239Ø</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>2576</td>
<td>61</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>26Ø1</td>
<td>22</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>2637</td>
<td>96</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>265V</td>
<td>1Ø3</td>
<td></td>
</tr>
<tr>
<td>2Ø</td>
<td>266C</td>
<td>1Ø4</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>2667</td>
<td>81</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>2671</td>
<td>82</td>
<td></td>
</tr>
<tr>
<td>3Ø</td>
<td>2717</td>
<td>69</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>2887</td>
<td>49</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>3Ø78</td>
<td>38</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>3192</td>
<td>16</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>3193</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>32Ø2</td>
<td>4Ø</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>32Ø5</td>
<td>43</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>321X</td>
<td>7Ø</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>321Z</td>
<td>71</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>322B</td>
<td>72</td>
<td></td>
</tr>
<tr>
<td>4Ø</td>
<td>322D</td>
<td>73</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>322E</td>
<td>74</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>3249</td>
<td>3Ø</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>337M</td>
<td>41</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>34ØC</td>
<td>76</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>341D</td>
<td>83</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>346F</td>
<td>44</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>346L</td>
<td>65</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>349H</td>
<td>84</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>349K</td>
<td>85</td>
<td></td>
</tr>
<tr>
<td>5Ø</td>
<td>351J</td>
<td>42</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>351Q</td>
<td>45</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>3931</td>
<td>28</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>4Ø2Ø</td>
<td>46</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>4138</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>451Ø</td>
<td>68</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>461W</td>
<td>19</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>463E</td>
<td>37</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>466Z</td>
<td>47</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>47ØP</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>6Ø</td>
<td>4717</td>
<td>26</td>
<td></td>
</tr>
</tbody>
</table>
TAEG REPORT NO. 40

61 4064 75
62 516U 4
63 5224 36
64 525U 98
65 526K 91
66 528E 88
67 528H 92
68 528J 93
69 5324 99
70 533P 94
71 533Z 100
72 534A 80
73 534Z 109
74 538P 90
75 538Q 95
76 539J 23
77 539Q 35
78 548M 58
79 548N 59
80 542N 101
81 544L 66
82 5535 5
83 5599 108
84 6053 31
85 6057 24
86 6069 32
87 6078 9
88 626B 33
89 6278 67
90 6287 10
91 6529 8
92 7365 86
93 7529 62
94 7540 77
95 7541 78
96 7654 6
97 7914 63
98 7915 64
99 7938 79
100 8178 106
101 9057 51
102 9059 51
103 9010 52
104 9133 54
105 9051 55
106 9162 61
107 9279 87
108 9238 53
109 9749 107
110 9750 48
111 9751 56
IERS551 - INSERT 222, DELETE 222
IERS541 - RCD IN OUT
IERS521 - EOJ
*** SORT COMPLETED OK, CODE = 0 ***

*** CIN DIRECTORY HAS BEEN SUCCESSFULLY REWRITTEN

*** CDP DIRECTORY HAS BEEN SUCCESSFULLY REWRITTEN

COURSE DIRECTORY CONSTRUCT PROGRAM IS NOW TERMINATING

D.2-37
PROGRAM NAME : P7A
ENTRY POINT : P7A
RUN EXECUTIVES(S) : RUNP7A
USER OUTPUT VIA : Terminal Only
USER PROMPTING : None

FUNCTION : Load Course Descriptor File (File Number 6)

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MREF</td>
<td>INOUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DIND</td>
<td>DIRECT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DESC</td>
<td>OUTPUT</td>
<td>6</td>
<td>DASD</td>
<td>ETAM/DESC</td>
<td>F, 213, 213</td>
</tr>
<tr>
<td>DCARD</td>
<td>INPUT</td>
<td>5</td>
<td>DASD</td>
<td>ETAM/SCDESC</td>
<td>F, 80, 80</td>
</tr>
</tbody>
</table>

Discussion

The output of this program is the Course Descriptor file (File Number 6). Note that the same descriptor card input file (File Number 5) was used in Program P14 to select the appropriate Course file CIN numbers. Accordingly, there will be no errors of mismatched CIN numbers in the run of this Program P7A.

Listing of EXECUTIVE = RUNP7A

```
&TYPE OFF
&COMMENT COURSE DESCRIPTORS FILE LOAD
&COMMENT NO ENTRY PARM
FILEDEF MREF DSK ETAM MREF RE F LR 80 BL 80
FILEDEF DIND DSK ETAM DIND RE F LR 80 BL 80
FILEDEF DCARD DSK ETAM SCDESC RE F LR 80 BL 80
FILEDEF DESC DSK ETAM DESC RE F LR 213 BL 213
LOAD P7A (CLEAR LIBE) PLILIB
START (BRIEFL)
FILEDEF " CLEAR
&EXIT
```

D.2-38
Program P7A Error Messages

The only error diagnosed by this program is that of an invalid numeric subscript occurring on an input card. This error is detected by comparing the given descriptor numbers with those indicated as legal in the Master Descriptor Index file (File Number 3). The following error message results in the event of a mis-compare; this error condition is not considered terminal, and the program continues with the next subscript for consideration.

** ERROR IN CIN -12345678- DESCRIPTOR SUBSCRIPT 999 DOES NOT EXIST **

Program P7A Specific User Instructions

None.

Program P7A Sample Run Output

The following represents an actual run of Program P7A. The output is not shown in its entirety. Note that this output is directed only to the online terminal printer device. The output shown represents the current contents of the Course Descriptor file.

RUNP7A

COURSE FILE DESCRIPTOR LOAD PROGRAM

CIN ASSIGNED DESCRIPTOR(S)

A 49$111 61 63 $4 11 12 13 2$ 21 22 33 42 45 48 53 65 66 62
A 2E$13 61 62 63 1$ 21 33 65 8$ 82
A 2G$14 61 62 63 12 33 41 42 44 45 48 52 53 65 69 82 84
A 4A$25 63 12 21 22 33 6$ 82
A 8C$15 61 65 12 2$ 32 8$ 82
A1$1$6 3$ 31 71 8$ 85
A1$2$6 3$ 31 64 65 71 85
A1$2$93 3$ 82
A1$2$95 5$ 11 12 13 2$ 21 31 64 65 82 83
A1$3$5 5$ 31 8$ 85
A2$1$1 5$ 2$ 21 22 3$ 32 63 64 65 66 85
A2$4$1 5$ 1$ 11 12 13 32 6$ 62 63 66 8$ 85

(Completion of text output presented on following page)
NOTE - Majority of text output has been eliminated for the sake of brevity. The last few lines of Program P7A output are as shown below.

```
K @2@4 01 03 04 11 12 13 20 22 32 49 65 66 82 84
K 2E1@78 01 02 03 05 10 12 21 33 42 45 49 53 65 66 72 82
K221@42 01 05 20 30 42 45 53 63 64 82 83
K222@35 02 03 20 30 41 44 49 52 61 62 64 67 69 72 86
K233@66 01 02 03 05 13 14 20 30 41 44 49 52 61 62 63 64 81 84
L1@1@24 05 80
L661@56 03 11 12 13 20 21 31 40 44 46 49 52 62 63 64 65 85
M1981@E 03 11 20 31 40 44 46 49 52 64 65 84 85
N1@555 05 11 12 13 20 21 31 42 45 48 53 64 65 71 84
N7@1@20 03 05 10 11 12 13 20 31 40 44 46 49 51 61 62 63 64 85
P @33@8 01 02 03 04 11 12 20 21 22 33 42 45 48 53 65 66 82 83
Q 2C@15 03 11 12 13 20 30 40 43 46 49 51 64 67 68 85
R21@61@4 05 20 20 32 42 45 48 64 83 89
S5@@29 01 02 03 04 10 12 13 20 32 44 65 66 84
83@@1@0 02 03 05 10 11 12 13 20 30 62 63 64 65 81 84
83@@1@2 03 04 05 11 12 13 20 21 30 32 62 63 64 65 69 81 83
```

Now at EOF on descriptor card input.

TOTAL NUMBER DESCRIPTR INPUT CARDS: 132
TOTAL UNIQUE CIN NUMBERS IN INPUT: 122
TOTAL NUMBER OUTPUT DESCRIPTR RECORDS: 122
TOTAL NUMBER CIN OR SUBSCRIPT ERRORS: 0

COURSE DESCRIPTOR LOAD PROGRAM IS TERMINATING
PROGRAM NAME: P7B

ENTRY POINT: IHECMS (Parameter Supplied by Executive RUNP7B at Entry)

RUN EXECUTIVES(S): RUNP7B, DEVICE, JCL

USER OUTPUT VIA: Terminal, Offline Printer, or Both

USER PROMPTING: At Executive Level:
(1) Output Device Select
(2) Print Option (A, B, or C) Select

FUNCTION: Print Data Contained Within or Associated With the Abbreviated Course File (File Number 7)

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPTR</td>
<td>OUTPUT</td>
<td>N/A : THIS IS THE OFFLINE PRINTER</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>FA, 121, 121</td>
</tr>
<tr>
<td>MREF</td>
<td>INPUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/COURSE</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>COURSE</td>
<td>DIRECT</td>
<td>7</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>INDESC</td>
<td>INPUT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DCDP</td>
<td>F, 1200, 1200</td>
</tr>
<tr>
<td>DCDP</td>
<td>INPUT</td>
<td>9</td>
<td>DASD</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Discussion

This module serves to provide printer dumps of the contents of the Abbreviated Course file. Due to the size of the output, offline print is recommended as the most economical method.

Listing of EXECUTIVE = RUNP7B

@TYPE OFF
@COMMENT G.P. PRINT EXEC FOR COURSES
@COMMENT ENTRY/DEVICE PROMPTS DONE HERE AND IN EXEC = DEVICE
@SPACE
@PRINT ABBR. COURSE FILE PRINT PROGRAM IS STARTING
@SPACE
@PRINT ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
@PRINT A = PRINT ALL COURSES, SORTED BY CIN
@PRINT B = PRINT ALL COURSES IN CDP ORDER
@PRINT C = LIST DESCRIPTORS ASSIGNED TO COURSE DATA

D.2-41
Program P7B Specific User Instructions

As shown in the samples following, the User will be prompted for both output device and type of output. The type of output is categorized by the following options in this program:

<table>
<thead>
<tr>
<th>Option</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Print the contents of each Abbreviated Course file record; the output is to be ordered by CIN number.</td>
</tr>
<tr>
<td>B</td>
<td>Print the same contents; output order is by CDP number.</td>
</tr>
<tr>
<td>C</td>
<td>Print the descriptors associated with Courses; the data for this output is retrieved from the Master Descriptor Index file (File Number 3).</td>
</tr>
</tbody>
</table>

NOTE - To print each of the Course CIN numbers and the actual descriptors assigned to each, use Option "C" of Program P20.
Program P7B Error Messages

The following error message can result during Program P7B execution:

** ERROR XX OCCURED ; NREC WAS YYYYY **

In this message, "XX" is the error number as explained below. The quantity "YYYYY" is the number of the input record on which the error was detected.

<table>
<thead>
<tr>
<th>XX</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>Print Option &quot;B&quot; was being processed. The CDP Directory File indicated that a record was to be accessed from the Abbreviated Course file. The record number of the new record was outside the bounds of the Course file. Note that the maximum number of allowed records appearing in the Course file is maintained in the first count (subscript = 1) of the Master Reference file (File Number 1).</td>
</tr>
<tr>
<td>02</td>
<td>Print Option &quot;C&quot; processing. Master Reference file (subscript = 4) contains the maximum usable record in the Descriptor Index (File Number 3) file. An attempt was made to read a record beyond this allowable limit.</td>
</tr>
</tbody>
</table>
Program P7B Sample Run Output - Print Option "A"

The following represents a sample of an actual run of Program P7B using Print Option "A". The appearance of the output resulting from Option "B" would be the same, except for ordering of the individual records. Accordingly, a "B" Option sample will not be shown.

RUNP7B

ABBR. COURSE FILE PRINT PROGRAM IS STARTING

ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
A = PRINT ALL COURSES, SORTED BY CIN
B = PRINT ALL COURSES IN CDP ORDER
C = LIST DESCRIPTORS ASSIGNED TO COURSE DATA

A

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT

TERM

$$$$EXECUTION:

PAGE : 1

<table>
<thead>
<tr>
<th>CIN</th>
<th>CDP</th>
<th>COURSE TITLE</th>
<th>NOBC</th>
<th>NEC</th>
<th>RMS</th>
<th>TPC</th>
<th>LEN</th>
<th>THRS</th>
<th>LHRS</th>
<th>ATTR</th>
<th>STPY</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>80111</td>
<td>8327</td>
<td>SWO ADV COMMAND</td>
<td>8000</td>
<td>PDB</td>
<td>8435</td>
<td>40</td>
<td>224</td>
<td>16</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td></td>
<td>PRIORITY :</td>
<td>TYPE CRS :</td>
<td>C2</td>
<td>SERVICE CDE :</td>
<td>1</td>
<td>COST/AOB :</td>
<td>0.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRAPS :</td>
<td>STATUS :</td>
<td>T</td>
<td>STATUS DTE :</td>
<td>8000</td>
<td>METH-I :</td>
<td>L</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A 2E013 | 8313 | TAL PCO/PO ORI | 8000 | 8000 | KBA | 8325 | 12 | 80 | 0 | 0.0 | 0.0 |
| PRIORITY : | 12 | TYPE CRS : | C2 | SERVICE CDE : | 1 | COST/AOB : | 0.00 |
| TRAPS : | STATUS : | STATUS DTE : | 62171 | METH-I : | P |

A 2G014 | 8143 | MINE C/M OFF | 9268 | 8000 | CCB | 84311 | 33 | 120 | 30 | 0.0 | 0.0 |
| PRIORITY : | 25 | TYPE CRS : | C2 | SERVICE CDE : | 1 | COST/AOB : | 0.00 |
| TRAPS : | STATUS : | STATUS DTE : | 72183 | METH-I : | L |

... Output terminated at this point ...
Program P7B Sample Run Output - Print Option "C"

The following is an actual Program P7B run using Print Option "C".

RUNP7B

ABBR. COURSE FILE PRINT PROGRAM IS STARTING

ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
A = PRINT ALL COURSES, SORTED BY CIN
B = PRINT ALL COURSES IN CDP ORDER
C = LIST DESCRIPTORS ASSIGNED TO COURSE DATA

C

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT

TERM

$$EXECUTION:

PAGE : 1

-- COURSES

::: REFERENCE KNOWLEDGE
  01 SYSTEM PURPOSES
  02 ORGANIZATIONAL ROLES
  03 CONTEXTS OF OPERATION
  04 ORGANIZATIONAL RULES
  05 OTHER

::: ENABLING KNOWLEDGE
  10 OPERATIONAL GOAL CRITERIA
  11 NOMENCLATURE, IDENTIFICATION, LOCATION
  12 PROCEDURAL DESCRIPTIONS
  13 JOB RELEVANT FACTS, RULES
  14 OTHER

::: TASK FORMATS
  20 PROCEDURE FORMATS
  21 DECISION FORMATS
  22 CONSTRUCTION FORMATS
  23 OTHER

::: GROSS JOB CATEGORIES
  30 OPERATIONS
  31 MAINTENANCE
  32 SERVICE & ADMINISTRATION
  33 COMMAND
  34 OTHER

D.2-45
OBJECTIVE TASK VARIABLES AS MANIFEST IN THE TRAINING

48 EQUIPMENT & OBJECTS USED: REAL
41 EQUIPMENT & OBJECTS USED: SIMULATED
42 EQUIPMENT & OBJECTS USED: SYMBOLIC
43 ENVIRONMENTS IN WHICH TASK IS TRAINED: REAL
44 ENVIRONMENTS IN WHICH TASK IS TRAINED: SIMULATED
45 ENVIRONMENTS IN WHICH TASK IS TRAINED: SYMBOLIC
46 TOOLS/INSTRUMENTS USED IN TRAINING: REAL
47 TOOLS/INSTRUMENTS USED IN TRAINING: SIMULATED
48 TOOLS/INSTRUMENTS USED IN TRAINING: SYMBOLIC
49 REF/ENABLING INFO IN DOING TASK: APPLIED
51 REF/ENABLING INFO IN DOING TASK: NOT APPLIED
52 CRITERIA OF TASK PERFORMANCE: REAL
53 CRITERIA OF TASK PERFORMANCE: SIMULATED
54 CRITERIA OF TASK PERFORMANCE: SYMBOLIC

TASK FUNCTIONS DOMINANT IN TRAINING

68 GOAL PROJECTION
61 SCAN-DETECT
62 IDENTIFY
64 INTERPRET
65 PROCEDURE FOLLOWING
66 DECIDE
67 CONSTRUCT, PLAN
68 TRACK
69 MOTOR PERFORMANCE
70 INTERPERSONAL INTERACTION
71 RECALL TASK-CYCLE INFORMATION
72 RECALL ENABLING INFORMATION
73 ADAPT IMPROVISATIONALLY/IMPROMPTU

STAGE OF LEARNING

80 ORIENTATION, FAMILIARIZATION
81 TASK NOMENCLATURE, IDENTS, LOCATIONS, FACTS, RULES
82 TASK FORMATS AT CONCEPTUAL LEVEL
83 PROCEDURES AT VERBAL LEVEL ONLY
84 TASK COMPONENTS WITH GUIDANCE
85 ENTIRE JOB-TASK PROCEDUREALLY AT BARELY ACCEPTABLE MASTERY
86 HIGHLY PROFICIENT IN WORK CONTEXT
87 UNUSUAL TASK CONDITIONS
88 PERFORMANCE AT KEY MAN LEVEL
89 REFRESHER LEARNING
PROGRAM NAME: P8
ENTRY POINT: P8
RUN EXECUTIVES(S): RUNP8
USER OUTPUT VIA: Terminal Only
USER PROMPTING: None

FUNCTION: Makeup Vehicle Abbreviated Data File, Directory, and Descriptor File (File Numbers 15, 16, and 12, Inclusive)

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MREF</td>
<td>INOUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DIND</td>
<td>DIRECT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>VDATA</td>
<td>INPUT</td>
<td>14</td>
<td>DASB</td>
<td>ETAM/SVDATA</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>VCDESC</td>
<td>INPUT</td>
<td>11</td>
<td>DASD</td>
<td>ETAM/SVCDESC</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DVEH</td>
<td>OUTPUT</td>
<td>16</td>
<td>DASD</td>
<td>ETAM/DVEH</td>
<td>F, 750, 750</td>
</tr>
<tr>
<td>VDESC</td>
<td>OUTPUT</td>
<td>12</td>
<td>DASD</td>
<td>ETAM/VDESC</td>
<td>F, 213, 213</td>
</tr>
<tr>
<td>VEHS</td>
<td>OUTPUT</td>
<td>15</td>
<td>DASD</td>
<td>ETAM/VEHS</td>
<td>F, 80, 80</td>
</tr>
</tbody>
</table>

Discussion

All three files of the Abbreviated Vehicle data base are created by Program P8. This includes the data file (File Number 15), the Vehicle Directory (File Number 16), and the Vehicle Descriptor file (File Number 12).
Listing of EXECUTIVE = RUNP8

@TYPE OFF
@COMMENT MAKE VEH DATA/DIRECTORY/_DESCRIPTOR FILES
@COMMENT NO ENTRY PARM
FILEDEF MREF DSK ETAM MREF RE F LR 8Ø BL 8Ø
FILEDEF DIND DSK ETAM DIND RE F LR 8Ø BL 8Ø
FILEDEF VDATA DSK ETAM VDATA RE F LR 8Ø BL 8Ø
FILEDEF VDESC DSK ETAM VDESC RE F LR 8Ø BL 8Ø
FILEDEF VDESCRIPTION DSK ETAM VDESCRIPTION RE F LR 213 BL 213
FILEDEF VEHS DSK ETAM VEHS RE F LR 8Ø BL 8Ø
FILEDEF DVEH DSK ETAM DVEH RE F LR 75Ø BL 75Ø
LOAD P8 (CLEAR LIBE) PLILD
START (BRIEF)
FILEDEF :: CLEAR
SEXIT

Program P8 Error Messages

Within the data input deck (File Number 11), a given Stock Number can exist only on one card. Since the deck is sorted, duplicates will be signaled by the following message, and then rejected:

** DUPLICATE STOCK NUMBER ON PRECEEDING CARD
RECORD WILL BE IGNORED **

NOTE - See sample output (to follow) for an example of this message.

Within the descriptor input deck, a given numeric descriptor can be illegal. Program P8 verifies the existence of each such descriptor through use of the Master Descriptor Index (File Number 3). If an error is detected, the following message will ensue:

** ERROR IN VEH -1234567890123- DESCRIPTOR SUBSCRIPT
999 DOES NOT EXIST **
Since both the Vehicle data cards and the Vehicle descriptor cards are input to P8, discrepancies can exist within the Stock Numbers. The following messages, if applicable, will be printed after the descriptor file has reached EOF and the EOF message printed.

** THE FOLLOWING DATA FILE VEH NUMBERS HAVE NO MATCH IN THE VEH DESCRIPTOR INPUT **

XXXXXXXXXXXXX
YYYYYYYYYYYYY
...

and/or the following

** THE FOLLOWING DESCRIPTOR CARD VEH NUMBERS HAVE NO MATCH IN THE VEH DATA FILE **

XXXXXXXXXXXXX
YYYYYYYYYYYYY

Either or both of the above error message sequences will be absent in the event of no detected errors.

Program P8 Specific User Instructions

Both the input Vehicle data file (File Number 14) and the Vehicle descriptor card file (File Number 11) must be sorted into ascending order by Stock Number before run of Program P8.

Note that both of these sorts are accomplished within the executive STARTUP.

Program P8 Sample Run Output

The following represents sections of an actual run of Program P8. Content sufficient to illustrate the output message sequence has been maintained.
$S$
**INITIALIZE OF VEHICLE DATA AND DESCRIPTOR FILES**

<table>
<thead>
<tr>
<th>NUM</th>
<th>STOCK NUMBER</th>
<th>DESIGNATOR</th>
<th>DEVICE NOMENCLATURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6910LLC000468</td>
<td>12CD-17</td>
<td>MATH, VECTOR ACCELERATION DEMONSTRATOR T/A</td>
</tr>
<tr>
<td>2</td>
<td>6910LLC000469</td>
<td>12CD-18</td>
<td>MATH, NOMOGRAPHIC PROJECTION DEMONSTRATOR T/A</td>
</tr>
<tr>
<td>3</td>
<td>6910LLC000470</td>
<td>12CD-19</td>
<td>MATH, PROJECTION, DEMONSTRATOR TRAINING AID</td>
</tr>
<tr>
<td>4</td>
<td>6910LLC000471</td>
<td>12CD-21</td>
<td>MATH, ORDER OF DIFFERENTIATION, DEM T/A</td>
</tr>
</tbody>
</table>

... Output interrupted at this point ...

57 69400052719ø1 26A-11 ELECTRONICS FUNDAMENTALS UNITS EFU-1

::: DUPLICATE STOCK NUMBER ON PRECEEDING CARD
RECORD WILL BE IGNORED :::

57 69400052719ø4 26A-11 LABORATORY CIRCUIT ANALYSIS UNIT CAU-1
58 6940005666627 26A-11 VOLTAGE REGULATOR LAB CKT ANAL UNIT CAU-3
59 6940005723ø15 26A-11 RADAR/MICROWAVE SYS CKT ANALYSIS UNIT CAU-5
6ø 6940005723ø18 26A-11 SONAR LAB CIRCUIT ANALYSIS UNIT CAU-6
61 6940006641435 15Z1-1 APS-T3, ULTRASONIC TRAINER
62 6940007123ø47 26A-11 TRANSISTOR LAB CIRCUIT ANALYSIS UNIT CAU-7
63 694000856112 26A-11 SYNCHRO LAB CKT ANALYSIS UNIT CAU-9

::: NOW AT EOF ON DATA CARD INPUT :::
A TOTAL OF 63 INPUT CARDS WERE READ

::: VEHICLE DIRECTORY HAS BEEN SUCCESSFULLY REWRITTEN :::

::: LOAD OF VEHICLE DESCRIPTOR FILE FOLLOWS :::

<table>
<thead>
<tr>
<th>STOCK NUMBER</th>
<th>DEVICE DESCRIPTORS</th>
</tr>
</thead>
<tbody>
<tr>
<td>6910LLC000468</td>
<td>Ø3 11 21 32 45 52 61 71 81</td>
</tr>
<tr>
<td>6910LLC000469</td>
<td>Ø3 11 21 32 45 52 61 71 81</td>
</tr>
<tr>
<td>6910LLC000470</td>
<td>Ø3 11 21 32 45 52 61 71 81</td>
</tr>
</tbody>
</table>

... Output interrupted at this point ...
TOTAL NUMBER DESCRIPTOR CARDS INPUT : 63
TOTAL NUMBER UNIQUE VEHICLE STOCK NUMBERS : 63
TOTAL MATCHES FOUND IN VEHICLE DATA DIRECTORY : 63

VEHICLE DATA/DIRECTORY/DESCRIPTOR LOAD PROGRAM IS NOW TERMINATING
PROGRAM NAME: P8A

ENTRY POINT: IHECMS (Runtime Parameter Supplied by Executive RUNP8A at Entry)

RUN EXECUTIVES(S): RUNP8A

USER OUTPUT VIA: Terminal, Offline Printer, or Both

USER PROMPTING: At Executive Level:
- 1) Output Device Select
- 2) Print Option (A or B) Select

FUNCTION: Print Data Within or Associated With the Abbreviated Vehicle File (File Number 15)

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSPRINT</td>
<td>OUTPUT</td>
<td>N/A : THIS IS THE OFFLINE PRINTER</td>
<td></td>
<td></td>
<td>FA, 121, 121</td>
</tr>
<tr>
<td>MREF</td>
<td>INPUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DIND</td>
<td>DIRECT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DVEH</td>
<td>INPUT</td>
<td>16</td>
<td>DASD</td>
<td>ETAM/DVEH</td>
<td>F, 750, 750</td>
</tr>
<tr>
<td>VEHS</td>
<td>DIRECT</td>
<td>15</td>
<td>DASD</td>
<td>ETAM/VEHS</td>
<td>F, 80, 80</td>
</tr>
</tbody>
</table>

Discussion

The contents of the Abbreviated Vehicle data file are dumped to the online terminal of offline printer (or both) by use of Program P8A.
Listing of EXECUTIVE = RUNP8A

$TYPE OFF
$COMMENT C.P. PRINT EXEC FOR VEHICLES
$COMMENT ENTRY/DEVICE PROMPTS DONE HERE AND IN EXEC = DEVICE
$SPACE
$PRINT ABBR. VEHICLE FILE PRINT PROGRAM IS STARTING
$SPACE
$PRINT ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
$PRINT A = PRINT ALL VEHICLE RECORDS IN SORTED ORDER
$PRINT B = LIST DESCRIPTORS ASSIGNED TO VEHICLE DATA
-STP1 &READ ARGS
  $IF $1 EQ QUIT &QUIT
  $ALPHA0 = $1
  $IF $1 EQ A &GOTO -STP2
  $IF $1 EQ B &GOTO -STP2
  $SPACE
  $PRINT UNABLE TO DETERMINE ENTRY TYPE - TRY AGAIN
  $SPACE
  &GOTO -STP1
-STP2 EXEC DEVICE
  $ALPHA2 = $ALPHA0 !! $ALPHA1
-STP3 $SPACE
EXEC JCL
FILEDEF SYSPRINT PTR RE FA LR 121 BL 121
LOAD P8A (CLEAR LINE) PLILIR
START THECMS $ALPHA2 (BRIEF)
FILEDEF " CLEAR
&EXIT
Program P8A Specific User Instructions

As shown in the samples to follow, the User will be prompted for both output device and type of output. The type of output is categorized by the following options in this program:

<table>
<thead>
<tr>
<th>Option</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Print the contents of each Abbreviated Vehicle file record. The order of output will be the natural order of the file, i.e., Vehicle Stock Number.</td>
</tr>
<tr>
<td>B</td>
<td>Print the descriptors associated with Vehicles. The data for this output is retrieved from the Descriptor Master Index file (File Number 3).</td>
</tr>
</tbody>
</table>

NOTE - To print each of the Vehicle Stock Numbers and the actual descriptors assigned to each unique Stock Number, use the "V" Option of Program P20.

Program P8A Error Messages

The following error message results during Program P8A execution only in the event of a serious logic defect in the program:

** ERROR XX OCCURED ; NREC WAS YYYYY **

In the message, "XX" is the error number as explained below. The quantity "YYYYY" indicates the number of the input record on which the error was detected.

<table>
<thead>
<tr>
<th>XX</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>While processing Print Option &quot;B&quot;, an attempt was made to access a record that was outside the bounds of the Descriptor Master Index file (ETAM/DIND - File Number 3). Note that the maximum number of records present in File Number 3 is indicated in the fourth (subscript = 4) count in the Master Reference File ETAM/MREF.</td>
</tr>
<tr>
<td>02</td>
<td>During Print Option &quot;A&quot; processing, an attempt was made to access a record outside the bounds of the Vehicle data file. Master Reference file (subscript = 7) indicates the maximum record number used in the Vehicle data file (File Number 15).</td>
</tr>
</tbody>
</table>
Program P8A Sample Run Output - Print Option "A"

The following listing is a portion of an actual run of Program P8 using Print Option "A".

RUNP8A

ABBR. VEHICLE FILE PRINT PROGRAM IS STARTING

ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
A = PRINT ALL VEHICLE RECORDS IN SORTED ORDER
B = LIST DESCRIPTORS ASSIGNED TO VEHICLE DATA

A

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT

TERM

$\$\$\$EXECUTION:

PAGE : 1

ITEM STOck Num DESIGNATOR DEVICE
--- ---------- --------------

1 6910LLC6463 12CD-17 MATH, VECTOR ACCELERATION DEMONSTRATOR T/A
2 6910LLC6469 12CD-18 MATH, NOMOGRAPHIC PROJECTION DEMONSTRATOR T/A
3 6910LLC4710 12CD-19 MATH, PROJECTION, DEMONSTRATOR TRAINING AID
4 6910LLC6471 12CD-21 MATH, ORDER OF DIFFERENTIATION, DEM T/A
5 6910LLC4720 12CD-22 MATH, FINITE ROTATION, DEMONSTRATOR T/A
6 6910LLC4730 12CD-24 MATH, DIRECTION COSINE, DEMONSTRATOR T/A
7 6910LLC4740 12CD-25 MATH, SKEW LINE, DEMONSTRATOR TRAINING AID
8 6910LLC9475 3E34-1 106MM RECOILLESS RIFLE, FIRING MECH, TRANSP,

... Output terminated at this point ...

D.2-55
Program P8A Sample Run Output - Print Option "B"

Using Print Option "B", the following represents a sample of the resulting output of Program P8A.

RUNP8A

ABBR. VEHICLE FILE PRINT PROGRAM IS STARTING

ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
A = PRINT ALL VEHICLE RECORDS IN SORTED ORDER
B = LIST DESCRIPTORS ASSIGNED TO VEHICLE DATA

B

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT

TERM

$$EXECUTION:

PAGE : 1

-- VEHICLES

::: VEHICLE TYPES
1 INSTRUCTOR
2 STATIC GRAPHICS
3 ANIMATED GRAPHICS
4 AUDIO
5 PHYSICAL MODELS
6 PROCEDURAL TRAINERS: SYMBOLIC
7 PROCEDURAL TRAINERS: PHYSICAL BUT NON-FUNCTIONAL
8 PROCEDURAL TRAINERS: FUNCTIONAL
9 TASK & SYSTEM SIMULATORS
10 REAL EQUIPMENT ITSELF

::: CLASS OF TRAINING OBJECTIVE
11 REFERENCE KNOWLEDGE
12 KNOWLEDGE, TASK SPECIFIC/ENABLING
13 TASK-SKILL FORMATS
14 SKILL TRAINING

::: VEHICLE PROPERTIES
21 VISUAL
22 AUDITORY
23 KINESTHETIC/VESTIBULAR
24 TACTILE
<table>
<thead>
<tr>
<th>Type of Content Displayed</th>
</tr>
</thead>
<tbody>
<tr>
<td>31 Text-Verbal</td>
</tr>
<tr>
<td>32 Diagrammatic</td>
</tr>
<tr>
<td>33 Abstracted Pictorial Representation</td>
</tr>
<tr>
<td>34 Pictorial Representations</td>
</tr>
<tr>
<td>35 Physical Representations</td>
</tr>
<tr>
<td>36 Other</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type of Presentational Sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>41 Library of Frames or Items</td>
</tr>
<tr>
<td>42 Presentation Sequence Not Applicable</td>
</tr>
<tr>
<td>43 Fixed Sequential Frames or Items</td>
</tr>
<tr>
<td>44 Random Selection of Frame Sequences</td>
</tr>
<tr>
<td>45 Dynamic Change of Content Within Frame</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Selection Source for Sequencing</th>
</tr>
</thead>
<tbody>
<tr>
<td>51 Internal Program</td>
</tr>
<tr>
<td>52 Instructor</td>
</tr>
<tr>
<td>53 Student Choice</td>
</tr>
<tr>
<td>54 Student Performance</td>
</tr>
<tr>
<td>55 Combinations of the Above</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Type of External Control Operated by Student</th>
</tr>
</thead>
<tbody>
<tr>
<td>61 Not Applicable Directly</td>
</tr>
<tr>
<td>62 Artificial or Symbolic Response</td>
</tr>
<tr>
<td>63 Representational Response by Symbolic Selection</td>
</tr>
<tr>
<td>64 Representational Response by Dummy Control Activation</td>
</tr>
<tr>
<td>65 Task-Manipulative Response, Non-Dynamic in Time and Force</td>
</tr>
</tbody>
</table>

Page: 2

66 Task-Manipulative Response, Dynamic and Interactive

<table>
<thead>
<tr>
<th>Feedback Presentation Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>71 Not Applicable</td>
</tr>
<tr>
<td>72 Selects Next Stimulus Item or Sequence</td>
</tr>
<tr>
<td>73 Gives Evaluation of Preceding Response</td>
</tr>
<tr>
<td>74 Selects and Presents Guidance Info</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Response Evaluation Logic</th>
</tr>
</thead>
<tbody>
<tr>
<td>81 Not Internal—Depends on Instructor or Student Evaluation</td>
</tr>
<tr>
<td>82 Evaluation Limited to Student's Immediate Response</td>
</tr>
<tr>
<td>83 Evaluation Extended to a Set of Student Responses</td>
</tr>
<tr>
<td>84 Tolerance Limits on Acceptable Student Response: Fixed</td>
</tr>
<tr>
<td>85 Tolerance Limits on Acceptable Student Response: Variable</td>
</tr>
</tbody>
</table>
PROGRAM NAME : P9  
ENTRY POINT : P9  
RUN EXECUTIVES(S) : RUNP9  
USER OUTPUT VIA : Terminal, Offline Printer, or Both  
USER PROMPTING : Program Level : Output Device Select  
FUNCTION : Makeup Task Abbreviated Data File, Directory, and Descriptor File (File Numbers 22, 23, and 19, incl.)  

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSPRINT</td>
<td>OUTPUT</td>
<td>N/A : THIS IS THE OFFLINE PRINTER</td>
<td>FA, 121, 121</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MREF</td>
<td>INOUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DIND</td>
<td>DIRECT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>TDATA</td>
<td>INPUT</td>
<td>21</td>
<td>DASD</td>
<td>ETAM/STDATA</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>TCDESC</td>
<td>INPUT</td>
<td>18</td>
<td>DASD</td>
<td>ETAM/STDSCR</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DTASK</td>
<td>OUTPUT</td>
<td>23</td>
<td>DASD</td>
<td>ETAM/DTASK</td>
<td>F, 1500, 1500</td>
</tr>
<tr>
<td>TASKS</td>
<td>OUTPUT</td>
<td>22</td>
<td>DASD</td>
<td>ETAM/TASKS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>TDESC</td>
<td>OUTPUT</td>
<td>19</td>
<td>DASD</td>
<td>ETAM/TDESC</td>
<td>F, 213, 213</td>
</tr>
</tbody>
</table>

Discussion

Program P9 serves to generate all three files of the Abbreviated Tasks data base. This includes the data file (File Number 22), the Tasks Directory (File Number 23), and the Tasks Descriptor file (File Number 19).
Listing of EXECUTIVE = RUNP9

&TYPE OFF
&COMMENT LOAD TASK DATA/DIRECTORY/DESCRIPTIONS
&COMMENT NO EXEC ENTRY PARMS
&COMMENT PROGRAM P9 WILL PROMPT FOR OUTPUT DEVICE,
&COMMENT TERMINAL, OFFLINE PRINTER, BOTH, QUIT.
FILEDEF MREF DSK ETAM MREF RE F LR 80
FILEDEF DIND DSK ETAM DIND RE F LR 80 BL 80
FILEDEF TDATA DSK ETAM TDATA RE F LR 80 BL 80
FILEDEF TCDESC DSK ETAM TCDESC RE F LR 80 BL 80
FILEDEF DTASK DSK ETAM DTASK RE F LR 500 BL 500
FILEDEF TASKS DSK ETAM TASKS RE F LR 80 BL 80
FILEDEF TDDESC DSK ETAM TDDESC RE F LR 213 BL 213
FILEDEF SYSPRINT PTR RE FA LR 121 BL 121
LOAD P9 (CLEAR LIBE) PLILIB
START (BRIEF)
FILEDEF ** CLEAR
EXIT

Program P9 Error Messages

Within the data input card deck (File Number 21), a given Rate/Rank/Jobtask Number cannot be duplicated on more than one card. Since this file is sorted by the contents of these data fields, duplicates can be detected by Program P9. In such a case, the following error message text will be output:

** DUPLICATE STOCK NUMBER ON PRECEDING CARD RECORD
WILL BE IGNORED **

Program P9 verifies the legality of each of the descriptors in the descriptor input file (File Number 18). The Master Descriptor Index (File Number 3) is used for this purpose. If an error is detected, the following message is output:

** ERROR IN TASK NUMBER -XXX YY ZZZZZ- DESCRIPTOR
SUBSCRIPT SSS DOES NOT EXIST **

where "XXX" is the Task Rate, "YY" the Rank, and "ZZZZZ" the Jobtask number.
Since both the Task data cards and descriptor cards are input to Program P9, it is possible for a discrepancy to exist within the key fields of each (first 13 characters of each card - includes Rate, Rank, and Jobtask number). If such discrepancies are detected, the occurrence is saved within the Program and the messages illustrated below will be printed following the message signifying end-of-file on the descriptor card input.

** THE FOLLOWING DATA FILE TASK NUMBERS
HAVE NO MATCH IN THE DESCRIPTOR CARD FILE **

XXX XXXXXXXXXXXXX
XXX XXXXXXXXXXXXX
... Etc.

and/or the following

** THE FOLLOWING DESCRIPTOR CARD TASK NUMBERS
HAVE NO MATCH IN THE DATA FILE **

YYY YYYYYYYYYYYY
YYY YYYYYYYYYYYY

Either or both of the above error message sequences will be absent in the event of no detected errors.

Program P9 Specific User Instructions

Both input files - the data file (File Number 21) and the descriptor input file (File Number 18) must be sorted into ascending order by the first 13 characters of each record before run of Program P9.

Note that both of the required sorts are accomplished within the executive STARTUP.
Program P9 Sample Run Output

The following represents sections of an actual run of Program P9. Content sufficient to illustrate the output message sequence has been maintained.

RUNP9

$$$

TASK DATA/DIRECTORY/DESCRIPTION LOAD PROGRAM IS NOW STARTING

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, OR BOTH?
ENTER TERM, PRTR, BOTH, OR QUIT
 TERM

** LOAD OF TASK ABBREVIATED DATA FILE FOLLOWS

<table>
<thead>
<tr>
<th>NUMBER</th>
<th>RATE</th>
<th>RANK</th>
<th>TASK</th>
<th>TASK TITLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>AN</td>
<td>20350</td>
<td>AIRCRAFT UNDER EMERGENCY CONDITIONS</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>AN</td>
<td>20351</td>
<td>IDENTIFY MARKINGS INDICATING DANGEROUS AREAS</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>AN</td>
<td>20352</td>
<td>IDENTIFY TOXIC PROPERTIES OF CLEANING MATERIALS</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>AN</td>
<td>20368</td>
<td>RECOGNIZE ARMED EJECTION SEATS</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>AN</td>
<td>24425</td>
<td>HANDLE AIRCRAFT BATTERIES</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>AN</td>
<td>25352</td>
<td>KNOW NAVAL AIR ARM GENERAL ORGANIZATION</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>AN</td>
<td>25353</td>
<td>IDENTIFY NAMES, FUNCTIONS OF NAVAL AIR GROUPS</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>AN</td>
<td>25354</td>
<td>IDENTIFY NAVAL AIRCRAFT SQUADRONS</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>AN</td>
<td>42353</td>
<td>STAND AIRCRAFT SECURITY WATCH</td>
<td></td>
</tr>
</tbody>
</table>

... Output interrupted at this point ...

The above text will be followed by:

** EOF ON DATA INPUT FILE
CARDS PROCESSED : XX,XXX **

** LOAD OF TASK DESCRIPTOR CARDS FOLLOWS **
(Followed by print of Task Descriptor Cards)

** EOF REACHED ON TASK DESCRIPTOR INPUT FILE
CARDS PROCESSED : XX,XXX **
PROGRAM NAME: P9A

ENTRY POINT: IHECMS (Execution Parameter Supplied by Executive RUNP9A at Entry)

RUN EXECUTIVES(S): RUNP9A

USER OUTPUT VIA: Terminal, Offline Printer, or Both

USER PROMPTING: At Executive Level: (1) Output Device Select
(2) Print Option Select

FUNCTION: Print Data Within or Associated with the Abbreviated Tasks File (File Number 22)

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MREF</td>
<td>INPUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DIND</td>
<td>DIRECT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DTASK</td>
<td>INPUT</td>
<td>23</td>
<td>DASD</td>
<td>ETAM/DTASK</td>
<td>F, 1500, 1500</td>
</tr>
<tr>
<td>TASKS</td>
<td>DIRECT</td>
<td>22</td>
<td>DASD</td>
<td>ETAM/TASKS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>SYSPRINT</td>
<td>OUTPUT</td>
<td>N/A : THIS IS THE OFFLINE PRINTER</td>
<td>FA</td>
<td></td>
<td>FA, 121, 121</td>
</tr>
</tbody>
</table>

Discussion

Program P9A to printer dump the contents of the Abbreviated Tasks data file (File Number 22). At User option, the dump may be directed to either (or both) the online terminal or the offline printer.

Listing of EXECUTIVE = RUNP9A

@TYPE OFF
@COMMENT G.P. PRINT EXEC FOR TASKS
@COMMENT ENTRY/DEVICE PROMPTS DONE HERE AND IN EYEC = DEVICE
@SPACE
@PRINT ABBR. TASKS FILE PRINT PROGRAM IS STARTING
@SPACE
@PRINT ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
@PRINT A = PRINT ALL TASK RECORDS IN SORTED ORDER
@PRINT B = LIST DESCRIPTORS ASSIGNED TO TASK DATA

D.2-62
-STP1 &READ ARGS
   &IF &I EQ QUIT &QUIT
   &ALPHA2 = &I
   &IF &I EQ A &GOTO -STP2
   &IF &I EQ B &GOTO -STP2
   &SPACE
   &PRINT UNABLE TO DETERMINE ENTRY TYPE - TRY AGAIN
   &SPACE
   &GOTO -STP1

-STP2 EXEC DEVICE
   &ALPHA2 = GALPHA0 !! &ALPHA1

-STP3 &SPACE
   EXEC JCL
   FILEDEF SYSPRINT PTR RE FA LR 121 BL 121
   LOAD P9A (CLEAR LIBE) PLILIB
   START IHECMS &ALPHA2 (BRIEF)
   FILEDEF = CLEAR
   &EXIT
**Program P9A Specific User Instructions**

As shown in the samples to follow, the User is prompted to select the output device and the type of output. Within Program P9A, the following output options are available:

<table>
<thead>
<tr>
<th>Option</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Print the contents of the Abbreviated Tasks file. The order of the output will be the natural order of the file, i.e., by the contents of the 13-character field containing the concatenation of Rate, Rank, and Jobtask Number.</td>
</tr>
<tr>
<td>B</td>
<td>Print the descriptors associated with Tasks. The data for this output is retrieved from the Descriptor Master Index file (File Number 3).</td>
</tr>
</tbody>
</table>

NOTE - To command the print of the actual descriptors assigned to each unique task, use the "T" Option of Program P20.

**Program P9A Error Messages**

The following error text can be generated by Program P9A:

** ERROR XX OCCURED ; NREC WAS YYYYY **

In the above message, "XX" is the error number as explained below. The quantity "YYYYY" indicates the number of the input record that was being processed when the error was detected.

<table>
<thead>
<tr>
<th>XX</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>During processing of Print Option &quot;B&quot;, an attempt was made to read a record that was outside the bounds of the Descriptor Master Index file (ETAM/DIND - File Number 3). The Master Reference file (ETAM/MREF - File Number 1) indicates the maximum number of records available in the ETAM/DIND file. This count is available in the fourth (subscript = 4) count of MREF.</td>
</tr>
<tr>
<td>02</td>
<td>While processing Print Option &quot;A&quot;, an attempt was made to access a record outside the bounds of the Tasks data file (File Number 22). The Master Reference file (subscript = 11) indicates the maximum record number that can be accessed in the Tasks data file.</td>
</tr>
</tbody>
</table>
Program P9A Sample Run Output - Print Option "A"

The following listing is an example of the output that results from selection of Print Option "A".

RUNP9A

ABBR. TASKS FILE PRINT PROGRAM IS STARTING

ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
A = PRINT ALL TASK RECORDS IN SORTED ORDER
B = LIST DESCRIPTORS ASSIGNED TO TASK DATA

A

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT
TERM

$$EXECUTION:

PAGE : 1

ITEM RATE RANK TASK COST-PER-BILLET

---- ----- ---- ------ ---------------------

1 AN 20350 $0.00 AIRCRAFT UNDER EMERGENCY CONDITIONS
2 AN 20351 $0.00 IDENTIFY MARKINGS INDICATING DANGEROUS AREAS
3 AN 20352 $0.00 IDENTIFY TOXIC PROPERTIES OF CLEANING MATERIALS
4 AN 20368 $0.00 RECOGNIZE ARMED EJECTION SEATS
5 AN 24425 $0.00 HANDLE AIRCRAFT BATTERIES
6 AN 25352 $0.00 KNOW NAVAL AIR ARM GENERAL ORGANIZATION
7 AN 25353 $0.00 IDENTIFY NAMES, FUNCTIONS OF NAVAL AIR GROUPS

... Output terminated at this point ...

D.2-65
Program P9A Sample Run Output - Print Option "B"
The following is a sample of the output (complete) from Program P9A that results from Print Option "B" selection.

RUNP9A

ABBR. TASKS FILE PRINT PROGRAM IS STARTING

ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
A = PRINT ALL TASK RECORDS IN SORTED ORDER
B = LIST DESCRIPTORS ASSIGNED TO TASK DATA

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT

$EXECUTION:

PAGE : 1

-- TASKS

::: ADMINISTRATIVE ROUTINE PAPERWORK
11 FORMS FILLING
12 DOCUMENT-FILE MANAGEMENT
13 DECODE-ENCODE
14 SCREEN-FILTER DISTRIBUTE
15 OTHER

::: ADMINISTRATIVE NON-ROUTINE PAPERWORK
11 CONSTRUCT MESSAGES-REPORTS
12 ANALYZE-INTERPRET
13 CONSTRUCT RECOMMENDATION-PROPOSAL
14 CONSTRUCT PLAN
15 OTHER

::: ADMINISTRATIVE OFFICE EQUIPMENT OPERATION
21 TYPEWRITER
22 TELEPHONE, ETC.
23 REPRODUCER
24 COMPUTER TERMINAL
25 TELETYPE
26 OTHER

D.2-66
**INTERPERSONAL:** INFORM-INSTRUCT-MANAGE
31 BRIEF-DEBRIEF
32 INSTRUCT-TRAIN
33 ASSIGN, MONITOR, COORDINATE
34 EVALUATE
35 OTHER

**TECHNICAL PROCEDURES**
41 SEQUENTIAL
42 STRATEGIC, ADAPTIVE
43 INTERPERSONAL, TEAM MEMBER
44 OTHER

**TECHNICAL TYPE OF PROCEDURE (MAIN EMPHASIS)**
51 SCAN-DETECT: SYMBOLIC (INCL. MAPS, RADAR, TRANSDUCED SIGNALS)
52 SCAN-DETECT: NATURAL
53 IDENTIFY: SYMBOLIC (INCL. TRANSDUCED SIGNALS)
54 IDENTIFY: NATURAL
55 INTERPRET: SYMBOLIC
56 INTERPRET: NATURAL
57 PERCEPTUAL-MOTOR
58 COGNITIVE OPERATIONS
59 MANUAL
60 COMMUNICATE

**TECHNICAL WITH OR WITHOUT EQUIPMENT**
61 WITH EQUIPMENT (PAPER IS EQUIPMENT)
62 WITHOUT EQUIPMENT

**TECHNICAL DECIDE**
71 DIAGNOSE-ANALYZE

PAGE : 2

72 SELECT-CHOOSE
73 UNDER STRESS OR LOAD
74 OTHER

**TECHNICAL CONSTRUCT-REPAIR, PLAN**
81 MANUAL CONSTRUCT OR REPAIR
82 COGNITIVE CONSTRUCT, PLAN

**TECHNICAL TRACK-AIM-STEER**
91 APPLICABLE
92 TIME STRESS
93 INFORMATION-LOAD STRESS

D.2-67
PROGRAM NAME : P14
ENTRY POINT : P14
RUN EXECUTIVES(S) : RUNP14
USER OUTPUT VIA : Terminal Only
USER PROMPTING : Executive Level Only
FUNCTION : Extract Abbreviated Course File Data from the NITRAS MCRF Extract Tape

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>NITRAS</td>
<td>INPUT</td>
<td>24</td>
<td>Tape</td>
<td>Not Applicable</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>FB, 810, 3240</td>
</tr>
<tr>
<td>DESCR</td>
<td>INPUT</td>
<td>5</td>
<td>DASD</td>
<td>ETAM/SCDESC</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Card Image</td>
</tr>
<tr>
<td>COURSE</td>
<td>OUTPUT</td>
<td>7</td>
<td>DASD</td>
<td>ETAM/COURSE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Card Image</td>
</tr>
</tbody>
</table>

Discussion
This program constructs the Abbreviated Course data base. Using the CIN numbers as indicated on the input descriptor cards (File Number 5, above), matching data is sought on the NITRAS MCRF extract tape. If a match occurs, the data is selected from the NITRAS tape.

If a CIN number remains unmatched on a descriptor card, a message is output to indicate the missing data. Samples of this condition follow.
Listing of EXECUTIVE = RUNP14

&TYPE OFF
&COMMENT MAKEUP OF ABBR, COURSE DATA BASE.
&COMMENT NO ENTRY PARMS.
&COMMENT NITRAS TAPE MUST BE MOUNTED PRIOR TO INVOCATION
&COMMENT OF THIS EXEC.
&SPACE
&PRINT IS NITRAS TAPE ALREADY MOUNTED AND MOUNT VERIFIED?
-SP1 &PRINT RESPOND YES OR NO
&SPACE
&READ ARCS
&IF &1 EO YES &GOTO -YES
&IF &1 EO NO &GOTO -NO
&PRINT UNABLE TO DETERMINE LAST ENTRY TYPE - TRY AGAIN
&SPACE
&GOTO -SP1
-NO &PRINT COMPLETE PROCEDURE FOR NITRAS TAPE MOUNT, AND
&PRINT INVOKE EXECUTIVE RUNP14 AGAIN.
-FIN &SPACE
&PRINT EXECUTIVE RUNP14 IS NOW TERMINATING
-OUT &SPACE
&QUIT
-YES &SPACE
TAPEIO F5P TAP2
&IF &INDEX0 EO & &GOTO -RUN
&PRINT TAPE FORWARD FILE SKIP BROKE; RETURN CODE WAS &INDEX0
&GOTO -FIN
-RUNIT FILEDEF NITRAS TAP2 RE FB LR 810 BL 3240 DEN 3
FILEDEF DESCRIPT DSK ETAM SCDESC RE F LR 80 BL 80
FILEDEF COURSE DSK ETAM COURSE2 RE F LR 80 BL 80
LOAD P14 (CLEAR LIBE) PLILIB
START (BRIEF)
FILEDEF & CLEAR
&EXIT

Program P14 Error Messages

The only error of significance in this module results after reaching
an end-of-file on the NITRAS tape, when more (one/more) descriptor cards
remain to be read. An instance of this is shown in the sample output
below. The error message follows the normal EOF ON NITRAS-TAPE INPUT
message for CIN number 83000010.
Program P14 Sample Run Output

The following represents an execution of Program P14. Only a cut of the output is shown in the interest of brevity. For a listing of the contents of the actual Course Abbreviated data base, see the output listings that have resulted from Program P7B output.

**RUNP14**

$$
**
ABBREVIATED COURSE DATA BASE EXTRACT PROGRAM
DATE : 04/18/77
TIME : 1350

<table>
<thead>
<tr>
<th>DESCRIPTOR CARD NUM</th>
<th>CIN COUNT</th>
<th>CIN</th>
<th>NITRAS REC NUM</th>
<th>COURSE CDP</th>
<th>COURSE SHORT TITLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>A000111</td>
<td>13</td>
<td>0327</td>
<td>SWO ADV COMMAND</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>A1010108</td>
<td>213</td>
<td>516U</td>
<td>WRA-4 CMB MA</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>214</td>
<td>5535</td>
<td>WRA-4 CMB /FINIS</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>215</td>
<td>7654</td>
<td>WRA-4 CMB MA</td>
</tr>
<tr>
<td>3 ***</td>
<td>3</td>
<td>A1020060</td>
<td></td>
<td></td>
<td>NO CIN MATCH IN NITRAS</td>
</tr>
<tr>
<td>4 ***</td>
<td>4</td>
<td>A1020093</td>
<td></td>
<td></td>
<td>NO CIN MATCH IN NITRAS</td>
</tr>
<tr>
<td>5 ***</td>
<td>5</td>
<td>A1020095</td>
<td></td>
<td></td>
<td>NO CIN MATCH IN NITRAS</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>A1930050</td>
<td>794</td>
<td>2213</td>
<td>INERT NAV PRINC</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>A2E0013</td>
<td>881</td>
<td>0303</td>
<td>TAL PCO/PXO ORI</td>
</tr>
</tbody>
</table>

ETC ...

** EOF ON NITRAS-TAPE INPUT **

132 *** 121 83000010 - NO CIN MATCH IN NITRAS -

* ** EOF ON CARD-DESCRIPTOR INPUT **

132 : TOTAL DESCRIPTOR CARDS READ
121 : TOTAL CIN NUMBERS INPUT
50 : TOTAL CIN ERRORS (NO MATCH IN NITRAS)
111 : TOTAL COURSES EXTRACTED
4,127 : LAST RECORD READ IN NITRAS

D.2-70
Program P14 Specific User Instructions

In regards the NITRAS MCRF Extract Tape:

* The tape must be in the format shown in the File Section for File Number 24.

* The tape must be sorted into ascending order by CIN number, by CDP number. If an IBM sort package is used for this purpose on the 360/370 Series IBM computers, the following OS SORT control card is required:

   SORT FIELDS=(5,8,CH,A,1,4,CH,A),FILSZ=E5000

* When the tape has been posted to the NCSS time-sharing vendor's computer site, have the local NCSS Representative obtain a NCSS tape library "Bin Number" for the tape.

* When ready to process the tape, precede the RUNP14 executive command by the following online terminal entry:

   MOUNT TAPE binnumber AS TAP2 RINGOUT

   Wait for a response from the NCSS I/O Operator before proceeding with execution of RUNP14.

* RUNP14 executive presumes a Standard-Label OS tape; a forward-skip file is executed to pass over the volume identification and data set label. This is accomplished in RUNP14 EXEC line:

   TAPEIO FSF TAP2

   If a non-labeled tape is to be processed, a change must be made to the executive.
PROGRAM NAME : P16
ENTRY POINT : P16
RUN EXECUTIVES(S) : RUNP16
USER OUTPUT VIA : Terminal Only
USER PROMPTING : None

FUNCTION : Construct Descriptor Master Index File (File Number 3)

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MREF</td>
<td>INOUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DIND</td>
<td>OUTPUT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>CARDS</td>
<td>INPUT</td>
<td>2</td>
<td>DASD</td>
<td>ETAM/DICRDS</td>
<td>F, 80, 80</td>
</tr>
</tbody>
</table>

Discussion

The Descriptor Master Index (File Number 3) contains card-image records in the following sequence:

<table>
<thead>
<tr>
<th>Record Number</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Header record, containing 40 binary halfword counts. Count(1) is the zero-origin maximum record number in the file (last record position used).</td>
</tr>
<tr>
<td>1 - 4</td>
<td>Course index contents; these cards will be read into the PL/1 structure called DINDEX.</td>
</tr>
<tr>
<td>5 - 8</td>
<td>Vehicle index contents; same comments as above.</td>
</tr>
<tr>
<td>9 - 12</td>
<td>Task index contents; same comments as above.</td>
</tr>
<tr>
<td>13 - N1</td>
<td>Card records as read from the input file (File Number 2) descriptive of Course type descriptors.</td>
</tr>
<tr>
<td>(N1+1) - N2</td>
<td>Vehicle descriptor cards.</td>
</tr>
<tr>
<td>(N2+1) - N3</td>
<td>Task descriptor cards; end-of-file follows.</td>
</tr>
</tbody>
</table>
Listing of EXECUTIVE = RUNP16

&TYPE OFF
&COMMENT MAKEUP OF MASTER DESCRIPTOR INDEX (DIND)
&COMMENT NO ENTRY PARMS
FILEDEF CARDS DSK ETAM DICRDS RE F LR 8$$ BL 8$$
FILEDEF MREF DSK ETAM MREF RE F LR 8$$ BL 8$$
FILEDEF DIND DSK ETAM DIND RE F LR 8$$ BL 8$$
LOAD P16 (CLEAR LIBE) PLILIB
START (BRIEF)
FILEDEF CARDS CLEAR
FILEDEF MREF CLEAR
FILEDEF DIND CLEAR
&EXIT

Program P16 Error Messages

The following general error message sequence can result from errors detected
in the run of P16:

** ERROR NUMBER X HAS OCCURED **
(Followed by descriptive line)

(Followed by the two standard terminus messages - as shown on the
sample run)

In the message first line, X has the following significance; X also
determines the exact contents of the message second line.

<table>
<thead>
<tr>
<th>Number X</th>
<th>Message Second Line</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>FIRST CARD NOT (--) CARD</td>
</tr>
<tr>
<td>2</td>
<td>DUPLICATE (--) CARD</td>
</tr>
<tr>
<td>3</td>
<td>TYPE OF (--) CARD INVALID</td>
</tr>
<tr>
<td>4</td>
<td>TYPE (**) CARD MUST FOLLOW TYPE (--)</td>
</tr>
<tr>
<td>5</td>
<td>CATEGORY HAS NO DESCRIPTORS</td>
</tr>
<tr>
<td>6</td>
<td>DESCRIPTOR NUMBER CONVERSION ERROR</td>
</tr>
<tr>
<td>7</td>
<td>CATEGORY TABLE SIZE EXCEEDED, MAX = 15</td>
</tr>
</tbody>
</table>
Program P16 Sample Error Message

** ERROR NUMBER 2 HAS OCCURRED **
DUPLICATE (--) CARD

** TOTAL DESCRIPTOR INDEX CARDS READ : 44
MASTER DESCRIPTOR INDEX PROCESSING IS NOW TERMINATING

Program P16 Sample Run Output

The following represents an actual run of P16. The output is shown in its entirety; this represents the contents of the Descriptor Index in the current ETAM implementation.

RUNP16

$$$$$
MASTER DESCRIPTOR INDEX CONSTRUCT PROGRAM IS NOW STARTING

DESCRIPTOR CARD FILE LISTING FOLLOWS

---------------

-- COURSES

::: REFERENCE KNOWLEDGE

01 SYSTEM PURPOSES
02 ORGANIZATIONAL ROLES
03 CONTEXTS OF OPERATION
04 ORGANIZATIONAL RULES
05 OTHER

::: ENABLING KNOWLEDGE

10 OPERATIONAL GOAL CRITERIA
11 NOMENCLATURE, IDENTIFICATION, LOCATION
12 PROCEDURAL DESCRIPTIONS
13 JOB RELEVANT FACTS, RULES
14 OTHER

::: TASK FORMATS

20 PROCEDURE FORMATS
21 DECISION FORMATS
22 CONSTRUCTION FORMATS
23 OTHER

::: CROSS JOB CATEGORIES

30 OPERATIONS
31 MAINTENANCE
32 SERVICE & ADMINISTRATION
33 COMMAND
34 OTHER
OBJECTIVE TASK VARIABLES AS MANIFEST IN THE TRAINING

EQUIPMENT & OBJECTS USED: REAL
EQUIPMENT & OBJECTS USED: SIMULATED
EQUIPMENT & OBJECTS USED: SYMBOLIC

ENVIRONMENTS IN WHICH TASK IS TRAINED: REAL
ENVIRONMENTS IN WHICH TASK IS TRAINED: SIMULATED
ENVIRONMENTS IN WHICH TASK IS TRAINED: SYMBOLIC

TOOLS/INSTRUMENTS USED IN TRAINING: REAL
TOOLS/INSTRUMENTS USED IN TRAINING: SIMULATED
TOOLS/INSTRUMENTS USED IN TRAINING: SYMBOLIC

PEF/ENABLING INFO IN DOING TASK: APPLIED
PEF/ENABLING INFO IN DOING TASK: NOT APPLIED
CRITERIA OF TASK PERFORMANCE: REAL
CRITERIA OF TASK PERFORMANCE: SIMULATED
CRITERIA OF TASK PERFORMANCE: SYMBOLIC

TASK FUNCTIONS DOMINANT IN TRAINING

GOAL PROJECTION
SCAN-DETECT
IDENTIFY
INTERPRET
PROCEDURE FOLLOWING
DECIDE
CONSTRUCT, PLAN
TRACK
MOTOR PERFORMANCE
INTERPERSONAL INTERACTION
RECALL TASK-CYCLE INFORMATION
RECALL ENABLING INFORMATION
ADAPT IMPROVISATIONALLY/IMPROMPTU
OTHER

STAGE OF LEARNING

ORIENTATION, FAMILIARIZATION
TASK NOMENCLATURE, IDENTs, LOCATIONS, FACTS, RULES
TASK FORMATS AT CONCEPTUAL LEVEL
PROCEDURES AT VERBAL LEVEL ONLY
TASK COMPONENTS WITH GUIDANCE
ENIRE JOB-TASK PROCEDURALLY AT BARELY ACCEPTABLE MASTERY
HIGHLY PROFICIENT IN WORK CONTEXT
UNUSUAL TASK CONDITIONS
PERFORMANCE AT KEY MAN LEVEL
REFRESHER LEARNING

VEHICLES

VEHICLE TYPES
INSTRUCTOR
STATIC GRAPHICS

D.2-75
ANIMATED GRAPHICS
PHYSICAL MODELS
PROCEDURAL TRAINERS: SYMBOLIC
PROCEDURAL TRAINERS: PHYSICAL BUT NON-FUNCTIONAL
PROCEDURAL TRAINERS: FUNCTIONAL
TASK & SYSTEM SIMULATORS
REAL EQUIPMENT ITSELF

CLASS OF TRAINING OBJECTIVE
REFERENCE KNOWLEDGE
KNOWLEDGE, TASK SPECIFIC/ENABLING
TASK-SKILL FORMATS
SKILL TRAINING

VEHICLE PROPERTIES
VISUAL
AUDITORY
KINESTHETIC/VESTIBULAR
TACTILE

TYPE OF CONTENT DISPLAYED
TEXT-VERBAL
DIAGRAMMATIC
ABSTRACTED PICTORIAL REPRESENTATION
PICTORIAL REPRESENTATIONS
PHYSICAL REPRESENTATIONS
OTHER

TYPE OF PRESENTATIONAL SEQUENCE
LIBRARY OF FRAMES OR ITEMS
PRESENTATION SEQUENCE NOT APPLICABLE
FIXED SEQUENTIAL FRAMES OR ITEMS
RANDOM SELECTION OF FRAME SEQUENCES
DYNAMIC CHANGE OF CONTENT WITHIN FRAME

SELECTION SOURCE FOR SEQUENCING
INTERNAL PROGRAM
INSTRUCTOR
STUDENT CHOICE
STUDENT PERFORMANCE
COMBINATIONS OF THE ABOVE

TYPE OF EXTERNAL CONTROL OPERATED BY STUDENT
NOT APPLICABLE DIRECTLY
ARTIFICIAL OR SYMBOLIC RESPONSE
REPRESENTATIONAL RESPONSE BY SYMBOLIC SELECTION
REPRESENTATIONAL RESPONSE BY DUMMY CONTROL ACTIVATION
TASK-MANIPULATIVE RESPONSE, NON-DYNAMIC IN TIME AND FORCE
TASK-MANIPULATIVE RESPONSE, DYNAMIC AND INTERACTIVE

D.2-76
FEEDBACK PRESENTATION LOGIC
71 NOT APPLICABLE
72 SELECTS NEXT STIMULUS ITEM OR SEQUENCE
73 GIVES EVALUATION OF PRECEDING RESPONSE
74 SELECTS AND PRESENTS GUIDANCE INFO

RESPONSE EVALUATION LOGIC
81 NOT INTERNAL—DEPENDS ON INSTRUCTOR OR STUDENT EVALUATION
82 EVALUATION LIMITED TO STUDENT'S IMMEDIATE RESPONSE
83 EVALUATION EXTENDED TO A SET OF STUDENT RESPONSES
84 TOLERANCE LIMITS ON ACCEPTABLE STUDENT RESPONSE: FIXED
85 TOLERANCE LIMITS ON ACCEPTABLE STUDENT RESPONSE: VARIABLE

-- TASKS

ADMINISTRATIVE ROUTINE PAPERWORK
41 FORMS FILLING
42 DOCUMENT-FILE MANAGEMENT
43 DECODE-ENCODE
44 SCREEN-FILTER DISTRIBUTE
45 OTHER

ADMINISTRATIVE NON-ROUTINE PAPERWORK
11 CONSTRUCT MESSAGES-REPORTS
12 ANALYZE-INTERPRET
13 CONSTRUCT RECOMMENDATION-PROPOSAL
14 CONSTRUCT PLAN
15 OTHER

ADMINISTRATIVE OFFICE EQUIPMENT OPERATION
21 TYPEWRITER
22 TELEPHONE, ETC.
23 REPRODUCER
24 COMPUTER TERMINAL
25 TELETYpe
26 OTHER

INTERPERSONAL: INFORM-INSTRUCT-MANAGE
31 BRIEF-DEBRIEF
32 INSTRUCT-TRAIN
33 ASSIGN, MONITOR, COORDINATE
34 EVALUATE
35 OTHER

TECHNICAL PROCEDURES
41 SEQUENTIAL
42 STRATEGIC, ADAPTIVE
43 INTERPERSONAL, TEAM MEMBER

D.2-77
44 OTHER

### TECHNICAL TYPE OF PROCEDURE (MAIN EMPHASIS)

- 51 SCAN-DETECT: SYMBOLIC (INCL. MAPS, RADAR, TRANSDUCED SIGNALS)
- 52 SCAN-DETECT: NATURAL
- 53 IDENTIFY: SYMBOLIC (INCL. TRANSDUCED SIGNALS)
- 54 IDENTIFY: NATURAL
- 55 INTERPRET: SYMBOLIC
- 56 INTERPRET: NATURAL
- 57 PERCEPTUAL-MOTOR
- 58 COGNITIVE OPERATIONS
- 59 MANUAL
- 60 COMMUNICATE

### TECHNICAL WITH OR WITHOUT EQUIPMENT

- 61 WITH EQUIPMENT (PAPER IS EQUIPMENT)
- 62 WITHOUT EQUIPMENT

### TECHNICAL DECIDE

- 71 DIAGNOSE-ANALYZE
- 72 SELECT-CHOOSE
- 73 UNDER STRESS OR LOAD
- 74 OTHER

### TECHNICAL CONSTRUCT-REPAIR, PLAN

- 81 MANUAL CONSTRUCT OR REPAIR
- 82 COGNITIVE CONSTRUCT, PLAN

### TECHNICAL TRACK-AIM-STEER

- 91 APPLICABLE
- 92 TIME STRESS
- 93 INFORMATION-LOAD STRESS

### TOTAL DESCRIPTOR INDEX CARDS READ: 181

MASTER DESCRIPTOR INDEX PROCESSING IS NOW TERMINATING
PROGRAM NAME: P17
ENTRY POINT: IHECMS (Execution Parameter Supplied by Executive RUNP17)
RUN EXECUTIVES(S): RUNP17, NONAME, IFNAME, DEVICE, JCL2
USER OUTPUT VIA: Terminal, Offline Printer, or Both
USER PROMPTING: At Executive Level: Output Device Select
FUNCTION: Load One/More Project Files With Input Data Supplied on Cards

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSPRINT</td>
<td>OUTPUT</td>
<td>N/A: THIS IS THE OFFLINE PRINTER</td>
<td></td>
<td>FA, 121, 121</td>
<td></td>
</tr>
<tr>
<td>CARDS</td>
<td>INPUT</td>
<td>25</td>
<td>DASD</td>
<td>Projectname/LOAD</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REC</td>
<td>OUTPUT</td>
<td>27</td>
<td>DASD</td>
<td>Projectname/REC</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REJ</td>
<td>OUTPUT</td>
<td>29</td>
<td>DASD</td>
<td>Projectname/REJ</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>REV</td>
<td>OUTPUT</td>
<td>28</td>
<td>DASD</td>
<td>Projectname/REV</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>ID</td>
<td>OUTPUT</td>
<td>26</td>
<td>DASD</td>
<td>Projectname/ID</td>
<td>F, 80, 80</td>
</tr>
</tbody>
</table>

The following files are not a functional component of the Range-Of-Effects (ROE) search process. Since they are card-image format, they can be loaded directly from Program P17 and printed from Program P1.

<table>
<thead>
<tr>
<th></th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>EXD</td>
<td>OUTPUT</td>
<td>33</td>
<td>DASD</td>
<td>Projectname/EXD</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>BPT</td>
<td>OUTPUT</td>
<td>34</td>
<td>DASD</td>
<td>Projectname/BPT</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>RKP</td>
<td>OUTPUT</td>
<td>35</td>
<td>DASD</td>
<td>Projectname/RKP</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>RRPJ</td>
<td>OUTPUT</td>
<td>36</td>
<td>DASD</td>
<td>Projectname/RRPJ</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>RRPK</td>
<td>OUTPUT</td>
<td>37</td>
<td>DASD</td>
<td>Projectname/RRPK</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>VQAL</td>
<td>OUTPUT</td>
<td>38</td>
<td>DASD</td>
<td>Projectname/VQAL</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>VARF</td>
<td>OUTPUT</td>
<td>39</td>
<td>DASD</td>
<td>Projectname/VARF</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>SCEN</td>
<td>OUTPUT</td>
<td>40</td>
<td>DASD</td>
<td>Projectname/SCEN</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>ALTP</td>
<td>OUTPUT</td>
<td>41</td>
<td>DASD</td>
<td>Projectname/ALTP</td>
<td>F, 80, 80</td>
</tr>
</tbody>
</table>
Discussion

This Program serves to generate those files of the ETAM system called "Project Files". In the current implementation, only those of card-image format are of concern. The following four card-image files are functional in the Range-of-Effects search portion of the ETAM system implementation:

<table>
<thead>
<tr>
<th>Filetype</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Project Description</td>
</tr>
<tr>
<td>REC</td>
<td>Course Range-of-Effects (ROE) Search Descriptors</td>
</tr>
<tr>
<td>REV</td>
<td>Vehicle ROE Search Descriptors</td>
</tr>
<tr>
<td>REJ</td>
<td>Task ROE Search Descriptors</td>
</tr>
</tbody>
</table>

In addition to the above, if the control-card input so signifies, the following card-image Project Files may be initialized:

- EXD: Search Extract Defaults
- BPT: Benefit Pattern
- RKP: Risk Profile
- RRJP: Risk Reduction Projects
- RRPK: Risk Reduction Packages
- VQAL: Variables Qualitative
- VARF: Variable References
- SCEN: Scenarios
- ALTP: Alternate Projects

All of the above card-image files can be printed utilizing Program Pl, either individually, or in a group using the "ALL" option.

Listing of EXECUTIVE = RUNP17

```
&TYPE OFF
&COMMENT THIS EXEC CONTROLS PROJECT FILE INIT/GENERATE
&COMMENT SINGLE ENTRY PARM REQUIRED - PROJECT NAME
&COMMENT EXEC = DEVICE WILL PROMPT FOR OUTPUT DEVICE
&IF &INDEX EO 0 &GOTO -STP1
EXEC NONAME RUNP17
-STP1 &ALPHA3 = &1
EXEC IFNAME
&SPACE
EXEC DEVICE
&SPACE
EXEC JCL2
FILEDEF SYSPRINT PTR RE FA LR 121 BL 121
FILEDEF CARDS DSK &ALPHA3 LOAD RE F LR 80 BL 80
LOAD P17 (CLEAR LIBE) PLILIB
START IHECMS &ALPHA1 (BRIEF)
FILEDEF " CLEAR
&EXIT
```
Program P17 Specific User Instructions

Since this program processes only those files associated with an individual ETAM project in a single run, the name of the project must be provided by the User when invoking the RUMP17 executive. On the sample runs to follow, this is demonstrated using the ETAM project name "PN123".

The format (contents) of the input control file (File Number 25) dictates the loading process within Program P17. See the File Section description of File Number 25 for specific card formats. The following notes apply to the ordering of the control cards within the input file.

1. Initialization of a particular file is signified in the control input by the occurrence of a control card with the characters "**" in record positions one and two. The Project File filetype appears in record positions four through seven. All cards following this card up to the next "**" card or end-of-file will be loaded into the indicated filetype.

2. More than one filetype may be initialized within one input control file.

3. If a given file is to be initialized, and the file already exists, then the old copy is replaced with the new.

4. If the first card of the control input is an "INIT" card, then any Project File not mentioned in the control input will be opened and one blank record inserted by Program P17.

Those files deemed "Project Files" in the ETAM system have been implemented in the NCSS time-sharing system as follows:

<table>
<thead>
<tr>
<th>NCSS Filename</th>
<th>NCSS Filetype</th>
</tr>
</thead>
<tbody>
<tr>
<td>The specific</td>
<td>ID, REC, REV, Etc.</td>
</tr>
<tr>
<td>ETAM Project</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td></td>
</tr>
</tbody>
</table>

Program P17 Error Messages

If the "INIT" option is used as in the instructions above, then the second card of the input control file must be an "**" type to signify the placement of the cards to follow. If the "INIT" option is omitted, then the first card must be an "**" type. If this condition is not met, the following error message will be output:

*** FIRST TITLE CARD IS MISSING ***
Once a "**" title card has been detected, the filetype name on the card will be verified; the following message test will appear if the filetype is illegal. This usually signifies a spelling error:

*** TITLE CARD FILENAME -filename- IS INVALID ***

The name in error is indicated in the message. All cards following this illegal title card will be printed on the selected output device(s), until the occurrence of the next title card or end-of-file.

As files are initialized, Program P17 maintains a list of the files that have been processed within the current run. If an attempt is made to initialize the same filetype two or more times within the same run, the following error text will be output:

*** FILENAME -filename- IS DUPLICATED ***

All cards following this error will be printed up to the occurrence of the next "**" title card or end-of-file.

Program P17 Sample Run Output

In the following example, the User has selected to print the contents of the initialization on the offline printer. The text shown below appears on the online terminal. Ten of the thirteen files have been provided card initialization; three were initialized with one blank card-image record. Since the blank initialization occurred, the control input for this run contained an "INIT" card as the first card.

```
RUNP17 PN123
PROJECT NAMED PN123 DOES NOT YET EXIST,
IS THIS CORRECT? (RESPOND YES OR QUIT)
YES

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT
PRTR
$EXECUTION:
PROJECT FILE INITIALIZATION PROGRAM IS STARTING
LOAD OF PROJECT FILETYPE : ID - PROJECT DESCRIPTION
LOAD OF PROJECT FILETYPE : REC - COURSES ROE SEARCH ARGUMENTS
LOAD OF PROJECT FILETYPE : BPT - BENEFIT PATTERN
LOAD OF PROJECT FILETYPE : RKP - RISK PROFILE
LOAD OF PROJECT FILETYPE : RRPJ - RISK REDUCTION PROJECTS
```
LOAD OF PROJECT FILETYPE: SCEN - SCENARIOS
LOAD OF PROJECT FILETYPE: RRPK - RISK REDUCTION PACKAGES
LOAD OF PROJECT FILETYPE: VQAL - VARIABLES QUALIFICATION
LOAD OF PROJECT FILETYPE: VARF - VARIABLE REFERENCES
LOAD OF PROJECT FILETYPE: ALTP - ALTERNATE PROJECTS

FILE: REJ  HAS BEEN INITIALIZED WITH ONE (1) BLANK RECORD
FILE: REV  HAS BEEN INITIALIZED WITH ONE (1) BLANK RECORD
FILE: EXD  HAS BEEN INITIALIZED WITH ONE (1) BLANK RECORD

""" ALL DATASETS INITIALIZED """

""" TOTAL NUMBER INPUT CARDS READ : 461 """

PROJECT FILE INITIALIZATION PROGRAM IS NOW TERMINATING
PROGRAM NAME : P18
ENTRY POINT : P18
RUN EXECUTIVES(S) : STARTUP, START1, START2
USER OUTPUT VIA : None
USER PROMPTING : None

FUNCTION: Initializes (zeros) counts in Master Reference File (File Number 1)

<table>
<thead>
<tr>
<th>Internal Filename</th>
<th>Refer To File</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>MREF</td>
<td>OUTPUT 1</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>F, 80, 80</td>
</tr>
</tbody>
</table>

Discussion

This file contains forty (40) binary halfword counts per record. Currently, there is only one record (one card-image) in use in this file. All forty quantities are zeroed by one run of P18, i.e., any previous contents will be destroyed.

Listing of EXECUTIVE = STARTUP

```
P STARTUP EXEC

&TYPE OFF
&COMMENT THIS IS THE SYSTEM STARTUP EXEC
&COMMENT NO INPUT PARAMETERS REQUIRED
-STEP1 STATE ETAM MREF P
&IF &INDEXØ EO Ø &GOTO -MDONE
LOAD P18 (CLEAR LIBE) PLILIB
FILEDEF MREF DSK ETAM MREF RE F LR 8Ø BL 8Ø
START (BRIEF)
&IF &INDEXØ EO Ø &GOTO -MGO
-ER1 EXEC BROKE 1 &INDEXØ
EXEC START1 4 MREF
&GOTO -STEP2
-MGO EXEC START1 2 MREF
&GOTO -STEP2
-MDONE EXEC START1 1 MREF
-STEP2 EXEC START2 CDESC SCDESC 2 1 8
-STEP3 EXEC START2 VDATA SVDATA 3 1 13
-STEP4 EXEC START2 VCDESC SVCDESC 4 1 13
-STEP5 EXEC START2 TDATA STDATA 5 1 13
-STEP6 EXEC START2 TDSCR STDSCR 6 1 13
&EXIT

D.2-84
```
Listing of EXECUTIVE = START1

`&TYPE OFF 
&COMMENT P1 IS INTERNAL MESSAGE ROUTER, RANGES 1 TO 5 
&COMMENT P2 IS FILETYPE NAME 
&IF &1 EQ 5 &GOTO -NYET 
&IF &1 EQ 4 &GOTO -BUST 
&IF &1 EQ 3 &GOTO -RGO 
&IF &1 EQ 2 &GOTO -GO 
&PRINT FILE &2 HAS BEEN PREVIOUSLY INITIALIZED 
&EXIT 
-GO &PRINT INITIALIZE OF FILE &2 WAS SUCCESSFUL 
&EXIT 
-RGO &PRINT REINITIALIZE OF FILE &2 WAS SUCCESSFUL 
&EXIT 
-BUST &PRINT FILE &2 INITIALIZATION FAILED 
&SPACE 
&EXIT 
-NYET &PRINT FILE &2 CANNOT BE INITIALIZED (NO INPUT DATA) 
&EXIT`

Listing of EXECUTIVE = START2

`&TYPE OFF 
&COMMENT P1 IS FT OF OFFLINE READ INPUT FILE 
&COMMENT P2 IS FT OF NEW FILE TO BE CREATED 
&COMMENT P3 IS ERROR NUMBER, IF BREAKS 
&COMMENT P4 AND P5 ARE THE SORT PARMs 
STATE ETAM &1 P 
&IF &INDEXØ NE Ø &GOTO -NONEW 
ERASE .ETAM &1 
&STACK &4 &5 
&STACK KT 
SORT ETAM &1 (BRIEF) 
&IF &INDEXØ EO Ø &GOTO -SGO 
-ERR EXEC BROKE &3 &INDEXØ 
EXEC START1 4 &2 
&EXIT &3 
-SGO STATE ETAM &2 P 
&IF &INDEXØ NE Ø &GOTO -NOREG 
&INDEX1 = 3 
ERASE ETAM &2 P 
&GOTO -SNEW 
-NOREG &INDEX1 = 2 
-SNEW ALTER .ETAM &1 P ETAM &2 P 
EXEC START1 &INDEX1 &2 
&EXIT 
-NONEW STATE ETAM &2 P 
&IF &INDEXØ EO Ø &GOTO -STP1 
EXEC START1 5 &2 
&EXIT 
-STP1 EXEC START1 1 &2 
&EXIT`
Program P18 Error Messages
None required.

Program P18 Sample Run Output
None generated by this program.
PROGRAM NAME: P20
ENTRY POINT: IHECMS (Execution Parameter Supplied by Executive RUNP20 at Entry)
RUN EXECUTIVES(S): RUNP20, DEVICE, JCL
USER OUTPUT VIA: Terminal Offline Printer, or Both
USER PROMPTING: At Executive Level: (1) Output Device Select
(2) Print Option Select (C, V, or T)
FUNCTION: Prints Course, Vehicle, or Task Entities and the Descriptors Associated With Each

<table>
<thead>
<tr>
<th>Internal Filename (DDNAME)</th>
<th>Usage</th>
<th>Refer To File Reference Number</th>
<th>Media</th>
<th>External NCSS Filename/Filetype</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSPRINT</td>
<td>OUTPUT</td>
<td>N/A : THIS IS THE OFFLINE PRINTER</td>
<td>DASD</td>
<td>ETAM/MREF</td>
<td>FA, 121, 121</td>
</tr>
<tr>
<td>MREF</td>
<td>INPUT</td>
<td>1</td>
<td>DASD</td>
<td>ETAM/DIND</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>DIND</td>
<td>DIRECT</td>
<td>3</td>
<td>DASD</td>
<td>ETAM/DESC</td>
<td>F, 213, 213</td>
</tr>
<tr>
<td>DESC</td>
<td>INPUT</td>
<td>6</td>
<td>DASD</td>
<td>ETAM/DCIN</td>
<td>F, 760, 760</td>
</tr>
<tr>
<td>DCIN</td>
<td>INPUT</td>
<td>8</td>
<td>DASD</td>
<td>ETAM/VDESC</td>
<td>F, 213, 213</td>
</tr>
<tr>
<td>VDESC</td>
<td>INPUT</td>
<td>12</td>
<td>DASD</td>
<td>ETAM/DVEH</td>
<td>F, 750, 750</td>
</tr>
<tr>
<td>DVEH</td>
<td>INPUT</td>
<td>16</td>
<td>DASD</td>
<td>ETAM/TDESC</td>
<td>F, 213, 213</td>
</tr>
<tr>
<td>TDESC</td>
<td>INPUT</td>
<td>19</td>
<td>DASD</td>
<td>ETAM/DTASK</td>
<td>F, 1500, 1500</td>
</tr>
<tr>
<td>DTASK</td>
<td>INPUT</td>
<td>23</td>
<td>DASD</td>
<td>ETAM/VEHS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>VEHS</td>
<td>DIRECT</td>
<td>15</td>
<td>DASD</td>
<td>ETAM/TASKS</td>
<td>F, 80, 80</td>
</tr>
<tr>
<td>TASKS</td>
<td>DIRECT</td>
<td>22</td>
<td>DASD</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Discussion

This program provides print facilities for the three types of ETAM entities - Courses, Vehicles, and Tasks. The entity type and output device type are selected by the User at the executive level.

Printer output from Program P20 is of the general format: entity description followed by all descriptors assigned to that particular entity. The order of output for Courses is by CIN number; for Vehicles, Stock Number order is maintained. For Tasks, the output is ordered by the contents of the 13-byte field of concatenated Rate, Rank, and Jobtask Number.

Listing of EXECUTIVE = RUNP20

```
&TYPE OFF
&COMMENT G.P. PRINT EXEC FOR COURSES, VEHICLES, TASKS ;
&COMMENT PRINTS DESCRIPTORS ASSIGNED TO EACH ENTITY
&COMMENT ENTRY/DEVICE PROMPTS DONE HERE AND IN EXEC = DEVICE
&SPACE
&PRINT DESCRIPTOR PRINT PROGRAM IS STARTING
&SPACE
&PRINT ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
&PRINT C = PRINT ALL DESCRIPTORS FOR COURSES, SORTED BY CIN
&PRINT V = PRINT ALL DESCRIPTORS ASSIGNED TO VEHICLES
&PRINT T = LIST DESCRIPTORS ASSIGNED TO TASK DATA

-STP1 &READ ARCS
 &IF &1 EO QUIT &QUIT
 &ALPHA0 = &1
 &IF &1 EO C &GOTO -STP2
 &IF &1 EO V &GOTO -STP2
 &IF &1 EO T &GOTO -STP2
 &SPACE
 &PRINT UNABLE TO DETERMINE ENTRY TYPE - TRY AGAIN
 &SPACE
 &GOTO -STP1

-STP2 EXEC DEVICE
 &ALPHA2 = &ALPHA0 $1 &ALPHA1

-STP3 &SPACE
EXEC &JCL
FILEDEF SYSPRINT PTR RE FA LR 121 BL 121
LOAD 'P22$ (CLEAR LIBE) PLLIB$ START IHECMS &ALPHA2 (BRIEFE)
FILEDEF & CLEAR
&EXIT
```
Program P20 Error Messages

Program P20 can produce the following error text (two lines):

** ERROR NUMBER XX HAS OCCURRED **

(Followed by a descriptive content line - varies with message number, as indicated below)

The error numbers associated with the above message have the following significance:

01 - The entry parameter supplied to Program P20 at entry time was not one of the three characters "C", "V", or "T".

The 2nd error line will contain the actual character(s) provided as an entry parameter by the external executive.

02 - A pointer discrepancy has been detected in the Descriptor Master Index file (File Number 3).

The second error line contains the expected and actual pointer values of concern.

03 - Pointer to the ETAM/DIND Descriptor Master Index file is outside the bounds of the file.

Second error line = same as that for error number 02.

04 - Number of categories in the ETAM/DIND file is greater than 15.

Second error line, same as for error number 02.

Program P20 Specific User Instructions

Three print options exist for the User in Program P20. These options are exercised by the selection of a single character to be entered via online terminal. The options are as follows:

<table>
<thead>
<tr>
<th>Character</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;C&quot;</td>
<td>Select Courses and their related descriptors for output</td>
</tr>
<tr>
<td>&quot;V&quot;</td>
<td>Select Vehicles, etc.</td>
</tr>
<tr>
<td>&quot;T&quot;</td>
<td>Select Tasks.</td>
</tr>
</tbody>
</table>

D.2-89
Program P20 Sample Run Output - Course Selection

Selection of the "C" options results in output as presented in the following sample. Shown are the prompting sequences and a few sample lines to illustrate the general format for Course output.

RUNP20

DESCRIPTOR PRINT PROGRAM IS STARTING

ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
C = PRINT ALL DESCRIPTORS FOR COURSES, SORTED BY CIN
V = PRINT ALL DESCRIPTORS ASSIGNED TO VEHICLES
T = LIST DESCRIPTORS ASSIGNED TO TASK DATA
C

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT
TERM

$EXECUTION:

PAGE 1: COURSE DESCRIPTOR ASSIGNMENTS

CIN : A 000111

*** REFERENCE KNOWLEDGE
  #1 SYSTEM PURPOSES
  #3 CONTEXTS OF OPERATION
  #4 ORGANIZATIONAL RULES

*** ENABLING KNOWLEDGE
  11 NOMENCLATURE, IDENTIFICATION, LOCATION
  12 PROCEDURAL DESCRIPTIONS
  13 JOB RELEVANT FACTS, RULES

*** TASK FORMATS
  20 PROCEDURE FORMATS
  21 DECISION FORMATS
  22 CONSTRUCTION FORMATS

*** GROSS JOB CATEGORIES
  33 COMMAND

*** OBJECTIVE TASK VARIABLES AS MANIFEST IN THE TRAINING
  42 EQUIPMENT & OBJECTS USED: SYMBOLIC
  45 ENVIRONMENTS IN WHICH TASK IS TRAINED: SYMBOLIC
  48 TOOLS/INSTRUMENTS USED IN TRAINING: SYMBOLIC
  53 CRITERIA OF TASK PERFORMANCE: SYMBOLIC

D.2-90
TAEG REPORT NO. 40

**TASK FUNCTIONS DOMINANT IN TRAINING**
- 65 DECIDE
- 66 CONSTRUCT, PLAN

**STAGE OF LEARNING**
- 82 TASK FORMATS AT CONCEPTUAL LEVEL

CIN : A 2E$\phi$13.

**REFERENCE KNOWLEDGE**
- $\phi$1 SYSTEM PURPOSES
- $\phi$2 ORGANIZATIONAL ROLES
- $\phi$3 CONTEXTS OF OPERATION

... Output interrupted at this point ...

Program P20 Sample Run Output - Vehicles Selection

Following is an illustration of the "V" option output:

RUNP20

DESCRIPTION PRINT PROGRAM IS STARTING

ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE)
C = PRINT ALL DESCRIPTORS FOR COURSES, SORTED BY CIN
V = PRINT ALL DESCRIPTORS ASSIGNED TO VEHICLES
T = LIST DESCRIPTORS ASSIGNED TO TASK DATA
V

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTR, BOTH, OR QUIT
TERM

$\$EXECUTION:

PAGE  1 : VEHICLE DESCRIPTOR ASSIGNMENTS

691MILCD0468 / 12CD-17 / MATH, VECTOR ACCELERATION DEMONSTRATOR T/A

**VEHICLE TYPES**
- $\phi$3 ANIMATED GRAPHS

**CLASS OF TRAINING OBJECTIVE**
- 11 REFERENCE KNOWLEDGE

**VEHICLE PROPERTIES**
- 21 VISUAL

... Output terminated at this point ...

D.2-91
Program P20 Sample Run Output - Tasks Option

Selection of the "T" option results in output that is in the following format:

```
RUNP2Q

DESORPTOR PRINT PROGRAM IS STARTING

ENTER SINGLE CHARACTER OPTION (OR QUIT TO TERMINATE).
C = PRINT ALL DESCRIPTORS FOR COURSES, SORTED BY CIN
V = PRINT ALL DESCRIPTORS ASSIGNED TO VEHICLES
T = LIST DESCRIPTORS ASSIGNED TO TASK DATA

OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
ENTER TERM, PRTP, BOTH, OR QUIT

EXECUTION:

PAGE 1 : JOB TASK DESCRIPTOR ASSIGNMENTS
AN   /   / 28352 AIRCRAFT UNDER EMERGENCY CONDITIONS

*** TECHNICAL PROCEDURES
   41 SEQUENTIAL
   42 STRATEGIC, ADAPTIVE

*** TECHNICAL TYPE OF PROCEDURE(MAIN EMPHASIS)
   57 PERCEPTUAL-MOTOR

*** TECHNICAL WITH OR WITHOUT EQUIPMENT
   36 WITH EQUIPMENT (PAPER IS EQUIPMENT)

*** TECHNICAL DECIDE
   73 UNDER STRESS OR LOAD

AN   /   / 28351 IDENTIFY MARKINGS INDICATING DANGEROUS AREAS

*** TECHNICAL TYPE OF PROCEDURE(MAIN EMPHASIS)
   53 IDENTIFY: SYMBOLIC(INCL. TRANSDUCED SIGNALS)

AN   /   / 28352 IDENTIFY TOXIC PROPERTIES OF CLEANING MATERIALS

*** TECHNICAL TYPE OF PROCEDURE(MAIN EMPHASIS)
   54 IDENTIFY: NATURAL

AN   /   / 28368 RECOGNIZE ARMED EJECTION SEATS

*** TECHNICAL Type OF PROCEDURE(MAIN EMPHASIS)
   54 IDENTIFY: NATURAL

... Output terminated at this point ...
```
NCSS EXECUTIVE SEQUENCES ASSOCIATED WITH MULTIPLE PROGRAMS

Each program of the ETAM system has an associated NCSS executive (exec) routine. These executives are listed in the Section covering the program of concern.

The executives within this Section are, in general, multiple usage modules not associated with a single ETAM program; a brief description and a listing of each follows.

**EXECUTIVE = BROKE**

This is a general-purpose routine used to print a standardized error message. The routine is designed to be invoked by a higher-level executive. Return is always to the caller-executive routine.

```plaintext
&TYPE OFF
&COMMENT P1 IS ERROR NUMBER
&COMMENT P2 IS INDEX0 VALUE THAT CAUSED ERROR
&SPACE
&PRINT """" ERROR NUMBER &1 HAS OCCURRED; INDEX0 VALUE WAS &2 """
&SPACE
&EXIT
TOP
```

**EXECUTIVE = DEVICE**

Many of the ETAM programs have been designed to direct output to the User's online terminal, offline printer, or both. This executive function is to prompt the User in the selection of the desired device(s). If the User elects the QUIT option, forced termination of the executive sequence at all levels follows.

```plaintext
&TYPE OFF
&COMMENT PROMPTING EXEC FOR DEVICE
&COMMENT NO ENTRY ARGUMENTS REQUIRED
&COMMENT IF QUIT ENTERED, EXEC SEQUENCE TERMINATED
&COMMENT NORMAL RETURN WITH $ALPHA1 SET TO ENTRY
-STOP $SPACE
&PRINT OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, BOTH, OR QUIT?
&PRINT ENTER TERM, PRTR, BOTH, OR QUIT
```

D.3-1
Those programs associated with a specific ETAM project must be furnished the name of the project. No operation can ensue, if the project name is omitted. This executive determines the existence of the project name provided.

EXECUTIVE = IFNAME

P IFNAME EXEC

&TYPE OFF
&COMMENT EXEC TO VALIDATE THE EXISTENCE OF A USER-SUPPLIED PROJECT NAME
&COMMENT ENTRY REQUIRES GLOBAL ALPHA VARIABLE NO. 3
&COMMENT TO BE INITIALIZED TO THE PROJECT FILE NAME
&SPACE
STATE &ALPHA3 ID P
&IF &INDEX1 EO 0 &GOTO -YES
-NO &PRINT PROJECT NAMED &ALPHA3 DOES NOT YET EXIST,
&GOTO -STP1
-YES &PRINT PROJECT NAMED &ALPHA3 ALREADY EXISTS,
-STP1 &PRINT IS THIS CORRECT? (RESPOND YES OR QUIT)
&SPACE
&READ ARGS
&IF &1 EO YES &EXIT
&IF &1 EO QUIT &EXIT
&SPACE
&PRINT CANNOT DETERMINE LAST ENTRY - TRY AGAIN
&SPACE
&GOTO -STP1
EXECUTIVE = JCL

This executive provides the FILEDEF statements required by an ETAM program making reference to any of the abbreviated data bases.

&TYPE OFF
&COMMENT G.P. FILEDEF EXEC FOR ETAM SYSTEM FILES
&COMMENT NO ENTRY/EXIT PARAMETERS OR ARGUMENTS
FILEDEF MREF DSK ETAM MREF RE F LR 80 BL 80
FILEDEF DIND DSK ETAM DIND RE F LR 80 BL 80
FILEDEF COURSE DSK ETAM COURSE RE F LR 80 BL 80
FILEDEF DCIN DSK ETAM DCIN RE F LR 760 BL 760
FILEDEF DCMP DSK ETAM DCMP RE F LR 1200 BL 1200
FILEDEF DESC DSK ETAM DESC RE F LR 213 BL 213
FILEDEF VEH S DSK ETAM VEH S RE F LR 80 BL 80
FILEDEF DVEH DSK ETAM DVEH RE F LR 750 BL 750
FILEDEF VDESC DSK ETAM VDESC RE F LR 213 BL 213
FILEDEF TASKS DSK ETAM TASKS RE F LR 80 BL 80
FILEDEF DTASK DSK ETAM DTASK RE F LR 1500 BL 1500
FILEDEF TDAT DSK ETAM TDAT RE F LR 213 BL 213
&EXIT

EXECUTIVE = JCL2

Similar to the JCL executive, this executive provides FILEDEF statements for programs requiring access to the files of a specific ETAM project.

&TYPE OFF
&COMMENT G.P. FILEDEFS FOR PROJECT FILES
&COMMENT SINGLE ENTRY PARM IS PROJECT FILETYPE IN
&COMMENT GLOBAL ALPHA VARIABLE 3
FILEDEF ID DSK GALPHA3 ID RE F LR 80 BL 80
FILEDEF REC DSK GALPHA3 REC RE F LR 80 BL 80
FILEDEF REJ DSK GALPHA3 REJ RE F LR 80 BL 80
FILEDEF REV DSK GALPHA3 REV RE F LR 80 BL 80
FILEDEF EXD DSK GALPHA3 EXD RE F LR 80 BL 80
FILEDEF BPT DSK GALPHA3 BPT RE F LR 80 BL 80
FILEDEF RKP DSK GALPHA3 RKP RE F LR 80 BL 80
FILEDEF RRPJ DSK GALPHA3 RRPJ RE F LR 80 BL 80
FILEDEF RRPK DSK GALPHA3 RRPK RE F LR 80 BL 80
FILEDEF VREAL DSK GALPHA3 VREAL RE F LR 80 BL 80
FILEDEF VARF DSK GALPHA3 VARF RE F LR 80 BL 80
FILEDEF SCEN DSK GALPHA3 SCEN RE F LR 80 BL 80
FILEDEF ALTP DSK GALPHA3 ALTP RE F LR 80 BL 80
FILEDEF RESU DSK GALPHA3 RESU RE FB LR 16 BL 800
FILEDEF REE DSK GALPHA3 REE RE FB LR 16 BL 800
FILEDEF SARG DSK GALPHA3 SARG RE F LR 240 BL 240
&EXIT

D.3-3
EXECUTIVE = NONAME

If a program associated with an ETAM project file is invoked without supplying the name of the project, this executive will diagnose the error. A message will be output to inform the User, and action at all levels will be terminated.

&TYPE OFF
&COMMENT OFF THIS EXEC FOR ERROR MESSAGE AND FORCED TERMINATE
&COMMENT IN THE EVENT OF NO PROJECT NAME SUPPLIED.
&COMMENT SINGLE ENTRY PARM IS NAME OF EXEC ENDING.
&SPACE
&PRINT *** NO PROJECT NAME SUPPLIED BY USER,
&PRINT EXEC &1 IS TERMINATING ABNORMALLY ***
&SPACE
&QUIT

EXECUTIVE = ETAM

This is the ETAM master control executive as discussed in Section III. Figure III-18 depicts the control logic implemented into this executive. The text output to the online terminal is illustrated in Figure III-19. Following this executive listing is a listing of NCSS file ETAM/TEXT1 - this contains output text to be used by the ETAM executive.

&TYPE OFF
&COMMENT THIS IS THE OVERALL SYSTEM EXEC
&COMMENT NO ENTRY PARMS REQUIRED
-S1 &SPACE
&PRINT ENTER PROJECT IDENTIFIER, E.G., PN123 (5 CHARACTERS)
&SPACE
&READ ARGS
&SPACE
&IF &INDEX EQ 1 &GOTO -S2
&PRINT CANNOT DECODE LAST ENTRY - TRY AGAIN
&GOTO -S1
-S2 &IF &1 EQ QUIT &QUIT
&ALPHA5 = &1
STATE &ALPHA5 ID P
&IF &INDEX$ EQ 0 &GOTO -S5
-S3 &SPACE
&PRINT PROJECT FILE DOES NOT EXIST
&PRINT GENERATE NEW FILE? ? ?
&PRINT 1) YES
&PRINT 2) NO
&PRINT ENTER: 1 OR 2
&SPACE
GREAD ARGS
GSPACE
IF \$1 EQ 2 &GOTO -S1
IF \$1 EQ QUIT &QUIT
IF \$1 EQ 1 &GOTO -S4
PRINTF CANNOT DECODE LAST ENTRY - TRY AGAIN
GOTO -S5
-S4 EXEC RUNP17 &ALPHA5
-S5 GSPACE
PRINTF SELECT OPTION,
PRINTF 1) ROE
PRINTF 2) ASSESS
PRINTF 3) PRINT
PRINTF 4) NEW PROJECT
PRINTF 5) INSTRUCTIONS
PRINTF 6) QUIT
PRINTF ENTER: 1 ... 6
GSPACE
GREAD ARGS
GSPACE
IF \$1 EQ QUIT &QUIT
IF \$1 EQ 6 &QUIT
IF \$1 EQ 4 &GOTO -S1
IF \$1 EQ 5 &GOTO -MENU
IF \$1 EQ 3 &GOTO -PGM1
IF \$1 EQ 2 &GOTO -ASSESS
IF \$1 EQ 1 &GOTO -ROE
PRINTF CANNOT DECODE LAST ENTRY - TRY AGAIN
GOTO -S5
-ROE GSPACE
PRINTF RANGE-OF-EFFECT, SELECT OPTION
PRINTF 1) EDIT DESCRIPTOR
PRINTF 2) SEARCH
PRINTF 3) PRINT SEARCH
PRINTF 4) EDIT SEARCH
PRINTF 5) PRINT EXTRACT
PRINTF 6) QUIT
PRINTF ENTER: 1 ... 6
GSPACE
GREAD ARGS
GSPACE
IF \$1 EQ QUIT &QUIT
IF \$1 EQ 6 &QUIT
IF \$1 EQ 1 &GOTO -PGM2
IF \$1 EQ 2 &GOTO -PGM3
IF \$1 EQ 3 &GOTO -PGM5A
IF \$1 EQ 4 &GOTO -PGM5B
IF \$1 EQ 5 &GOTO -PGM5C
PRINTF CANNOT DECODE LAST ENTRY - TRY AGAIN
GOTO -ROE

D.3-5
- PGM2 EXEC RUNP2 SALPHA5
  &GOTO -MROE
- PGM3 EXEC RUNP3 SALPHA5
  &GOTO -MROE
- PGM5A EXEC RUNP5A SALPHA5
  &GOTO -MROE
- PGM5B EXEC RUNP5B SALPHA5
  &GOTO -MROE
- PGM5C EXEC RUNP5C SALPHA5
  -MROE &SPACE
  &PRINT CONTINUE ROE TO SELECT OTHER ROUTINES? ? ?
  &PRINT 1) YES
  &PRINT 2) NO
  &PRINT ENTER: 1 OR 2
  &SPACE
  &READ ARGS
  &SPACE
  &IF $1 EQ QUIT &QUIT
  &IF $1 EQ 1 &GOTO -ROE
  &IF $1 EQ 2 &GOTO -S5
  &PRINT CANNOT DECODE LAST ENTRY - TRY AGAIN
  &GOTO -MROE
- ASSESS &SPACE
  &PRINT ASSESSMENT - SELECT OPTION
  &PRINT 1) MODEL
  &PRINT 2) RISK
  &PRINT 3) DECISION
  &PRINT 4) FINANCIAL
  &PRINT 5) QUIT
  &PRINT ENTER: 1 ... 5
  &SPACE
  &READ ARGS
  &SPACE
  &IF $1 EQ QUIT &QUIT
  &IF $1 EQ 5 &QUIT
  &IF $1 EQ 1 &GOTO -NONE
  &IF $1 EQ 2 &GOTO -NONE
  &IF $1 EQ 3 &GOTO -NONE
  &IF $1 EQ 4 &GOTO -NONE
  &PRINT CANNOT DECODE LAST ENTRY - TRY AGAIN
  &GOTO -ASSESS
- NONE &SPACE
  &PRINT PROGRAM NOT INSTALLED.
- MASS &SPACE
  &PRINT CONTINUE ASSESS TO SELECT OTHER ROUTINES? ? ?
  &PRINT 1) YES
  &PRINT 2) NO
  &PRINT ENTER: 1 OR 2
  &SPACE
  &READ ARGS
  &SPACE
  &IF $1 EQ QUIT &QUIT
  &IF $1 EQ 1 &GOTO -ASSESS
  &IF $1 EQ 2 &GOTO -S5
  &PRINT CANNOT DECODE LAST ENTRY - TRY AGAIN
  &GOTO -MASS
FILE = ETAM/TEXT1

This file contains text to be output by the master control executive named ETAM.

P ETAM TEXT1

ETAM INSTRUCTIONS

OPTIONS 1 THRU 6 PERMIT SELECTION OF THE FOLLOWING ETAM FUNCTIONS.

1 WILL CAUSE ENTRY INTO THE ETAM RANGE-OF-EFFECT FUNCTION FOR:
   RUNNING DESCRIPTOR EDIT PROGRAM
   SEARCHING ABBREVIATED DATA BASES
   PRINTING SEARCH RESULTS
   RUNNING SEARCH EDIT PROGRAM
   RUNNING PRINT EXTRACT PROGRAM

2 WILL CAUSE ENTRY INTO THE ETAM ASSESSMENT FUNCTION FOR:
   SETTING UP MODEL INPUTS FOR MODEL RUN
   PACKAGING RISK REDUCTION PROJECTS
   RUNNING DECISION ANALYSIS PROGRAMS
   RUNNING FINANCIAL ANALYSIS PROGRAMS

3 WILL PERMIT PRINTING OF PROJECT DATA BASE FILES

4 WILL CAUSE A RETURN TO THE BEGINNING OF THE ETAM CONTROL EXEC SO THAT A DIFFERENT PROJECT ID CAN BE ENTERED

5 WILL CAUSE THESE ETAM INSTRUCTIONS TO BE REPRINTED

6 WILL END ETAM EXEC CONTROL AND RETURN TO NCSS CONTROL
The current ETAM system has been implemented to provide those features up to and including the Range-of-Effects search process. This has resulted in the identification of forty-one (41) unique data files.

In conformity with the rules of the NCSS time-sharing vendor's online system, a file is made unique by establishing a unique "filename" and/or "filetype". Two filenames are used by the ETAM system, as follows:

**Filename = ETAM:** For those applications and data required by all ETAM projects in common. For instance, all ETAM projects access the same Abbreviated Course data base.

**Filename = Project name:** These files are unique to each given ETAM project. Convention dictates that this project name be of the format "PNxxx" where "xxx" is numeric.
TAEG REPORT NO. 40

FILE REFERENCE NUMBER: 1

NCSS FILENAME/FILETYPE: ETAM/MREF

MEDIA: DASD

FORMAT: F, 80, 80

CREATED/GENERATED BY: Program P18 (zeros contents); for value insertion, see table, below

CONTENTS PRINTED BY: Not Applicable

FILE FUNCTION: Record counts are saved in this file to be used, as required, by various programs of the ETAM system.

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>Fixed Binary Count</td>
<td>2</td>
<td>Fixed Bin</td>
<td>See below, for meaning</td>
</tr>
<tr>
<td>3-80</td>
<td>Repeat of (1-2) Above, Occurs 40 Times Total</td>
<td>78</td>
<td>Fixed Bin</td>
<td></td>
</tr>
</tbody>
</table>

This file contains counts to be used as required by the various programs of the ETAM system. MREF contains a single 80-byte (card image) record; as indicated above, space has been reserved for forty (40) unique fixed binary counts. Current usage is indicated in the table below. Indicated are the programs in which the individual quantities are established and the quantity meaning. Note that some are 0-origin (zero-origin), whereas others are 1-origin (one-origin) in their context.

<table>
<thead>
<tr>
<th>Subscript</th>
<th>Count Origin</th>
<th>Count Related to Filetype</th>
<th>File Ref</th>
<th>Set in Program</th>
<th>Count Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>COURSE</td>
<td>7</td>
<td>P7</td>
<td>Total number of records in the Abbreviated course file</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>DCIN</td>
<td>8</td>
<td>P7</td>
<td>Maximum subscript used in the course file CIN directory</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>DCDP</td>
<td>9</td>
<td>P7</td>
<td>Total number of entries in the course file CDP directory; this quantity should correspond to the subscript (1) value, above</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>DIND</td>
<td>3</td>
<td>P16</td>
<td>Highest subscript used in the Descriptor Master Index</td>
</tr>
<tr>
<td>Subscript</td>
<td>Count Origin</td>
<td>Count Related to Filetype</td>
<td>File Ref</td>
<td>Set in Program</td>
<td>Count Meaning</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------</td>
<td>---------------------------</td>
<td>----------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>SCDESC</td>
<td>5</td>
<td>P7A</td>
<td>Total number of course file descriptor cards</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>DESC</td>
<td>6</td>
<td>P7A</td>
<td>Total number course descriptor records</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>VEHS</td>
<td>15</td>
<td>P8</td>
<td>Total number records in the Abbreviated vehicle data file</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>DVEH</td>
<td>16</td>
<td>P8</td>
<td>Maximum subscript set in the vehicle file directory</td>
</tr>
<tr>
<td>9</td>
<td>1</td>
<td>VDESC</td>
<td>12</td>
<td>P8</td>
<td>Total number of records in the vehicle descriptor file</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>SVCDESC</td>
<td>11</td>
<td>P8</td>
<td>Total number vehicle descriptor cards</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>TASKS</td>
<td>22</td>
<td>P9</td>
<td>Total number record in the Abbreviated vehicle file</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>DTASK</td>
<td>23</td>
<td>P9</td>
<td>Maximum subscript used in the task file directory</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>TDESC</td>
<td>19</td>
<td>P9</td>
<td>Total number records in the task descriptor file</td>
</tr>
<tr>
<td>14</td>
<td>1</td>
<td>STDSCR</td>
<td>18</td>
<td>P9</td>
<td>Total number task descriptor cards</td>
</tr>
</tbody>
</table>
One input card deck contains descriptor decks for all three entities, in following order: Courses, Vehicles, Tasks.

One descriptor deck for a single entity is composed as follows:

Card #1  
Field A = '**'  
Field B = 'COURSES', 'VEHICLE', or 'TASKS'; left justified; remainder of Field B is blank

Card #2  
Field A = Descriptor Number  
Field B = Descriptor Title

The combination of category card (#2, above) followed by as many descriptor cards (#3, above) as appropriate is repeated.

All descriptor numbers must be in ascending sort order, starting with number = one.
The following is a map of the Master Descriptor Index in memory. File number 3 is card image, therefore, 4 80-byte records are required to load the following structure.

Records Nos. 0 thru 3 are the Course index; records 4 thru 7 load the Vehicle index; records 8 thru 11 load the Task index.

The Master Index description cards (see File number 2) are then loaded into this file sequentially, starting at record location 12.

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>Start Card Number for this Entity Category</td>
<td>2</td>
<td>Fixed Bin</td>
<td>Card number = relative number within this file</td>
</tr>
<tr>
<td>3-4</td>
<td>End Card Number for this Entity</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>5-6</td>
<td>Total Number of Descriptor Categories for this Entity</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>7-8</td>
<td>Total Number of Descriptors Active for this Entity</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>9-10</td>
<td>Pointer to 1st Category Card for this Entity</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>11-12</td>
<td>Start Descriptor Number within the Current Category</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>13-14</td>
<td>Last Descriptor Number within Current Category</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>15-98</td>
<td>Pos (9-14) Repeated, 84 Occurs 15 Times Total</td>
<td>84</td>
<td>Fixed Bin</td>
<td>Unused portion of this array is zeroed</td>
</tr>
</tbody>
</table>
The CDESC card file is the data as transmitted to the NCSS time-sharing vendor. This file may be in unsorted format.

The SCDESC file has been sorted into ascending sequence by CIN number. This sort is accomplished in the executive sequence STARTUP.

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-8</td>
<td>CIN Number</td>
<td>8</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Filler</td>
<td>1</td>
<td>Char</td>
<td>Blank</td>
</tr>
<tr>
<td>10-11</td>
<td>2-Digit Descriptor</td>
<td>2</td>
<td>F(2)</td>
<td>P'99'</td>
</tr>
<tr>
<td>12-80</td>
<td>Pos 9-11, Repeated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>23 times</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Notes - Max of 24 descriptors per card; if full card not required, leave remainder blank.

- Each descriptor is a two-digit number; supply lead zero, if applicable.
FILE REFERENCE NUMBER : 6
NCSS FILENAME/FILETYPE : ETAM/DESC
MEDIA : DASD
FORMAT : F, 213, 213
CREATED/GENERATED BY : Program P7A
CONTENTS PRINTED BY : Program P20 (Option C)

FILE FUNCTION : Descriptor data to match the contents of the Abbreviated course data file (File Number 7)

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-8</td>
<td>CIN Number</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>9-13</td>
<td>Filler</td>
<td>5</td>
<td>Char</td>
<td>Blank</td>
</tr>
<tr>
<td>14-213</td>
<td>Descriptor Flags, Occurs (100) Times</td>
<td>200</td>
<td>Fixed Bin</td>
<td>Initially, all flags are zeroed</td>
</tr>
</tbody>
</table>

Note - Descriptor Flags (subscripted 1 to 100, incl.) are NON-ZERO for those descriptors applicable to the Course keyed by the CIN portion of the record.
FILE REFERENCE NUMBER : 7
NCSS FILENAME/FILETYPE : ETAM/COURSE

FILE FUNCTION : Abbreviated Course Data File

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>CDP</td>
<td>4</td>
<td>Char</td>
<td>Blank, if missing</td>
</tr>
<tr>
<td>5-12</td>
<td>CIN</td>
<td>8</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>13-28</td>
<td>Course Short Title</td>
<td>16</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>29-32</td>
<td>NOBC</td>
<td>4</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>33-36</td>
<td>NEC</td>
<td>4</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>37-38</td>
<td>Priority Code</td>
<td>2</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>39-41</td>
<td>RMS Cost Code</td>
<td>3</td>
<td>Char</td>
<td>Last 3 digits only</td>
</tr>
<tr>
<td>42-43</td>
<td>Type Course</td>
<td>2</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>SVC Code</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>Method of Instruction</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>Status Code</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>47-51</td>
<td>Status Date</td>
<td>5</td>
<td>P</td>
<td>P'(5)9'</td>
</tr>
<tr>
<td>52</td>
<td>TRAPS Indicator</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>53-57</td>
<td>TPC</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>58-60</td>
<td>Established Attrition Rate</td>
<td>3</td>
<td>P</td>
<td>P'999V9'</td>
</tr>
<tr>
<td>61-63</td>
<td>Established Setback Rate</td>
<td>3</td>
<td>P</td>
<td>P'999V9'</td>
</tr>
<tr>
<td>64-66</td>
<td>Course Length</td>
<td>3</td>
<td>P</td>
<td>P'999'</td>
</tr>
<tr>
<td>67-69</td>
<td>Total Hours</td>
<td>3</td>
<td>P</td>
<td>P'999'</td>
</tr>
<tr>
<td>Record Position</td>
<td>Data Name</td>
<td>Length</td>
<td>Type</td>
<td>Value/Comments</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
<td>--------</td>
<td>------</td>
<td>---------------</td>
</tr>
<tr>
<td>70-72</td>
<td>Lab Hours</td>
<td>3</td>
<td>P</td>
<td>P'999'</td>
</tr>
<tr>
<td>73-79</td>
<td>Cost Per AOB*</td>
<td>7</td>
<td>P</td>
<td>P'(5)9V99'</td>
</tr>
<tr>
<td>80</td>
<td>Filler</td>
<td>1</td>
<td>Char</td>
<td>Blank</td>
</tr>
</tbody>
</table>

*This field currently not loaded; not available on the NITRAS MCRF Extract Tape.*
FILE REFERENCE NUMBER: 8
NCSS FILENAME/FILETYPE: ETAM/DCIN
MEDIA: DASD
FORMAT: F, 760, 760
CREATED/GENERATED BY: Program P7
CONTENTS PRINTED BY: Not Applicable

FILE FUNCTION: Abbreviated Course File Directory, ordered by CIN Number

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-8</td>
<td>CIN</td>
<td>8</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>9-10</td>
<td>Pointer to Relative Record in Abbreviated Course File (File Number 7)</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>11-760</td>
<td>Repeat of Pos 1-10, Occurs (76) Times Total</td>
<td>750</td>
<td></td>
<td>Unused elements of the Array are blanked (CIN) or zeroed (pointer)</td>
</tr>
</tbody>
</table>
FILE REFERENCE NUMBER : 9
NCSS FILENAME/FILETYPE : ETAM/DCDP

MEDIA : DASD
FORMAT : F, 1200, 1200

CREATED/GENERATED BY : Program P7
CONTENTS PRINTED BY : Not Applicable

FILE FUNCTION : Abbreviated Course File Directory, ordered by CDP Number

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>CDP</td>
<td>4</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>5-6</td>
<td>Pointer to Related</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Record in Abbreviated</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Course File (File</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Number 7)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7-1200</td>
<td>Repeat of Pos (1-6),</td>
<td>1194</td>
<td></td>
<td>Unused elements in the array are blanked (CDP) and zeroed (pointer)</td>
</tr>
</tbody>
</table>
TAEG REPORT NO. 40

FILE REFERENCE NUMBER: 10, 11

NCSS FILENAME/FILETYPE: (10) ETAM/VCDESC; (11) ETAM/SVCDESC

MEDIA: DASD

FORMAT: F, 80, 80

CREATED/GENERATED BY: Not Applicable

CONTENTS PRINTED BY: Not Applicable

FILE FUNCTION: Input card deck with Descriptors as assigned to Vehicle data

The VCDESC card file is the Vehicle Descriptor data as transmitted to the NCSS time-sharing vendor. This file may be in unsorted format.

The SVCDESC file has been sorted into ascending sequence by Vehicle Stock Number. The sort is accomplished by running the executive sequence STARTUP.

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-13</td>
<td>Vehicle Stock Number</td>
<td>13</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Filler</td>
<td>1</td>
<td>Char</td>
<td>Blank</td>
</tr>
<tr>
<td>15-16</td>
<td>2-Digit Descriptor</td>
<td>2</td>
<td>F(2)</td>
<td>p'99'</td>
</tr>
<tr>
<td>17-79</td>
<td>Pos 14-16, Repeated 21 Times</td>
<td>63</td>
<td></td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>Filler</td>
<td>1</td>
<td>Char</td>
<td>Blank</td>
</tr>
</tbody>
</table>

Notes - Max of 22 descriptors per card; if full card not required, leave remainder blank

- Each descriptor is a two-digit number; supply lead zero, if applicable

D.4-12
FILE REFERENCE NUMBER : 12
NCSS FILENAME/FILOTYPE : ETAM/VDESC
MEDIA : DASD
FORMAT : F, 213, 213
CREATED/GENERATED BY : Program P8
CONTENTS PRINTED BY : Program P20 (Option V)

FILE FUNCTION : Descriptor data to match the contents of the Abbreviated Vehicle data file (File Number 15)

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-13</td>
<td>Vehicle Stock Number</td>
<td>13</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14-213</td>
<td>Descriptor Flags, Occurs (100) Times</td>
<td>200</td>
<td>Fixed Bin</td>
<td>Initially, all 100 flags are zeroed</td>
</tr>
</tbody>
</table>

Note - Descriptor Flags (subscripted 1 to 100, incl.) are NON-ZERO for those descriptors applicable to the Vehicle keyed by the first 13 characters of the record.
FILE REFERENCE NUMBER : 13, 14

NCSS FILENAME/FIETYPE : (13) ETAM/VDATA; (14) ETAM/SVDATA

MEDIA : DASD

FORMAT : F, 80, 80

CREATED/GENERATED BY : Not Applicable

CONTENTS PRINTED BY : Not Applicable

FILE FUNCTION : Input card deck with data for the load of the Abbreviated Vehicle data file

The VDATA card deck is the input file as transmitted to the NCSS time-sharing vendor. This file may be in unsorted order.

The SVDATA file has been sorted into ascending sequence by the Vehicle Stock Number portion of the record. This sort is accomplished by running the executive sequence STARTUP.

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-13</td>
<td>Vehicle Stock Number</td>
<td>13</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14-22</td>
<td>Vehicle Designator</td>
<td>9</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>23-69</td>
<td>Vehicle Nomenclature</td>
<td>47</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>70-79</td>
<td>Vehicle Cost</td>
<td>10</td>
<td>F(10,2)</td>
<td>P'(8)ZVZZ'; leave blank, if unused</td>
</tr>
<tr>
<td>80</td>
<td>Filler</td>
<td>1</td>
<td>Char</td>
<td>Blank</td>
</tr>
</tbody>
</table>
FILE REFERENCE NUMBER : 15
NCSS FILENAME/FILETYPE : ETAM/VEHS
MEDIA : DASD
FORMAT : F, 80, 80
CREATED/GENERATED BY : Program P8
CONTENTS PRINTED BY : Program P8A
FILE FUNCTION : Abbreviated Vehicle Data File

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-13</td>
<td>Vehicle Stock Number</td>
<td>13</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14-22</td>
<td>Device Designator</td>
<td>9</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>23-69</td>
<td>Device Name</td>
<td>47</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>70-79</td>
<td>Device Cost</td>
<td>10</td>
<td>F(10,2)</td>
<td>P'(8)9V99' This field currently not available for all records</td>
</tr>
<tr>
<td>80</td>
<td>Filler</td>
<td>1</td>
<td>Char</td>
<td>Blank</td>
</tr>
</tbody>
</table>
TAEG REPORT NO. 40

FILE REFERENCE NUMBER : 16
NCSS FILENAME/FILETYPE : ETAM/DVEH
MEDIA : DASD
FORMAT : F, 750, 750
CREATED/GENERATED BY : Program P8
CONTENTS PRINTED BY : Not Applicable

FILE FUNCTION : Abbreviated Vehicle Directory, ordered by Vehicle Stock Number

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-13</td>
<td>Vehicle Stock Number</td>
<td>13</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14-15</td>
<td>Pointer to Relative Record in Abbreviated Vehicle File (File Number 15)</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>16-750</td>
<td>Repeat of Pos (1-15), above, Occurs 50 Times Total</td>
<td>734</td>
<td></td>
<td>Unused elements of the array are blanked (Vehicle Stock Number) or zeroed (pointer)</td>
</tr>
</tbody>
</table>
FILE REFERENCE NUMBER: 17, 18

NCSS FILENAME/FILETYPE: (17) ETAM/TDSCR; (18) ETAM/STDSCR

MEDIA: DASD

FORMAT: F, 80, 80

CREATED/GENERATED BY: Not Applicable

CONTENTS PRINTED BY: Not Applicable

FILE FUNCTION: Input card deck with Descriptors as assigned to Task data

The TDSCR file is the Task Descriptor data as transmitted to the NCSS time-sharing vendor. This file may be in unsorted sequence.

The STDSCR file has been sorted into ascending sequence by the field represented by the first 13 characters of the record. This sort is accomplished by running the executive sequence STARTUP.

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>Rating</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>6-7</td>
<td>Rank</td>
<td>2</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>8-13</td>
<td>Jobtask Number</td>
<td>6</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14-15</td>
<td>2-Digit Descriptor</td>
<td>2</td>
<td>F(2)</td>
<td>P'99'</td>
</tr>
<tr>
<td>16</td>
<td>Filler</td>
<td>1</td>
<td>Char</td>
<td>Blank</td>
</tr>
<tr>
<td>17-79</td>
<td>Pos 14-16, Repeated</td>
<td>63</td>
<td></td>
<td>Blank</td>
</tr>
<tr>
<td></td>
<td>21 Times</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>80</td>
<td>Filler</td>
<td>1</td>
<td>Char</td>
<td>Blank</td>
</tr>
</tbody>
</table>

Notes - Max of 22 descriptors per card; if full card not required, leave remainder blank
- Each descriptor is a two-digit number; supply lead zero, if applicable.
FILE REFERENCE NUMBER: 19

NCSS FILENAME/FILETYPE: ETAM/TDESC

MEDIA: DASD

FORMAT: F, 213, 213

CREATED/GENERATED BY: Program P9

CONTENTS PRINTED BY: Program P20 (Option T)

FILE FUNCTION: Descriptor data to match the contents of the Abbreviated Task data file (File Number 22)

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>Rating</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>6-7</td>
<td>Rank</td>
<td>2</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>8-13</td>
<td>Job Task Number</td>
<td>6</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14-213</td>
<td>Descriptor Flags, Occurs 100 Times</td>
<td>200</td>
<td>Fixed Bin</td>
<td>Initially, all 100 flags are zeroed</td>
</tr>
</tbody>
</table>

Note - Descriptor Flags (subscripted 1 to 100, incl.) are NON-ZERO for those descriptors applicable to the Task keyed by the first 13 characters of the record.
FILE REFERENCE NUMBER: 20, 21

NCSS FILENAME/FILETYPE: (20) ETAM/TDATA; (21) ETAM/STDATA

MEDIA: DASD

FORMAT: F, 80, 80

CREATED/GENERATED BY: Not Applicable

CONTENTS PRINTED BY: Not Applicable

FILE FUNCTION: Input data deck for the load of the Abbreviated Task data file

The TDATA deck is the input data as transmitted to the NCSS time-sharing vendor. This file may be unsorted.

The STDATA file has been sorted into ascending sequence by the field represented by the first 13 characters of the record. This sort is accomplished by running the executive sequence STARTUP.

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>Rating</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>6-7</td>
<td>Rank</td>
<td>2</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>8-13</td>
<td>Jobtask Number</td>
<td>6</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14-63</td>
<td>Job Description</td>
<td>50</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>64-69</td>
<td>Filler</td>
<td>6</td>
<td>Char</td>
<td>Blank</td>
</tr>
<tr>
<td>70-79</td>
<td>Billet Cost</td>
<td>10</td>
<td>F(10,2)</td>
<td>P'(8)ZVZZ'; leave blank, if unused</td>
</tr>
<tr>
<td>80</td>
<td>Filler</td>
<td>1</td>
<td>An</td>
<td>Blank</td>
</tr>
</tbody>
</table>
FILE REFERENCE NUMBER : 22
NCSS FILENAME/FILETYPE : ETAM/TASKS
MEDIA : DASD
FORMAT : F, 80, 80
CREATED/GENERATED BY : Program P9
CONTENTS PRINTED BY : Program P9A
FILE FUNCTION : Abbreviated Task Data File

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>Rating</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>6-7</td>
<td>Rank</td>
<td>2</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>8-13</td>
<td>Jobtask Number</td>
<td>6</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14-63</td>
<td>Job Description</td>
<td>50</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>64-69</td>
<td>Filler</td>
<td>6</td>
<td>Char</td>
<td>Blank</td>
</tr>
<tr>
<td>70-79</td>
<td>Billet Cost</td>
<td>10</td>
<td>F(10,2)</td>
<td>P'(0)9V99'</td>
</tr>
<tr>
<td>80</td>
<td>Filler</td>
<td>1</td>
<td>Char</td>
<td>Blank</td>
</tr>
</tbody>
</table>
FILE REFERENCE NUMBER: 23
NCSS FILENAME/FILETYPE: ETAM/DTASK
MEDIA: DASD
FORMAT: F, 1500, 1500
CREATED/GENERATED BY: Program P9
CONTENTS PRINTED BY: Not Applicable

FILE FUNCTION: Abbreviated Task File Directory, ordered by the first 13 characters of the Task File record (includes RATING, RANK, and JOBTASK Number)

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-5</td>
<td>Rating</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>6-7</td>
<td>Rank</td>
<td>2</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>8-13</td>
<td>Jobtask Indicator</td>
<td>6</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>14-15</td>
<td>Pointer to Relative Record in Abbreviated Task File (File #22)</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>16-1500</td>
<td>Repeat of (1-15), Above, Occurs 100 Times Total</td>
<td>1484</td>
<td></td>
<td>Unused elements of the array are blanked (all char fields) or zeroed (pointer)</td>
</tr>
</tbody>
</table>
FILE REFERENCE NUMBER : 24
NCSS FILENAME/FILETYPE : Not Applicable
MEDIA : TAPE
FORMAT : FB, 810, 3240, DENSITY = 1600BPI, STANDARD LABEL
CREATED/GENERATED BY : Not Applicable
CONTENTS PRINTED BY : Not Applicable
FILE FUNCTION : Abbreviated Course file input is selected from this tape

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-4</td>
<td>CDP</td>
<td>4</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>5-12</td>
<td>CIN</td>
<td>8</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>13-28</td>
<td>Course Short Title</td>
<td>16</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>29-32</td>
<td>NOBC</td>
<td>4</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>33-36</td>
<td>NEC</td>
<td>4</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>37-39</td>
<td>Off. Crs. Code</td>
<td>3</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>40-44</td>
<td>Priority Des</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>45-48</td>
<td>RMS Cost Code</td>
<td>4</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>49-50</td>
<td>Type Course</td>
<td>2</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>SVC Supp</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>Method Inst.</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>53-67</td>
<td>Filler</td>
<td>15</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>Dept-Code</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>69-71</td>
<td>Filler</td>
<td>3</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>72</td>
<td>Status-Code</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>73-75</td>
<td>Status-Date</td>
<td>3</td>
<td>P</td>
<td>S9(5) COMP-3</td>
</tr>
<tr>
<td>76-83</td>
<td>Prereq-CIN</td>
<td>8</td>
<td>Char</td>
<td></td>
</tr>
</tbody>
</table>
### TAEG REPORT NO. 40

<table>
<thead>
<tr>
<th>Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>84-85</td>
<td>Est. Attr. Rate</td>
<td>2</td>
<td>P</td>
<td>S99V9 COMP-3</td>
</tr>
<tr>
<td>86-87</td>
<td>Est. Stbk. Rate</td>
<td>3</td>
<td>P</td>
<td>S99V9 COMP-3</td>
</tr>
<tr>
<td>88-90</td>
<td>Theory Hrs.</td>
<td>3</td>
<td>P</td>
<td>S9(5) COMP-3</td>
</tr>
<tr>
<td>91-93</td>
<td>Lab Hrs.</td>
<td>3</td>
<td>P</td>
<td>S9(5) COMP-3</td>
</tr>
<tr>
<td>94-96</td>
<td>Filler</td>
<td>3</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>97</td>
<td>TRAPS Ind.</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>98-102</td>
<td>TPC</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>103-107</td>
<td>Stu. UIC</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>108-112</td>
<td>Staff UIC</td>
<td>5</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>113-140</td>
<td>CRS Contact Hrs</td>
<td>28</td>
<td>P</td>
<td>S99 COMP-3</td>
</tr>
<tr>
<td></td>
<td>Contact Ratio</td>
<td>2</td>
<td>P</td>
<td>S99 COMP-3</td>
</tr>
<tr>
<td>141-143</td>
<td>Total Cont Hrs</td>
<td>3</td>
<td>P</td>
<td>S9(5) COMP-3</td>
</tr>
<tr>
<td>144</td>
<td>CFY-Cross Util</td>
<td>1</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>145-147</td>
<td>CFY-Crs Length</td>
<td>3</td>
<td>P</td>
<td>S9(5) COMP-3</td>
</tr>
<tr>
<td>148-150</td>
<td>CFY-Class Frequency</td>
<td>3</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>151-153</td>
<td>CFY-Pers Input</td>
<td>3</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>154-156</td>
<td>CFY-Pers Freq</td>
<td>3</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>157-159</td>
<td>CFY-Equip Input</td>
<td>3</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>160-162</td>
<td>CFY-Equip Freq</td>
<td>3</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>163-165</td>
<td>CFY-Space Input</td>
<td>3</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>166-168</td>
<td>CFY-Space Freq</td>
<td>3</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>169-183</td>
<td>Filler</td>
<td>15</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>184-223</td>
<td>FY+1 Capacities</td>
<td></td>
<td></td>
<td>same as Col. 144-183</td>
</tr>
<tr>
<td>224-263</td>
<td>FY+2 Capacities</td>
<td></td>
<td></td>
<td>same as Col. 144-183</td>
</tr>
<tr>
<td>264-286</td>
<td>CFY OF Plan-USN</td>
<td>3</td>
<td>P</td>
<td>S9(5) Comp-3</td>
</tr>
<tr>
<td>267-269</td>
<td>CFY OF Plan-USNOB</td>
<td>3</td>
<td>P</td>
<td></td>
</tr>
<tr>
<td>270-272</td>
<td>CFY OF Plan-USNR</td>
<td>3</td>
<td>P</td>
<td></td>
</tr>
</tbody>
</table>

D.4-23
<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>273-275</td>
<td>CFY OF Plan-USNRR</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>276-278</td>
<td>CFY OF Plan-USMC</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>279-281</td>
<td>CFY OF Plan-USCG</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>282-284</td>
<td>CFY OF Plan-USA</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>285-287</td>
<td>CFY OF Plan-USAF</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>288-290</td>
<td>CFY OF Plan-NATG</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>291-293</td>
<td>CFY OF Plan-FORMAT</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>294-296</td>
<td>CFY OF Plan-DOD</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>297-299</td>
<td>CFY OF Plan-NDOD</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>300-302</td>
<td>CFY OF Plan-WOM</td>
<td>3</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>303-341</td>
<td>CFY EN Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>342-380</td>
<td>FY+1 OF Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>381-419</td>
<td>FY+1 EN Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>420-458</td>
<td>FY+2 OF Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>459-497</td>
<td>FY+2 EN Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>498-536</td>
<td>FY+3 OF Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>537-575</td>
<td>FY+3 EN Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>576-614</td>
<td>FY+4 OF Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>615-653</td>
<td>FY+4 EN Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>654-692</td>
<td>FY+5 OF Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>693-731</td>
<td>FY+5 EN Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>732-770</td>
<td>FY+6 OF Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>771-809</td>
<td>FY+6 EN Plan</td>
<td>39</td>
<td>p</td>
<td></td>
</tr>
<tr>
<td>810</td>
<td>Filler</td>
<td></td>
<td></td>
<td>These fields are all formatted the same as CFY OF PLAN in Col. 264-302 (13.S9(5) COMP-3)</td>
</tr>
</tbody>
</table>

D.4-24
FILE FUNCTION: Load card file for any/all files of a given project

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>This is the load deck for one or more project files; Program P17 loads the indicated files by reference to control cards present in the load deck. The control cards are:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-4</td>
<td>Control Field</td>
<td>4</td>
<td>Char</td>
<td>&quot;INIT&quot;</td>
</tr>
<tr>
<td>5-80</td>
<td>Filler</td>
<td>76</td>
<td>Char</td>
<td>Blank</td>
</tr>
<tr>
<td>File Card:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1-2</td>
<td>Control Field</td>
<td>2</td>
<td>Char</td>
<td>&quot;**&quot;</td>
</tr>
<tr>
<td>4-7</td>
<td>Filetype; legal</td>
<td>4</td>
<td>Char</td>
<td></td>
</tr>
<tr>
<td>Files are:</td>
<td>ID</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>REC</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>REV</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>REJ</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>EXD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>BPT</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RKP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RRPJ</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RRPK</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VQAL</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>VARF</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SCEN</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ALTP</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
FILE REFERENCE NUMBER : 26

NCSS FILENAME/FILETYPE : Projectname/ID

MEDIA : DASD

FORMAT : F, 80, 80

CREATED/GENERATED BY : Program P17

CONTENTS PRINTED BY : Program P17, P1

FILE FUNCTION : Project Identification Information

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-80</td>
<td>Descriptive information related to a specific project</td>
<td>80</td>
<td>Char</td>
<td></td>
</tr>
</tbody>
</table>

All of the records of this file are free-format except for the first card-image; the format for the first card is:

1-1 Project ID 10

11-80 Project Title 70
NCSS FILENAME/FILETYPE: (27) Projectname/REC  
(28) Projectname/REV  
(29) Projectname/REJ

CREATED/GENERATED BY: Program P17

FILE FUNCTION: Search Descriptor Arguments for Courses (REC), Vehicles (REV), and Tasks (REJ)

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-10</td>
<td>Descriptor Number</td>
<td>10</td>
<td>Char</td>
<td>1/2 Digit Number, Optionally Signed; If Signed, No Blank Between Sign and Leftmost Digit</td>
</tr>
<tr>
<td>11-80</td>
<td>Filler</td>
<td>70</td>
<td>Char</td>
<td>Can Be Used as Comment, If Desired- Not Machine Processable</td>
</tr>
</tbody>
</table>
FILE REFERENCE NUMBER : 30
NCSS FILENAME/FILETYPE : Projectname/SARG

MEDIA : DASD
FORMAT : F,240,240
CREATED/GENERATED BY : Program P2
CONTENTS PRINTED BY : Not Applicable

FILE FUNCTION : Search Argument input to the Range-of-Effects Program P3

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-8 below</td>
<td>Record Type</td>
<td>8</td>
<td>Char</td>
<td>See notes,</td>
</tr>
<tr>
<td>9-10</td>
<td>Total Number of Active Descriptor Search Arguments</td>
<td>2</td>
<td>Fixed Bin</td>
<td></td>
</tr>
<tr>
<td>11-240</td>
<td>Fixed Bin Array, Occurs 115 Times</td>
<td>230</td>
<td>Fixed Bin</td>
<td></td>
</tr>
</tbody>
</table>

Notes: Record type will be indicated by one of the words 'courses', 'vehicles', 'tasks'; there are 3 records in this file, in order as shown.

- For each entity (Courses, Vehicles or Tasks), the numeric array will be zeroed except for those positions which indicate a search designator. The subscript of a non-zero position indicates the designator number. The non-zero contents will be positive for a normal argument, or negative for a "NOT" argument.
FILE REFERENCE NUMBER: 31, 32

NCSS FILENAME/FIETYPE: (31) Projectname/RESU
(32) Projectname/REE

MEDIA: DASD

FORMAT: F, 16, 16

CREATED/GENERATED BY: (RESU) by Program P3
(REE) by Program P5B

CONTENTS PRINTED BY: P1

FILE FUNCTION: RESU is the un-edited output of the Range-of-Effects Search Program P3; REE is the edited output of Program P5B

<table>
<thead>
<tr>
<th>Record Position</th>
<th>Data Name</th>
<th>Length</th>
<th>Type</th>
<th>Value/Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Record Type</td>
<td>1</td>
<td>Char</td>
<td>P'9' 1 = Course 2 = Vehicle 3 = Task</td>
</tr>
<tr>
<td>2</td>
<td>Record Status</td>
<td>1</td>
<td>Char</td>
<td>P'9' 1 = Original Search Result 2 = Added Via Interactive Program 5B 3 = Deleted Via Program P5B</td>
</tr>
<tr>
<td>3</td>
<td>Abbreviated Data Status</td>
<td>1</td>
<td>Char</td>
<td>P'9' 1 = Abbreviated Data is Available 2 = No Data Match in Current Abbr File</td>
</tr>
<tr>
<td>4-16</td>
<td>Entity Identifier, As Follows:</td>
<td>13</td>
<td>Char</td>
<td>Courses: 4-11 CIN 8 12-15 CDP 4 if available 16 Filler 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Vehicles: 4-16 Veh Stock No.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Tasks : 4-8 Rating 5 9-10 Rank 2 11-16 Jobtask 6 Number</td>
</tr>
</tbody>
</table>

D.4-29
FILE REFERENCE NUMBER: 33, 34, 35, 36, 37, 38, 39, 40, 41

NCSS FILENAME/FILETYPE:
(33) Projectname/EXD
(34) Projectname/BPT
(35) Projectname/RKP
(36) Projectname/RRPJ
(37) Projectname/RRPK
(38) Projectname/VQAL
(39) Projectname/VARF
(40) Projectname/SCEN
(41) Projectname/ALTP

MEDIA: DASD

FORMAT: F, 80, 80

CREATED/GENERATED BY: Program P17

CONTENTS PRINTED BY: Program P1

FILE FUNCTION: Miscellaneous card-image files; can be initialized by Program P17

None of these files perform a functional task in the Range-of-Effects search portion of ETAM. Their function is explained in previous section of this document that present the system design for those segments of ETAM that are to follow the ROE search.

Since these are card-files, they may initialized by providing proper input to the Project File Initialization Program (P17).
SECTION D.5

MISCELLANEOUS CONTROL CARD SOURCE LISTINGS

Following are listings of the control card sequences used to build the current ETAM Range-of-Effects system at the NCSS time-sharing vendor.

All control decks are NCSS filename = ETAM. They are presented within this section in alphabetic order by filetype. Refer to the File Reference Number as indicated in the following table for the specific format of each control deck.

<table>
<thead>
<tr>
<th>Filename</th>
<th>Filetype</th>
<th>Reference Number</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>ETAM</td>
<td>CDESC</td>
<td>4</td>
<td>Course Descriptor data, as transmitted to the NCSS time-sharing vendor's computer site</td>
</tr>
<tr>
<td>ETAM</td>
<td>COURSE</td>
<td>7</td>
<td>This is the Abbreviated Course File; normally, this is output from Program P14. To conserve time, use of the complete NITRAS MCRF Extract tape and P14 were bypassed</td>
</tr>
<tr>
<td>ETAM</td>
<td>DICRDS</td>
<td>2</td>
<td>Control card input for the generation of the Descriptor Master Index File (File Number 3, ETAM/DIND)</td>
</tr>
<tr>
<td>ETAM</td>
<td>TDATA</td>
<td>20</td>
<td>Data load for the Abbreviated Task data file (File Reference 22)</td>
</tr>
<tr>
<td>ETAM</td>
<td>TDSCR</td>
<td>17</td>
<td>Data load for the Descriptor information associated with Task type entities</td>
</tr>
<tr>
<td>ETAM</td>
<td>VCDESC</td>
<td>10</td>
<td>Descriptor load data for entity type Vehicles</td>
</tr>
<tr>
<td>ETAM</td>
<td>VDATA</td>
<td>13</td>
<td>Data load for the Abbreviated Vehicle data file (File Reference 15)</td>
</tr>
</tbody>
</table>

D.5-1
<table>
<thead>
<tr>
<th>PAGE NUMBER</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOURCE CARD LISTING FOR</td>
<td></td>
</tr>
<tr>
<td>OFFLINE READ ETAM CODESC</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OFFLINE READ</th>
<th>ETAM CODESC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td></td>
</tr>
</tbody>
</table>

**TAEG REPORT NO. 40**

**D.5-2**
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>69</td>
<td>01 INSTRUCTOR</td>
</tr>
<tr>
<td>70</td>
<td>02 STATIC GRAPHICS</td>
</tr>
<tr>
<td>71</td>
<td>03 ANIMATED GRAPHICS</td>
</tr>
<tr>
<td>72</td>
<td>04 AUDIO</td>
</tr>
<tr>
<td>73</td>
<td>05 PHYSICAL MODELS</td>
</tr>
<tr>
<td>74</td>
<td>06 PROCEDURAL TRAINERS: SYMBOLIC</td>
</tr>
<tr>
<td>75</td>
<td>07 PROCEDURAL TRAINERS: PHYSICAL BUT NON-FUNCTIONAL</td>
</tr>
<tr>
<td>76</td>
<td>08 PROCEDURAL TRAINERS: FUNCTIONAL</td>
</tr>
<tr>
<td>77</td>
<td>09 TASK &amp; SYSTEM SIMULATORS</td>
</tr>
<tr>
<td>78</td>
<td>10 REAL EQUIPMENT ITSELF</td>
</tr>
<tr>
<td>79</td>
<td>11 REFERENCE KNOWLEDGE</td>
</tr>
<tr>
<td>80</td>
<td>12 KNOWLEDGE, TASK SPECIFIC/ENABLING</td>
</tr>
<tr>
<td>81</td>
<td>13 TASK-SKILL FORMATS</td>
</tr>
<tr>
<td>82</td>
<td>14 SKILL TRAINING</td>
</tr>
<tr>
<td>83</td>
<td>** VEHICLE PROPERTIES</td>
</tr>
<tr>
<td>84</td>
<td>21 VISUAL</td>
</tr>
<tr>
<td>85</td>
<td>22 AUDITORY</td>
</tr>
<tr>
<td>86</td>
<td>23 KINESCEPTIC/VESTIBULAR</td>
</tr>
<tr>
<td>87</td>
<td>24 TACTILE</td>
</tr>
<tr>
<td>88</td>
<td>** TYPE OF CONTENT DISPLAYED</td>
</tr>
<tr>
<td>89</td>
<td>31 TEXT-VERBAL</td>
</tr>
<tr>
<td>90</td>
<td>32 DIAGRAMMATIC</td>
</tr>
<tr>
<td>91</td>
<td>33 ABSTRACTED PICTORIAL REPRESENTATION</td>
</tr>
<tr>
<td>92</td>
<td>34 PICTORIAL REPRESENTATIONS</td>
</tr>
<tr>
<td>93</td>
<td>35 PHYSICAL REPRESENTATIONS</td>
</tr>
<tr>
<td>94</td>
<td>36 OTHER</td>
</tr>
<tr>
<td>95</td>
<td>** TYPE OF PRESENTATIONAL SEQUENCE</td>
</tr>
<tr>
<td>96</td>
<td>41 LIBRARY OF FRAMES OR ITEMS</td>
</tr>
<tr>
<td>97</td>
<td>42 PRESENTATION SEQUENCE NOT APPLICABLE</td>
</tr>
<tr>
<td>98</td>
<td>43 FIXED SEQUENTIAL FRAMES OR ITEMS</td>
</tr>
<tr>
<td>99</td>
<td>44 RANDOM SELECTION OF FRAME SEQUENCES</td>
</tr>
<tr>
<td>100</td>
<td>45 DYNAMIC CHANGE OF CONTENT WITHIN FRAME</td>
</tr>
<tr>
<td>101</td>
<td>** SELECTION SOURCE FOR SEQUENCING</td>
</tr>
<tr>
<td>102</td>
<td>51 INTERNAL PROGRAM</td>
</tr>
<tr>
<td>103</td>
<td>52 INSTRUCTOR</td>
</tr>
<tr>
<td>104</td>
<td>53 STUDENT CHOICE</td>
</tr>
<tr>
<td>105</td>
<td>54 STUDENT PERFORMANCE</td>
</tr>
<tr>
<td>106</td>
<td>55 COMBINATIONS OF THE ABOVE</td>
</tr>
<tr>
<td>107</td>
<td>** TYPE OF EXTERNAL CONTROL OPERATED BY STUDENT</td>
</tr>
<tr>
<td>108</td>
<td>61 NOT APPLICABLE DIRECTLY</td>
</tr>
<tr>
<td>109</td>
<td>62 ARTIFICIAL OR SYMBOLIC RESPONSE</td>
</tr>
<tr>
<td>110</td>
<td>63 REPRESENTATIONAL RESPONSE BY SYMBOLIC SELECTION</td>
</tr>
<tr>
<td>111</td>
<td>64 REPRESENTATIONAL RESPONSE BY DUMMY CONTROL ACTIVATION</td>
</tr>
<tr>
<td>112</td>
<td>65 TASK-MANIPULATIVE RESPONSE, NON-DYNAMIC IN TIME AND FORCE</td>
</tr>
<tr>
<td>113</td>
<td>66 TASK-MANIPULATIVE RESPONSE, DYNAMIC AND INTERACTIVE</td>
</tr>
<tr>
<td>114</td>
<td>** FEEDBACK PRESENTATION LOGIC</td>
</tr>
<tr>
<td>115</td>
<td>71 NOT APPLICABLE</td>
</tr>
<tr>
<td>116</td>
<td>72 SELECTS NEXT STIMULUS ITEM OR SEQUENCE</td>
</tr>
<tr>
<td>117</td>
<td>73 GIVES EVALUATION OF PRECEDING RESPONSE</td>
</tr>
<tr>
<td>118</td>
<td>74 SELECTS AND PRESENTS GUIDANCE INFO</td>
</tr>
<tr>
<td>119</td>
<td>** RESPONSE EVALUATION LOGIC</td>
</tr>
<tr>
<td>120</td>
<td>81 NOT INTERNAL-DEPENDS ON INSTRUCTOR OR STUDENT EVALUATION</td>
</tr>
<tr>
<td>121</td>
<td>82 EVALUATION EXTENDED TO A SET OF STUDENT RESPONSES</td>
</tr>
<tr>
<td>122</td>
<td>83 EVALUATION EXTENDED TO A SET OF STUDENT RESPONSES</td>
</tr>
<tr>
<td>123</td>
<td>84 TOLERANCE LIMITS ON ACCEPTABLE STUDENT RESPONSE: FIXED</td>
</tr>
<tr>
<td>124</td>
<td>85 TOLERANCE LIMITS ON ACCEPTABLE STUDENT RESPONSE: VARIABLE</td>
</tr>
<tr>
<td>125</td>
<td></td>
</tr>
<tr>
<td>126</td>
<td>** ADMINISTRATIVE ROUTINE PAPERWORK</td>
</tr>
<tr>
<td>127</td>
<td>01 FORMS FILLING</td>
</tr>
<tr>
<td>128</td>
<td>02 DOCUMENT-FILE MANAGEMENT</td>
</tr>
<tr>
<td>129</td>
<td>03 DECODE-ENCODE</td>
</tr>
<tr>
<td>130</td>
<td>04 SCREEN-FILTER</td>
</tr>
<tr>
<td>131</td>
<td>05 OTHER</td>
</tr>
<tr>
<td>132</td>
<td>** ADMINISTRATIVE NON-ROUTINE PAPERWORK</td>
</tr>
<tr>
<td>133</td>
<td>11 CONSTRUCT MESSAGES-REPORTS</td>
</tr>
<tr>
<td>134</td>
<td>12 ANALYZE-INTERPRET</td>
</tr>
<tr>
<td>135</td>
<td>13 CONSTRUCT RECOMMENDATION-PROPOSAL</td>
</tr>
<tr>
<td>136</td>
<td>14 CONSTRUCT PLAN</td>
</tr>
</tbody>
</table>

---

**TAEG REPORT NO. 40**

*Source Card Listing for Offline Read 'Etam DiCrds*
<table>
<thead>
<tr>
<th>Source Card Listing for Offline Read ETAM DICRDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>138 15 OTHER</td>
</tr>
<tr>
<td>139 22 Administrative Office Equipment Operation</td>
</tr>
<tr>
<td>140 21 Typewriter</td>
</tr>
<tr>
<td>141 23 Telephone, etc.</td>
</tr>
<tr>
<td>142 24 Computer Terminal</td>
</tr>
<tr>
<td>143 25 Teletype</td>
</tr>
<tr>
<td>144 26 OTHER</td>
</tr>
<tr>
<td>145 ** Interpersonal: Inform-Instruct-Manage</td>
</tr>
<tr>
<td>146 31 Brief-Debrief</td>
</tr>
<tr>
<td>147 32 Instruct-Train</td>
</tr>
<tr>
<td>148 33 Assign, Monitor, Coordinate</td>
</tr>
<tr>
<td>149 34 Evaluate</td>
</tr>
<tr>
<td>150 35 OTHER</td>
</tr>
<tr>
<td>151 ** Technical Procedures</td>
</tr>
<tr>
<td>152 41 Sequential</td>
</tr>
<tr>
<td>153 42 Strategic, Adaptive</td>
</tr>
<tr>
<td>154 43 Interpersonal, Team Member</td>
</tr>
<tr>
<td>155 44 OTHER</td>
</tr>
<tr>
<td>156 ** Technical Type of Procedure (Main Emphasis)</td>
</tr>
<tr>
<td>157 51 Scan-Detect: Symbolic (Incl. Maps, Radar, Transduced Signals)</td>
</tr>
<tr>
<td>158 52 Scan-Detect: Natural</td>
</tr>
<tr>
<td>159 53 Identify: Symbolic (Incl. Transduced Signals)</td>
</tr>
<tr>
<td>160 54 Identify: Natural</td>
</tr>
<tr>
<td>161 55 Interpret: Symbolic</td>
</tr>
<tr>
<td>162 56 Interpret: Natural</td>
</tr>
<tr>
<td>163 57 Perceptual-Motor</td>
</tr>
<tr>
<td>164 58 Cognitive Operations</td>
</tr>
<tr>
<td>165 59 Manual</td>
</tr>
<tr>
<td>166 60 Communicate</td>
</tr>
<tr>
<td>167 ** Technical With or Without Equipment</td>
</tr>
<tr>
<td>168 61 With Equipment (Paper Is Equipment)</td>
</tr>
<tr>
<td>169 62 Without Equipment</td>
</tr>
<tr>
<td>170 ** Technical Decide</td>
</tr>
<tr>
<td>171 71 Diagnose-Analyze</td>
</tr>
<tr>
<td>172 72 Select-Choose</td>
</tr>
<tr>
<td>173 73 Under Stress or Load</td>
</tr>
<tr>
<td>174 74 OTHER</td>
</tr>
<tr>
<td>175 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>176 81 Manual Construct or Repair</td>
</tr>
<tr>
<td>177 82 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>178 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>179 91 Applicable</td>
</tr>
<tr>
<td>180 92 Time Stress</td>
</tr>
<tr>
<td>181 93 Information-Load Stress</td>
</tr>
<tr>
<td>182 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>183 94 Manual Construct or Repair</td>
</tr>
<tr>
<td>184 95 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>185 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>186 96 Applicable</td>
</tr>
<tr>
<td>187 97 Time Stress</td>
</tr>
<tr>
<td>188 98 Information-Load Stress</td>
</tr>
<tr>
<td>189 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>190 99 Manual Construct or Repair</td>
</tr>
<tr>
<td>191 100 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>192 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>193 101 Applicable</td>
</tr>
<tr>
<td>194 102 Time Stress</td>
</tr>
<tr>
<td>195 103 Information-Load Stress</td>
</tr>
<tr>
<td>196 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>197 104 Manual Construct or Repair</td>
</tr>
<tr>
<td>198 105 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>199 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>200 106 Applicable</td>
</tr>
<tr>
<td>201 107 Time Stress</td>
</tr>
<tr>
<td>202 108 Information-Load Stress</td>
</tr>
<tr>
<td>203 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>204 109 Manual Construct or Repair</td>
</tr>
<tr>
<td>205 110 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>206 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>207 111 Applicable</td>
</tr>
<tr>
<td>208 112 Time Stress</td>
</tr>
<tr>
<td>209 113 Information-Load Stress</td>
</tr>
<tr>
<td>210 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>211 114 Manual Construct or Repair</td>
</tr>
<tr>
<td>212 115 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>213 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>214 116 Applicable</td>
</tr>
<tr>
<td>215 117 Time Stress</td>
</tr>
<tr>
<td>216 118 Information-Load Stress</td>
</tr>
<tr>
<td>217 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>218 119 Manual Construct or Repair</td>
</tr>
<tr>
<td>219 120 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>220 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>221 121 Applicable</td>
</tr>
<tr>
<td>222 122 Time Stress</td>
</tr>
<tr>
<td>223 123 Information-Load Stress</td>
</tr>
<tr>
<td>224 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>225 124 Manual Construct or Repair</td>
</tr>
<tr>
<td>226 125 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>227 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>228 126 Applicable</td>
</tr>
<tr>
<td>229 127 Time Stress</td>
</tr>
<tr>
<td>230 128 Information-Load Stress</td>
</tr>
<tr>
<td>231 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>232 129 Manual Construct or Repair</td>
</tr>
<tr>
<td>233 130 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>234 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>235 131 Applicable</td>
</tr>
<tr>
<td>236 132 Time Stress</td>
</tr>
<tr>
<td>237 133 Information-Load Stress</td>
</tr>
<tr>
<td>238 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>239 134 Manual Construct or Repair</td>
</tr>
<tr>
<td>240 135 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>241 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>242 136 Applicable</td>
</tr>
<tr>
<td>243 137 Time Stress</td>
</tr>
<tr>
<td>244 138 Information-Load Stress</td>
</tr>
<tr>
<td>245 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>246 139 Manual Construct or Repair</td>
</tr>
<tr>
<td>247 140 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>248 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>249 141 Applicable</td>
</tr>
<tr>
<td>250 142 Time Stress</td>
</tr>
<tr>
<td>251 143 Information-Load Stress</td>
</tr>
<tr>
<td>252 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>253 144 Manual Construct or Repair</td>
</tr>
<tr>
<td>254 145 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>255 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>256 146 Applicable</td>
</tr>
<tr>
<td>257 147 Time Stress</td>
</tr>
<tr>
<td>258 148 Information-Load Stress</td>
</tr>
<tr>
<td>259 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>260 149 Manual Construct or Repair</td>
</tr>
<tr>
<td>261 150 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>262 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>263 151 Applicable</td>
</tr>
<tr>
<td>264 152 Time Stress</td>
</tr>
<tr>
<td>265 153 Information-Load Stress</td>
</tr>
<tr>
<td>266 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>267 154 Manual Construct or Repair</td>
</tr>
<tr>
<td>268 155 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>269 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>270 156 Applicable</td>
</tr>
<tr>
<td>271 157 Time Stress</td>
</tr>
<tr>
<td>272 158 Information-Load Stress</td>
</tr>
<tr>
<td>273 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>274 159 Manual Construct or Repair</td>
</tr>
<tr>
<td>275 160 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>276 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>277 161 Applicable</td>
</tr>
<tr>
<td>278 162 Time Stress</td>
</tr>
<tr>
<td>279 163 Information-Load Stress</td>
</tr>
<tr>
<td>280 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>281 164 Manual Construct or Repair</td>
</tr>
<tr>
<td>282 165 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>283 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>284 166 Applicable</td>
</tr>
<tr>
<td>285 167 Time Stress</td>
</tr>
<tr>
<td>286 168 Information-Load Stress</td>
</tr>
<tr>
<td>287 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>288 169 Manual Construct or Repair</td>
</tr>
<tr>
<td>289 170 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>290 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>291 171 Applicable</td>
</tr>
<tr>
<td>292 172 Time Stress</td>
</tr>
<tr>
<td>293 173 Information-Load Stress</td>
</tr>
<tr>
<td>294 ** Technical Construct-Repair, Plan</td>
</tr>
<tr>
<td>295 174 Manual Construct or Repair</td>
</tr>
<tr>
<td>296 175 Cognitive Construct, Plan</td>
</tr>
<tr>
<td>297 ** Technical Track-Aim-Steer</td>
</tr>
<tr>
<td>298 176 Applicable</td>
</tr>
<tr>
<td>299 177 Time Stress</td>
</tr>
<tr>
<td>300 178 Information-Load Stress</td>
</tr>
<tr>
<td>PAGE NUMBER : 1</td>
</tr>
<tr>
<td>----------------</td>
</tr>
<tr>
<td>SOURCE CARD LISTING FOR</td>
</tr>
<tr>
<td>OFFLINE READ ETAM TDATA</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OFFLINE READ</th>
<th>ETAM TDATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SM 3 34009 REEVE AND SPLICE MAST YARDS</td>
</tr>
<tr>
<td>2</td>
<td>SM 3 42020 RENDER PASSING AND SIDE HONORS</td>
</tr>
<tr>
<td>3</td>
<td>SM 3 50303 USE MAINTENANCE REQUIREMENT CARDS (MRC)</td>
</tr>
<tr>
<td>4</td>
<td>SM 3 50928 COMPLETE MAINTENANCE DATA FORMS</td>
</tr>
<tr>
<td>5</td>
<td>SM 3 52001 PERFORM ROUTINE MAINTENANCE ON OPTICAL EQUIPMENT</td>
</tr>
<tr>
<td>6</td>
<td>SM 3 77264 SECURITY OF COMMUNICATIONS CLASSIFIED MATERIAL</td>
</tr>
<tr>
<td>7</td>
<td>SM 3 84225 OBSERVE COMM. TRANSMISSION SECURITY REQUIREMENTS</td>
</tr>
<tr>
<td>8</td>
<td>SM 3 86015 RECOGNIZE, USE AND INTERPRET EMERGENCY SIGNALS</td>
</tr>
<tr>
<td>9</td>
<td>SM 3 86016 TRANSMIT, RECEIVE FLASHING LIGHT CODE AT 6 GPM</td>
</tr>
<tr>
<td>10</td>
<td>SM 3 86017 TRANSMIT, RECEIVE FLASHING LIGHT PLAIN LANGUAGE</td>
</tr>
<tr>
<td>11</td>
<td>SM 3 86018 TRANSMIT, RECEIVE Semaphore PLAIN LANGUAGE</td>
</tr>
<tr>
<td>12</td>
<td>SM 3 86020 SELECT AND DISPLAY FLAGS AND PENNANTS</td>
</tr>
<tr>
<td>13</td>
<td>SM 3 86024 EXERCISE RULES OF VISUAL RESPONSIBILITY</td>
</tr>
<tr>
<td>14</td>
<td>SM 3 86025 IDENTIFY FLAGS AND ENSIGNS OF MARITIME NATIONS</td>
</tr>
<tr>
<td>15</td>
<td>SM 3 86031 PREPARE A VISUAL MESSAGE FOR TRANSMISSION</td>
</tr>
<tr>
<td>16</td>
<td>SM 3 86035 RECEIVE, SEND MESSAGES ACCORDING TO PRECEDENCE</td>
</tr>
<tr>
<td>17</td>
<td>SM 3 86038 SIGNALMAN DUTIES AS A MEMBER OF A BOATCREW</td>
</tr>
<tr>
<td>18</td>
<td>SM 3 86365 OPERATE ELECTRICAL AND ELECTRONIC VISUAL SIGNALS</td>
</tr>
<tr>
<td>19</td>
<td>SM 3 86366 OPERATE INFRARED AND FLASHING LIGHT EQUIPMENT</td>
</tr>
<tr>
<td>20</td>
<td>SM 3 86430 ROUTINE UPKEEP OF VISUAL SIGNALING EQUIPMENT</td>
</tr>
<tr>
<td>21</td>
<td>SM 3 86605 FILE RECORDS AND CORRESPONDENCE</td>
</tr>
<tr>
<td>22</td>
<td>SM 3 86606 ORDER AND MAINTAIN STOCK OF OFFICE SUPPLIES</td>
</tr>
<tr>
<td>23</td>
<td>SM 3 88822 PERFORM AS MAST YEOMAN</td>
</tr>
<tr>
<td>24</td>
<td>SM 3 88824 PERFORM RECEPTIONIST DUTIES</td>
</tr>
<tr>
<td>25</td>
<td>SM 3 88825 OPERATE DUPLICATING EQUIPMENT</td>
</tr>
<tr>
<td>26</td>
<td>SM 3 88826 OPERATE DUPLICATING EQUIPMENT AT 30 WORDS PER MINUTE</td>
</tr>
<tr>
<td>27</td>
<td>SM 3 88827 TRANSCRIBE OFFICER MESSAGE ORDERS</td>
</tr>
<tr>
<td>28</td>
<td>SM 3 88830 LEAVE REGULATIONS, PREPARE LEAVE AUTHORIZATIONS</td>
</tr>
<tr>
<td>29</td>
<td>SM 3 88831 TYPE CORRESPONDENCE AND MESSAGES</td>
</tr>
<tr>
<td>30</td>
<td>SM 3 88834 ADMINISTRATIVE OPERATIONAL RECORDS AND REPORTS</td>
</tr>
<tr>
<td>31</td>
<td>SM 3 88835 INITIATE JOB ORDERS AND WORK REQUESTS</td>
</tr>
<tr>
<td>32</td>
<td>SM 3 88843 MAINTAIN OFFICER, WARRANT OFFICER SERVICE RECORDS</td>
</tr>
<tr>
<td>33</td>
<td>SM 3 88849 TYPE AT 30 WORDS PER MINUTE</td>
</tr>
<tr>
<td>34</td>
<td>SM 3 88852 TYPE INSTRUCTIONS AND NOTICES</td>
</tr>
<tr>
<td>35</td>
<td>SM 3 88854 HANDLE OFFICIAL DUTIES DURING MAN OVERBOARD EXERCISE</td>
</tr>
<tr>
<td>36</td>
<td>SM 3 88855 IDENTIFY FLAGS AND ENSIGNS OF MARITIME NATIONS</td>
</tr>
<tr>
<td>37</td>
<td>SM 3 88856 TRANSMIT, RECEIVE FLASHING LIGHT PLAIN LANGUAGE</td>
</tr>
<tr>
<td>38</td>
<td>SM 3 88857 TRANSMIT, RECEIVE Semaphore PLAIN LANGUAGE</td>
</tr>
<tr>
<td>39</td>
<td>SM 3 88858 PERFORM DUTIES AS MAST YEOMAN</td>
</tr>
<tr>
<td>40</td>
<td>SM 3 88859 PREPARE TEMAOD ORDERS FOR DECORATIONS AND AWARDS</td>
</tr>
<tr>
<td>41</td>
<td>SM 3 88860 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>42</td>
<td>SM 3 88864 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>43</td>
<td>SM 3 88866 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>44</td>
<td>SM 3 88868 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>45</td>
<td>SM 3 88869 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>46</td>
<td>SM 3 88870 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>47</td>
<td>SM 3 88871 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>48</td>
<td>SM 3 88872 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>49</td>
<td>SM 3 88873 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>50</td>
<td>SM 3 88874 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>51</td>
<td>SM 3 88875 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>52</td>
<td>SM 3 88876 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>53</td>
<td>SM 3 88877 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>54</td>
<td>SM 3 88878 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>55</td>
<td>SM 3 88879 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>56</td>
<td>SM 3 88880 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>57</td>
<td>SM 3 88881 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>58</td>
<td>SM 3 88882 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>59</td>
<td>SM 3 88883 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>60</td>
<td>SM 3 88884 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>61</td>
<td>SM 3 88885 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>62</td>
<td>SM 3 88886 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>63</td>
<td>SM 3 88887 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>64</td>
<td>SM 3 88888 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>65</td>
<td>SM 3 88889 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>66</td>
<td>SM 3 88890 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>67</td>
<td>SM 3 88891 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
<tr>
<td>68</td>
<td>SM 3 88892 HANDLE OBSELETE FILES AND RECORDS</td>
</tr>
</tbody>
</table>

---

D.5-9
TAEG REPORT NO. 40

PAGE NUMBER : 2

SOURCE CARD LISTING FOR
OFFLINE READ  ETAM TDATA

69  YN  1  38629  APPLY PRINCIPLES OF WORK SIMPLIFICATION
70  YN  1  38630  MONITOR THE REPORTS CONTROL SYSTEM
71  YN  1  38631  PREPARE MANPOWER AUTHORIZATION CHANGE REQUESTS
72  YN  1  38636  REVIEW COMPLETED JOB ORDERS AND WORK REQUESTS
73  YN  1  38655  PAPERWORK FOR INFORMAL/FORMAL INVESTIGATIONS
74  YN  1  38656  TYPE AT 50 WORDS PER MINUTE
75  YN  1  38658  CARRY OUT CASUALTY REPORTING, ASSISTING PROCEDURES
76  YN  1  38663  PAPERWORK FOR SUMMARY AND SPECIAL COURTS-MARTIAL
77  YN  1  38690  PREPARE CORRESPONDENCE AND MESSAGES
78  YN  1  38691  DRAFT INSTRUCTIONS AND NOTICES
79  YN  1  46361  SUPERVISE ALL ASPECTS OF PUBLICATION HANDLING
80  YN  1  65480  ASSIST THE CASUALTY ASSISTANCE CALLS OFFICER
81  YN  1  77285  MONITOR CLASSIFIED MATERIALS SECURITY COMPLIANCE
82  YN  C  38639  APPLY OFFICE MANAGEMENT CONCEPTS AND TECHNIQUES
83  YN  C  38780  MONITOR CORRESPONDENCE AND FORMS TICKLER SYSTEM
84  YN  C  38786  ADMINISTER CLASSIFIED MATERIAL CONTROL SYSTEM
85  YN  C  38788  DOCUMENTS THAT REQUIRE OFFICIAL SIGNATURE
86  YN  C  38861  ORGANIZATION BOOK, BILLS AND PERMANENT DIRECTIVES
87  YN  C  38862  MANAGEMENT OF ADMINISTRATIVE OFFICE
88  YN  C  38863  MONITOR ALL REPORTS AND OUTGOING CORRESPONDENCE
89  YN  C  38864  MONITOR OFFICER FITNESS REPORTS
90  YN  C  38866  INTERPRET AND ANALYZE MANPOWER AUTHORIZATIONS
91  YN  C  44462  ESTABLISH TRAINING IN JUMPS AND MAPMS PROCEDURES
92  YN  C  65490  ADVISE PAY, ALLOWANCES AND TRAVEL MATTERS
93  YN  C  38547  MONITOR OPERATIONS IN OWN AREA OF RESPONSIBILITY
94  YN  C  38548  PREPARE LOCAL DIRECTIVES AND INSTRUCTIONS
95  YN  C  38549  INSURE MAXIMUM PERSONNEL UTILIZATION
96  YN  C  38551  SUPERVISE USE OF OFFICE MATERIALS AND EQUIPMENT
97  YN  C  38554  ADMINISTER RECORDS DISPOSAL PROGRAM
98  YN  C  38556  MONITOR NAVAL ACTIVITY ADMIN. STAFF FUNCTIONS
99  YN  C  38557  MONITOR ALL ASPECTS OF RECORDS, FORMS MGMT. PGM.
100 YN  C  38558  ALL ASPECTS OF OFFICER SERVICE RECORDS MAINTENANCE
101 YN  C  38437  TRAINING, EFFECTIVENESS, IMPROVEMENT INITIATION
102 YN  C  38543  CONTROL ADMINISTRATION ACTIVITIES
103 YN  C  38548  FUTURE PLANNING IN OWN AREA OF RESPONSIBILITY
104 YN  C  38549  ESTABLISH GOALS IN OWN AREA OF RESPONSIBILITY
105 YN  C  38542  REVIEW PERSONNEL, EQUIP. MATERIAL REQUIREMENTS
106 YN  C  38559  PERFORM ORGANIZATIONAL ANALYSES
107 YN  C  38529  DEVELOP OPERATING BUDGETS, MONITOR EXPENDITURES
108 BM  3  11511  LOCATE DAMAGE-CONTROL FITTINGS AND EQUIPMENT
109 BM  3  16061  SERVE AS GUN CAPTAIN
110 BM  3  16063  ELEMENTARY FIRE CONTROL
111 BM  3  30091  RIGGING FOR HANDLING CARGO
112 BM  3  30101  RUNNING RIGGING FOR HANDLING CARGO
113 BM  3  30102  Erect station markers for replenishment at sea
114 BM  3  30103  Serve as Signalman for Winchman or Craneman
115 BM  3  30124  Handling and cargo-handling equipment
116 BM  3  30204  Member of an anchoring and mooring detail
117 BM  3  30203  Characteristics of navigational lights and aids
118 BM  3  30202  Paint and metal surfaces for painting
119 BM  3  30201  Prepare and apply paints and primers
120 BM  3  30200  Use stencils for painting letters, numerals
121 BM  3  30209  Miscellaneous knots, hitches and splices
122 BM  3  30208  Tackles and purchases
123 BM  3  30207  Break-out and fake-down hawser and wire rope
124 BM  3  30222  Ground tackle, rigging, hoisting, stowing
125 BM  3  30233  Handling canvas and leather
126 BM  3  30258  OPERATE VARIOUS TYPES OF LANDING CRAFT
127 BM  3  30267  SERVE AS BOAT COXSWAIN ON LANDING CRAFT
128 BM  3  30268  BOAT LOG COMPASS AND CHARTS
129 BM  3  30270  EQUIPMENT ON SHIP'S POWERBOATS, LCVP'S, LCM'S
130 BM  3  34271  BOAT HAILS, RECALLS SALUTES, AND ETIQUETTE
131 BM  3  34262  Conduct minor inspection, maintenance of boats
132 BM  3  34263  Maneuvering small boat in a restricted space
133 BM  3  34410  USE OF NAVIGATIONAL SOUND SIGNALS
134 BM  3  34411  INTERPRET RULES FOR AVOIDING COLLISIONS
135 BM  3  34412  INTERPRET DISTRESS SIGNALS
136 BM  3  34413  GENERAL PRUDENTIAL RULE, GOOD SEAMANSHIP
137 BM  3  34414  RULES TO DETERMINE SAFE WORKING LOADS

D.5-10
OPERATE A SEWING MACHINE FOR SEWING CANVAS
MAINTENANCE OPERATIONS WITH CANVAS
STAND WATCH AS BOATSWAIN'S MATE OF THE WATCH
DIRECT HELICOPTERS USING STD. HAND SIGNALS
INTERPRET SIGNAL FLAGS AND PENNANTS
SEND AND RECEIVE SEMAPHORE AND FLASHING LIGHT
STANDING AND RUNNING BLOCKS AND TACKLES
SHARPEN SCRAPERS AND CHIPPING HAMMERS
MAINTENANCE OF PAINT-SPRAYING EQUIPMENTS
LUBRICATE DECK MACHINERY
DIRECT A DAMAGE-CONTROL REPAIR PARTY SUBGROUP
SERVE AS MOUNT CAPTAIN
SUPERVISE LOADING, DISCHARGING, STOWING OF CARGO
SUPERVISE RIGGING FOR FUELING AT SEA
RIG YARD- AND- STAY AND SWINGING DERRICKS
USE OF GROUND TACKLE FOR MOORING AND ANCHORING
MAINTAIN ANCHORS AND SECURING EQUIPMENT
RECOGNIZE STORM INDICATIONS
USE BAROMETER AND WET- AND DRY- BULB THERMOMETERS
SUPERVISE SHIP'S DECK EQUIPMENT MAINTENANCE
PAINT REMOVAL, SURFACE PREPARATION, PAINTING
PAINTS UNDER DIRECT SUPERVISION
RIGGING OF ACCOMMODATION LADDER AND BOAT BOOM
RIG FOR TOWING AND BEING TOWED
OPERATE BOOM RAMPS ON LANDING CRAFT
OPERATE STERN ANCHOR WINCH
HANDLING OF WIRE DURING BEACHING AND RETRACTING
INSPECTION, INFLATABLE LIFEBOATS AND EQUIPMENT
USE OF INFLATABLE LIFEBOAT AND EQUIPMENT
BOAT HANDLING ON A SHIP IN PORT AND UNDERWAY
REEVE A SET OF BOAT FALLS
SUPERVISE SPlicing AND EYE SPlicing
MAINTAIN PAINTING RECORDS AND REPORTS
PERSONNEL ASSIGNMENTS FOR PAINTING TASKS
INSTRUCT MEN IN DECK WATCH SECTION
INVENTORY AND SPARE PART SUPPORT IN COSAL
ORDER REPAIR PARTS USING COSAL
CHANGE THE ALLOWANCE PARTS LIST
SUPERVISE GENERAL UPKEEP OF DECK EQUIPMENT
DIRECT A DAMAGE-CONTROL REPAIR PARTY
SUPERVISE TRANSFER OF PERSONNEL, CARGO AND MAIL
DIRECT PROCEDURES FOR MOORING TO A BUOY
DIRECT PREPARATIONS FOR CARRYING OUT ANCHORS
PREPARE FOR MAKING MEDITERRANEAN MOOR
DETERMINE NAVIGATIONAL POSITION
ASSIST IN LEAD DANGER BEARING
SMALL BOAT HANDLING, EFFECTS OF GETTING ALONGSIDE
IDENTIFY METHODS OF MINESweepING
EQUIPMENT RIGGING, UNRIGGING
IDENTIFY PROCEDURES AND USE OF PONTOON GEAR
IDENTIFY DUTIES, RESPONSIBILITIES OF BEACHMASTER
ORGANIZE, SUPERVISE PAINTSHOP AND/OR PAINTLOCKER
PLAN, ORGANIZE, ADMINISTER DAILY WORK ASSIGNMENTS
SUPERVISE MAINT OF REQUIRED RECORDS AND LOGS
TRAIN AND SUPERVISE IN BOATSWAIN'S MATES DUTIES
POST CHANGES AND ADDITIONS TO COSAL
ORGANIZE, SUPERVISE A DAMAGE-CONTROL REPAIR PARTY
SUPERVISE HANDLING OF AMMUNITION AND FLAMMABLES
GROUND TACKLE MAINTENANCE
SUPERVISE ORDINARY MOORING WITH ANCHORS
FORMULATE METHODS OF ASSISTING A STRANDED VESSEL
IDENTIFY BOATSWAIN'S MATE DUTIES, LCU, WARPING TUG
USE MANEUVERING BOARD
USE TIDE AND CURRENT TABLES
USE NAVIGATIONAL AIDS TO MANEUVER YARD CRAFT
USE THE THREE-MINUTE RULE
DIFFERENTIAL INTERNA/:ION, INLAND RULES OF ROAD
POST SECAS CHANGES AND SUBMIT REPORT TO VFO
207 BM CS11516 ASSIST DAMAGE CONTROL OFFICER
208 BM CS11517 IDENTIFY STABILITY FACTORS
209 BM CS34076 COORDINATE ALL TYPES OF CARGO
210 BM CS34077 COORDINATE REPLENISHMENT OF SHIPS AT SEA
211 BM CS34080 COORDINATE ANCHORING, MOORING DURING STORMS
212 BM CS35476 TRACK OPERATIONS IN OWN AREA OF RESPONSIBILITY
213 BM CS35478 PREPARE LOCAL DIRECTIVES AND INSTRUCTIONS
214 BM CS35479 PREPARE CORRESPONDENCE
215 BM CS35480 ASSURE MAXIMUM PERSONNEL UTILIZATION
216 BM CS34375 ORGANIZE AND SCHEDULE TRAINING PROGRAMS
217 BM CM34082 COORDINATE HANDLING OF BOATS AND LANDING CRAFT
218 BM CM34083 COORDINATE SALVAGE AND RESCUE OPERATIONS
219 BM CM35483 PLAN, ORGANIZE, IMPLEMENT AND CONTROL ACTIVITIES
220 BM CM35489 FORECAST FUTURE PLAN AND ACTION REQUIREMENTS
221 BM CM35490 ESTABLISH GOALS, OBJECTIVES AND PRIORITIES
222 BM CM35492 REVIEW PERSONNEL, EQUIP, MATERIAL REQUIREMENTS
223 BM CM35045 DEPT. LONG-RANGE PLANNED MAINTENANCE PROGRAM
224 BM CM52298 DEVELOP OPERATING BUDGETS, MONITOR EXPENDITURES
225 AN 20350 AIRCRAFT UNDER EMERGENCY CONDITIONS
226 AN 20351 IDENTIFY MARKINGS INDICATING DANGEROUS AREAS
227 AN 20352 IDENTIFY TOXIC PROPERTIES OF CLEANING MATERIALS
228 AN 20368 RECOGNIZE ARMED EJECTION SEATS
229 AN 23525 IN-USE AIRCRAFT BATTERIES
230 AN 25352 KNOW NAVAL AIR ARM GENERAL ORGANIZATION
231 AN 25353 IDENTIFY NAMES, FUNCTIONS OF NAVAL AIR GROUPS
232 AN 25354 IDENTIFY NAVAL AIRCRAFT SQUADRONS
233 AN 25354 IDENTIFY NAVAL AIRCRAFT SQUADRONS
234 AN 40490 FABRICATE TIE-DOWN LINES
235 AN 42353 STAND AIRCRAFT SECURITY WATCH
236 AN 46410 USE PUBLICATIONS IN AIRCRAFT MAINTENANCE
237 AN 46330 HOT BRAKES FIRE EQUIMENT IDENTIFICATION
238 AN 62350 DEFINE COMMON AVIATION TERMS AND NOMENCLATURE
239 AN 62351 IDENTIFY PURPOSE OF BASIC AIRCRAFT INSTRUMENTS
240 AN 62592 USE AIRFRAME STATION NUMBERING SYSTEM
241 AN 62353 DESIGNATION SYSTEM FOR IDENTIFYING NAVAL AIRCRAFT
242 AN 62354 IDENTIFY BASIC TYPES OF AIRCRAFT POWERPLANTS
243 AN 62355 IDENTIFY AC ELECTRONIC ARMAMENT EQUIPMENTS
244 AN 62357 AC HYDRAULIC, ELECTRICAL PNEUMATIC CABLE SYSTEMS
245 AN 62358 IDENTIFY MAJOR STRUCTURAL COMPONENTS OF AIRCRAFT
246 AN 62359 STAND FIRE WATCH DURING FUELING, STARTING
247 AN 62361 COLOR MARKING SYSTEM FOR AIRFIELD VEHICLES
248 AN 62361 DIRECT AIRCRAFT DURING TAXI
249 AN 62362 SERVE AS A MEMBER OF AN AIRCRAFT HANDLING TEAM
250 AN 62363 SERVE AS A MEMBER OF AN AIRCRAFT HANDLING TEAM
251 AN 62364 STAND COCKPIT BRAKE WATCH, TOWING OR PUSHING
252 AN 62365 OPERATING MOBILE EQUIPMENT AROUND AIRCRAFT
253 AN 62369 CARE OF AIRCRAFT HANDLING EQUIPMENT
254 AN 62368 IDENTIFY, CARE OF HANDLING LAND, CARRIER AIRCRAFT
255 AN 62368 RECOGNIZE AIRCRAFT CARRIER CLOTHING COLOR CODES
256 AN 62369 IDENTIFY, AVIATION SUPPORT EQUIPMENT
257 AN 62371 OIL, FUEL HYDRAULICS, NITROGEN/AIR SERVICING
258 AN 62373 SERVICE AIRCRAFT SERVICING EQUIPMENT
259 AN 62376 COLOR CODES USED, AIRCRAFT LINES, TUBING HOSES
260 AN 70361 CORROSION AND RUST INSPECTION OF AIRCRAFT
261 AN 79550 CLEAN AND WAX AIRCRAFT
262 AN 79550 CLEAN AND WAX AIRCRAFT
263 AN 79551 PLEXIGLASS/FIBERGLASS/RUBBER/FABRIC CLEAN AGENTS
264 AN 96455 USE COMMON MEASURING TOOLS
265 AN 94553 USE COMMON MEASURING TOOLS
266 AN 94556 INSTALL COMMON AIRCRAFT SAFETY AND LOCKING DEVICES
267 AN 94557 USE COMMON AIRCRAFT SCREWS, NUTS AND BOLTS
268 AN 99556 HANDLING OF AVIATION-TYPE INFLATABLE LIFE VEST
269 AN 99551 ATTACH AND ADJUST PARACHUTE HARNESS
270 AN 99552 AX EQUIPMENT LIFE RAFTS
271 AN 99552 AX EQUIPMENT LIFE RAFTS
272 SM 32000 REEVE AND SPlice Halyards
273 SM 42020 RENDER PASSING AND SIDE HONORS
274 SM 50303 USE MAINTENANCE REQUIREMENT CARDS (MRC)
COMPLETE MAINTENANCE DATA FORMS
INVENTORY INSTALLED EQUIPMENT
PERFORM ROUTINE MAINTENANCE ON OPTICAL EQUIPMENT
SAFEKEEPING AND STORAGE OF CLASSIFIED MATERIAL
COMMUNICATIONS SECURITY REGULATIONS, PROCEDURES
RECOGNIZE, USE, INTERPRET EMERGENCY SIGNALS
TRANSMIT, RECEIVE CODE GROUPS BY FLASHING LIGHT
TRANSMIT, RECEIVE PLAIN LANGUAGE BY FLASHING LIGHT
TRANSMIT, RECEIVE PLAIN LANGUAGE BY SEMAPHORE
DISTINGUISH FLAGHOIST SIGNALS
SELECT AND DISPLAY FLAGS AND PENNANTS
EXERCISE RULES OF VISUAL RESPONSIBILITY
DIRECT DAY LIFEBOAT, NIGHT MAN OVERBOARD EXERCISES
IDENTIFY FLAGS AND ENSIGNS OF MARITIME NATIONS
PREPARE A VISUAL MESSAGE FOR TRANSMISSION
HANDLE MESSAGES ACCORDING TO PRECEDENCE
PERFORM CHALLENGE REPLY PROCEDURES
SIGNALMAN DUTIES AS MEMBER OF A BOATCREW
ELECTRICAL AND ELECTRONIC VISUAL SIGNAL EQUIPMENT
OPERATE INFRARED AND FLASHING LIGHT EQUIPMENT
MAINTENANCE OF VISUAL SIGNALING EQUIPMENT
DEVELOP AND FULL-DRESS A SHIP
RECOGNIZE AND USE RULES GOVERNING FLAG DISPLAY
MAINTAIN VISUAL SIGNAL LOG, VISUAL TRAFFIC FILES
IDENTIFY SIGNAL AND PERSONAL FLAGS AND PENNANTS
LOCATE A SHIP USING FORMATION DIAGRAM
CORRECT AND MAINTAIN SIGNAL PUBLICATIONS
COMPLETE PMS FEEDBACK REPORTS
ORDER COSAL REPAIR PARTS AND SPECIAL TOOLS
CONSTRUCT AND USE ALL SIGNALING CALL SIGNS
ENCODE, DECODE NAVAL, INTERNATIONAL SIGNALS
CLASSIFIED MATERIAL DISPOSAL AND DESTRUCTION
ENSURE CORRECT CLASSIFIED MATERIAL TRANSPORTATION
TRANSMIT, RECEIVE FLASHING LIGHT CODE GROUPS
TRANSMIT, RECEIVE FLASHING LIGHT PLAIN LANGUAGE
TRANSMIT, RECEIVE SEMAPHORE PLAIN LANGUAGE
SURVEY AND REQUISITION SIGNAL EQUIPMENT
PREPARE SIGNALS, VISUAL COMM, STANDING ORDERS
PREPARE CRITIQUES OF VISUAL SIGNALING DRILLS
INSTRUCT IN IDENTIFICATION OF SHIPS AND AIRCRAFT
INSTRUCT IN VISUAL COMMUNICATIONS PROCEDURES
INSTRUCT IN VISUAL COMMUNICATIONS AUTHENTICATION
REVIEW COMPLETED MDCS FORMS
PREPARE WEEKLY SCHEDULES OF PREVENTIVE MAINTENANCE
POST CHANGES AND ADDITIONS TO COSAL
MAINTAIN CUSTODY CARDS AND INVENTORY FILES
VERIFY SPARE PART SUPPORT IN COSAL
ADMINISTER SECURITY CLASSIFICATION MANAGEMENT
INSTITUTE RELIEVER OF VISUAL MESSAGES
DIRECT WARTIME TRANSITION, VISUAL SIGNAL METHODS
CLASSIFY VARIOUS USES OF MESSAGES
USE RESERVE ON BOARD SIGNALING, PUBLICATIONS
TRANSMIT, RECEIVE FLASHING LIGHT CODE GROUPS
TRANSMIT, RECEIVE SEMAPHORE PLAIN LANGUAGE
PREPARE QUARTERLY SCHEDULES OF PREVENTIVE MAINT
POST SECAS CHANGES AND SUBMIT REPORT TO VFO
ADMINISTER COMMAND SECURITY EDUCATION PROGRAM
PERFORM AS TACTICAL COMMUNICATIONS WATCH OFFICER
SUPERVISE VISUAL COMMUNICATIONS ACTIVITIES
SUPERVISE SIGNAL BRIDGE PERSONNEL
INTERPRET COMM SECTION ORGANIZATION FUNCTIONS
MANAGE IN OWN AREA OF RESPONSIBILITY
PREPARE LOCAL DIRECTIVES AND INSTRUCTIONS
PREPARE CORRESPONDENCE
IMPLEMENT MAXIMUM PERSONNEL UTILIZATION
SUPERVISE SIGNAL EQUIP TECHNICAL INFORMATION
OBSERVE, EVALUATE VISUAL COMMUNICATIONS EXERCISES
ORGANIZE, IMPLEMENT COMMUNICATIONS SECTION
ADMINISTER VISUAL COMM EMERGENCY DRILLS
ADMINISTER LONG-RANGE PLANNED MAINT PROGRAMS
ADMINISTER COMMAND SECURITY PROGRAM
TEAS ELECTRO-HYDRAULIC AND SPECIAL SYSTEMS
PLANT, ORGANIZE, IMPLEMENT, CONTROL ACTIVITIES
FORECAST FUTURE REQUIREMENTS
ESTABLISH GOALS, OBJECTIVES AND PRIORITIES
REVIEW PERSONNEL, EQUIP., MATERIAL REQUIREMENTS
DEVELOP OPERATING BUDGETS, MONITOR EXPENDITURES
LOCATE PRINCIPAL VALVES, FIRE-MAIN SYSTEM
OPERATE AND SECURE FIRE AND FLASHING PUMP
OPERATE INTERNAL-COMBUSTION ENGINE-DRIVEN PUMPS
USE RADIO INSTRUMENTS FOR MONITORING
USE FIREFIGHTING EQUIPMENT
BLEED HYDRAULICS SYSTEMS
INSTALL SEALS, PACKINGS AND WIPERS IN HYDRAULICS
USE OF COMMON TEST EQUIPMENTS
PERFORM ELECTRICAL CHARGING, REGULATING SYSTEMS
TECHNICAL DRAWING USAGE ABOARD SHIP
CONDUCT BOILER FEED WATER TESTS
CORRECT COMMON BOILER MALFUNCTIONS
STEAM OPERATED DISTILLING PLANTS
REMOVE SCALE FROM EVAPORATOR TUBES MECHANICALLY
STAND WATCH ON REFRIGERATION, AIR CONDITIONING
LOW-, MEDIUM-, HIGH-PRESSURE AIR COMPRESSORS
ADJUST COMPRESSOR FORCED-FEED LUBRICATORS
OVERHAUL MANUALLY OPERATED VALVES
OPERATE AND MAINTAIN HYDRAULIC EQUIPMENT
PERFORM SHIFING OF STEERING UNIT PUMPS
STEERING TRANSFER FROM PILOT HOUSE TO AFT STEERING
TRANSFER TO MANUAL STEERING
INSPECT, LUBRICATE AND TEST GALLEY EQUIPMENT
INSPECT, LUBRICATE AND TEST LAUNDRY EQUIPMENT
INSPECT, LUBRICATE AND TEST DECK MACHINERY
MAINTAIN MECHANICAL LUBRICATORS
REPACK ADJUST STUFFING BOXES ON PUMPS AND SHAFTS
COMPLETE OPERATIONAL MAINTENANCE OF DIESEL ENGINES
OPERATE INTERNAL COMBUSTION ENGINES
LUBRICATION ON INTERNAL COMBUSTION ENGINES
CARRYOUT TURNING OVER OF MAIN ENGINES
ALIGN LUBRICATION OIL SYSTEM
USE STANDBY LUBRICATION OIL PUMPS
PURGE DIESEL ENGINE FUEL INJECTION SYSTEM
ENTRIES TO MAIN PROPULSION/DIESEL GENERATOR LOG
MAINTAIN ENGINEER'S BELL BOOK
OPERATION OF LUBRICATION SYSTEM, OPERATING ENGINE
ALIGN STARTING SYSTEM (AIR/HYDRAULIC/ELECTRICAL)
ALIGN COOLING SYSTEM
ALIGN FUEL SYSTEM
CHEMICAL TESTS, LUBE OIL AND DISTILLATE FUELS
TESTS ON DIESEL ENGINE CLOSED-COOLING WATER SYSTEM
OIL AND WATER TESTS, LIGHTING OFF AND SECURING
ALIGN, OPERATE LUBE OIL AND FUEL OIL PURIFIERS
CHANGE OIL IN DIESEL ENGINES
MAINTENANCE OF FLANGE SHIELDS ON PIPING SYSTEMS
LUBRICATION SYSTEM MAINTENANCE
SELECT AND USE GRINDING COMPOUNDS
INTERPRET READINGS OF STANDARD ENGINE INSTRUMENTS
USE OF STANDARD MACHINERY SUPPLIES
REDUCTION GEARS, THRUST BEARINGS
LUBE OIL SAMPLES FROM DIESEL ENGINE FOR ANALYSIS
MAKE UP RIGID TUBING AND LOW PRESSURE HOSES
MAIN AND MAIN TECHNICAL AND MAINT MANUALS
COMPLETE MAINTENANCE DATA FORMS
USE MAINTENANCE REQUIREMENT CARDS (MRC)
IDENTIFY CATEGORIES OF MATERIAL IN COSAL
USE AND CARE FOR MEASURING INSTRUMENTS
ADJUST FUEL COMBUSTION TO MINIMIZE AIR POLLUTION
PREVENTION OF OIL SPILL ACCIDENTS
SAFETY IN HIGH NOISE AND HEAT STRESS AREAS
INSPECT AND REPAIR PORTABLE PUMPS
IDENTIFY ORGANIZATION OF A REPAIR PARTY
OPERATE AUXILIARY BOILER
REMOVE SCALE FROM DISTILLING UNITS CHEMICALLY
TEST EVAPORATORS AND CONDENSERS FOR LEAKS
CONDUCT STANDARD DISTILLING PLANT TESTS
HALIDE TORCH TESTS ON REFRIGERATION UNITS
OIL CHANGES IN REFRIGERATION COMPRESSORS
SUCTION, DISCHARGE VALVE TESTS, REFRIG COMPRESSORS
TEST AND RECHARGE REFRIGERATION UNITS
TEST, RENEW OIL SEALS IN REFRIGERATION UNITS
SUCTION, DISCHARGE VALVES MAINT ON AIR COMPRESSORS
DETERMINE CLEARANCES IN PUMPS
OVERHAUL PUMPS
CHECK FOR ALIGNMENT OF PUMP DRIVING UNITS
REPLACE ROTARY SEALS
TEST HYDROSTATICALLY, REPAIR PIPING SYSTEMS
LUBRICATION STEERING GEAR
REFRIGERATION AUTOMATIC CONTROLS
CORRECT INEFFECTIVE REFRIGERATION SYSTEMS OPS
CLEAN, INSPECT, TEST HEAT EXCHANGERS
MAINT OF EXHAUST SILENCERS AND SPARK ARRESTERS
OPERATE A HEAT RECOVERY DISTILLING PLANT
INTERNAL COMBUSTION ENGINE MAINTENANCE
REFACE, RESET INTAKE AND EXHAUST VALVES
TEST UNIT INJECTORS AND/OR FUEL INJECTION NOZZLES
MAINTENANCE OF FUEL OIL INJECTORS
PREVENTION OF CRANKCASE EXPLOSIONS
TROUBLESHOOT GASOLINE ENGINES
CHEMICAL ANALYSIS OF FUELS AND OILS
ENGINE AND DRIVE SHAFT ADJUSTMENTS, SMALL CRAFT
CHECK OIL CLEARANCE IN BEARINGS
MAINT INTERNAL COMBUSTION ENGINE TURBO-CHARGERS
TROUBLESHOOT INTERNAL COMBUSTION ENGINE BLOWERS
OPERATE AND MAINTAIN VERTICAL CONVEYORS
MAINT OF LUBE OIL AND FUEL OIL PURIFIERS
EXTERNAL ADJUSTMENTS ON DIESEL ENGINES
MAINT TREND ANALYSIS DATA ON DIESEL ENGINE OPS
RECORD RESULTS OF EQUIPMENT TESTS
RECORD TRAINING DATA
TROUBLESHOOT AUXILIARY SYSTEMS
DYE-PENETRANT AND MAGNAFLUX TESTS, METAL SURFACES
COMPLETE PMS FEEDBACK REPORTS
REPAIR EQUIPMENT AND SPARE PART SUPPORT IN COSAL
ORDER REPAIR PARTS AND SPECIAL TOOLS USING COSAL
MECHANICAL MAINTENANCE TEST EQUIPMENT
PLAIN TURNING AND CUTTING ON AN ENGINE LATHE
SUPERVISE A DAMAGE CONTROL PARTY
MAINTENANCE AND TESTS ON AUXILIARY BOILERS
PLUG AND/OR REPLACE HEAT EXCHANGER TUBES
AFTERCOOLER, OIL COOLER MAINTENANCE
OVERHAUL OF AIR COMPRESSORS
REDUCING, RELIEF, TEMPERATURE CONTROL VALVE MAINT
HYDRAULIC SYSTEM MAINTENANCE
STEAM OPERATED DISTILLING PLANT MAINTENANCE
VAPOR COMPRESSION DISTILLING PLANT MAINTENANCE
INTERNAL COMBUSTION ENGINE MAINTENANCE
ADJUSTMENT OF TEMPERATURE REGULATING VALVES
GRAPHICALLY DETERMINE PROPER ENGINE OPERATION
REPAIR OF INTERNAL COMBUSTION ENGINES
MAIN ENGINE BEARING AND THRUST CLEARANCES
CLEARANCES LOBES; TIMING, DRIVE GEAR CHECK
inspect, TAKE CLEARANCES ON TURBO-CHARGERS
CHECK THRUST BEARING READINGS
TAEG REPORT NO. 40

PAGE NUMBER : 8

SOURCE CARD LISTING FOR
OFFLINE READ ETAM TDATA

483 EN 1 31097 CHECK DAMAGE TO SHAFTS AND THRUST BEARINGS
484 EN 1 38253 MONITOR GOVERNORS AND OVERSPEED TRIPS
485 EN 1 31454 COMPLETE MOT OF GOVERNORS AND OVERSPEED TRIPS
486 EN 1 35323 MAINTAIN ENGINE ROOM RECORDS
487 EN 1 51167 REVIEW COMPLETED MDCS FORMS
488 EN 1 51168 PREPARE WEEKLY SCHEDULES OF PREVENTIVE MAINTENANCE
489 EN 1 54827 POST CHANGES AND ADDITIONS TO COSAL
490 EN 1 30359 MECHANICAL OPERATION TESTS
491 EN 1 30374 SUPERVISE AND TRAIN IN MECHANICAL MAINTENANCE
492 EN 1 31045 DIESEL AND GASOLINE ENGINE MAINTENANCE
493 EN 1 31046 CONDUCT DIESEL ENGINE POST-OVERHAUL CHECKOUT
494 EN 1 31047 MEASURE CRANKSHAFT DEFLECTION
495 EN 1 31077 SHIPS EXTERNAL DRIVE CHECKS WHEN IN DRYDOCK
496 EN 1 31095 MAIN REDUCTION GEAR BACKLASH, ALIGNMENT CHECKS
497 EN 1 35430 REVIEW POST REPAIR TRIAL REPORTS
498 EN 1 35431 REVIEW ALL ENGINE ROOM RECORDS
499 EN 1 35885 SUPERVISE ALL ADMINISTRATION FOR ENGINEERING DEPT
500 EN 1 35886 INSPECT RESULTS OF EQUIPMENT TESTS
501 EN 1 42270 INTERPRET DUTIES, ENGINEERING OFFICER OF THE WATCH
502 EN 1 42279 ENGINE ROOM WATCH ON DIESEL-DRIVEN SHIP
503 EN 1 42567 INTERPRET DUTIES, ENGINEERING OFFICER OF THE WATCH
504 EN 1 51069 ESTIMATE REPAIRS FOR AUX, MAIN PROPELLION MACH
505 EN 1 51169 PREPARE QUARTERLY PM SCHEDULES
506 EN 1 "SHIP TO SHOP" WORK AND REPORTS
507 EN 1 51476 ANALYZE CHEMICAL OIL ANALYSIS REPORTS
508 EN 1 51477 INTERPRET DIESEL ENGINE TREND ANALYSIS REPORTS
509 EN 1 51478 INTERPRET OIL CHEMICAL TESTS
510 EN 1 35887 SUBMIT CHANGES TO SECAS, SUBMIT REPORTS TO VFO
511 EN 1 35888 SUPERVISE OIL SPILL CONTAINMENT EXERCISES
512 EN 1 35899 TRAINING OF PERSONNEL IN MECHANICAL MAINTENANCE
513 EN 1 35887 ANALYSIS OF DAILY OPERATING RECORDS
514 EN 1 35875 ESTIMATE FEASIBILITY OF REPAIRS
515 EN 1 35877 MONITOR MAINTAIN OF EQUIPMENT, MATERIAL LOGS
516 EN 1 35872 ADVISE PERSONNEL IN REPAIR PAPERWORK
517 EN 1 35874 MANAGEMENT IN OWN AREA OF RESPONSIBILITY
518 EN 1 35878 PREPARE LOCAL DIRECTIVES AND INSTRUCTIONS
519 EN 1 35879 PREPARE CORRESPONDENCE
520 EN 1 35880 ESTABLISH MAXIMUM PERSONNEL UTILIZATION
521 EN 1 35881 REVISE WATCHSTANDING QUALIFICATIONS
522 EN 1 35882 REVIEW PUBLICATIONS REQUIREMENTS
523 EN 1 35883 ORGANIZE AND SCHEDULE TRAINING PROGRAMS
524 EN 1 35884 ORGANIZE AND SCHEDULE WORK
525 EN 1 35885 ADMINISTER LONG-RANGE PLANNED MAINTENANCE PROGRAM
526 EN 1 35886 MONITOR ENVIRONMENTAL POLLUTION CONTROL PROGRAMS
527 EN 1 35887 ORGANIZE AND SCHEDULE TRAINING PROGRAMS
528 EN 1 35888 ORGANIZE AND SCHEDULE WORK
529 EN 1 35889 ADMINISTER LONG-RANGE PLANNED MAINTENANCE PROGRAM
530 EN 1 35890 MONITOR DAILY OPERATING RECORDS
531 EN 1 35891 PREPARE MONTHLY SUMMARY, DIESEL-DRIVEN SHIPS
532 EN 1 35892 ORGANIZE ENGINE PUBLICATION DISTRIBUTION
533 EN 1 35893 PLAN, ORGANIZE, IMPLEMENT, CONTROL ACTIVITIES
534 EN 1 35894 FORECAST FUTURE REQUIREMENTS
535 EN 1 35895 ESTABLISH OBJECTIVES AND PRIORITIES
536 EN 1 35896 REVIEW PERSONNEL, EQUIPMENT, MATERIAL REQUIREMENTS
537 EN 1 35897 MONITOR ALL "SHIP TO SHOP" WORK AND REPORTS
538 EN 1 35898 ORGANIZE AND SCHEDULE WORK
539 EN 1 35899 COORDINATE REPAIRS BETWEEN SHIP AND SHIPYARD
540 EN 1 35900 DEVELOP OPERATING BUDGETS, MONITOR EXPENDITURES
541 EN 1 35901 MONITOR ALL "SHIP TO SHOP" WORK AND REPORTS
542 EN 1 35902 DEVELOP AND MONITOR SAFETY PROGRAMS
543 EN 1 35903 MONITOR ALL "SHIP TO SHOP" WORK AND REPORTS
544 GMG 1 14564 INTERPRET COLOR-CODING OF ELECTRONIC PARTS
545 GMG 1 14565 TEST AND REPLACE ELECTRONIC PARTS
546 GMG 1 15305 IDENTIFY ELECTRONIC CIRCUIT PROTECTIVE DEVICES
547 GMG 1 15305 INSPECT AND SERVICE HYDRAULIC EQUIPMENT
548 GMG 1 15305 TEST HYDRAULIC ORDNANCE EQUIPMENT
549 GMG 1 15305 EXAMINE TYPES AND PRINCIPLES, PRESSURE GAUGES
550 GMG 1 15305 MAIN PIPE PIPES, FITTINGS, SEALS AND GASKETS
551 GMG 1 16023 IDENTIFY TYPES OF EXPLOSIVE ORDNANCE
552 GMG 1 16033 HANDLE, STOW ORDNANCE AND PYROTECHNICS

D.5-16
552 GMG 3 16034 STOW, ISSUE, INVENTORY SMALL ARMS AND AMMUNITION
553 GMG 3 16034 OPERATE AMMUNITION HOIST UNITS
554 GMG 3 16334 SERVICE ORDNANCE HANDLING AND ASSOCIATED EQUIPMENT
555 GMG 3 16398 INSTALL, SET, REMOVE PROJECTILE FUZING DEVICES
556 GMG 3 16399 ASSEMBLE AND USE LANDING PARTY EQUIPMENT
557 GMG 3 16400 OPERATE AND MAINTAIN LINE-THROWING GUNS
558 GMG 3 16402 PERFORM FUZING OF HAND GRENADES
559 GMG 3 16403 MONITOR, REPLACE RELATIVE HUMIDITY INDICATORS
560 GMG 3 16404 LD: MAGAZINE TEMPERATURES, INSPECT MAGAZINE AREA
561 GMG 3 16405 OPERATE MAGAZINE FIREFIGHTING SYSTEMS
562 GMG 3 16407 INSPECT ORDNANCE ITEMS
563 GMG 3 16408 OPERATE PYROTECHNIC EQUIPMENT
564 GMG 3 17079 MAINTAIN SMALL ARMS
565 GMG 3 17081 MAINTAIN LANDING PARTY EQUIPMENT
566 GMG 3 17365 PERFORM DAILY TRANSMISSION TESTS
567 GMG 3 18017 OPERATE MECHANICAL AND ELECTRICAL TEST EQUIPMENT
568 GMG 3 18022 OPERATE PRESSURE GAUGES, INTERPRET READINGS
569 GMG 3 24023 MAKE ELECTRICAL CABLE AND WIRING CHECKS
570 GMG 3 24026 COMMUTATOR, SLIPPING, BRUSH ASSEMBLY MAINTENANCE
571 GMG 3 24390 ELECTRICAL/ELECTRONIC TERMS/UNITS OF MEASURE
572 GMG 3 24497 IDENTIFY TYPES OF D.C. MOTORS AND GENERATORS
573 GMG 3 24500 CLEAN ELECTRICAL CONTACTS AND SWITCHES
574 GMG 3 24503 MECHANICAL WIRE CONNECTING, INCLUDING SOLDERING
575 GMG 3 24504 MECHANICAL WIRE CONNECTING, CLEANING AND REPLACING BLOWN FUSES
576 GMG 3 24505 POINT-TO-POINT VOLTAGE, RESISTANCE MEASUREMENTS
577 GMG 3 28016 IDENTIFY MECHANICAL, HYDRAULIC, ELECTRONIC SYMBOLS
578 GMG 3 28019 USE AND READ DIAGRAMS AND DRAWINGS
579 GMG 3 28020 TRACE CIRCUITS ON SCHEMATICS AND DRAWINGS
580 GMG 3 28023 MAINTAIN ONBOARD SYSTEMS BREECH MECHANISMS
581 GMG 3 28027 REPAIR SEALING SURFACES, MATING AREAS, THREADS
582 GMG 3 28045 MAINTAIN FASTENING HARDWARE AND SEALS
583 GMG 3 28054 INSTALL ALIGNMENT EQUIPMENT ON GUN MOUNTS
584 GMG 3 40525 CUT, BEND, PREPARE PIPING AND TUBING
585 GMG 3 40630 USE PUMPS RELATED TO OPS/TEST/REPAIR, ORDNANCE EQPT
586 GMG 3 40631 USE AND DISPOSAL OF MISFIRING ROUNDS
587 GMG 3 40632 COMPLETE MAINTENANCE DATA FORMS
588 GMG 3 50126 USE MAINTENANCE REQUIREMENT CARDS
589 GMG 3 50128 USE MAINTENANCE REQUIREMENT CARDS
590 GMG 3 54192 PROCESS AND DOCUMENT REPAIRABLE ITEMS
591 GMG 3 54759 MAINTAIN CONSUMABLE SUPPLIES, OBTAIN REPLACEMENTS
592 GMG 3 54760 IDENTIFY MATERIALS CONTAINED IN COSAL
593 GMG 3 54761 INVENTORY TOOLS AND PORTABLE TESTING EQUIPMENT
594 GMG 3 59272 CORROSION PREVENTION, MOISTURE PROTECTIVE MATERIAL
595 GMG 3 92336 USE, MAINTAIN FIXED/PORTABLE PNEU/ELEC POWER TOOLS
596 GMG 3 94424 USE, MAINTAIN HARDTOOL
597 GMG 3 94501 OPERATE TORQUING TOOLS
598 GMG 3 94508 OPERATE TORQUING TOOLS
599 GMG 3 14043 SERVOAMPLIFIERS MAINTENANCE
600 GMG 3 14566 ANALYZE ELECTRONIC SYSTEMS MALFUNCTIONS
601 GMG 3 14572 CONTROL INTERFACE DSOT
602 GMG 3 14798 CHECK ALARMS, SENSSING DEVICES FOR OPERATION
603 GMG 3 15023 PERFORM HYDRAULIC TESTS, ORDNANCE EQUIPMENT
604 GMG 3 15309 INSTALL, MAINTAIN COMPONENTS IN HYDRAULIC SYSTEMS
605 GMG 3 15311 ANALYZE HOPE MALFUNCTIONS AND MAKE REPAIRS
606 GMG 3 15312 RECOIL, COUNTER RECOIL, GAS EJECTOR SYSTEMS
607 GMG 3 16004 USE DEMOLITION CHARGES FOR EMERGENCY DESTRUCTION
608 GMG 3 16175 ASSEMBLE EXPLOSIVE ORDNANCE ITEMS FOR DANGER
609 GMG 3 16504 MOVEMENT COMPONENTS IN BLOCKING AMMUNITION, EXPLOSIVES
610 GMG 3 16509 HANDLE DAMAGED EXPLOSIVE COMPONENTS
611 GMG 3 16599 CHECK ALARMS AND SENSING DEVICES
612 GMG 3 17366 WEAPONS ALIGNMENT AND ALIGNMENT
613 GMG 3 17367 REPLACE DEFECTIVE COMPONENTS OR MODULES
614 GMG 3 18419 INSTALL, OPERATE GUN MOUNT DYNAMIC TEST EQUIPMENT
615 GMG 3 18426 TEST, REPLACE ELECTRICAL CIRCUIT COMPONENTS
616 GMG 3 18427 INSTALL, ALIGN, ZERO SYNCHROS
617 GMG 3 24393 ANALYZE ORDNANCE ELECTRICAL SYSTEMS MALFUNCTIONS
618 GMG 3 28017 PREPARE BASIC DRAWINGS AND MATHEMATICAL LAYOUTS
619 GMG 3 28020 USE MICROFICHE, LD DRAWINGS, READ/PRINT MACHINES
620 GMG 3 29237 CHECK ACCURACY OF TRAIN AND ELEVATION INDICATORS
TAEG REPORT NO. 40

PAGE NUMBER : 10

SOURCE CARD LISTING FOR
OFFLINE READ ETAM TO DATA

621 GMG 2 29243 TEST AND MAINTAIN ORDNANCE WATER SYSTEM
622 GMG 2 29245 PREPARE ONBOARD GUNS AND EQUIPMENT FOR FIRING
623 GMG 2 29246 PERFORM POST FIRING PROCEDURES ON GUNS, EQUIPMENT
624 GMG 2 29248 TEST FIRING CUTOUT ASSEMBLIES, INTERPRET MEANINGS
625 GMG 5 29249 INSPECT FUZE-SETTING DEVICES, INTERPRET MEANINGS
626 GMG 5 29250 INSPECT TEST ASSEMBLIES (LESS POWER DRIVES)
627 GMG 2 29251 MAINTAIN, REPAIR ASSEMBLIES (LESS POWER DRIVES)
628 GMG 2 29252 MAINTAIN, CHECK ORDNANCE OPERATION LOGS
629 GMG 2 36450 MAINTAIN SHOP AND EQUIPMENT WORKLOGS
630 GMG 2 36451 MAINTAIN EQUIPMENT HISTORIES
631 GMG 2 51114 COMPLETE PMS FEEDBACK REPORTS
632 GMG 5 54708 SUBMIT DAILY AMMUNITION TRANSACTION REPORTS
633 GMG 2 54813 CORRECT THE COSAL REPORTS
634 GMG 2 54814 INVENTORY EQUIPMENT, SPARE PART SUPPORT IN COSAL
635 GMG 2 54815 ORDER PARTS AND TOOLS USING COSAL
636 GMG 2 14062 DIAGNOSE ELECTRONIC MALFUNCTIONS
637 GMG 2 15313 TEST, ADJUST, REPAIR HYDRAULIC MECHANISMS
638 GMG 1 16023 SUPERVISE CREWS IN ORDNANCE HANDLING
639 GMG 1 18018 ANALYZE, INTERPRET DATA, GUN MOUNT DYNAMIC TEST
640 GMG 1 18021 OPERATE AN OSCILLOSCOPE
641 GMG 1 28018 PREPARE ADVANCED DRAWINGS AND MATHEMATICAL LAYOUTS
642 GMG 1 29253 INSPECT AND TEST POWER DRIVES
643 GMG 2 29255 MAINTAIN AND REPAIR POWER DRIVES
644 GMG 2 29256 REPAIR AND ADJUST FIRING CUTOUT ASSEMBLIES
645 GMG 2 29256 REPAIR AND ADJUST FUZE-SETTING DEVICES
646 GMG 2 29257 TEST, ADJUST, REPAIR ALARMS AND SENSING DEVICES
647 GMG 2 29552 OVERHAUL AND REPAIR GUN MOUNTS
648 GMG 1 28344 COMPLETE WEAPONS SYSTEM DEFICIENCY REPORTS
649 GMG 1 28345 PREPARE PERIODIC PERFORMANCE REPORTS
650 GMG 1 28346 MAINTAIN AMMUNITION RECORDS, INVENTORIES
651 GMG 1 29834 PREPARE WEEKLY SCHEDULES OF PREVENTIVE MAINTENANCE
652 GMG 1 29888 ORGANIZE, SUPERVISE MAINTENANCE OF A TECHNICAL LIBRARY
653 GMG 1 51167 REVIEW COMPLETED MDCS FORMS
654 GMG 1 14868 EFFECT CHANGES TO SECAS SYSTEM
655 GMG 1 15484 INITIATE SURVEYS AND DAMAGED SHIPMENT REPORTS
656 GMG 1 15484 SUPERVISE INVENTORIES AND MAINTAIN CUSTOM RECORDS
657 GMG 1 198255 PREPARE ENVIRONMENTAL POLLUTION CONTROL REPORTS
658 GMG 1 198255 MAINTAIN ENVIRONMENTAL POLLUTION CONTROL REPORTS
659 GMG C 14042 REPAIRS TO INDICATOR AND RECEIVER REGULATORS
660 GMG C 14044 SUPERVISE MAINT OF ORDNANCE ELECTRIC/ELECTRONIC SYS
661 GMG C 14045 INSPECT REPAIRS ELECTRIC/ELECTRONIC EQUIP
662 GMG C 14047 ANALYZE ELECTRIC/ELECTRONIC DISCREPANCY TRENDS
663 GMG C 15032 SUPERVISE ASHORE STOWAGE OF EXPLOSIVE ORDNANCE
664 GMG C 15611 SUPERVISE AND INSPECT ORDNANCE STOWAGE
665 GMG C 17426 INSPECT COMPLETED MODERNIZATIONS
666 GMG C 17427 SUPERVISE SAFETY PROCEDURES FOR INSPECTIONS
667 GMG C 20276 SUPERVISE WEAPONS DEPARTMENT REPORTS
668 GMG C 20276 IMPLEMENT LOADING AND STOWAGE PLANS
669 GMG C 28895 SUPERVISE WEAPONS DEPARTMENT REPORTS
670 GMG C 28897 SUPERVISE SAFETY PROCEDURES FOR INSPECTIONS
671 GMG C 51079 SUPERVISE WORK AND ESTABLISH WORK PRIORITIES
672 GMG C 54766 PERFORM ARMAMENT INVENTORIES
673 GMG C 54766 SUPERVISE SHIPBOARD ENVIRONMENTAL POLLUTION CONT
674 GMG C 98336 SUPERVISE REPAIRMENT OF GUN, MISSILE BATTERIES
675 GMG C 98336 REPAIRMENT OF GUN, MISSILE BATTERIES
676 GMG CS35441 SUPERVISE, SUBMIT ALL ADMINISTRATIVE REPORTS
677 GMG CS35446 REPORT OPERATIONS IN OWN AREA OF RESPONSIBILITY
678 GMG CS35446 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
679 GMG CS35446 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
680 GMG CS35446 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
681 GMG CS35446 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
682 GMG CS35446 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
683 GMG CS35446 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
684 GMG CS35446 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
685 GMG CS35446 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
686 GMG CS35446 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
687 GMG CM20391 SUPERVISE LOCAL DIRECTIVES AND INSTRUCTIONS
688 GMG CM35446 PLAN, ORGANIZE, IMPLEMENT, CONTROL ACTIVITIES
689 GMG CM35446 FORECAST FUTURE REQUIREMENTS
TAEG REPORT NO. 40

SOURCE CARD LISTING FOR OFFLINE READ ETAM DATA

CM35490 690 GM ESTABLISH GOALS, OBJECTIVES AND PRIORITIES
CM35492 691 GM REVIEW PERSONNEL, EQUIPMENT, MATERIAL REQMNTS
CM35551 692 GM SUPERVISE OPERATIONS AND PROCEDURES
CM35552 693 GM PREPARE MAINT PERIODIC AND RECURRING REPORTS
CM35553 694 GM DEVELOP ACCOUNTING PROCEDURES FOR ALL WORK
CM51156 695 GM COORDINATE SURFACE WEAPONS SYSTEMS OVERHAUL
CM51157 696 GM ANALYZE DISCREPANCY TRENDS IN EQUIPMENT RECORDS
CM52298 697 GM DEVELOP OPERATING BUDGETS, MONITOR EXPENDITURES
<table>
<thead>
<tr>
<th>Page</th>
<th>Source Card Listing for Offline Read ETAM TDSCR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>SM 3 34009 41 59</td>
</tr>
<tr>
<td>2</td>
<td>SM 3 42020 41 42 62</td>
</tr>
<tr>
<td>3</td>
<td>SM 3 42021 41 58</td>
</tr>
<tr>
<td>4</td>
<td>SM 3 50303 01</td>
</tr>
<tr>
<td>5</td>
<td>SM 3 50928 01</td>
</tr>
<tr>
<td>6</td>
<td>SM 3 54722 01</td>
</tr>
<tr>
<td>7</td>
<td>SM 3 72001 41 52 59 61</td>
</tr>
<tr>
<td>8</td>
<td>SM 3 77254 01</td>
</tr>
<tr>
<td>9</td>
<td>SM 3 84225 53 55 60 61</td>
</tr>
<tr>
<td>10</td>
<td>SM 3 86015 53 55 60 61</td>
</tr>
<tr>
<td>11</td>
<td>SM 3 86016 55 60 61</td>
</tr>
<tr>
<td>12</td>
<td>SM 3 86017 55 60 61</td>
</tr>
<tr>
<td>13</td>
<td>SM 3 86018 55 60 61</td>
</tr>
<tr>
<td>14</td>
<td>SM 3 86021 55 61</td>
</tr>
<tr>
<td>15</td>
<td>SM 3 86022 41 60</td>
</tr>
<tr>
<td>16</td>
<td>SM 3 86023 41 56</td>
</tr>
<tr>
<td>17</td>
<td>SM 3 86024 42 60 62 91</td>
</tr>
<tr>
<td>18</td>
<td>SM 3 86025 53</td>
</tr>
<tr>
<td>19</td>
<td>SM 3 86031 41 60 82</td>
</tr>
<tr>
<td>20</td>
<td>SM 3 86035 41 60</td>
</tr>
<tr>
<td>21</td>
<td>SM 3 86036 41 60</td>
</tr>
<tr>
<td>22</td>
<td>SM 3 86038 43 60</td>
</tr>
<tr>
<td>23</td>
<td>SM 3 86355 52 54 61</td>
</tr>
<tr>
<td>24</td>
<td>SM 3 86366 41 42 57 61</td>
</tr>
<tr>
<td>25</td>
<td>SM 3 94530 41 59 61</td>
</tr>
<tr>
<td>26</td>
<td>YN 3 38605 02</td>
</tr>
<tr>
<td>27</td>
<td>YN 3 38606 01</td>
</tr>
<tr>
<td>28</td>
<td>YN 3 38822 05</td>
</tr>
<tr>
<td>29</td>
<td>YN 3 38823 01 02 04 26</td>
</tr>
<tr>
<td>30</td>
<td>YN 3 38824 35</td>
</tr>
<tr>
<td>31</td>
<td>YN 3 38825 23</td>
</tr>
<tr>
<td>32</td>
<td>YN 3 38826 21</td>
</tr>
<tr>
<td>33</td>
<td>YN 3 38827 03</td>
</tr>
<tr>
<td>34</td>
<td>YN 3 38830 01 12</td>
</tr>
<tr>
<td>35</td>
<td>YN 3 38831 21</td>
</tr>
<tr>
<td>36</td>
<td>YN 3 38834 11</td>
</tr>
<tr>
<td>37</td>
<td>YN 3 38835 01</td>
</tr>
<tr>
<td>38</td>
<td>YN 3 38843 02</td>
</tr>
<tr>
<td>39</td>
<td>YN 3 38852 11 21</td>
</tr>
<tr>
<td>40</td>
<td>YN 3 38958 01</td>
</tr>
<tr>
<td>41</td>
<td>YN 3 46355 02 12</td>
</tr>
<tr>
<td>42</td>
<td>YN 3 77269 41 42 61</td>
</tr>
<tr>
<td>43</td>
<td>YN 3 77284 03</td>
</tr>
<tr>
<td>44</td>
<td>YN 3 94620 21 23 41 59</td>
</tr>
<tr>
<td>45</td>
<td>YN 2 25269 53 55</td>
</tr>
<tr>
<td>46</td>
<td>YN 2 38619 02</td>
</tr>
<tr>
<td>47</td>
<td>YN 2 38620 02 04</td>
</tr>
<tr>
<td>48</td>
<td>YN 2 38627 02 04</td>
</tr>
<tr>
<td>49</td>
<td>YN 2 38836 01 04 11 21</td>
</tr>
<tr>
<td>50</td>
<td>YN 2 38840 21</td>
</tr>
<tr>
<td>51</td>
<td>YN 2 38841 32</td>
</tr>
<tr>
<td>52</td>
<td>YN 2 38842 02</td>
</tr>
<tr>
<td>53</td>
<td>YN 2 38844 02 04</td>
</tr>
<tr>
<td>54</td>
<td>YN 2 38846 12</td>
</tr>
<tr>
<td>55</td>
<td>YN 2 38848 13</td>
</tr>
<tr>
<td>56</td>
<td>YN 2 38849 11</td>
</tr>
<tr>
<td>57</td>
<td>YN 2 38850 11</td>
</tr>
<tr>
<td>58</td>
<td>YN 2 38851 01</td>
</tr>
<tr>
<td>59</td>
<td>YN 2 38853 04</td>
</tr>
<tr>
<td>60</td>
<td>YN 2 38854 02 04</td>
</tr>
<tr>
<td>61</td>
<td>YN 2 38855 01</td>
</tr>
<tr>
<td>62</td>
<td>YN 2 38857 02</td>
</tr>
<tr>
<td>63</td>
<td>YN 2 38959 01 11</td>
</tr>
<tr>
<td>64</td>
<td>YN 2 46360 01</td>
</tr>
<tr>
<td>65</td>
<td>YN 2 65015 12 13 35</td>
</tr>
<tr>
<td>66</td>
<td>YN 2 65500 12 35</td>
</tr>
<tr>
<td>67</td>
<td>YN 2 65501 12 35</td>
</tr>
<tr>
<td>68</td>
<td>YN 2 65501 12 35</td>
</tr>
<tr>
<td>PAGE NUMBER</td>
<td>2</td>
</tr>
<tr>
<td>-------------</td>
<td>---</td>
</tr>
<tr>
<td>SOURCE CARD LISTING FOR</td>
<td>OFFLINE READ ETAM TDSCR</td>
</tr>
<tr>
<td>69</td>
<td>YN</td>
</tr>
<tr>
<td>70</td>
<td>YN</td>
</tr>
<tr>
<td>71</td>
<td>YN</td>
</tr>
<tr>
<td>72</td>
<td>YN</td>
</tr>
<tr>
<td>73</td>
<td>YN</td>
</tr>
<tr>
<td>74</td>
<td>YN</td>
</tr>
<tr>
<td>75</td>
<td>YN</td>
</tr>
<tr>
<td>76</td>
<td>YN</td>
</tr>
<tr>
<td>77</td>
<td>YN</td>
</tr>
<tr>
<td>78</td>
<td>YN</td>
</tr>
<tr>
<td>79</td>
<td>YN</td>
</tr>
<tr>
<td>80</td>
<td>YN</td>
</tr>
<tr>
<td>81</td>
<td>YN</td>
</tr>
<tr>
<td>82</td>
<td>YN</td>
</tr>
<tr>
<td>83</td>
<td>YN</td>
</tr>
<tr>
<td>84</td>
<td>YN</td>
</tr>
<tr>
<td>85</td>
<td>YN</td>
</tr>
<tr>
<td>86</td>
<td>YN</td>
</tr>
<tr>
<td>87</td>
<td>YN</td>
</tr>
<tr>
<td>88</td>
<td>YN</td>
</tr>
<tr>
<td>89</td>
<td>YN</td>
</tr>
<tr>
<td>90</td>
<td>YN</td>
</tr>
<tr>
<td>91</td>
<td>YN</td>
</tr>
<tr>
<td>92</td>
<td>YN</td>
</tr>
<tr>
<td>93</td>
<td>YNCS</td>
</tr>
<tr>
<td>94</td>
<td>YNCS</td>
</tr>
<tr>
<td>95</td>
<td>YNCS</td>
</tr>
<tr>
<td>96</td>
<td>YNCS</td>
</tr>
<tr>
<td>97</td>
<td>YNCS</td>
</tr>
<tr>
<td>98</td>
<td>YNCS</td>
</tr>
<tr>
<td>99</td>
<td>YNCS</td>
</tr>
<tr>
<td>100</td>
<td>YNCS</td>
</tr>
<tr>
<td>101</td>
<td>YNCS</td>
</tr>
<tr>
<td>102</td>
<td>YNCS</td>
</tr>
<tr>
<td>103</td>
<td>YNCS</td>
</tr>
<tr>
<td>104</td>
<td>YNCS</td>
</tr>
<tr>
<td>105</td>
<td>YNCS</td>
</tr>
<tr>
<td>106</td>
<td>YNCS</td>
</tr>
<tr>
<td>107</td>
<td>YNCS</td>
</tr>
<tr>
<td>108</td>
<td>BM</td>
</tr>
<tr>
<td>109</td>
<td>BM</td>
</tr>
<tr>
<td>110</td>
<td>BM</td>
</tr>
<tr>
<td>111</td>
<td>BM</td>
</tr>
<tr>
<td>112</td>
<td>BM</td>
</tr>
<tr>
<td>113</td>
<td>BM</td>
</tr>
<tr>
<td>114</td>
<td>BM</td>
</tr>
<tr>
<td>115</td>
<td>BM</td>
</tr>
<tr>
<td>116</td>
<td>BM</td>
</tr>
<tr>
<td>117</td>
<td>BM</td>
</tr>
<tr>
<td>118</td>
<td>BM</td>
</tr>
<tr>
<td>119</td>
<td>BM</td>
</tr>
<tr>
<td>120</td>
<td>BM</td>
</tr>
<tr>
<td>121</td>
<td>BM</td>
</tr>
<tr>
<td>122</td>
<td>BM</td>
</tr>
<tr>
<td>123</td>
<td>BM</td>
</tr>
<tr>
<td>124</td>
<td>BM</td>
</tr>
<tr>
<td>125</td>
<td>BM</td>
</tr>
<tr>
<td>126</td>
<td>BM</td>
</tr>
<tr>
<td>127</td>
<td>BM</td>
</tr>
<tr>
<td>128</td>
<td>BM</td>
</tr>
<tr>
<td>129</td>
<td>BM</td>
</tr>
<tr>
<td>130</td>
<td>BM</td>
</tr>
<tr>
<td>131</td>
<td>BM</td>
</tr>
<tr>
<td>132</td>
<td>BM</td>
</tr>
<tr>
<td>133</td>
<td>BM</td>
</tr>
<tr>
<td>134</td>
<td>BM</td>
</tr>
<tr>
<td>135</td>
<td>BM</td>
</tr>
<tr>
<td>136</td>
<td>BM</td>
</tr>
<tr>
<td>137</td>
<td>BM</td>
</tr>
<tr>
<td>BM</td>
<td>Listing</td>
</tr>
<tr>
<td>----</td>
<td>---------</td>
</tr>
<tr>
<td>138</td>
<td>3 40001 57 61 81</td>
</tr>
<tr>
<td>139</td>
<td>3 40511 57 61 81</td>
</tr>
<tr>
<td>140</td>
<td>3 42024 41 52 54 56 60 72</td>
</tr>
<tr>
<td>141</td>
<td>3 62001 42 60 62</td>
</tr>
<tr>
<td>142</td>
<td>3 86049 53 55</td>
</tr>
<tr>
<td>143</td>
<td>3 86050 60 61</td>
</tr>
<tr>
<td>144</td>
<td>3 94014 57 59 61 81</td>
</tr>
<tr>
<td>145</td>
<td>3 94592 57 61</td>
</tr>
<tr>
<td>146</td>
<td>3 94593 41 57 61</td>
</tr>
<tr>
<td>147</td>
<td>3 94633 41 59 61</td>
</tr>
<tr>
<td>148</td>
<td>2 11513 33 42 58 61 71 72</td>
</tr>
<tr>
<td>149</td>
<td>2 16062 33 41 58 61</td>
</tr>
<tr>
<td>150</td>
<td>2 34014 33 42 58 61 72</td>
</tr>
<tr>
<td>151</td>
<td>2 34015 33 42 58 61 72</td>
</tr>
<tr>
<td>152</td>
<td>2 34019 41 58 59</td>
</tr>
<tr>
<td>153</td>
<td>2 34021 33 41 58</td>
</tr>
<tr>
<td>154</td>
<td>2 34027 41 52 54 57 59 61</td>
</tr>
<tr>
<td>155</td>
<td>2 34032 54 56</td>
</tr>
<tr>
<td>156</td>
<td>2 34033 54 58</td>
</tr>
<tr>
<td>157</td>
<td>2 34042 33 41 57 61</td>
</tr>
<tr>
<td>158</td>
<td>2 34043 33 33 41 59 61</td>
</tr>
<tr>
<td>159</td>
<td>2 34044 72</td>
</tr>
<tr>
<td>160</td>
<td>2 34051 33 41 59 61</td>
</tr>
<tr>
<td>161</td>
<td>2 34054 58 59 91</td>
</tr>
<tr>
<td>162</td>
<td>2 34059 41 57 61</td>
</tr>
<tr>
<td>163</td>
<td>2 34060 42 57 91</td>
</tr>
<tr>
<td>164</td>
<td>2 34061 52 54 58 60</td>
</tr>
<tr>
<td>165</td>
<td>2 34066 33 52 54 61 81</td>
</tr>
<tr>
<td>166</td>
<td>2 34069 33 41 57 59 61</td>
</tr>
<tr>
<td>167</td>
<td>2 34415 33 41 57 61</td>
</tr>
<tr>
<td>168</td>
<td>2 34416 57 59 61</td>
</tr>
<tr>
<td>169</td>
<td>2 34417 33 57 59 61 81</td>
</tr>
<tr>
<td>170</td>
<td>2 35326 01 01</td>
</tr>
<tr>
<td>171</td>
<td>2 35327 01 02</td>
</tr>
<tr>
<td>172</td>
<td>2 35328 14</td>
</tr>
<tr>
<td>173</td>
<td>2 42342 31 32 33 42 58 62</td>
</tr>
<tr>
<td>174</td>
<td>2 54119 01</td>
</tr>
<tr>
<td>175</td>
<td>2 54200 01</td>
</tr>
<tr>
<td>176</td>
<td>2 54201 01</td>
</tr>
<tr>
<td>177</td>
<td>2 94005 33 58 71 72</td>
</tr>
<tr>
<td>178</td>
<td>1 11514 33 42 58 61 72 81</td>
</tr>
<tr>
<td>179</td>
<td>1 34016 33 58 82</td>
</tr>
<tr>
<td>180</td>
<td>1 34025 33 42 59 61 82</td>
</tr>
<tr>
<td>181</td>
<td>1 34026 41 59 61 82</td>
</tr>
<tr>
<td>182</td>
<td>1 34028 41 59 61 82</td>
</tr>
<tr>
<td>183</td>
<td>1 34029 41 57 58 61</td>
</tr>
<tr>
<td>184</td>
<td>1 34030 41 41 57 58 61</td>
</tr>
<tr>
<td>185</td>
<td>1 34035 52 54 56 58</td>
</tr>
<tr>
<td>186</td>
<td>1 34055 53 58</td>
</tr>
<tr>
<td>187</td>
<td>1 34056 42 50 61 82</td>
</tr>
<tr>
<td>188</td>
<td>1 34062 53 72</td>
</tr>
<tr>
<td>189</td>
<td>1 34063 53 82</td>
</tr>
<tr>
<td>190</td>
<td>1 34418 33 82</td>
</tr>
<tr>
<td>191</td>
<td>1 35350 33 82</td>
</tr>
<tr>
<td>192</td>
<td>1 35351 01 02 33</td>
</tr>
<tr>
<td>193</td>
<td>1 44062 32 33</td>
</tr>
<tr>
<td>194</td>
<td>1 54653 04</td>
</tr>
<tr>
<td>195</td>
<td>C 11515 33 59 61 72 82</td>
</tr>
<tr>
<td>196</td>
<td>C 16060 33 34 42 59</td>
</tr>
<tr>
<td>197</td>
<td>C 34022 33 41 57 59 61</td>
</tr>
<tr>
<td>198</td>
<td>C 34023 33 41 42 59 61</td>
</tr>
<tr>
<td>199</td>
<td>C 34057 53</td>
</tr>
<tr>
<td>200</td>
<td>C 34065 53</td>
</tr>
<tr>
<td>201</td>
<td>C 34074 42 58 61</td>
</tr>
<tr>
<td>202</td>
<td>C 34075 58</td>
</tr>
<tr>
<td>203</td>
<td>C 34078 42 55 58 61 91</td>
</tr>
<tr>
<td>204</td>
<td>C 34079 42 58</td>
</tr>
<tr>
<td>205</td>
<td>C 34081 53 58</td>
</tr>
<tr>
<td>206</td>
<td>C 54658 11</td>
</tr>
<tr>
<td>Page</td>
<td>Type</td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>207</td>
<td>BM</td>
</tr>
<tr>
<td>208</td>
<td>BM</td>
</tr>
<tr>
<td>209</td>
<td>BM</td>
</tr>
<tr>
<td>210</td>
<td>BM</td>
</tr>
<tr>
<td>211</td>
<td>BM</td>
</tr>
<tr>
<td>212</td>
<td>BM</td>
</tr>
<tr>
<td>213</td>
<td>BM</td>
</tr>
<tr>
<td>214</td>
<td>BM</td>
</tr>
<tr>
<td>215</td>
<td>BM</td>
</tr>
<tr>
<td>216</td>
<td>BM</td>
</tr>
<tr>
<td>217</td>
<td>BM</td>
</tr>
<tr>
<td>218</td>
<td>BM</td>
</tr>
<tr>
<td>219</td>
<td>BM</td>
</tr>
<tr>
<td>220</td>
<td>BM</td>
</tr>
<tr>
<td>221</td>
<td>BM</td>
</tr>
<tr>
<td>222</td>
<td>BM</td>
</tr>
<tr>
<td>223</td>
<td>BM</td>
</tr>
<tr>
<td>224</td>
<td>BM</td>
</tr>
<tr>
<td>225</td>
<td>AN</td>
</tr>
<tr>
<td>226</td>
<td>AN</td>
</tr>
<tr>
<td>227</td>
<td>AN</td>
</tr>
<tr>
<td>228</td>
<td>AN</td>
</tr>
<tr>
<td>229</td>
<td>AN</td>
</tr>
<tr>
<td>230</td>
<td>AN</td>
</tr>
<tr>
<td>231</td>
<td>AN</td>
</tr>
<tr>
<td>232</td>
<td>AN</td>
</tr>
<tr>
<td>233</td>
<td>AN</td>
</tr>
<tr>
<td>234</td>
<td>AN</td>
</tr>
<tr>
<td>235</td>
<td>AN</td>
</tr>
<tr>
<td>236</td>
<td>AN</td>
</tr>
<tr>
<td>237</td>
<td>AN</td>
</tr>
<tr>
<td>238</td>
<td>AN</td>
</tr>
<tr>
<td>239</td>
<td>AN</td>
</tr>
<tr>
<td>240</td>
<td>AN</td>
</tr>
<tr>
<td>241</td>
<td>AN</td>
</tr>
<tr>
<td>242</td>
<td>AN</td>
</tr>
<tr>
<td>243</td>
<td>AN</td>
</tr>
<tr>
<td>244</td>
<td>AN</td>
</tr>
<tr>
<td>245</td>
<td>AN</td>
</tr>
<tr>
<td>246</td>
<td>AN</td>
</tr>
<tr>
<td>247</td>
<td>AN</td>
</tr>
<tr>
<td>248</td>
<td>AN</td>
</tr>
<tr>
<td>249</td>
<td>AN</td>
</tr>
<tr>
<td>250</td>
<td>AN</td>
</tr>
<tr>
<td>251</td>
<td>AN</td>
</tr>
<tr>
<td>252</td>
<td>AN</td>
</tr>
<tr>
<td>253</td>
<td>AN</td>
</tr>
<tr>
<td>254</td>
<td>AN</td>
</tr>
<tr>
<td>255</td>
<td>AN</td>
</tr>
<tr>
<td>256</td>
<td>AN</td>
</tr>
<tr>
<td>257</td>
<td>AN</td>
</tr>
<tr>
<td>258</td>
<td>AN</td>
</tr>
<tr>
<td>259</td>
<td>AN</td>
</tr>
<tr>
<td>260</td>
<td>AN</td>
</tr>
<tr>
<td>261</td>
<td>AN</td>
</tr>
<tr>
<td>262</td>
<td>AN</td>
</tr>
<tr>
<td>263</td>
<td>AN</td>
</tr>
<tr>
<td>264</td>
<td>AN</td>
</tr>
<tr>
<td>265</td>
<td>AN</td>
</tr>
<tr>
<td>266</td>
<td>AN</td>
</tr>
<tr>
<td>267</td>
<td>AN</td>
</tr>
<tr>
<td>268</td>
<td>AN</td>
</tr>
<tr>
<td>269</td>
<td>AN</td>
</tr>
<tr>
<td>270</td>
<td>AN</td>
</tr>
<tr>
<td>271</td>
<td>SM</td>
</tr>
<tr>
<td>272</td>
<td>SM</td>
</tr>
<tr>
<td>273</td>
<td>SM</td>
</tr>
<tr>
<td>274</td>
<td>SM</td>
</tr>
<tr>
<td>275</td>
<td>SM</td>
</tr>
<tr>
<td>PAGE NUMBER</td>
<td>SOURCE CARD LISTING FOR OFFLINE READ ETAM TOSCR</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------------------------------------------</td>
</tr>
<tr>
<td>276 SM</td>
<td>3 54722 01 52 59 61</td>
</tr>
<tr>
<td>277 SM</td>
<td>3 72301 41 55 60 61</td>
</tr>
<tr>
<td>278 SM</td>
<td>3 77264 02 55 60 61</td>
</tr>
<tr>
<td>279 SM</td>
<td>3 86015 53 55 60 61</td>
</tr>
<tr>
<td>280 SM</td>
<td>3 86016 55 60 61</td>
</tr>
<tr>
<td>281 SM</td>
<td>3 86017 55 60 61</td>
</tr>
<tr>
<td>282 SM</td>
<td>3 86018 55 60 61</td>
</tr>
<tr>
<td>283 SM</td>
<td>3 86021 55 61</td>
</tr>
<tr>
<td>284 SM</td>
<td>3 86022 41 60</td>
</tr>
<tr>
<td>285 SM</td>
<td>3 86023 41 54</td>
</tr>
<tr>
<td>286 SM</td>
<td>3 86024 42 60 62 91</td>
</tr>
<tr>
<td>287 SM</td>
<td>3 86025 53</td>
</tr>
<tr>
<td>288 SM</td>
<td>3 86031 41 60 82</td>
</tr>
<tr>
<td>289 SM</td>
<td>3 86035 41 60</td>
</tr>
<tr>
<td>290 SM</td>
<td>3 86036 41 60</td>
</tr>
<tr>
<td>291 SM</td>
<td>3 86038 43 60</td>
</tr>
<tr>
<td>292 SM</td>
<td>3 86039 43 55</td>
</tr>
<tr>
<td>293 SM</td>
<td>3 86044 43 60</td>
</tr>
<tr>
<td>294 SM</td>
<td>3 86045 43 55</td>
</tr>
<tr>
<td>295 SM</td>
<td>3 86046 43 55</td>
</tr>
<tr>
<td>296 SM</td>
<td>3 86047 43 55</td>
</tr>
<tr>
<td>297 SM</td>
<td>3 86048 43 55</td>
</tr>
<tr>
<td>298 SM</td>
<td>3 86049 43 55</td>
</tr>
<tr>
<td>299 SM</td>
<td>3 86050 43 55</td>
</tr>
<tr>
<td>300 SM</td>
<td>3 86051 43 55</td>
</tr>
<tr>
<td>301 SM</td>
<td>3 86052 43 55</td>
</tr>
<tr>
<td>302 SM</td>
<td>3 86053 43 55</td>
</tr>
<tr>
<td>303 SM</td>
<td>3 86054 43 55</td>
</tr>
<tr>
<td>304 SM</td>
<td>3 86055 43 55</td>
</tr>
<tr>
<td>305 SM</td>
<td>3 86056 43 55</td>
</tr>
<tr>
<td>306 SM</td>
<td>3 86057 43 55</td>
</tr>
<tr>
<td>307 SM</td>
<td>3 86058 43 55</td>
</tr>
<tr>
<td>308 SM</td>
<td>3 86059 43 55</td>
</tr>
<tr>
<td>309 SM</td>
<td>3 86060 43 55</td>
</tr>
<tr>
<td>310 SM</td>
<td>3 86061 43 55</td>
</tr>
<tr>
<td>311 SM</td>
<td>3 86062 43 55</td>
</tr>
<tr>
<td>312 SM</td>
<td>3 86063 43 55</td>
</tr>
<tr>
<td>313 SM</td>
<td>3 86064 43 55</td>
</tr>
<tr>
<td>314 SM</td>
<td>3 86065 43 55</td>
</tr>
<tr>
<td>315 SM</td>
<td>3 86066 43 55</td>
</tr>
<tr>
<td>316 SM</td>
<td>3 86067 43 55</td>
</tr>
<tr>
<td>317 SM</td>
<td>3 86068 43 55</td>
</tr>
<tr>
<td>318 SM</td>
<td>3 86069 43 55</td>
</tr>
<tr>
<td>319 SM</td>
<td>3 86070 43 55</td>
</tr>
<tr>
<td>320 SM</td>
<td>3 86071 43 55</td>
</tr>
<tr>
<td>321 SM</td>
<td>3 86072 43 55</td>
</tr>
<tr>
<td>322 SM</td>
<td>3 86073 43 55</td>
</tr>
<tr>
<td>323 SM</td>
<td>3 86074 43 55</td>
</tr>
<tr>
<td>324 SM</td>
<td>3 86075 43 55</td>
</tr>
<tr>
<td>325 SM</td>
<td>3 86076 43 55</td>
</tr>
<tr>
<td>326 SM</td>
<td>3 86077 43 55</td>
</tr>
<tr>
<td>327 SM</td>
<td>3 86078 43 55</td>
</tr>
<tr>
<td>328 SM</td>
<td>3 86079 43 55</td>
</tr>
<tr>
<td>329 SM</td>
<td>3 86080 43 55</td>
</tr>
<tr>
<td>330 SM</td>
<td>3 86081 43 55</td>
</tr>
<tr>
<td>331 SM</td>
<td>3 86082 43 55</td>
</tr>
<tr>
<td>332 SM</td>
<td>3 86083 43 55</td>
</tr>
<tr>
<td>333 SM</td>
<td>3 86084 43 55</td>
</tr>
<tr>
<td>334 SM</td>
<td>3 86085 43 55</td>
</tr>
<tr>
<td>335 SM</td>
<td>3 86086 43 55</td>
</tr>
<tr>
<td>336 SM</td>
<td>3 86087 43 55</td>
</tr>
<tr>
<td>337 SM</td>
<td>3 86088 43 55</td>
</tr>
<tr>
<td>338 SM</td>
<td>3 86089 43 55</td>
</tr>
<tr>
<td>339 SM</td>
<td>3 86090 43 55</td>
</tr>
<tr>
<td>340 SM</td>
<td>3 86091 43 55</td>
</tr>
<tr>
<td>341 SM</td>
<td>3 86092 43 55</td>
</tr>
<tr>
<td>342 SM</td>
<td>3 86093 43 55</td>
</tr>
<tr>
<td>343 SM</td>
<td>3 86094 43 55</td>
</tr>
<tr>
<td>344 SM</td>
<td>3 86095 43 55</td>
</tr>
<tr>
<td>PAGE NUMBER</td>
<td>SOURCE CARD LISTING FOR</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------</td>
</tr>
<tr>
<td>414 EN</td>
<td>3 98024 42 58 61</td>
</tr>
<tr>
<td>415 EN</td>
<td>3 98025 71</td>
</tr>
<tr>
<td>416 EN</td>
<td>3 98026 42 82</td>
</tr>
<tr>
<td>417 EN</td>
<td>3 98335 42 82</td>
</tr>
<tr>
<td>418 EN</td>
<td>2 11093 52 54 61 81</td>
</tr>
<tr>
<td>419 EN</td>
<td>2 11447 54 72</td>
</tr>
<tr>
<td>420 EN</td>
<td>2 30319 41 57 61</td>
</tr>
<tr>
<td>421 EN</td>
<td>2 30324 42 57 61</td>
</tr>
<tr>
<td>422 EN</td>
<td>2 30325 52 54</td>
</tr>
<tr>
<td>423 EN</td>
<td>2 30326 42 54 57 61</td>
</tr>
<tr>
<td>424 EN</td>
<td>2 30331 41 59 61</td>
</tr>
<tr>
<td>425 EN</td>
<td>2 30332 41 54 57 61</td>
</tr>
<tr>
<td>426 EN</td>
<td>2 30333 41 54 57 61</td>
</tr>
<tr>
<td>427 EN</td>
<td>2 30336 41 54 57 61</td>
</tr>
<tr>
<td>428 EN</td>
<td>2 30338 41 54 57 61</td>
</tr>
<tr>
<td>429 EN</td>
<td>2 30339 54 61</td>
</tr>
<tr>
<td>430 EN</td>
<td>2 30340 41 54 59 61</td>
</tr>
<tr>
<td>431 EN</td>
<td>2 30341 41 52 54 61</td>
</tr>
<tr>
<td>432 EN</td>
<td>2 30344 41 57 59 61</td>
</tr>
<tr>
<td>433 EN</td>
<td>2 30347 42 59 54 61</td>
</tr>
<tr>
<td>434 EN</td>
<td>2 30358 41 59 61</td>
</tr>
<tr>
<td>435 EN</td>
<td>2 30372 41 57 59 61</td>
</tr>
<tr>
<td>436 EN</td>
<td>2 30374 61 71</td>
</tr>
<tr>
<td>437 EN</td>
<td>2 30911 41 52 54 61</td>
</tr>
<tr>
<td>438 EN</td>
<td>2 30912 41 52 54 61</td>
</tr>
<tr>
<td>439 EN</td>
<td>2 30913 41 57 61</td>
</tr>
<tr>
<td>440 EN</td>
<td>2 30928 42 57 59 61</td>
</tr>
<tr>
<td>441 EN</td>
<td>2 30930 42 57 59 61</td>
</tr>
<tr>
<td>442 EN</td>
<td>2 30966 41 52 54 61</td>
</tr>
<tr>
<td>443 EN</td>
<td>2 30981 41 57 59 61</td>
</tr>
<tr>
<td>444 EN</td>
<td>2 30984 41 57 59 61</td>
</tr>
<tr>
<td>445 EN</td>
<td>2 30985 41 57 61</td>
</tr>
<tr>
<td>446 EN</td>
<td>2 30986 41 57 61</td>
</tr>
<tr>
<td>447 EN</td>
<td>2 30987 41 57 61</td>
</tr>
<tr>
<td>448 EN</td>
<td>2 30988 41 57 61</td>
</tr>
<tr>
<td>449 EN</td>
<td>2 30989 41 57 61</td>
</tr>
<tr>
<td>450 EN</td>
<td>2 30990 41 57 61</td>
</tr>
<tr>
<td>451 EN</td>
<td>2 30991 41 57 61</td>
</tr>
<tr>
<td>452 EN</td>
<td>2 30992 41 57 61</td>
</tr>
<tr>
<td>453 EN</td>
<td>2 30993 41 57 61</td>
</tr>
<tr>
<td>454 EN</td>
<td>2 30994 41 57 61</td>
</tr>
<tr>
<td>455 EN</td>
<td>2 30995 41 57 61</td>
</tr>
<tr>
<td>456 EN</td>
<td>2 30996 41 57 61</td>
</tr>
<tr>
<td>457 EN</td>
<td>2 30997 41 57 61</td>
</tr>
<tr>
<td>458 EN</td>
<td>2 30998 41 57 61</td>
</tr>
<tr>
<td>459 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>460 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>461 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>462 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>463 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>464 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>465 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>466 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>467 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>468 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>469 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>470 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>471 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>472 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>473 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>474 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>475 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>476 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>477 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>478 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>479 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>480 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>481 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>482 EN</td>
<td>2 30999 41 57 61</td>
</tr>
<tr>
<td>Page Number: 8</td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td></td>
</tr>
<tr>
<td><strong>SOURCE CARD LISTING FOR</strong></td>
<td></td>
</tr>
<tr>
<td><strong>OFFLINE READ ETAM TDSCR</strong></td>
<td></td>
</tr>
</tbody>
</table>

<p>| EN | 483 | 131253 41 52 54 57 61 |
| 484 | 31494 41 52 54 59 61 81 |
| 485 | 35323 01 02 |
| 486 | 51167 12 |
| 487 | 51168 14 |
| 488 | 54827 01 04 |
| 489 | C 30359 41 53 54 61 |
| 490 | C 30674 32 33 41 57 59 61 81 |
| 491 | C 31045 41 52 54 57 61 |
| 492 | C 31046 41 52 54 57 61 |
| 493 | C 31047 41 54 57 58 61 |
| 494 | C 31077 11 41 52 54 61 |
| 495 | C 31095 41 52 54 56 61 |
| 496 | C 35432 01 33 42 58 61 82 |
| 497 | C 35433 12 |
| 498 | C 35885 31 32 33 41 61 |
| 499 | C 35886 12 |
| 500 | C 42278 31 |
| 501 | C 42279 33 41 42 61 |
| 502 | C 42367 31 82 |
| 503 | C 51069 61 82 |
| 504 | C 51169 02 14 |
| 505 | C 51475 12 |
| 506 | C 51476 55 58 71 |
| 507 | C 51477 55 71 |
| 508 | C 51478 55 71 |
| 509 | C 54804 11 13 |
| 510 | C 98021 33 42 58 61 71 72 |
| 511 | C 351410 42 58 60 62 |
| 512 | C 351455 42 55 61 71 |
| 513 | C 351455 58 61 71 82 |
| 514 | C 353072 01 02 33 |
| 515 | C 353562 01 02 04 31 32 |
| 516 | C 35479 11 13 14 |
| 517 | C 35479 11 |
| 518 | C 35480 14 33 34 |
| 519 | C 35588 12 13 14 |
| 520 | C 35888 12 |
| 521 | C 44375 13 14 32 34 |
| 522 | C 451073 33 |
| 523 | C 51206 33 82 |
| 524 | C 59820 12 |
| 525 | C 59828 13 14 33 |
| 526 | C 59828 13 |
| 527 | C 20060 13 14 33 |
| 528 | C 31457 12 |
| 529 | C 35365 11 |
| 530 | C 35366 04 11 12 13 |
| 531 | C 35483 12 14 33 |
| 532 | C 35489 12 14 |
| 533 | C 35490 12 13 14 |
| 534 | C 35492 12 14 |
| 535 | C 46616 12 |
| 536 | C 51072 12 |
| 537 | C 51075 12 |
| 538 | C 51179 35 |
| 539 | C 552298 12 13 14 |
| 540 | C 554806 42 61 71 82 |
| 541 | GMG 3 14086 41 53 61 |
| 542 | GMG 3 14566 53 61 |
| 543 | GMG 3 14566 41 53 61 |
| 544 | GMG 3 14570 53 54 61 |
| 545 | GMG 3 15305 41 52 53 54 59 61 |
| 546 | GMG 3 15307 41 53 54 55 56 60 61 |
| 547 | GMG 3 15314 54 55 58 61 |
| 548 | GMG 3 15315 41 52 54 59 61 |
| 549 | GMG 3 16027 53 54 55 56 61 |
| 550 | GMG 3 16033 42 57 59 61 |
| 551 | GMG 3 16034 01 41 59 61 |</p>
<table>
<thead>
<tr>
<th>PAGE NUMBER : 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOURCE CARD LISTING FOR</td>
</tr>
<tr>
<td>OFFLINE READ ETAM TDESCR</td>
</tr>
<tr>
<td>552</td>
</tr>
<tr>
<td>553</td>
</tr>
<tr>
<td>554</td>
</tr>
<tr>
<td>555</td>
</tr>
<tr>
<td>556</td>
</tr>
<tr>
<td>557</td>
</tr>
<tr>
<td>558</td>
</tr>
<tr>
<td>559</td>
</tr>
<tr>
<td>560</td>
</tr>
<tr>
<td>561</td>
</tr>
<tr>
<td>562</td>
</tr>
<tr>
<td>563</td>
</tr>
<tr>
<td>564</td>
</tr>
<tr>
<td>565</td>
</tr>
<tr>
<td>566</td>
</tr>
<tr>
<td>567</td>
</tr>
<tr>
<td>568</td>
</tr>
<tr>
<td>569</td>
</tr>
<tr>
<td>570</td>
</tr>
<tr>
<td>571</td>
</tr>
<tr>
<td>572</td>
</tr>
<tr>
<td>573</td>
</tr>
<tr>
<td>574</td>
</tr>
<tr>
<td>575</td>
</tr>
<tr>
<td>576</td>
</tr>
<tr>
<td>577</td>
</tr>
<tr>
<td>578</td>
</tr>
<tr>
<td>579</td>
</tr>
<tr>
<td>580</td>
</tr>
<tr>
<td>581</td>
</tr>
<tr>
<td>582</td>
</tr>
<tr>
<td>583</td>
</tr>
<tr>
<td>584</td>
</tr>
<tr>
<td>585</td>
</tr>
<tr>
<td>586</td>
</tr>
<tr>
<td>587</td>
</tr>
<tr>
<td>588</td>
</tr>
<tr>
<td>589</td>
</tr>
<tr>
<td>590</td>
</tr>
<tr>
<td>591</td>
</tr>
<tr>
<td>592</td>
</tr>
<tr>
<td>593</td>
</tr>
<tr>
<td>594</td>
</tr>
<tr>
<td>595</td>
</tr>
<tr>
<td>596</td>
</tr>
<tr>
<td>597</td>
</tr>
<tr>
<td>598</td>
</tr>
<tr>
<td>599</td>
</tr>
<tr>
<td>600</td>
</tr>
<tr>
<td>601</td>
</tr>
<tr>
<td>602</td>
</tr>
<tr>
<td>603</td>
</tr>
<tr>
<td>604</td>
</tr>
<tr>
<td>605</td>
</tr>
<tr>
<td>606</td>
</tr>
<tr>
<td>607</td>
</tr>
<tr>
<td>608</td>
</tr>
<tr>
<td>609</td>
</tr>
<tr>
<td>610</td>
</tr>
<tr>
<td>611</td>
</tr>
<tr>
<td>612</td>
</tr>
<tr>
<td>613</td>
</tr>
<tr>
<td>614</td>
</tr>
<tr>
<td>615</td>
</tr>
<tr>
<td>616</td>
</tr>
<tr>
<td>617</td>
</tr>
<tr>
<td>618</td>
</tr>
<tr>
<td>619</td>
</tr>
<tr>
<td>620</td>
</tr>
</tbody>
</table>
TAEG REPORT NO. 40

SOURCE CARD LISTING FOR
OFFLINE READ ETAM TDSGR

<table>
<thead>
<tr>
<th>PAGE NUMBER</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMG</td>
<td>2 29245 41 57 59 61</td>
</tr>
<tr>
<td>GMG</td>
<td>2 29246 41 52 54 56 61</td>
</tr>
<tr>
<td>GMG</td>
<td>2 29248 41 54 56 61</td>
</tr>
<tr>
<td>GMG</td>
<td>2 29249 41 52 54 61</td>
</tr>
<tr>
<td>GMG</td>
<td>2 29250 41 52 54 61</td>
</tr>
<tr>
<td>GMG</td>
<td>2 29251 41 54 56 61 81</td>
</tr>
<tr>
<td>GMG</td>
<td>2 29252 01 02 12</td>
</tr>
<tr>
<td>GMG</td>
<td>2 36450 01 02</td>
</tr>
<tr>
<td>GMG</td>
<td>2 36451 01 02</td>
</tr>
<tr>
<td>GMG</td>
<td>2 51114 01 11</td>
</tr>
<tr>
<td>GMG</td>
<td>2 54708 01</td>
</tr>
<tr>
<td>GMG</td>
<td>2 54813 01</td>
</tr>
<tr>
<td>GMG</td>
<td>2 54814 01 02 53 54</td>
</tr>
<tr>
<td>GMG</td>
<td>2 54815 01</td>
</tr>
<tr>
<td>GMG</td>
<td>1 14018 61 71</td>
</tr>
<tr>
<td>GMG</td>
<td>1 15313 41 51 52 53 54 57 61 81</td>
</tr>
<tr>
<td>GMG</td>
<td>1 16023 33 42 61</td>
</tr>
<tr>
<td>GMG</td>
<td>1 18018 61 71</td>
</tr>
<tr>
<td>GMG</td>
<td>1 18021 41 53 55 61</td>
</tr>
<tr>
<td>GMG</td>
<td>1 28018 42 53 55 57 58 82</td>
</tr>
<tr>
<td>GMG</td>
<td>1 29253 41 52 54 57 61</td>
</tr>
<tr>
<td>GMG</td>
<td>1 29254 41 52 54 61 81</td>
</tr>
<tr>
<td>GMG</td>
<td>1 29255 41 54 57 61 81</td>
</tr>
<tr>
<td>GMG</td>
<td>1 29256 41 54 57 61 81</td>
</tr>
<tr>
<td>GMG</td>
<td>1 29551 41 54 57 61 81</td>
</tr>
<tr>
<td>GMG</td>
<td>1 29552 41 54 57 59 61 81</td>
</tr>
<tr>
<td>GMG</td>
<td>1 38344 01 12</td>
</tr>
<tr>
<td>GMG</td>
<td>1 38345 01 11 12</td>
</tr>
<tr>
<td>GMG</td>
<td>1 38346 01 02</td>
</tr>
<tr>
<td>GMG</td>
<td>1 38687 01 02</td>
</tr>
<tr>
<td>GMG</td>
<td>1 38888 14 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 51167 12</td>
</tr>
<tr>
<td>GMG</td>
<td>1 51168 14 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 54816 01 12 14</td>
</tr>
<tr>
<td>GMG</td>
<td>1 54813 11 12</td>
</tr>
<tr>
<td>GMG</td>
<td>1 54844 01 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 98255 11</td>
</tr>
<tr>
<td>GMG</td>
<td>1 14042 42 51 52 53 54 58 61 71</td>
</tr>
<tr>
<td>GMG</td>
<td>1 14044 33 34 53 55 58 61 72</td>
</tr>
<tr>
<td>GMG</td>
<td>1 14045 51 52 53 54 55 56 61 72</td>
</tr>
<tr>
<td>GMG</td>
<td>1 14046 12 14</td>
</tr>
<tr>
<td>GMG</td>
<td>1 16032 33 61</td>
</tr>
<tr>
<td>GMG</td>
<td>1 16411 33 61</td>
</tr>
<tr>
<td>GMG</td>
<td>1 17426 71</td>
</tr>
<tr>
<td>GMG</td>
<td>1 17427 42 51 52 53 54 55 56 58 61</td>
</tr>
<tr>
<td>GMG</td>
<td>1 20276 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 38892 01 11</td>
</tr>
<tr>
<td>GMG</td>
<td>1 38893 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 38897 12</td>
</tr>
<tr>
<td>GMG</td>
<td>1 51079 14 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 51169 14</td>
</tr>
<tr>
<td>GMG</td>
<td>1 54576 01 53 54 61</td>
</tr>
<tr>
<td>GMG</td>
<td>1 98396 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 17417 33 61</td>
</tr>
<tr>
<td>GMG</td>
<td>1 35341 11 12</td>
</tr>
<tr>
<td>GMG</td>
<td>1 35476 12 13 35</td>
</tr>
<tr>
<td>GMG</td>
<td>1 35478 13 14</td>
</tr>
<tr>
<td>GMG</td>
<td>1 35479 11</td>
</tr>
<tr>
<td>GMG</td>
<td>1 35480 14 33 34</td>
</tr>
<tr>
<td>GMG</td>
<td>1 44375 15 14 32</td>
</tr>
<tr>
<td>GMG</td>
<td>1 51153 33 34</td>
</tr>
<tr>
<td>GMG</td>
<td>1 51154 33 34</td>
</tr>
<tr>
<td>GMG</td>
<td>1 51206 12 14 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 52196 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 52921 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 35483 14 33</td>
</tr>
<tr>
<td>GMG</td>
<td>1 35489 12 14</td>
</tr>
<tr>
<td>GMG</td>
<td>1 35490 12 14</td>
</tr>
<tr>
<td>Number</td>
<td>Code</td>
</tr>
<tr>
<td>--------</td>
<td>------</td>
</tr>
<tr>
<td>690</td>
<td>GM</td>
</tr>
<tr>
<td>691</td>
<td>GM</td>
</tr>
<tr>
<td>692</td>
<td>GM</td>
</tr>
<tr>
<td>693</td>
<td>GM</td>
</tr>
<tr>
<td>694</td>
<td>GM</td>
</tr>
<tr>
<td>695</td>
<td>GM</td>
</tr>
<tr>
<td>696</td>
<td>GM</td>
</tr>
</tbody>
</table>
TAEG REPORT NO. 40
PAGE NUMBER

I

1

SOURCE CARD LISTING FOR
OFFLINE REAO
ETAM VCDESC
l
2
3
<»
5
6
7
8
9
10

n

13
14
15
16

\l

IS
21
22

n

25
26
27
28

I?31
32
33
3*
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52

8
55

56
57
58
59
60
61
62
63
64

OFFLINE REAO
692DLLC003292
6910LLC003318
6930LLC001008
691DLLC001564
6930LLC001407
6910LLC002061
6910LLC002062
6910005723079
6910LLC002063
6910LLC003395
6913LLC003396
6910LLC003397
6910LLC003398
6913LLC001403
6910LLC003399
6910LLC003400
691DLLC003401
6910LLC003402
6910LLC003403
6910LLC000468
691DLLC00D469
6910LLC000470
6910LLC00U70
6910LLC000471
6910LLC000472
6910005231416
6910LLC000473
6910LLC000474
6940LLC000371
6940006641435
6910LLC003636
6910LLC003665
6910LLC001614
6910LLC001615
6910LLC001616
6910LLC001617
6910LLC003437
6940LLC003443
6910LLC003444
6940LLC003445
6940LLC003447
6940LLC003448
6940LLC003449
6940005271901
6940005271904
6910005271905
6910005606623
6940005606627
6910005606628
6940005723015
6910005723016
6910005723017
6940005723018
6910006583654
6940007123047
6940008506112
6910LLC001162
6910LLC002134
6920008289165
6920006499282
6910LLC000900
6930LLC000593
6910LLC002009

ETAM VCDESC
13 12 21 22
02 12 21 31
09 14 21 22
02 11 21 33
09
03
\\
03 ii 21 32
03 11 21 32
03 11
32
03 11
32
03 11 21 32
03 11 21 32
03 11 21 32
03 11 21 32
03 11 21 32
03
21 32
03
21 32
03 11 21 32
03 U 21 32
03 11 21 32
03 11 21 32
21 32
21 32
03 11 21 32
03 11 21 32
03 11 21 32
03 11 21 32

11

I

54
43
33
53
35
61
61

il

t\
11 H
61 71 81

1t

°ol 11

II fi li

&
09 14 21 33
05 12 21 35
03
05

j il

11 21
11
8
05 11 il
05 11 21
81
1111
05 11 21

36
32

32
32
32
32

32
32

32
05 11 21 32
05
21
05
21
05 11 21 32
05 11 21 32

11

ff

8?
11
05 11

21 32
21 32

21 32
11 21 32

32
32
11
U 21 32
05 11 21 32

P

il

l\ 11 il
°ol
li il
05 12 21

71
71
71
71
71
71
71

32
32
32
35

22
03
21 32
09 \\ 21 32
02 12 13 14

73
61
45
71
52
81
81

81
7]
5«
71 81
81
65 71 81

81
81
81
81
81
81
81

If ff
71
71
71
71
71
71
71
71

81
81
81
81
81
81
81
81
81
66 72 81
66 72 83
61 71 81

HU

i\
| H
61 71 81

61 71 81
61 71 81
61 71 81

t\61 |71 11
81
61 71 81
t\ { Sl8

11

8f 1111

21
21

l\

61
61
61
61
61
61
61
61
61
61
61
61
61
61
61
61
61

65
52
34
61
45
71
71

Hit

8
81
8
81
8

i\U

t\n

8
81
8
81
61 71 81
61718
61718.
66 73 8
54 65 7
81
61 71 8
55 63 fj!
32 33 34
44 51 54 62 63 64 72 82 83

H?t

a

0.5-31


TAEG REPORT NO. 40

SOURCE CARD LISTING FOR OFFLINE READ ETAM VDATA

1 OFFLINE READ ETAM VDATA
2 6920LCC00329323116-3 BOOBY TRAP SIMULATOR, TELEPHONE
3 6910LCC0033183479-1 TRANSPARENCY, MB8 TANK; TRACKS, SUSPENSION 33100
4 6933LCC0010008230-1 F-4B AIRCRAFT COCKPIT PROCEDURES TRAINER 84200
5 6910LCC001564114-1 AERODYNAMICS DISPLAY SET 18950000
6 6933LCC00140796-1 EJECTION SEAT, UNIVERSAL 36400
7 6900LCC0206112-01A TOTAL DIFFERENTIAL (MATH) TRAINING AID 10000
8 6940LCC0037114021-8 ASW SYSTEM SIMULATOR-JEZEBEL 64000
9 6900006641435151-1 APS-T3, ULTRASONIC TRAINER 83100
10 6910LCC003368518F228-1 FUEL PUMP, AUTOMATIC TRAINING AID 271100
11 6910LCC003366518F228-1 FLASH BLINDNESS INDENTIFICATION TRAINER 271100
12 694000527190126A11 ELECTRONIC CIRCUIT TRAINER 102000
13 6900008289165352E-1 TARGET SCORING MECHANISM, SINGLE POP-UP 15153180
14 692000649292565C-1 SIMULATOR, SMALL ARMS; FLASH-NOISE 4220000
15 6910LCC0009003E34-1 106MM RECOILLESS RIFLE, Firing Mech. Transp. 15153180
16 6930LCC0009531514-1 R-4B, AIRCRAFT TACTICS TRAINER 32500
17 6910LCC0020911H2-1 SIMULATOR, GENERAL PURPOSE 4220000
18 690000572307912-0-3 GRADIENT OF A FUNCTION DEMONSTRATOR 52500
19 69100572307912-0-3 TRIPLE VECTOR PRODUCT DEMONSTRATOR 52500
20 6910LCC00206312-0-6 STOKES THEOREM DEMONSTRATOR 27500
21 6910LCC00046912-0-18 QUINCEX BINOMIAL EXPANSION DEMONSTRATOR 5500
22 6910LCC00393612-0-6 ORTHOGRAPHIC PROJECTION DEMONSTRATOR 38000
23 6910LCC00393172-0-6 TRIGONOMETRIC FUNCTION TRAINING AID 2000000
24 6910LCC00393172-0-6 COMPONENT INTEGRATOR DEMONSTRATOR T/A 800000
25 6910LCC00393172-0-6 TRIGONOMETRIC TRAINER DEMONSTRATION AID 2000000
26 6910LCC00393172-0-6 CURVE DEMONSTRATOR TRAINING AID 150000
27 6910LCC00340012-0-13 VECTOR DISTRIBUTION LAWS TRAINING AID 38000
28 6910LCC00340212-0-15 CYLINDRICAL COORDINATES DEMONSTRATOR T/A 38000
29 6910LCC00340212-0-15 SPHERICAL COORDINATES DEMONSTRATOR T/A 21000
30 6910LCC00340212-0-15 MATH, VECTOR ACCELERATION DEMONSTRATOR T/A 50000
31 6910LCC00340212-0-15 MATH, QUADRIC SURFACE EXPANSION DEMONSTRATOR 50000
32 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
33 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
34 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
35 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
36 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
37 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
38 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
39 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
40 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
41 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
42 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
43 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
44 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
45 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
46 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
47 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
48 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
49 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
50 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
51 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
52 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
53 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
54 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
55 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
56 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
57 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
58 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
59 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
60 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
61 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
62 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
63 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
64 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000
65 6910LCC00340212-0-15 MATH, ORDER OF DIFFERENTIATION DEMONSTRATOR T/A 150000

D.5-32
The following represents the source program listings of the eighteen PL/1 program modules involved in the ETAM project up through the Range-of-Effect search and search results print.

In the pages following, the programs are source-listed in alphabetic order, P1, P2, P3, ..., P20. The order presented immediately below is the logical order of processing.

<table>
<thead>
<tr>
<th>Program Name</th>
<th>Program Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P18</td>
<td>Initialization of the Master Reference data set, ETAM/MREF</td>
</tr>
<tr>
<td>P16</td>
<td>Initialization of the Descriptor Master Index, ETAM/DIND</td>
</tr>
<tr>
<td>P14</td>
<td>Initialization of the Course data file, ETAM/COURSE</td>
</tr>
<tr>
<td>P7</td>
<td>Setup of the Course CIN and CDP directories (ETAM/DCIN and ETAM/DCDP, respectively)</td>
</tr>
<tr>
<td>P7A</td>
<td>Course Descriptor file load (ETAM/DESC)</td>
</tr>
<tr>
<td>P7B</td>
<td>Printer dump of the contents of the Abbreviated Course data base by option:</td>
</tr>
<tr>
<td></td>
<td>A = All Course data, sorted by CIN</td>
</tr>
<tr>
<td></td>
<td>B = All Course data, sorted by CDP</td>
</tr>
<tr>
<td></td>
<td>C = Course descriptor data</td>
</tr>
<tr>
<td>P8</td>
<td>Load of the Abbreviated Vehicle data file (ETAM/VEHS), Vehicle Directory (ETAM/DVEH) and Vehicle Descriptor File (ETAM/VDESC).</td>
</tr>
<tr>
<td>P8A</td>
<td>Print of Abbreviated Vehicle data base contents, via Option as follows:</td>
</tr>
<tr>
<td></td>
<td>A = Print all Vehicle records, ordered by Stock Number</td>
</tr>
<tr>
<td></td>
<td>B = Print Descriptor data assigned to Vehicles</td>
</tr>
<tr>
<td>P9</td>
<td>Load the Abbreviated Tasks data file (ETAM/TASKS), Tasks Directory (ETAM/DTASK), and Tasks Descriptor File (ETAM/TDESC).</td>
</tr>
<tr>
<td>P9A</td>
<td>Print, under option control, of the contents of the Abbreviated Tasks data base. Options are as follows:</td>
</tr>
<tr>
<td></td>
<td>A = Print all Task data</td>
</tr>
<tr>
<td></td>
<td>B = Print Descriptor data assigned to Tasks</td>
</tr>
<tr>
<td>Program Name</td>
<td>Program Description</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>P20</td>
<td>Print, via selection, the exact Descriptor assignments to each Course, Vehicle, or Task of the system. Options are as follows:</td>
</tr>
<tr>
<td></td>
<td>C - Print Courses</td>
</tr>
<tr>
<td></td>
<td>V - Print Vehicles</td>
</tr>
<tr>
<td></td>
<td>T - Print Tasks</td>
</tr>
<tr>
<td>P17</td>
<td>Initialize the files associated with a new ETAM project. This includes files named &quot;projectname&quot;/</td>
</tr>
<tr>
<td></td>
<td>ID</td>
</tr>
<tr>
<td></td>
<td>REC</td>
</tr>
<tr>
<td></td>
<td>REV</td>
</tr>
<tr>
<td></td>
<td>REJ</td>
</tr>
<tr>
<td>P2</td>
<td>Interactive edit of the Range-of-Effects (ROE) search arguments in files projectname/ REC, REV, and REJ</td>
</tr>
<tr>
<td>P3</td>
<td>Range-of-Effects (ROE) search module</td>
</tr>
<tr>
<td>P5A</td>
<td>Print of raw (un-edited) results of an ROE search</td>
</tr>
<tr>
<td>P5B</td>
<td>Interactive edit of the ROE search results</td>
</tr>
<tr>
<td>P5C</td>
<td>Print of the ROE extract results</td>
</tr>
<tr>
<td>P1</td>
<td>Print contents of any of the files associated with a given project.</td>
</tr>
</tbody>
</table>
SOURCE CARD LISTING FOR

PI:

PROC OPTIONS (MAIN); /* GENERAL-PURPOSE PROJECT FILE PRINT - THU/24/MAR/77 */

/* REVISIONS: */

24/MAR/THU/77 - VERY FIRST IMPLEMENTATION.

TUE/29/MAR/77 - MODS PRIOR TO TRANSMIT TO NCSS.

DECLARE ( DINO, COURSE, VEHS, TASKS )

FILE RECORD DIRECT KEYED ENV ( REGIONAL(1) ) ;

DECLARE DTYPE (21) CHAR (4) STATIC INIT /* FIRST 10 ARE */

'ID', 'EXD', 'BPT', 'RRP', /* STRICTLY CARDS */

'RPP', 'VQAL', 'VARF',

'SCEN', 'ALTP',

'REC', 'REV', 'REJ', 'RESU', 'REE', /* FROM HERE ON */

'EXC', 'EXJ', 'EXV', 'CMR', /* SPECIAL FORMATTING */

'FTR', 'FNR' ) ;

DECLARE TTLS (21) CHAR (40) STATIC INIT ( /*ID - PROJECT DESCRIPTION, */

'EXD - EXTRACT DEFAULTS',

'BPT - BENEFIT PATTERN',

'RRP - RISK PROFILE',

'RPP - RISK REDUCTION PROJECTS',

'PRP - RISK REDUCTION PACKAGES',

'VQAL - VARIABLES QUALITATIVE',

'VARF - VARIABLE REFERENCES',

'SCEN - SCENARIOS',

'ALTP - ALTERNATE PROJECTS',

'REC - COURSES ROE SEARCH ARGUMENTS',

'REV - VEHICLES ROE SEARCH ARGUMENTS',

'REJ - TASKS ROE SEARCH ARGUMENTS',

'RESU - ROE TOTAL SEARCH RESULTS',

'REE - ROE SEARCH RESULTS',

'EXC - EXTRACT DB - COURSES',

'EXJ - EXTRACT DB - JOBTASKS',

'EXV - EXTRACT DB - VEHICLES',

'CMR - COST MODEL RESULTS',

'TRER - TREE RESULTS',

'FINR - FINANCIAL RESULTS' ) ;

DECLARE I DINDEX STATIC; /* OVERALL DESCRIPTOR INDEX */

2 DSTART FIXED BIN; /* CKF.DESC.INDEX */

2 DEND FIXED BIN,

2 NCAT FIXED BIN,

2 NDESC FIXED BIN,

2 CATEGORIES (15),

3 PCAT FIXED BIN,

3 SCAT FIXED BIN,

3 ECAT FIXED BIN,

2 PODESC (111) FIXED BIN,

COVER (4) CHAR (80) BASED (PCOV) ;

DECLARE T2 CHAR (62) STATIC INIT (}
SOURCE CARD LISTING FOR
PI:
/* GENERAL-PURPOSE PROJECT FILE PRINT - THU/24/MAR/77 */

DECLARE DLINE8 CHAR (8) DEF DLINE POS (8),
DLINECST CHAR (36) DEF DLINE POS (34),
DLINE28 CHAR (4) DEF DLINE POS (28),
DLINE33 CHAR (36) DEF DLINE POS (33),
DLINE36 CHAR (36) DEF DLINE POS (36),
DLINE40 CHAR (40) DEF DLINE POS (40),
CARD7 CHAR (70) DEF CARD POS (8),
NODIR CHAR (35) STATIC INIT
( "** DIRECTORY SEARCH WAS NEGATIVE **" ),
NOAB CHAR (36) STATIC INIT
( "NO ABBREVIATED FILE DATA AVAILABLE" ) ;
DECLARE ( DISP, PTR, DPAGE, I, J, K, L, LT, NC, KALL, KDEV, KTYPE, N, NER, NFIRST, NLINE,
NV, NPAGE, RR, VTYPE, NV, NDF, NJ ) FIXED BIN STATIC,
DTTL CHAR (23) STATIC INIT
( "PAGE NUMBER : XXXX" ),
DDPGE PICTURE 'ZZ9' DEF DTTL POS (20),
MAXLTB FIXED BIN STATIC INIT ( 60 ) ;
DECLARE C1L80 CHAR (70) DEF CARD POS (11) ;
DECLARE 1 MAST STATIC,
2 REFS (40) FIXED BIN,
RESULT BASED (PR),
2 RTYP PICTURE '9',
2 ROSTAT PICTURE '9',
2 RESID CHAR (13),
1 RESC BASED (PR),
2 F1, 2 F2, 2 F3 \ CHAR (1),
2 RESCIN CHAR (8),
2 RESCDP CHAR (4),
1 REST BASED (PR),
2 F1, 2 F2, 2 F3 \ CHAR (1),
2 RATE CHAR (5),
2 RANK CHAR (2),
2 JOBTASK CHAR (6),
S16 CHAR (16) STATIC,
V1 (3) LABEL ;

DECLARE 1 ACRS STATIC, /* ABBR. COURSE FILE */
2 SETL \ /* CKF.ACOURSES */
3 COP CHAR (4),
3 CN CHAR (8),
3 CST CHAR (16),
3 VDBC CHAR (4),
3 VEC CHAR (4),
2 PC CHAR (2),
2 SET2,
DECLARE 1 CINDER (304) STATIC, /* CRS.CIN.DIR */
2 CHCIN CHAR (8), /* CKF.CRS.DIRCIN */
2 CINPTR FIXED BIN,
LOADCIN CHAR (760) BASED (PLCIN),
COVCIN (4) CHAR (760) BASED (PTCIN);

DECLARE 1 CDPDIR (1200) STATIC, /* CRS.CDP.DIR */
2 CHCDP CHAR (4), /* CKF.CRS.DIRCDP */
2 CDPTR FIXED BIN,
LOADCDP CHAR (1200) BASED (PLCDP),
COVCDP (6) CHAR (1200) BASED (PTCDP);

DECLARE 1 AVEHICLES STATIC, /* ABBR.VEH.FILE */
2 STOCKN CHAR (13), /* CKF.AVEHS */
2 DEVDESIG CHAR (9),
2 DEVNAME CHAR (47),
2 DEV_COST PICTURE '9999999999',
2 CC80 CHAR (1),
VCOST FIXED DEC (10,2) STATIC,
VOVER CHAR (80) BASED (PV80);

DECLARE 1 VEHDIR (200) STATIC, /* VEHICLE.DIR */
2 CHVEH CHAR (13), /* CKF.VEH.DIR */
2 VEHPTR FIXED BIN,
COVVEH (4) CHAR (750) BASED (PTVEH),
LOADVEH CHAR (750) BASED (PLVEH);

DECLARE 1 ATASKS STATIC, /* ABBR. TASK.FILE */
2 RATING CHAR (7), /* CKF.ATASKS */
2 JOBTASK CHAR (6),
2 TASKTTL CHAR (50),
2 FILL CHAR (6),
2 BILCOST PICTURE '9999999999',
2 CCGO CHAR (1);

DECLARE 1 TASKDIR (1200) STATIC, /* JOBTASK.DIR */
2 CHTASK CHAR (13), /* CKF.TASKS.DIR */
2 TASKPTR FIXED BIN,
COVTASK (12) CHAR (1500) BASED (PTTASK),
LOADTASK CHAR (1500) BASED (PLTASK);
START: NTYPE, NC, NJ, NV, NR, NN, NDF,
    NPAGE, DPAGE, LT, NFIRST,
    N = 0;
    MAXLTB = 60; /* MAX NUMBER LINES PER TERMINAL PAGE */
    LT = 100; /* TERMINAL LINE COUNTER */
    DISP = 1; /* START OFF WITH DISPLAY ACTIVE */
    PTR = 2; /* START WITH PRINTER INACTIVE */

FLCIN, FLCDP, FLVEH, FLTASK = '0'B;

PCOV = ADDR (DINDEX);
PR = ADDR (S16);
PTCIN = ADDR (CINDER);
PTCDP = ADDR (CDPDIR);
PV80 = ADDR (AVEHICLES);
PTVEH = ADDR (VEHDIR);
PTTASK = ADDR (TASKDIR);

COMMENCE:
OPEN FILE (MREF) RECORD INPUT,
FILE (ID) RECORD INPUT,
FILE (COURSE) INPUT,
FILE (DCIN) RECORD INPUT,
FILE (DCDP) RECORD INPUT,
FILE (VEHS) RECORD INPUT,
FILE (DVEH) RECORD INPUT,
FILE (TASKS) INPUT,
FILE (DTASK) RECORD INPUT,
FILE (REE) RECORD INPUT;
ON ENDFILE (REE) GO TO EOJ;
READ FILE (MREF) INTO (MAST);
READ FILE (ID) INTO (CARD);
T2NAME = C110;
DISPLAY ( ' ' ) ;
DISPLAY ( 'GENERAL-PURPOSE PROJECT FILE PRINT' ) ;
DISPLAY ( 'PROGRAM IS NOW STARTING...' ) ;
DISPLAY ( ' ' ) ;

/*****************************/
/* THIS IS THE CENTRAL PROMPTING AND DECODE SECTION */

PROMPTA: CARD = ' ' ;
PROMB: DLINE = '*** ENTER FILETYPE FOR OUTPUT, MENU, OR QUIT **' ;

DISPLAY ( DLINE ) REPLY ( CARD ) ;
J = INDEX ( CARD, ' ' ) ;
IF ( J = 0 ) THEN GO TO BUMKEY ;
NAME4 = SUBSTR ( CARD, 1, J ) ;
KDEV = 0 ;
REROUTE = PROMPTA ;
LPF: DD J = 1 TO 21 ;
KTYPE = J ;
IF ( NAME4 = DTYPE(J) ) THEN GO TO FINDF ;
END LPF ;
/* DISPLAY MENU OUTPUT HERE - TO TERMINAL ONLY */
IF ( NAME4 = 'MENU' ) THEN GO TO TRYQUIT ;
DLINE = 'PROJECT FILETYPES FOR OUTPUT ARE : ' ;
SOURCE CARD LISTING FOR

GENERAL-PURPOSE PROJECT FILE PRINT - THU/24/MAR/77

DISPLAY (','); DISPLAY (DLINE);
DISPLAY ('');

MOUT:
DJ J = 1 TO 21;
DISPLAY (5) I I TTS(J);
END MOUT:
DISPLAY ('');
GO TO PROMT A;

TRYQUIT: IF (NAME4 = 'QUIT') THEN GO TO FEND;
IF (NAME4 = 'ALL') THEN GO TO FINDALL;

BUMKEY: DLINE = '*** LAST ENTRY UNRECOGNIZED - TRY AGAIN **';
DISPLAY ('');
DISPLAY (DLINE);
GO TO PROMT A;

/* HERE WHEN FOUND A VALID FILE TYPE; START PROCESSING */

FINDF: IF (KTYPE > 15) THEN

BUMFILE:
DJ:
DLINE = '*** FILETYPE ' I I NAME4 I I ' IS NOT YET '
| 'SERVICABLE, TRY AGAIN **';
IF (PTR = 1) THEN
PUT LIST (DLINE) SKIP (2);
DISPLAY ('');
DISPLAY (DLINE);
GO TO REROJTE;
END BUMFILE;

/* NOW, DETERMINE THE VALID OUTPUT DEVICE(S) */

IF (KDEV > 1) THEN GO TO PROCESS;
KDEV = KDEV + 1;
NPAGE = 0;
DISP, PTR = 1;
DISPLAY ('');
DISPLAY ('OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, '
| 'BOTH, OR QUIT' );

TP2:
DISPLAY ('ENTER TERM, PRTR, BOTH, OR QUIT' ) REPLY (DLINE);
DISPLAY ('');
IF (DL4 = 'TERM') THEN GO TO STERM;
IF (DL4 = 'PRTR') THEN GO TO OPRTR;
IF (DL4 = 'BOTH') THEN GO TO SPRTR;
IF (DL4 = 'QUIT') THEN GO TO FEND;
DISPLAY ('');
DISPLAY ('UNABLE TO DETERMINE ENTRY TYPE - TRY AGAIN' );
DISPLAY ('');
GO TO TP2;

OPRTR: DISP = 2;

SPRTR: OPEN FILE (SYSPRINT) LINESIZE (120) PAGESIZE (60);

ON ENDPAGE (SYSPRINT)
BEGIN;
NPAGE = NPAGE + 1;
PJT EDIT ('PAGE NUMBER : ', NPAGE)
(PAGE, LINE(2), X(9), A, F(4));
TAEG REPORT NO. 40

SOURCE CARD LISTING FOR
P1: /* GENERAL-PURPOSE PROJECT FILE PRINT - THU/24/MAR/77 */

345 IF ( KTYPE > 15 ) THEN GO TO SKP;
346 PUT EDIT( 'LISTING OF PROJECT FILE CONTENTS',
347 T2, ' ' ) ;
348 IF ( KTYPE <= 17 ) THEN GO TO SKP;
349 IF ( KTYPE > 10 ) & ( KTYPE < 14 ) THEN GO TO SKP;
350 PJT SKIP (1) ;
351 IF ( NTYPE = 3 ) THEN GO TO TASKTTL ;
352 IF ( NTYPE = 2 ) THEN GO TO VEHTTL ;
353 CRSTTL: PJT EDIT ( *** THE FOLLOWING IS EXTRACT DATA FOR COURSES ***,'
354 "RECORD",
355 "NUM ORIGIN CIN CPD COURSE TITLE"
356 ( X(9), A, SKIP(2), X(9), A, 3 ( SKIP(1), A ) ) ;
357 VEHTTL: PJT EDIT ( *** THE FOLLOWING IS EXTRACT DATA FOR VEHICLES ***','
358 "RECORD",
359 "NUM ORIGIN STOCK NUMBER DESIGNATOR"
360 ( X(9), A, SKIP(2), X(9), A, 3 ( SKIP(1), A ) ) ;
361 TASKTTL: PJT EDIT ( *** THE FOLLOWING IS EXTRACT DATA FOR TASKS ***','
362 "RECORD",
363 "NUM ORIGIN RATING RANK JOB TASK"
364 ( X(9), A, SKIP(2), X(9), A, 3 ( SKIP(1), A ) ) ;
365 SKP: PUT SKIP (1) ;
366 END;
367 GO TO STARTP ;
368 STERM: PTR * 2 ;
369 STARTP: IF ( DISP = 2 ) THEN GO TO PROCESS ;
370 ON CONDITION (NEWDP)
371 BEGIN ;
372 IF ( LT >= 66 ) THEN GO TO LPB ;
373 IF ( LT < 61 ) THEN LT = 61 ;
374 LPA:
375 DISPLAY ( ' ' ) ;
376 END LPA ;
377 LPB:
378 DISPLAY ( ' ' ) ;
379 DPAGE = DPAGE + 1 ;
380 DDPGE = DPAGE ;
381 DISPLAY ( ' ' ) ;
382 LT = 3 ;
SOURCE CARD LISTING FOR
P1: /* GENERAL-PURPOSE PROJECT FILE PRINT - THU/24/MAR/77 */

IF ( KTYPE > 15 ) THEN GO TO NEWEND;
DISPLAY ( 'LISTING OF PROJECT FILE CONTENTS' );
DISPLAY ( T2 ); DISPLAY ( ' ' );
LT = LT + 3;
IF ( KTYPE <= 10 ) THEN GO TO NEWEND;
IF ( KTYPE > 10 ) & ( KTYPE < 14 ) THEN GO TO NEWEND;

IF ( NTYPE = 3 ) THEN DO:
DISPLAY ( 'THE FOLLOWING IS ROE RESULT DATA FOR TASKS' );
DISPLAY ( 'NUM ORIGIN RATING RANK JOB TASK' );
GO TO DSKIP;
END;

IF ( NTYPE = 2 ) THEN DO:
DISPLAY ( 'THE FOLLOWING IS ROE RESULT DATA FOR VEHICLES' );
DISPLAY ( 'NUM ORIGIN STOCK NUMBER DESIGNATOR' );
GO TO DSKIP;
END;

DISPLAY ( 'THE FOLLOWING IS ROE RESULT DATA FOR COURSES' );
DISPLAY ( 'NUM ORIGIN CIN CDP COURSE TITLE' );
DISPLAY ( ' ' );

DSKIP:
DISPLAY ( ' ' );
LT = LT + 4;
NEWEND:
END;

/**************************************************************/
/* NOW, NEW FILE IS SETUP */

PROCESS:
NFIRST = 0;
NR = 0;
FLAGS ( KTYPE ) = '1'B;
T2TTL = TTL5 ( KTYPE );
OPEN FILE ( PROJ ) RECORD INPUT TITLE ( DTYPE(KTYPE) );
ON ENDFILE ( PROJ ) GO TO TALLY;

IF ( PTR = 1 ) THEN SIGNAL ENDPAGE ( SYSPRINT );
IF ( DISP = 2 ) THEN GO TO DOIT;
SIGNAL CONDITION ( NEWDP );

DOIT:
IF ( KTYPE > 10 ) THEN GO TO WAY ( KTYPE );
/**************************************************************/
/* IF CARD-IMAGE PRINTING ALONE REQUIRED, IT IS DONE HERE */
RRR1:
READ FILE ( PROJ ) INTO ( CARD );
NR = NR + 1;
IF ( PTR = 1 ) THEN PUT LIST ( CARD ) SKIP ( 1 );

IF ( DISP = 2 ) THEN GO TO RRR1;
LT = LT + 1;
IF ( LT > MAXLTB ) THEN SIGNAL CONDITION ( NEWDP );
DISPLAY ( CARD );
GO TO RRR1;
/**************************************************************/
/* IN THIS SECTION, FILETYPES REC, REV, REJ ARE PROCESSED: */

WAY(11): DO J = 1 TO 4;
           READ FILE (DIND) INTO (CARD) KEY (J);
           COVER (J) = CARD;
           END;
           GO TO RECVJ;

/* NOW, SETUP FILE (REV) PROCESSING */

WAY(12): DO J = 5 TO 8;
           READ FILE (DIND) INTO (CARD) KEY (J);
           COVER (J - 4) = CARD;
           END;
           GO TO RECVJ;

/* FINALLY, SETUP OF FILE (REJ) ENSUETH */

WAY(13): DO J = 9 TO 12;
           READ FILE (DIND) INTO (CARD) KEY (J);
           COVER (J - 8) = CARD;
           END;
           /************/*
           /* HERE, ONE OF THE FILES (REC), (REV), (REJ) IS PRINTED */

RECVJ: READ FILE (PROJ) INTO (CARD);
         NR = NR + 1;
         IF ( C110 = ' ' ) THEN GO TO RECVJ;
         GET STRING (C110) EDIT (N) ( F(10) );
         J = ABS (N);
         IF ( PDESC(J) = 0 ) THEN GO TO FIXSR;
         READ FILE (DIND) INTO (CARD) KEY ( PDESC(J) );
         PUT STRING (DLINE) EDIT ( N, SUBSTR(CARD, 3, 70) )
               ( X(5), F(4), X(1), A ) ;
         IF ( PTR = 1 ) THEN PUT LIST (DLINE) SKIP (2);
         IF ( DISP = 2 ) THEN GO TO RECVJ;
         IF ( LT > MAXLTB ) THEN SIGNAL CONDITION (NEWDP);
         DISPLAY (' ** DETECTED IN ' || NAME4 || ' FILE ** ');
         LT = LT + 2;
         GO TO RECVJ;

FIXSR: DLINE = ' ** ILLEGAL DESCRIPTOR - DETECTED IN ' || NAME4 || ' FILE ** ';
         DISPLAY ( ' ' );
         DISPLAY ( ' (10)' );
         DISPLAY ( ' ' );
         DISPLAY ( ' ' );
         LT = LT + 3;
         PUT LIST ( DLINE ) SKIP (2);
         GO TO RECVJ;

/*********************/

/* HERE, EOF SENSED WHILE PROCESSING INPUT FILE */
TALLY: CLOSE FILE (PROJ) ;

PUT STRING (DLINE) EDIT ("A TOTAL OF", NR, "RECORDS ARE PRESENT") (X(10), A, F(4), A) ;
CARD = "(10)" | | "WITIIN THIS FILE" ;

IF ( PTR = 1 ) THEN PUT EDIT 
( DLINE, CARD ) ( SKIP(3), A, SKIP(1), A ) ;
IF ( DISP = 2 ) THEN 
DO ;
DISPLAY ("NOW LISTING FILETYPE = " || DTYPE (KTYPE) ) ;
GO TO TALLY2 ;

DISPLAY ("" ) ;
TALLY2: DISPLAY ( DLINE ) ; DISPLAY ("" ) ;
GO TO REROUTE ;

/*****************************/

/* HERE, WHEN PROCESSING ALL PROJECT FILES, THE ORDER OF 
PROCESSING WILL BE THAT OF VECTOR (DTYPE(21)) */

FINDALL: REROUTE = TALLYALL ;
KALL, KTYPE, KDEV = 1 ;
NEXALL: NAME4 = DTYPE (KTYPE) ;
GO TO FINDF ;
TALLYALL: IF ( KTYPE >= 21 ) THEN GO TO PREDITA ;
KTYPE = KTYPE + 1 ;
KALL = KALL + 1 ;
GO TO NEXALL ;

/*****************************/

/* HEREIN, IS THE PROCESSING OF FILETYPE = REE / RESU */

WAY(14): WAY(15): 
RD1: READ FILE (PROJ) INTO (S16) ;
NR = NR + 1 ;
IF ( RTYP = NTYPE ) THEN GO TO STEP1 ;
NTYPE = RTYP ;
IF ( PTR = 1 ) & ( NFIRST > 0 )
THEN SIGNAL ENDPAGE (SYSPRINT) ;
IF ( DISP = 1 ) & ( NFIRST > 0 )
THEN SIGNAL CONDITION (NEWDP) ;
NFIRST = NFIRST + 1 ;

STEP1: GO TO VI (NTYPE) ;

/* HERE, A COURSE RECORD IS PROCESSED */

VI(1): NC = NC + 1 ;
NLINE = 1 ;
DLINE = "" ;
PUT STRING (DLINE) EDIT (NR, RESCIN, RESCDP)
( P'Z', Z9, X(12), A, X(2), A ) ;
SOURCE CARD LISTING FOR

/* GENERAL-PURPOSE PROJECT FILE PRINT - THU/24/MAR/77 */

STP1: IF ( RSTAT = 1 ) THEN DLINE8 = 'ORIGINAL';
     ELSE IF ( RSTAT = 2 ) THEN DLINE8 = 'ADDED';
     ELSE DLINE8 = 'DELETED';

     IF ( RESCDP = ' ' ) THEN DLINE28 = 'NONE';
     IF ( RDSTAT = 2 ) THEN NOCDATA:     DO ;
         NN = NN + 1 ;
         DLINECST = NOAB ;
         GO TO OUTLOOP ;
         END NOCDATA ;

     YECDP = RESCDP ;
     LOOKU = STEP2 ;
     GO TO FINDCDP ;

STEP2: IF ( CDPREC = 0 ) THEN
     DO ;
         NDF = NDF + 1 ;
         DLINE36 = NODIR ;
         GO TO OUTLOOP ;
     END LP1 ;
     READ FILE (COURSE) INTO (ACRS) KEY (CDPREC) ;
     DLINECST = CST ;
     GO TO OUTLOOP ;

     /* HERE, VEHICLE TYPE RECORDS ARE PROCESSED */

V1(2):  NV = NV + 1 ;
     NLINE = 1 ;

     PUT STRING (DLINE) EDIT ( NR, RESID )
         ( P'Z.99', X(12), A(13) ) ;

STEP2: IF ( RSTAT = 1 ) THEN DLINE8 = 'ORIGINAL';
     ELSE IF ( RSTAT = 2 ) THEN DLINE8 = 'ADDED';
     ELSE DLINE8 = 'DELETED';

     IF ( RDSTAT = 2 ) THEN
         NOVDATA:     DO ;
             NN = NN + 1 ;
             DLINE36 = NOAB ;
             GO TO OUTLOOP ;
             END NOVDATA ;

     YEVEH = RESID ;
     LOOKU = STEP3 ;
     GO TO FINDVEH ;

STEP3: IF ( VEHREC = 0 ) THEN
     DO ;
         NDF = NDF + 1 ;
         DLINE36 = NODIR ;
         GO TO OUTLOOP ;
     END LP2 ;
     READ FILE (VEHS) INTO (AVEHICLES) KEY (VEHREC) ;
     DLINE33 = DEVDESIG ;
     CARD = ' ' ;
     CARD7 = DEVNAME ;
     NLINE = 2 ;
     GO TO OUTLOOP ;
/* Now, process a task record here */

VI(3):  NJ * NJ

DLINE » 1 ;

PUT STRING (DLINE) EDIT ( NR, RATE, RANK, REST*JOBTASK )

( PIZ, 229, X(12), A, X(4), A, X(3), A ) ;

IF ( RSTAT = 1 ) THEN DLINE8 = 'ORIGINAL' ;

ELSE IF ( RSTAT = 2 ) THEN DLINE8 = 'ADDED' ;

ELSE DLINE8 = 'DELETED' ;

STEP5: IF ( ROSTAT = 2 ) THEN

NOTDATA: DO ;

NN = NN + 1 ;

DLINE40 = NOAB ;

GO TO OUTLOOP ;

END NOTDATA ;

YETASK = RESID ;

LOOKU = STEP4 ;

GO TO FINDTASK ;

STEP4: IF ( TASKREC = 0 ) THEN

LP3: DO ;

NDF = NDF + 1 ;

DLINE40 = NODIR ;

GO TO OUTLOOP ;

END LP3 ;

READ FILE (TASKS) INTO (ATASKS) KEY (TASKREC) ;

/* Here is the printed live output sequence */

OUTLOOP: IF ( PTR = 1 ) THEN PUT LIST (DLINE) SKIP (2) ;

IF ( PTR = 1 ) & ( NLINE = 2 ) THEN PUT LIST (CARD) SKIP (1) ;

IF ( DISP = 2 ) THEN GO TO RD1 ;

IF ( LT > MAXLTB ) THEN SIGNAL CONDITION (NEWDP) ;

DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;

LT = LT + 2 ;

IF ( NLINE = 2 ) THEN GO TO RD1 ;

DISPLAY ( CARD ) ;

LT = LT + 1 ;

GO TO RD1 ;

*******************************************************************************/

DECLARE ( FLCIN, FLCDP, FLVEH, FLTASK ) BIT (1) STATIC,

YECIN CHAR (8) STATIC,

YECOP CHAR (4) STATIC,

YETASK, YEVEH CHAR (13) STATIC,

CINREC, CDPREC, VEHREC, TASKREC, VEHMAX, TASKMAX,

CINMAX, CDPMAX ) FIXED BIN STATIC,
/* GENERAL-PURPOSE PROJECT FILE PRINT - THU/24/MAR/77 */

LOOKU

/* CIN DIRECTORY LOOKUP SUBROUTINE
ENTRY: DESIRED CIN IN YECIN
EXIT: CINREC = 0 (NO FIND) OR REG NUMBER
RETURN VECTOR IS LOOKU */

FINDCIN: IF ( FLCIN ) THEN GO TO CIN2 ;

CINMAX = REFS (2) ;
FLCIN = '1'B ;

CINLOAD:
DO J = 1 TO 4 ;
PLCIN = ADDR ( COVCIN (J) ) ;
READ FILE (DCIN) INTO (LOADCIN) ;
END CINLOAD ;

CIN2:
DO I = 1 TO CINMAX ;
IF ( YECIN = CHCIN(I) ) THEN GO TO ECIN2 ;
CINREC = CINPTR (I) ;
GO TO LOOKU ;

ECIN2:
END CIN2 ;

NFCIN: CINREC = 0 ;
GO TO LOOKU ;

/* CDP DIRECTORY LOOKUP SUBROUTINE
ENTRY: DESIRED CDP IN YECOP
EXIT: CDPREC = 0 (NO FIND), OR REC NUMBER
RETURN VECTOR IS LOOKU */

FINDCDP: IF ( FLCOP ) THEN GO TO CDP2 ;

CDPMAX = REFS (3) ;
FLCDP = '1'B ;

CDPLOAD:
DO J = 1 TO 4 ;
PLCDP = ADDR ( COVCOP (J) ) ;
READ FILE (DCDP) INTO (LOADCDP) ;
END CDPLOAD ;

CDP2:
DO I = 1 TO CDPMAX ;
IF ( YECDP = CHCDP(I) ) THEN GO TO ECDP2 ;
CDPREC = CDPTR (I) ;
GO TO LOOKU ;
END CDP2 ;

NFCDP: CDPREC = 0 ;
GO TO LOOKU ;

/* VEHICLE ID LOOKUP SUBROUTINE
ENTRY: DESIRED VEH ID IN YEVEH
EXIT: VEHREC = 0 (NO FIND), OR REC NUMBER
RETURN VECTOR IS LOOKU */

FINDVEH: IF ( FLVEH ) THEN GO TO VEH2 ;

VEHMAX = REFS (8) ;
FLVEH = '1'B ;

VEHLOAD:
DO J = 1 TO 4 ;
PLVEH = ADDR ( COVVEH (J) ) ;
READ FILE (DVEH) INTO (LOADVEH) ;
END VEHLOAD ;

VEH2:
DO I = 1 TO VEHMAX ;
IF ( YEVEH = CHVEH(I) ) THEN GO TO EVEH2 ;
VEHREC = VEMPTR (I) ;
SOURCE CARD LISTING FOR
PI:  /* GENERAL-PURPOSE PROJECT FILE PRINT - THU/24/MAR/77 */

GO TO LOOKU ;

EVEH2:  END VEH2 ;

NFVEH:  VEHREC = 0 ;
         GO TO LOOKU ;

/* TASK ID DIRECTORY LOOKUP SUBROUTINE
ENTRY: DESIRED TASK ID (13-CHAR) IN YETASK
EXIT: TASKREC = 0 (NO FIND) OR REC NUMBER
RETURN VECTOR IS LOOKU */

FINDTASK: IF ( FLTASK ) THEN GO TO TASK2 ;

       TASKMAX = REFS (12) ;
       FLTASK = '1'B ;

       TASKLOAD:  DO J = 1 TO 12 ;
                  PLTASK = ADDR ( COVTASK (J)) ;
                  READ FILE (DTASK) INTO (LOADTASK) ;
                  ENO TASKLOAD ;

       TASK2:  DO I = 1 TO TASKMAX ;
                IF ( YETASK = CHTASK(I) ) THEN GO TO ETASK2 ;
                TASKREC = TASKPTR (I) ;
                GO TO LOOKU ;

       ETASK2:  ENO TASK2 ;

       NFTASK:  TASKREC = 0 ;
                GO TO LOOKU ;

/*********************************************************************************/

/* IF GET HERE, PROGRAM IS TERRIBLE BROKE */

BUSTED:  PUT STRING (CARD) EDIT
       ( ' ** ERROR NUMBER ', NER, ' HAS OCCURRED ** ' )
       ( A, F(2), A ) ;
       PUT LIST (CARD) SKIP (3) ;
       PUT LIST (DLINE) SKIP (2) ;
       DISPLAY ( ' ' ) ;
       DISPLAY (DLINE) ;
       DISPLAY ( ' ' ) ;

/*********************************************************************************/

/* HERE IS EOF WINDUP PROCESSING */

EOJ:  EOF:

FEND:  CLOSE FILE (MREF), FILE (ID), FILE (REE),
       FILE (COURSE), FILE (DCOP),
       FILE (VEHS), FILE (DVEH),
       FILE (TASKS), FILE (DTASK) ;

       IF ( PTR = 1 ) THEN CLOSE FILE (SYSPRINT) ;

       DISPLAY ( ' ' ) ;
       DISPLAY ( ' ' ) ;
       DISPLAY ( 'PROJECT-FILE PRINT PROGRAM IS NOW TERMINATING' ) ;
       DISPLAY ( ' ' ) ;

       END PI ;
PROC (INPARM) OPTIONS (MAIN);

/* REVISIONS: TUE/08/FEB/77 - FIRST DEBUG VERSION AT CKF.
  WED/23/FEB/77 - INTRODUCED REC, REV, REJ PROJECT FILES.
  WED/02/MAR/77 - ENTERED CHECKS TO VALIDATE THE INCOMING
  DESCRIPTORS FROM PROJECT FILES REC, REV, AND REJ; EDIT PROJECT FILES IF INVALID.
  MON/21/MAR/77 - MODS PRIOR TO TRANSMIT TO NCSS. */

DECLARE DIND FILE RECORD DIRECT
  KEYED ENV ( REGIONAL(1) ) ;

DECLARE ( NC, EOF, KEOF, NRD, J, K, L, FCON, CFLAG, ND,
  JE, KMAX, NER, KD, NTYPE, NUM, L1, LIA, N,
  JD ) FIXED BIN STATIC,

FNME CHAR (3) STATIC,
ROUTE LABEL,
( RD(3), SRET ) LABEL ;

DECLARE ( CARD, DLINE ) CHAR (80) STATIC,
  C14 CHAR (4) DEF CARD POS (1),
  C16 CHAR (6) DEF CARD POS (1),
  C10 CHAR (10) DEF CARD,
  C580 CHAR (75) DEF CARD POS (5),
  C780 CHAR (74) DEF CARD POS (7),
  C1080 CHAR (70) DEF CARD POS (11),

INPARM CHAR (100) VARYING,
  1 POVER BASED (PP),
  2 PARM CHAR (1),
  2 PARM2 CHAR (4),

RNUMS (3,3) FIXED BIN /* MASTER INDEX REC NOS */
  STATIC INIT ( I, 4, G, 0, /* COURSES */
  5, 8, 4, /* VEHICLES */
  9, 12, 8 ), /* TASKS */

TYPE3 (3) CHAR (8) STATIC INIT
  ( *COURSES*, *VEHICLES*, *TASKS* ),

FNAMES (3) CHAR (8) STATIC INIT
  ( 'REC', 'REV', 'REJ' ),

A80 (80) CHAR (1) BASED (PA80),
  I CARG, VARG, TARG ) CHAR (240) STATIC,

PARGS (3) POINTER,

WFLAGS (3) FIXED BIN STATIC,

TTL4 CHAR (10) STATIC ;

*******************************************************************************/

DECLARE 1 DINDEX STATIC; /* OVERALL DESCRIPTOR INDEX */

D.6-16
TAEG REPORT NO. 40

PAGE NUMBER : 2

SOURCE CARD LISTING FOR
P2: /* SEARCH DESIGNATORS IA EDITOR - TUE/08/FEB/77 */

SOURCE CARD LISTING FOR
P2: /* SEARCH DESIGNATORS IA EDITOR - TUE/08/FEB/77 */

DECLARE 1 MAST STATIC,
2 REFS (40) FIXED BIN ;

DECLARE 1 TTLCARD STATIC, /* PROJECT TITLE DATA BASE */
2 PNAME CHAR (8), /* CKF.TITLE */
2 PTTL CHAR (72),
1 SARGS BASED (PSARG), /* ROE SEARCH ARGUMENTS */
2 SID CHAR (8), /* CKF.SEARCH.ARGS */
2 SSTAT FIXED BIN,
2 STAB (115) FIXED BIN ;

******************************************************************************/

START: NC, MEGF, KEOF, NRD, NTYPE,
J, K = 0 ;
WFLAGS = 0 ;
SRET = STEP1 ; /* SET FOR INITIAL SELECT LOOP */

PARGS(1) = ADDR (CARG) ;
PARGS(2) = ADDR (VARG) ;
PP = ADDR (INPARM) ;
PCOV = ADDR (DINDEX) ;
PA80 = ADDR (CARD) ;

COMMENCE: OPEN FILE (DIND) INPUT,
FILE (MREF) RECORD INPUT,
FILE (ID) RECORD INPUT,
FILE (SARG) RECORD OUTPUT ;

READ FILE (MREF) INTO (MAST) ;
KMAX = REFS (4) ; /* MAX SS USED IN MASTER INDEX */

FINDT: READ FILE (ID) INTO (CARD) ;
TTL4 * C110 ;

DISPLAY ( • )
DISPLAY ( (10) • | • SEARCH DESCRIPTOR EDITOR RUN
| | FOR PROJECT : • | | TTL4 ) ;

DISPLAY ( C1080 ) ;
DISPLAY ( • ) ;

******************************************************************************/

/* PROJECT FILES REC, REV, REJ TRANSFERRED TO THE SEARCH
ARGUMENT FILE IN THIS SECTION */

OPEN FILE (REC) RECORD UPDATE,
FILE (REV) RECORD UPDATE,
FILE (REJ) RECORD UPDATE ;

ON ENDFILE (REC) GO TO L2 ;
ON ENDFILE (REV) GO TO L3 ;
ON ENDFILE (REJ) GO TO L4 ;

ON CONVERSION GO TO FIXSR ;

/* HERE, COJRSSES PROCESSED FROM FILE = REC */

PSARG = ADDR (CARG) ;
SID = 'COURSES' ;
SSTAT = 0 ; STAB = 0 ;
ROUTE = RW1 ;
FNME = 'REC' ;
DO J = 1 TO 4;
READ FILE (DIND) INTO (CARD) KEY (J);
COVER (J) = CARD;
END LOOP1;

READ FILE (REC) INTO (CARD);
IF (C110 = ' ') THEN GO TO R1;
GET STRING (C110) EDIT (N) (F(10));
J = ABS (N);
IF (PDESC(J) = 0) THEN GO TO FIXSR;
SSTAT = SSTAT + 1;
STAB (ABS(N)) = N;
GO TO R1;

REWRITE FILE (REC) FROM (CARD);
GO TO R1;

/* HERE, VEHICLES PROCESSED FROM REV FILE */

PSARG = ADDR (VARG);
SID = 'VEHICLES';
SSTAT = 0; STAB = 0;
ROUTE = RW2;
FNME = 'REV';

DO J = 5 TO 8;
READ FILE (DIND) INTO (CARD) KEY (J);
COVER (J - 4) = CARD;
END LOOP2;

READ FILE (REV) INTO (CARD);
IF (C110 = ' ') THEN GO TO R2;
GET STRING (C110) EDIT (N) (F(10));
J = ABS (N);
IF (PDESC(J) = 0) THEN GO TO FIXSR;
SSTAT = SSTAT + 1;
STAB (ABS(N)) = N;
GO TO R2;

REWRITE FILE (REV) FROM (CARD);
GO TO R2;

/* FINALLY, TASKS ARE PROCESSED FROM FILE = REJ */

PSARG = ADDR (TARG);
SID = 'TASKS';
SSTAT = 0; STAB = 0;
ROUTE = RW3;
FNME = 'REJ';

DO J = 9 TO 12;
READ FILE (DIND) INTO (CARD) KEY (J);
COVER (J - 8) = CARD;
END LOOP3;

READ FILE (REJ) INTO (CARD);
IF (C110 = ' ') THEN GO TO R3;
GET STRING (C110) EDIT (N) (F(10));
J = ABS (N);
IF (PDESC(J) = 0) THEN GO TO FIXSR;
SSTAT = SSTAT + 1;
STAB (ABS(N)) = N;
GO TO R3;

REWRITE FILE (REJ) FROM (CARD);
SOURCE CARD LISTING FOR
P2: /* SEARCH DESIGNATORS IA EDITOR — TUE/08/FEB/77 */

GO TO R3 ;

*/* WITHIN THIS SUBROUTINE, AN INVALID DESIGNATOR (CONVERSION
OR RANGE) IS ELIMINATED FROM REC, REJ, REV */*

FIXSR: DLINE = ' ** ILLEGAL DESCRIPTOR -' || C110 ||
         * - DETECTED IN ' || FNME || ' FILE **';
         DISPLAY ( DLINE ) ;
         DLINE = ' DESCRIPTOR WILL BE IGNORED AND REMOVED FROM THE
         || FNME || ' FILE';
         DISPLAY ( DLINE ) ;
         DISPLAY ( DLINE ) ;
         CARD = ' ' ;
         GO TO ROUTE ;

*/* HERE, FINAL SETUP, PRIOR TO ENTERING MAINLINE */*

L4:   CLOSE FILE (REC), FILE (REV), FILE (REJ) ;

ON CONVERSION
BEGIN ;

DISPLAY ( ' ' ) ;
DISPLAY ( ' ** KEYBOARD ENTRY -' || SUBSTR (CARD, K, 3) ||
' - CONTAINED ILLEGAL NUMERIC CHARACTER' ) ;

GO TO PROMPTA ;

EVD ;

épo ui, Initial Loop, awaiting Type Designator Select */

STEP1: DISPLAY ( ' ** ' ) ;
       DISPLAY ( ' ** MAKE INITIAL ENTRY SELECT ' ) REPLY (CARD) ;
       NRD = 1 ;
       IF ( C16 = ' SELECT' ) THEN GO TO STEP1 ;

LP1:   DO J = 1 TO 3 ;
       NTYPE = J ;
       IF ( INDEX ( C780, NTYPE3(J) ) > 0 ) THEN GO TO STEP2 ;
       END LP1 ;

DISPLAY ( ' ' ) ;
DISPLAY ( ' ** SELECTION ERROR — TRY AGAIN ' ) ;
GO TO SRET ;

*/* HERE, A SELECTION HAS BEEN MADE; INDEX IS IN NTYPE */

STEP2: PSARG = PARGS ( NTYPE ) ;

LP2:   DO J = RNUNS ( NTYPE, 1 ) TO RNUNS ( NTYPE, 2 ) ;
       READ FILE (DIND) INTO (CARD) KEY ( J ) ;
       CJVER ( J - RNUNS (NTYPE, 3) ) = CARD ;
       END LP2 ;

DISPLAY ( ' ' ) ;
DISPLAY ( ' ** ' || SID ||
' SEARCH DESCRIPTORS SELECTED ' ) ;
 IF ( SSTAT = 0 ) THEN
NONEMSG: DO J ;
DISPLAY ( ' ' ) ;
DISPLAY ( ' ** NO DESCRIPTORS NOW ACTIVE ' ) ;
  GO TO PROMPTA ;
SOURCE CARD LISTING FOR
P2: /* SEARCH DESIGNATORS IA EDITOR - TUE/08/FEB/77 */

END NONEMSG;

DISPLAY ( ' ' ) ;
DISPLAY ( '*** | SSTAT | ' ) ;
DISPLAY ( ' DESCRIPTORS NOW SELECTED' ) ;

CENTRAL ACTION PROMT AND REPLY DECODE FOLLOWS */

PRIMPTA: DISPLAY ( ' ' ) ;
DISPLAY ( '*** MAKE NEXT ACTION REQUEST' ) ;
REPLY ( CARD ) ;

NRD = 2 ;
IF ( C14 = 'ADD' ) THEN GO TO ADD ;
IF ( C16 = 'DELETE' ) THEN GO TO DELETE ;
IF ( C16 = 'SELECT' ) THEN GO TO SELECT ;
IF ( C14 = 'FILE' ) THEN GO TO FILE ;
IF ( C14 = 'QUIT' ) THEN GO TO QUIT ;

DISPLAY ( ' ' ) ;
DISPLAY ( '*** UNIDENTIFIED ACTION REQUEST - TRY AGAIN' ) ;
GO TO PRIMPTA ;

HERE IS PROCESSING FOR DELETE OPTION
ENTRIES ARE - DELETE (BLANK) - MEANS GET RID OF ALL
DESIGNATORS OF SELECTED TYPE.
- DELETE N - MEANS DELETE ONLY DESIGNATOR NUMBERED (N).

DELETE: IF ( C780 = ' ' ) THEN
DELALL: /* BLANK DELETE SERVICED HERE */
WFLAGS(NTYPE) = 1 ;
SSTAT = 0 ; STAB = 0 ;
DISPLAY ( ' ' ) ;
DISPLAY ( '*** ALL | SID | ' ) ;
DISPLAY ( ' SEARCH DESIGNATORS DELETED' ) ;
GO TO PRIMPTA ;
END DELALL ;

IF ( STAB(N) =* 0 ) THEN GO TO STEP12 ;

STEP3: FCON = 1 ;
GET STRING ( SUBSTR(CARD,K,4) ) EDIT ( NUM ) ( F(4) ) ;
N = ABS ( NUM ) ;
IF ( N > 100 ) | ( N = 0 ) | ( PDESC(N) = 0 ) THEN
BUMD: DO ;
DISPLAY ( ' ' ) ;
DISPLAY ( '*** | NUM | ' ) ;
DISPLAY ( ' IS NOT A LEGAL DESCRIP'TOR FOR | SID | ' ) ;
DISPLAY ( ' TYPE' ) ;
GO TO PRIMPTA ;
END BUMD ;
IF ( STAB(N) =* 0 ) THEN GO TO STEP12 ;
PUT STRING (DLINE) EDIT ( ' ** DESIGNATOR NUMBER ' ) ;
SOURCE CARD LISTING FOR
PZ:  /* SEARCH DESIGNATOR IA EDITOR - TUE/08/FEB/77 */
N, i HAS NOT BEEN SELECTED; IT CANNOT BE DELETED
) ;
DISPLAY ( DLINE ) ;
GO TO PROMPTA ;

STEP12: SSTAT = SSTAT - 1 ;
STAB ( N ) = 0 ;
WFLAGS ( NTYPE ) = 1 ;
DISPLAY ( '*' | SID | ' DESIGNATOR NUMBER | ' |
NUM | ' i HAS BEEN DELETED' ) ;
GO TO PROMPTA :

PRINT OPTION SERVICED IN THIS SECTION */
/* STAB SS */
CFLAG = 0 ;  /* CATEGORY PRINT FLAG; 0 MEANS DO IT */
NC = 1 ;  /* CATEGORIES SS */

STEP4: IF ( ND >= 100 ) THEN GO TO PROMPTA ;
ND = ND + 1 ;
IF ( STAB ( ND ) = 0 ) THEN GO TO STEP4 ;
JD = STAB ( ND ) ;  /* FOUND A NON-0 DESIGNATOR */
KD = PDESC ( ND ) ;  /* PTR TO DESCRIPTOR CARD */
IF ( KD <= KMAX ) THEN GO TO STEP5 ;  
ER1: NER = 1 ;
ER1A: PJT STRING ( DLINE ) DATA ( ND, JD, KD, KMAX ) ;
GO TO BUSTED ;

STEP5: IF ( ND >= SCAT ( NC ) ) THEN GO TO STEP7 ;
NC = NC + 1 ;
CFLAG = 0 ;
IF ( NC <= 15 ) THEN GO TO STEP5 ;
ER2: NER = 2 ;
GO TO ER1A ;

STEP7: IF ( ND <= SCAT ( NC ) ) THEN GO TO STEP8 ;
STEP8: IF ( CFLAG = 1 ) THEN GO TO DETOUT ;  
/* HERE, CATEGORY LINE IS OUTPUT */
CATOUT: READ FILE ( DIND ) INTO ( DLINE ) KEY ( PCAT ( NC ) ) ;
CFLAG = 1 ;
DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;
/* NOW, DETAIL LINE IS OUTPUT */
DETOUT: READ FILE ( DIND ) INTO ( DLINE ) KEY ( KD ) ;
IF ( JD < 0 ) THEN DLINE = ' ' | DLINE ;
DISPLAY ( DLINE ) ;
GO TO STEP4 ;

ADD:   DD J = 5 TO 77 ;
K = J ;
IF ( ABO(J) -= ' ' ) THEN GO TO STEP9 ;
SOURCE CARD LISTING FOR
P2: /* SEARCH DESIGNATORS IA EDITOR - TUE/08/FEB/77 */
   END ADD:

STEP9: FCON = 2;
GET STRING ( SUBSTR(CARD,K,4) ) EDIT (NUM) ( F(4) ) ;
N = ABS(NUM);
J = PDESC (N);
IF ( N = 0 ) | ( N > 100 ) | ( J = 0 )
   THEN GO TO BUMD ;
IF ( J <= KMAX ) THEN GO TO STEP10 ;
ER3: NER = 3 ;
JE = SSTAT ;
PUT STRING (DLINE) DATA ( NUM, J, JE, N ) ;
   GO TO BUSTED ;
STEP10: IF ( STAB(N) = 0 ) THEN GO TO STEP13 ;
STAB(N) = NUM ;
   GO TO STEP14 ;
STEP13: STAB(N) = NUM ;
SSTAT = SSTAT + 1 ;
STEP14: WFLAGS(INTYPE) = 1 ;
READ FILE (DINO) INTO (DLINE) KEY (J) ;
IF ( NUM < 0 ) THEN DLINE = '.|.' || DLINE ;
DISPLAY (' *** ADDED DESIGNATOR = ' || DLINE ) ;
DISPLAY (' *** TOTAL DESIGNATORS SELECTED ARE: ' || SSTAT ) ;
   GO TO PROMPTA ;

/**************************************************************************
   /* SELECT OPTION IMPLEMENTED IN THIS SECTION */
SELECT: GO TO LP1 ;
/**************************************************************************
   /* FILE OPTION IMPLEMENTED HERE */
   /* UPDATED ARRAYS WRITTEN BACK OUT TO REC, REV, REJ PLUS
SEARCH ARGUMENT FILE */
FILE: DO J = 1 TO 3 ;
   PSARG = PARGS(J) ;
K = J - 1 ;
WRITE FILE (SARG) FROM (SARGS) ;
IF ( WFLAGS(J) > 0 ) THEN DO ;
DISPLAY (' ') ;
DISPLAY ( ' *** MODIFIED | | SID | | DESIGNATORS HAVE BEEN SAVED' ) ;
   END ;
LP5: IF ( WFLAGS(J) > 0 ) THEN
   DO ;
OPEN FILE (PROJ) RECORD OUTPUT
   TITLE ( FNAMES(J) ) ;
LP6: DO L = 1 TO 100 ;
/* SEARCH DESIGNATORS IA EDITOR - TUE/08/FEB/77 */

IF ( STAB(L) = 0 ) THEN GO TO LP6END;

CARD = "";
PUT STRING (CARD) EDIT
( STAB(L) )( F(4) ) ;
WRITE FILE (PROJ) FROM (CARD);

SSTAT = SSTAT - 1 ;
END LP6 ;

LIA = SSTAT ;
IF ( SSTAT = 0 ) THEN GO TO STEP11 ;

ER4:
NER = 4 ;
PUT STRING (DLINE) DATA ( J, L1, LIA ) ;
GO TO BUSTED ;

STEP11:
CLOSE FILE (PROJ) ;
END LP5 ;
END FILE ;
GO TO EOJ ;

************

/* THIS IS THE ERROR PROCESSOR */

BUSTED: PUT STRING (DLINE) EDIT
( "** ERROR NUMBER \', NER, \' HAS OCCURRED **" )
( A, F(2), A ) ;
DISPLAY ( '"\', '"' ) ; DISPLAY ( DLINE ) ;
GO TO PROMPTA ;

************

/* THIS IS THE BITTER ENDE */

QUIT: DISPLAY ( "" ) ;
DISPLAY ( '"SEARCH DESIGNATOR EDITOR IS NOW TERMINATING"' ) ;

LP4:
IF ( WFLAGS(J) = 0 ) THEN GO TO LP4END ;

DISPLAY ( ' ' ) ;
DISPLAY ( ' ' ) ;

LP4END:
END LP4 ;

EOJ: CLOSE FILE (DIND), FILE (MREF), FILE (TTL),
FILE (SARG) ;

END P2 ;
P3: /* RANGE-OF-EFFECTS (ROE) SEARCH MODULE - WED/02/MAR/77 */

PROC OPTIONS (MAIN);

/* REVISIONS:
WED/02/MAR/77 - VERYFIRST EDITION.
SAT/05/MAR/77 - FINAL (HOPEFULLY) DEBUG EFFORTS.
SUN/20/MAR/77 - MODS PRIOR TO TRANSMIT TO NCSS.
*/

DECLARE ( DIND, COURSE, VEHS, TASKS ) FILE RECORD DIRECT KEYED ENV ( REGIONAL (1) ),

TI CHAR (38) STATIC INIT ( "RANGE-OF-EFFECTS SEARCH FOR PROJECT: " ),
MAXCSS (3) FIXED BIN STATIC INIT ( 2, 8, 12 ),
MAXRSS (3) FIXED BIN STATIC INIT ( 1, 7, 11 ),
( AFIRST, NR, SAVEX, HIT, NOUT, BOT, TOP, J, J1, J2, K, K1, K2, L, MAXD, MAXR, N, NER, NNR, NTYPE, NNF, MAXD2, KSAVE, NC, NF ) FIXED BIN STATIC,

( CARD, DLINE ) CHAR (180) STATIC, C110 CHAR (10) DEF CARD, C1180 CHAR (70) DEF CARD POS (11), CRSCIN CHAR (8) DEF CARD POS (5), CRSCDP CHAR (4) DEF CARD, IDB CHAR (8) STATIC, VI (3) LABEL ;

DECLARE RESULT STATIC, /* ROE SEARCH RESULTS RECORD */
2 RTYP PICTURE '9', /* 1 = CRS, 2 = VEH, 3 = JOB */
2 RSTAT PICTURE '9', /* 1 = ORIGINAL SEARCH, 2 = ADDED, 3 = DELETED */
2 ROSTAT PICTURE '9', /* 1 = ABBR DATA FOUND, 2 = NO DATA AVAILABLE */
2 RESID CHAR (13), /* ENTITY IDENTIFIER */
1 RESID2 BASED (PRID), /* OVERLAY FOR COURSE CIN/CDP */
2 RESCP CHAR (8),
2 RESCDP CHAR (4),

DNAMES (3) CHAR (13) STATIC INIT ('DESC', 'VDESC', 'TDESC'),
CHAR13 CHAR (13) STATIC, /* SEARCH ARRAY FOR ANDED-DEScriptors */
VJ13 CHAR (13) BASED (P13);

DECLARE AND STATIC, /* SEARCH ARRAY FOR ORED-DEScriptors */
2 NAND FIXED BIN,
½ TAND (100) FIXED BIN,
2 CAT (100) FIXED BIN,
1 OR (15) STATIC, /* SEARCH ARRAY FOR ORED- */
2 NOR FIXED BIN,
2 NCAT FIXED BIN,
2 TOR (15) FIXED BIN;

DECLARE TTLCARD STATIC, /* PROJECT TITLE DATA BASE */
2 PNAME CHAR (10), /* CKF.ID */
2 PTTL CHAR (70),
SOURCE CARD LISTING FOR P3: /* RANGE-OF-EFFECTS (ROE) SEARCH MODULE - WED/02/MAR/77 */

DECLARE 1 SARGS STATIC, /* ROE SEARCH ARGUMENTS */
  2 SID CHAR (8),
  2 SSTAT FIXED BIN,
  2 STAB (115) FIXED BIN,
RNUMS (3,3) FIXED BIN /* MASTER INDEX REC NOS */
STATIC INIT ( 5, 8, 4; /* COURSES */
  9, 12, 8 ); /* VEHICLES */
/

DECLARE 1 CIINDES STATIC, /* COURSE DESCRIPTORS */
  2 CINOS CHAR (8),
  2 FILL CHAR (5),
  2 DTAB (100) FIXED BIN ;

DECLARE 1 CINDER (304) STATIC, /* CRS CIN DIRECTORY */
  2 CIIN CHAR (8),
  2 CINPTR FIXED BIN, LUADCIN CHAR (760) BASED (PLCIN),
COVCIN (4) CHAR (760) BASED (PTCIN) ;

DECLARE 1 DINDEX STATIC, /* OVERALL DESCRIPTOR INDEX */
  2 DSTART FIXED BIN, /* CKF.DESC.INDEX */
  2 DEVO FIXED BIN,
  2 NCAT FIXED BIN,
  2 NCAT FIXED BIN,
  2 CATEGORIES (15),
  3 PCAT FIXED BIN,
  3 SCAT FIXED BIN,
  3 ECAT FIXED BIN,
  2 PDESC (111) FIXED BIN,
COVER (4) CHAR (80) BASED (PCOV) ;

DECLARE 1 MAST STATIC, /* VEHICLE DESCRIPTOR FILE */
  2 REFS (40) FIXED BIN ;

DECLARE 1 VEHDEC STATIC, /* VEHICLE DIRECTORY */
  2 VEHD CHAR (13),
  2 VETAB (100) FIXED BIN ;

DECLARE 1 VEHDIR (200) STATIC, /* VEHICLE DIRECTORY */
  2 CVHEH CHAR (13),
  2 CVEHPT FIXED BIN,
COVVEH (4) CHAR (750) BASED (PTVEH),
LOADVEH CHAR (750) BASED (PLVEH) ;

DECLARE 1 TASK.DESC STATIC, /* TASK DESCRIPTOR FILE */
  2 TASKOS CHAR (13),
  2 TATAB (100) FIXED BIN ;

DECLARE 1 TASKDIR (1200) STATIC, /* JOB TASK FILE DIRECTORY */
  2 CHTASK CHAR (13),
  2 CHTASK FIXED BIN,
COVTASK (12) CHAR (1500) BASED (PTTASK),
LOADTASK CHAR (1500) BASED (PLTASK) ;

COMMENCE:
PRID = ADDR (RESID) ;
P13 = ADDR (CINOS) ;
P1C1N = ADDR (CINDER) ;
PCOV = ADDR (DINDEX) ;
SOURCE CARD LISTING FOR P3: /* RANGE-OF-EFFECTS (ROE) SEARCH MODULE - WED/02/MAR/77 */

PTVEH = ADDR (VEHDIR) ;
PTTASK = ADDR (TASKDIR) ;

OPEN FILE (DIND) INPUT,
FILE (MREF) RECORD INPUT,
FILE (ID) RECORD INPUT,
FILE (SARG) RECORD INPUT,
FILE (RESU) RECORD OUTPUT,
FILE (COURSE) INPUT,
FILE (DCIN) RECORD INPUT,
FILE (VEHS) INPUT,
FILE (DVEH) RECORD INPUT,
FILE (TASKS) INPUT,
FILE (DTASK) RECORD INPUT ;

FILE DIRECTORIES LOADED IN THIS SECTION */

CINLOAD:  DO J = 1 TO 4 ;
           PLCIN = ADDR ( COVCIN (J) ) ;
           READ FILE (DCIN) INTO (LOADCIN) ;
           END CINLOAD ;

VEHLOAD:  DO J = 1 TO 4 ;
           PLVEH = ADDR ( COVVEH (J) ) ;
           READ FILE (DVEH) INTO (LOADVEH) ;
           END VEHLOAD ;

TASKLOAD: DO J = 1 TO 12 ;
           PLTASK = ADDR ( COVTASK (J) ) ;
           READ FILE (DTASK) INTO (LOADTASK) ;
           END TASKLOAD ;

/* NOW, OUTPUT THE PROJECT TITLE */

READ FILE (ID) INTO (CARD) ;
DISPLAY ( ' ' ) ;
DISPLAY ( 'RANGE-OF-EFFECTS (ROE) SEARCH ' ) ;
DLINE = T1 || C110 ;
DISPLAY ( ' ' ) ;
DISPLAY ( DLINE ) ;

DLINE = C1180 ;
DISPLAY ( DLINE ) ;

/* FINAL SETUP BEFORE PROGRAM MAINLINE */

NTYPE = 1 ; /* START WITH COURSES */
NOUT = 0 ; /* TOTAL # OUTPUT (HIT) RECORDS */
RSTAT = 1 ; /* RESULT RECORD = ALL ORIGINAL */

READ FILE (MREF) INTO (MAST) ;

/* THIS IS START OF MAINLINE LOOP */

START:  RTYP = NTYPE ; /* RESULT RECORD TYPE */
NC, NF, NNF = 0 ; /* DESCRIPTOR RECORD COUNTS */

READ FILE (SARG) INTO (SARGS) ;

PUT STRING (DLINE) EDIT ( ' ** NOW PROCESSING ', SID, 
" SEARCH - ", SSTAT, ' DESCRIPTORS SELECTED' ) 
( A, A(8), A, F(3), A ) ;
DISPLAY ( ' ' ) ;
DISPLAY ( DLINE ) ;
SOURCE CARD LISTING FOR
P3: /* RANGE-OF-EFFECTS (ROE) SEARCH MODULE - WED/02/MAR/77 */

IF ( SSTAT <= 0 ) THEN
  DO ;
  DLINE = '*** NO SEARCH DESCRIPTORS SPECIFIED FOR
       TYPE DATA - SEARCH IS BYPASSED' ;
  DISPLAY ( | | SID | | ) ;
  DISPLAY ( DLINE ) ;
  GO TO TALLY ;
END LP1 :

/* HERE, MASTER DESCRIPTOR INDEX IS LOADED FOR APPROPRIATE NTYPE */
LOADIND: DO J = RNUMS(NTYPE,1) TO RNUMS(NTYPE,2) ;
  READ FILE (DIND) INTO (CARD) KEY (J) ;
  C3VER ( J - RNUMS(NTYPE,3) ) = CARD ;
END LOADIND ;
NAND = 0 ;
NOR = 0 ;
MAXD = REFS ( MAXCSS(NTYPE) ) ; /* MAX DIRECTORY SS */
MAXD2 = MAXD / 2 ; /* HALFWAY POINT FOR TASKS TYPE ONLY */
MAXR = REFS ( MAXRSS(NTYPE) ) ; /* TOTAL # DATA RECS */
*************************************************************************/

/* NOW, LOAD UP ARRAYS PRIOR TO SEARCH OF DESIGNATOR DATASET. 
FIRST, SETUP THE -OR- ARRAY FOR HANDLING DESIGNATORS 
WITHIN A SINGLE CATEGORY */
NOR = 0 ; /* ALL COUNT FIELDS = 0 */
L = 0 ; /* OR STRUCTURE SS */
LP2: DO J = 1 TO 15 ; /* CATEGORY SS */
  AFIRST = 0 ; /* HAVE NOT STARTED A CAT YET */
  K1 = SCAT (J) ;
  K2 = ECAT (J) ;
  LP3: DO K = K1 TO K2 ; /* INDIVIDUAL CAT BOUNDS */
    IF ( STABU ) THEN GO TO LP3END ;
    IF ( AFIRST = 0 ) THEN GO TO STEPL :
      AFIRST = 1 ;
      SAVEK = STAB (K) ;
      KSAVE = K ;
      GO TO LP3END ;
    IF ( AFIRST < 0 ) THEN GO TO STEP2 ;
      L = L + 1 ;
      NOR (L) = 1 ;
      TOR (L,1) = SAVEK ;
      STAB (KSAVE) = 0 ;
      AFIRST = -1 ;
      OR - NCAT (L) = J ;
    STEP2: NOR(L) = NOR(L) + 1 ;
      TOR (L, NOR(L) ) = STAB (K) ;
      STAB (K) = 0 ;
    LP3END: END LP3 ;
END LP2 ;

/* NOW, SETUP THE -AND- STRUCTURE TO ENCOMPASS DESIGNATORS 
THAT ARE TO BE ANDED BETWEEN CATEGORIES */
SOURCE CARD LISTING FOR
P3: /* RANGE-OF-EFFECTS (ROE) SEARCH MODULE - WED/02/MAR/77 */

NAND = 0 ;  /* AND STRUCTURE COUNTER */
K = 0 ;  /* CATEGORY SS, 1 TO 15 */

DO J = 1 TO 100 ;  /* STAB SS */
IF ( STAB(J) = 0 ) THEN GO TO LP4END ;

K = K + 1 ;
IF ( J >= SCAT(K) ) THEN GO TO STEP4 ;

NER = 1 ;
ER1A: PUT STRING (DLINE) DATA ( NAND, J, K, SCAT(K) ) ;
GO TO BUSTED ;

STEP4: IF ( J > ECAT(K) ) THEN GO TO STEP6 ;
NAND = NAND + 1 ;
IF ( NAND <= 100 ) THEN GO TO STEP5 ;

NER = 2 ;
GO TO ER1A ;

STEP5: TAND (NAND) = STAB (J) ;
CAT (NAND) = K ;
GO TO LP4END ;

STEP6: IF ( K < 15 ) THEN GO TO STEP3 ;
NER = 3 ;
GO TO ER1A ;

LP4END: END LP4 ;

OPEN FILE (DESIGS) RECORD INPUT TITLE ( DNAMES(NTYPE) ) ;
ON ENDFILE (DESIGS) GO TO EOFUESC ;
NC, NF, NNF = 0 ;

RD1: READ FILE (DESIGS) INTO (CINDESC) ;
NC = NC + 1 ;
RDSTAT = 1 ;  /* ASSUME ABBR, DATA EXISTS */
SEARCH: HIT = 0 ;  /* NOGO, JUST STARTING */

IF ( NAND = 0 ) THEN GO TO SEARCHOR ;

ANDSEARCH: DO J = 1 TO NAND ;
N = ABS ( TAND(J) ) ;
IF ( TAND(J) < 0 ) THEN GO TO TESTNOT ;

TESTPOS: IF ( DTAB(N) > 0 ) THEN GO TO GOAND ;
GO TO ALTAND ;

TESTNOT: IF ( DTAB(N) <= 0 ) THEN GO TO GOAND ;
GO TO NOGO ;

ALTAND: K1 = SCAT ( CAT(J) ) ;
K2 = ECAT ( CAT(J) ) ;
DO K = K1 TO K2 ;
IF ( DTAB(K) <= 0 ) THEN GO TO NOGO ;
END LP5 ;

GOAND: END ANDSEARCH ;
SOURCE CARD LISTING FOR
P3: /* RANGE-OF-EFFECTS (ROE) SEARCH MODULE - WED/02/MAR/77 */
SEARCHOR: HIT = 1; /* HERE, WE MADE IT ONE WAY OR TOOTHER */
/* NOW, WE TACKLE THE OR SEARCH */
IF ( L > 0 ) /* L IS COUNT OF OR STRUCTURES */
    THEN GO TO ORSEARCH;
IF ( HIT = 0 ) THEN GO TO NOGO;
ELSE GO TO GO;
ORSEARCH: DO J = 1 TO L; /* OR-STRUCTURE SS */
    K1 = NOR (J); /* TOT DESCS WITHIN CAT */
    DO K = 1 TO K1;
    N = TOR (J,K);
    IF ( N > 0 ) THEN GO TO POSTEST;
NEGTEST: IF ( DTAB(ABS(N)) <= 0 ) THEN GO TO ORGO;
    GO TO LP6END;
POSTEST: IF ( DTAB(N) > 0 ) THEN GO TO ORGO;
LP6END: END LP6;
LP7: END LP7;
ORGO: END ORSEARCH;
/* IF FALL THRU FROM ABOVE, ALL IS GO */
GO: GO TO V1 (NTYPE);
/* HERE, COJARSE RESULTS PROCESSED */
V1(1): J1 = MAXD; /* MAX SS USED IN CRS DIRECTORY */
LP8: DO J = 1 TO J1; /* LOOK UP FOUND CIN */
    K = J;
    IF ( CINDS = CHCIN(J) )
        THEN GO TO CRSGO;
LP8END: END LP8;
CRSNOGO: /* NO FIND IF FALL OUTTA ABOVE */
    RESID = "";
    RDSTAT = 2;
    RESCIN = CINDS;
STEP7: WRITE FILE (RESU) FROM (RESULT);
    NOUT = NOUT + 1;
    NNF = NNF + 1;
    GO TO RD1;
CRSGO: /* HERE, CRS DIRECTORY SEARCH WAS GO */
    RDSTAT = 1;
    K1 = K;
    K = CINPTR (K);
    IF ( K <= MAXR ) THEN GO TO STEP8;
ER4: NER = 4;
SOURCE CARD LISTING FOR
P3: /* RANGE-OF-EFFECTS (ROE) SEARCH MODULE - WED/02/MAR/77 */

PJT STRING (DLINE) DATA ( K, K1, REFS(1), CINDS ) ;
GO TO BUSTED ;

STEP8: READ FILE (COURSE) INTO (CARD) KEY ( K ) ;
IF ( CINDS = CRSCIN ) THEN GO TO RO1 ;
RESID = * ;
RESCIN = CRSCIN ;
RESCDP = CRSCDP ;
WRITE FILE (RESU) FROM (RESULT) ;
NOUT = NOUT + 1 ;
NF = NF + 1 ;
K = K + 1 ;
IF ( K > REFS(1) ) THEN GO TO RD1 ;
GO TO STEP8 ;

/* HERE, VEHICLE RESULTS ARE OUTPUT */

V1(2): J1 = MAXD ;
RESID = VJ13 ;

LP9: DO J = 1 TO J1 ;
K = J ;
IF ( VJ13 = CHVEH(J) ) THEN GO TO VEHGO ;

VEHGO: /* IF FALL OUTTA ABOVE, NO FIND */
RDSTAT = 2 ;
GO TO RD7 ;

/* HERE, FOUND WANTED VEHICLE IN DIRECTORY */

ER5: NER = 5 ;
PUT STRING (DLINE) DATA ( K, VEHPTR(K), REFS(7), RESID ) ;
GO TO BUSTED ;

STEP9: WRITE FILE (RESU) FROM (RESULT) ;
NOUT = NOUT + 1 ;
NF = NF + 1 ;
GO TO RD1 ;

/* FINALLY, THE TASK RESULTS ARE PROCESSED */

V1(3): RESID = VJ13 ;
IF ( VJ13 = ChTASK(MAXD2) ) THEN DO ;
BOT = MAXD2 ; TOP = MAXD ; END ;
ELSE DO ;
BOT = 1 ; TOP = MAXD2 ; END ;

LP10: DO J = BOT TO TOP ;
K = J ;
IF ( VJ13 = ChTASK(J) ) THEN GO TO TASKGO ;

/* IF FALL OUTTA ABOVE, NO FIND AT ALL */
SOURCE CARD LISTING FOR
P3: /* RANGE-OF-EFFECTS (ROE) SEARCH MODULE - WED/02/MAR/77 */

GO TO VEHNOGO;

TASKGO: /* HERE, TASK DIRECTORY SEARCH WAS GOLDEN */

ROSTAT = 1;

IF ( TASKPTR(K) <= REFS(11) ) THEN GO TO STEP9;

ER6: NER = 6;

PUT STRING (DLINE) DATA ( K, TASKPTR(K), REFS(11), RESID );

GO TO BUSTED;

/***************************************************************************/

/* HERE, WHEN EOF REACHED ON DESCRIPTOR INPUT DATASET */

EOFDESC: CLOSE FILE (DESIGS);

PUT STRING (DLINE) EDIT

( "TOTAL NUMBER DESCRIPTOR RECORDS EXAMINED : ", NC ) ;

DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;

PUT STRING (DLINE) EDIT

( "TOTAL RECORDS SELECTED WITH MATCHING DATA : ", NF ) ;

DISPLAY ( DLINE ) ;

PUT STRING (DLINE) EDIT

( "TOTAL RECORDS SELECTED WITHOUT ABBREVIATED DATA : ", NNF ) ;

DISPLAY ( DLINE ) ;

PUT STRING (DLINE) EDIT

( "ACCUMULATED SEARCH OUTPUT RECORDS, THUS FAR : ", NOUT ) ;

DISPLAY ( DLINE ) ;

/* NOW, DETERMINE IF ANOTHER CYCLE IS REQUIRED */

TALLY: IF ( NTYPE >= 3 ) THEN GO TO EOJ;

NTYPE = NTYPE + 1;

GO TO START;

***************************************************************************/

/* HERE, PROGRAM IS IN STAGE OF TERMINAL ILLNESS */

BUSTED: PUT STRING (CARD) EDIT

( "** ERROR NUMBER ', NER, ' HAS OCCURRED **" ) ;

DISPLAY ( ' ' ) ;

DISPLAY ( CARD ) ;

DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;

***************************************************************************/

/* THIS IS THE BITTER ENDE */

EOJ: OLINE = "** DESIGNATOR SEARCH PROGRAM IS TERMINATING" ;

DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;

PUT STRING (DLINE) EDIT

( "TOTAL ACCUMULATED SEARCH OUTPUT RECORDS : ", NOUT ) ;

DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;

DISPLAY ( ' ' ) ;
SOURCE CARD LISTING FOR P3: /* RANGE-OF-EFFECTS (ROE) SEARCH MODULE - WED/02/MAR/77 */

CLOSE FILE (DIND), FILE (COURSE),
FILE (MREF), FILE (DCIN),
FILE (SARG), FILE (RESU), FILE (ID);
EVD P3 ;
SOURCE CARD LISTING FOR P5A: /* PRINT OF RAW ROE RESULTS FILE - MON/28/FEB/77 */

PROC (INPARM) OPTIONS (MAIN):

/* REVISIONS:
WED/02/MAR/77 - FIRST IMPLEMENTATION.
SAT/05/MAR/77 - FINAL (HOPEFULLY) DEBUG EFFORTS.
THU/24/MAR/77 - MODS PRIOR TO TRANSMIT TO NCSS. */

DECLARE 1 TTLCARD STATIC, /* PROJECT TITLE DATA BASE */
  2 PNAME CHAR (10), /* CKF.ID */
  2 PTTL CHAR (70),
  INPARM CHAR (100) VARYING,
  1 POVER BASED (PP),
  2 PARM2 CHAR (4),
  ( DISP, PTR, DPAGE, I, J, K, L, LT, NC, NN, NPAGE, NR, NTYPE, NV, NDF, NJ ) FIXED BIN STATIC,
  DTTL CHAR (23) STATIC INIT ( ' PAGE NUMBER : XXXX' ),
  DDGPE PICTURE *ZZZ9* DEF DTTL POS (20),
  MAXLTB FIXED BIN STATIC INIT ( 60 ) ;

DECLARE VINTAGE (3) CHAR (8) STATIC INIT ('COURSES', 'VEHICLES', 'TASKS'),
  DTTL2 CHAR (39) STATIC INIT ('RANGE-OF-EFFECTS RESULTS FOR : XXXXXXXX' ),
  DZTY CHAR (8) DEF DTTL2 POS (32),
  DTTL3 CHAR (39) STATIC INIT ( (39)'-' ) ;

DECLARE ( COURSE, VEHS, TASKS ) FILE RECORD DIRECT KEYED ENV ( REGIONAL1 ) ;

DECLARE ( CARD, DLINE ) CHAR (80) STATIC,
  C110 CHAR (10) DEF CARD,
  C1180 CHAR (70) DEF CARD POS (11) ;

DECLARE 1 MAST STATIC,
  2 REFS (40) FIXED BIN,
  1 RESULT BASED (PR),
  2 RTYP PICTURE '9',
  2 RSTAT PICTURE '9',
  2 RSTAT PICTURE '9',
  2 RESID CHAR (13),
  1 RESC BASED (PR),
  1 F1, 2 F2, 2 F3 ) CHAR (1),
  2 RESCIN CHAR (8),
  2 RESCDP CHAR (4),
  1 REST BASED (PR),
  1 F1, 2 F2, 2 F3 ) CHAR (1),
  2 RATE CHAR (5),
  2 RANK CHAR (2),
  2 JOBTASK CHAR (6),
  S16 CHAR (16) STATIC,
  V1 (3) LABEL ;
DECLARE T1 RECNO CHAR (32) STATIC INIT (COURSE' ),
    T1U CHAR (37) STATIC INIT (CDP ' ),
    T2 RECNO STOCK NUMBER FILE NUM TRAINING DEVICE' ),
    T2U CHAR (46) STATIC INIT (FILE_NUM ' ),
    T3 RECNO RATE RANK JOB TASK' ),
    T3U CHAR (29) STATIC INIT (JOB_TASK ' ) ;
DECLARE WINDUPS (6) CHAR (44) STATIC INIT (TOTAL NUMBER COURSE RECORDS PROCESSED :,
    TOTAL NUMBER VEHICLE RECORDS PROCESSED :,
    TOTAL NUMBER JOBTASK RECORDS PROCESSED :,
    TOTAL NUMBER ROE RESULT RECORDS READ :,
    RECORDS WITHOUT MATCHING ABBREVIATED DATA :,
    TOTAL NUMBER DIRECTORY SEARCH FAILURES : ),
WIND (6) FIXED BIN STATIC ;
/*****************************/
DECLARE 1 ACRS STATIC , /* ABBR. COURSE FILE */
    2 SET1 , /* CKF.ACOURSES */
        3 CDP CHAR (4),
        3 CIN CHAR (8),
        3 CST CHAR (16),
        3 VOB CHAR (4),
        3 VEC CHAR (4),
        2 PC CHAR (2),
        2 SET2 ,
        3 RMS CHAR (3),
        3 TYCRS CHAR (2),
        3 SVC CHAR (1),
        3 MI CHAR (1),
        2 STCD CHAR (1),
        2 STDE PICTURE ' (5)9* ,
        2 SET3 ,
        3 TRAPS CHAR (1),
        3 TPC CHAR (5),
        2 ATTR PICTURE '99V9*',
        2 STBK PICTURE '99V9*',
        2 CLEN PICTURE '999*',
        2 THRS PICTURE '999*',
        2 CA3B PICTURE ' (5)9V99*',
        2 CCB0 CHAR (1) ;
DECLARE 1 CINDER (304) STATIC , /* CRS CIN DIRECTORY */
    2 CHCIN CHAR (8), /* CKF.CRS.DIRCIN */
    2 CIVPTR FIXED BIN ,
    2 LOADCIN CHAR (760) BASED (PLCIN) ,
    2 COVCIN (4) CHAR (760) BASED (PTCIN) ;
DECLARE 1 CPDIR (1200) STATIC , /* CRS CDP DIRECTORY */
    2 CHCOP CHAR (4), /* CKF.CRS.DIRCDP */
    2 CPDPtr FIXED BIN ,
    2 LOADCP CHAR (1200) BASED (PLCDP) ,
    2 COVCOP (6) CHAR (1200) BASED (PTCDP) ;
D.6-34
SOURCE CARD LISTING FOR
PSA: /* PRINT OF RAW ROE RESULTS FILE - MON/28/FEB/77 */

DECLARE 1 AVEHICLES STATIC; /* ABBR. VEH FILE */
  2 STOCKN CHAR (13), /* CKF.AVEHS */
  2 DEVOESIG CHAR (9),
  2 DEVNAME CHAR (47),
  2 DEVCOST PICTURE '8V99',
  2 CC90 CHAR (1),
  VCOST FIXED DEC (10,2) STATIC;
  VOVER CHAR (80) BASED (PV80);

DECLARE 1 VEHDR (200) STATIC, /* VEHICLE DIRECTORY */
  2 CHVEH CHAR (13), /* CKF.VEH.DIR */
  2 VEMPTR FIXED BIN,
  COVVEH (4) CHAR (750) BASED (PTVEH),
  LOADVEH CHAR (750) BASED (PLVEH);

DECLARE 1 ATASKS STATIC; /* ABBR. TASK FILE */
  2 RATING CHAR (7), /* CKF.ATASKS */
  2 JOBTASK CHAR (6),
  2 TASKTTL CHAR (50),
  2 FILL CHAR (6),
  2 BILCOST PICTURE '8V99',
  2 CC80 CHAR (1),

DECLARE 1 TASKDIR (1200) STATIC, /* JOBTASK FILE DIRECTORY */
  2 CHTASK CHAR (13), /* CKF.TASKS.DIR */
  2 TASKPTR FIXED BIN,
  COVTASK (12) CHAR (1500) BASED (PTTASK),
  LOADTASK CHAR (1500) BASED (PLTASK);

START: NTYPE, NC, NJ, NV, N3, NN, NDF,
  NPAGE, DPAGE, LT,
  J, K = 0;
  FLCIN, FLCDP, FLVEH, FLTASK = 'O'B;
  PP = ADDR (INPARM);
  PR = ADDR (S16);
  PTCIN = ADDR (CINDER);
  PTCDP = ADDR (CDPDIR);
  PV80 = ADDR (AVEHICLES);
  PTVEH = ADDR (VEHDR);
  PTTASK = ADDR (TASKDIR);

COMENCE:
  OPEN FILE (MREF) RECORD INPUT,
  FILE (ID) RECORD INPUT,
  FILE (COURSE) INPUT,
  FILE (DCIN) RECORD INPUT,
  FILE (DCDP) RECORD INPUT,
  FILE (VEHS) INPUT,
  FILE (DVEH) RECORD INPUT,
  FILE (TASKS) INPUT,
  FILE (OTASK) RECORD INPUT,
  FILE (RESU) RECORD INPUT;
  ON ENDFILE (RESU) GO TO EOF;
  READ FILE (MREF) INT (MAST);
SOURCE CARD LISTING FOR
PSA: /* PRINT OF RAW ROE RESULTS FILE - MON/28/FEB/77 */

/*****************************/

/* HERE, DEVICE ASSIGNMENTS ARE VERIFIED */

FIRST: DISP, PTR = 1 ;
IF ( PARM2 = 'BOTH' ) THEN GO TO STEPA ;
IF ( PARM2 = 'TERM' )
THEN PTR = 2 ;
ELSE DISP = 2 ;

/* HERE, PRINTER DATASET IS INITIALIZED */

STEPA: IF ( PTR = 2 ) THEN GO TO STEPB ;
OPEN FILE (SYSPRINT)
LINESIZE (120) PAGESIZE (60) :
ON ENDPAGE (SYSPRINT)
BEGIN ;
NPAGE = NPAGE + 1 ;
PUT EDIT ( 'PAGE NUMBER : ', NPAGE )
( PAGE, X(10), A, F(*) ) ;
IF ( NTYPE = 1 ) THEN
DO ;
PUT EDIT ( T1, T1U, * * ) ( R(FMT1) ) ;
END ;
FMT1: FORMAT ( 3 ( SKIP(1), A ) ) ;
IF ( NTYPE = 2 )
THEN PUT EDIT ( T2, T2U, * * ) ( R(FMT1) ) ;
ELSE PUT EDIT ( T3, T3U, * * ) ( R(FMT1) ) ;
SKP: PUT SKIP (1) ;
END ;

/* HERE, INITIALIZATION OF TERMINAL DATASET PROCEEDS */

STEPB: IF ( DISP = 2 ) THEN GO TO STEPC ;
ON CONDITION (NEWDP)
BEGIN ;
IF ( LT < 61 ) THEN LT = 61 ;
IF ( LT < 66 ) THEN
DO L = LT TO 66 ;
DISPLAY ( " " ) ;
END LPA ;
DISPLAY ( " " ) ;
DPAGE = DPAGE + 1 ;
DPAGE = DPAGE ;
DISPLAY ( DTTL ) ;
DISPLAY ( " " ) ;
LT = 3 ;
DISPLAY ( DTTL2 ) ;
DISPLAY ( DTTL3 ) ;
LT = LT + 3 ;
EVD ;

/* NOW, MASTER TITLE STRIP IS OUTPUT */

STEC: READ FILE (ID) INTO (CARD) ;
SOURCE CARD LISTING FOR P5A: /* PRINT OF RAW ROE RESULTS FILE - MON/28/FEB/77 */

DLINE = (10) ; || ** INITIAL SEARCH RESULTS FOR PROJECT -
| | C110 ; || **
CARD = (12) ; | C1180 ;
DISPLAY ( * * ) ; DISPLAY ( DLINE ) ;
DISPLAY ( CARD ) ; DISPLAY ( * * ) ;
IF ( PTR = 2 ) THEN GO TO RD1 ;
PUT EDIT ( DLINE, CARD ) ( SKIP(3), A, SKIP(1), A ) ;
GO TO RD1 ;

*******************************************************************************
/* HEREIN, IS START OF MAINLINE LOOP */

RD1: READ FILE (RESU) INTO (S16) ;
NR = NR + 1 ;
IF ( RTYP = NTYPE ) THEN GO TO STEP1 ;
NTYPE = RTYP ;
D2TY = VINTAGE (NTYPE) ;
IF ( PTR = 1 ) THEN SIGNAL ENDPAGE (SYSPRINT) ;
IF ( DISP = 1 ) THEN SIGNAL CONDITION (NEWDP) ;

STEP1: GO TO VI (NTYPE) ;

/* HERE, A COURSE RECORD IS PROCESSED */

VI(1): NC = NC + 1 ;
IF ( RDSTAT = 2 ) THEN
NOCDATA: DO ;
PUT STRING (DLINE) EDIT
( NR, RESCIN, "NOTE - NO MATCHING NITRAS DATA", 
" AVAILABLE FOR THIS CIN" )
( P'ZZ,ZZ9*, X(1), A(8), X(2), A, A ) ;
GO TO OUTLOOP ;
END NOCDATA ;
YECDP = RESCDP ;
LOOKU = STEP2 ;
GO TO FINDCDP ;

STEP2: IF ( CDPREC = 0 ) THEN
LP1: DO ;
NDF = NDF + 1 ;
PUT STRING (DLINE) EDIT
( NR, RESCIN, RESCDP, "** CDP COULD NOT", 
" BE LOCATED IN CDP DIRECTORY" )
( P'ZZ,ZZ9*, X(1), A(8), X(1), A(4), 
X(2), A, A ) ;
GO TO OUTLOOP ;
END LP1 ;
READ FILE (COURSE) INTO (ACRS) KEY (CDPREC) ;
PUT STRING (DLINE) EDIT ( NR, RESCIN, RESCDP, CST )
( P'ZZ,ZZ9*, X(1), A(8), X(1), A(4), X(1), A(16) ) ;
GO TO OUTLOOP ;
SOURCE CARD LISTING FOR
PSA: /* PRINT OF RAW ROE RESULTS FILE - MON/28/FEB/77 */

/* HERE, VEHICLE TYPE RECORDS ARE PROCESSED */

V1(2): NV = NV + 1;

IF ( RDSTAT = 2 ) THEN

NOVDATA: DO:

NV = NN + 1;

PUT STRING (DLINE) EDIT

( NR, RESID, '** NO ABBREVIATED VEHICLE DATA **',
  'AVAILABLE **'
  ( P'ZZ,ZZ9', X(1), A(13), X(2), A, A ) ;

GO TO OUTLOOP ;

END NOVDATA ;

YEVEH = RESID ;
LOOKU = STEP3 ;
GO TO FINDVEH ;

STEP3: PUT STRING (DLINE) EDIT ( NR, RESID )

( P'ZZ,ZZ9', X(1), A(13) ) ;

IF ( VEHREC = 0 ) THEN

DO : NDF = NDF + 1 ;

SUBSTR ( DLINE, 22 ) =

'** STOCK NUMBER COULD NOT BE LOCATED IN DIRECTORY **'

GO TO OUTLOOP ;

END LP2 ;

READ FILE (VEHS) INTO (AVEHICLES) KEY (VEHREC) ;

SUBSTR ( DLINE, 22 ) =

DEVDESIG || || DEVNAME ;

GO TO OUTLOOP ;

/* NOW, PROCESS A TASK RECORD HERE */

V1(3): NJ = NJ + 1;

PUT STRING (DLINE) EDIT ( NR, RATE, RANK, REST*JOBTASK )

( P'ZZ,ZZ9', X(1), A(5), X(1), A(2), X(1), A(6) ) ;

IF ( RDSTAT = 2 ) THEN

NOTDATA: DO :

NV = NN + 1;

SUBSTR ( DLINE, 26 ) =

'** NO ABBREVIATED TASK DATA IS AVAILABLE **'

GO TO OUTLOOP ;

END NOTDATA ;

YEVEH = RESID ;
LOOKU = STEP4 ;
GO TO FINDTASK ;

STEP4: IF ( TASKREC = 0 ) THEN

DO : NDF = NDF + 1 ;

SUBSTR ( DLINE, 26 ) =

'** TASK ID COULD NOT BE LOCATED IN DIRECTORY **' ;
SOURCE CARD LISTING FOR
P5A: /* PRINT OF RAW ROE RESULTS FILE - MON/28/FEB/77 */

GO TO OUTLOOP;

END LP3;
READ FILE (TASKS) INTO (ATASKS) KEY (TASKREC);
SUBSTR (DLINE, 26) = TASKTTL;

GO TO OUTLOOP;

/* HERE IS THE PRINTED LINE OUTPUT SEQUENCE */

OUTLOOP: IF (PTR = 1)
THEN PUT LIST (DLINE) SKIP (1);
IF (DISP = 2) THEN GO TO RD1;
IF (LT > MAXLTB) THEN SIGNAL CONDITION (NEWDP);
DISPLAY (DLINE);
LT = LT + 1;
GO TO RD1;

DECLARE (FLCIN, FLCDP, FLVEH, FLTASK) BIT (1) STATIC,
YECIN CHAR (8) STATIC,
YECDP CHAR (4) STATIC,
(YETASK, YEVEH) CHAR (13) STATIC,
(CINREC, CDPREC, VEHREC, TASKREC, VEHMAX, TASKMAX,
CINMAX, CDPMAX) FIXED BIN STATIC,
LOOKU LABEL;

/* CIN DIRECTORY LOOKUP SUBROUTINE
ENTRY: DESIRED CIN IN YECIN
EXIT: CINREC = 0 (NO FIND) OR REG NUMBER
RETURN VECTOR IS LOOKU */

FINDCIN: IF (FLCIN) THEN GO TO CIN2;
CINMAX = REFS (2);
FLCIN = '1'B;

CINLOAD: DO J = 1 TO 4;
PLCIN = ADDR (COVCIN (J));
READ FILE (DCIN>) INTO (LOADCIN);
END CINLOAD;

CIN2: DO I = 1 TO CINMAX;
IF (YECIN ^= CHCIN(I)) THEN GO TO ECIN2;
CINREC = CINPTR (I);
GO TO LOOKU;

ECIN2: END CIN2;

NFCIN: CINREC = 0;
GO TO LOOKU;

/* CDP DIRECTORY LOOKUP SUBROUTINE
ENTRY: DESIRED CDP IN YECDP
EXIT: CDPREC = 0 (NO FIND), OR REC NUMBER
RETURN VECTOR IS LOOKU */

FINDCDP: IF (FLCDP) THEN GO TO CDP2;
CDPMAX = REFS (3);
FLCDP = '1'B;

CDPLOAD: DO J = 1 TO 4;
PLCDP = ADDR (COVCDP (J));
SOURCE CARD LISTING FOR
PSA: /* PRINT OF RAW ROE RESULTS FILE - MON/28/FEB/77 */

READ FILE (DCDP) INTO (LOADCDP) ;
END CDPLOAD ;

CDP2: DO I = 1 TO CDPMAX ;
IF ( YECDP = CHCDP(I) ) THEN GO TO ECDP2 ;
CDPREC = CDPPTR(I) ;
GO TO LOOKU ;
ECDP2: END CDP2 ;
NFCDP: CDPREC = 0 ;
GO TO LOOKU ;

/* VEHICLE ID LOOKUP SUBROUTINE
ENTRY : DESIRED VEH ID IN YEVEH
EXIT : VEHREC = 0 (NO FIND), OR REC NUMBER
RETURN VECTOR IS LOOKU */
FINDVEH: IF ( FLVEH ) THEN GO TO VEH2 ;
VEHMAX = REFS(8) ;
FLVEH = '1'B ;
VEHLOAD: DO J = 1 TO 4 ;
PLVEH = ADDR( COVVEH(J) ) ;
READ FILE (DVEH) INTO (LOADVEH) ;
END VEHLOAD ;
VEH2: DO I = 1 TO VEHMAX ;
IF ( YEVEH = CHVEH(I) ) THEN GO TO EVEH2 ;
VEHREC = VEHPTR(I) ;
GO TO LOOKU ;
EVEH2: END VEH2 ;
NFVEH: VEHREC = 0 ;
GO TO LOOKU ;

/* TASK ID DIRECTORY LOOKUP SUBROUTINE
ENTRY : DESIRED TASK ID (13-CHAR) IN YETASK
EXIT : TASKREC = 0 (NO FIND) OR REC NUMBER
RETURN VECTOR IS LOOKU */
FINDTASK: IF ( FLTASK ) THEN GO TO TASK2 ;
TASKMAX = REFS(12) ;
FLTASK = '1'B ;
TASKLOAD: DO J = 1 TO 12 ;
PLTASK = ADDR( COVTASK(J) ) ;
READ FILE (DTASK) INTO (LOADTASK) ;
END TASKLOAD ;
TASK2: DO I = 1 TO TASKMAX ;
IF ( YETASK = CHTASK(I) ) THEN GO TO ETASK2 ;
TASKREC = TASKPTR(I) ;
GO TO LOOKU ;
ETASK2: END TASK2 ;
NFTASK: TASKREC = 0 ;
GJ TO LOOKU ;

/*********************/
/* HERE IS EOF WINDUP PROCESSING */
EOF: WIND(1) = NC ;
WIND(2) = NV ;
WIND(3) = NJ ;
WIND(4) = NR ;
WIND(5) = NN ;
WIND(6) = NDF ;
IF ( PTR = 1 ) THEN PUT SKIP (3) ;
SOURCE CARD LISTING FOR P5A: /* PRINT OF RAW ROE RESULTS FILE - MON/28/FEB/77 */

DISPLAY ( "=" "=" ); DISPLAY ( "=" "=" );
DISPLAY ( "** RESULTS FILE NOW AT END-OF-FILE **" );
DISPLAY ( "=" "=" );

SUMMARY: DJ J = 1 TO 6 ;

PUT STRING (DLINE) EDIT ( WINDUPS(J), WIND(J) ) ( A(44), P'ZZ,ZZ9' ) ;
IF ( PTR = 1 )
    THEN PUT LIST ( (10) ' ' ) DLINE ) SKIP (1) ;
DISPLAY ( DLINE ) ;
END SUMMARY ;

CLOSE FILE (MREF), FILE (ID), FILE (RESU),
FILE (COURSE), FILE (DCDP),
FILE (VEHS), FILE (DVEH),
FILE (TASKS), FILE (DTASK) ;
IF ( PTR = 1 ) THEN CLOSE FILE (SYPRIINT) ;
DISPLAY ( "=" "=" );
END P5A ;
TAEG REPORT NO. 40

SOURCE CARD LISTING FOR P5B: /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */

PROC OPTIONS (MAIN);

/* REVISIONS:
  FRI/11/MAR/77 - VERYFIRST IMPLEMENTATION.
  MON/21/MAR/77 - MODS PRIOR TO TRANSMIT TO NCSS. */

DECLARE (TOTREC, MOD, N, N1, N2, L1, L2, L3, NREC, NRECA, NADD,
  RLINE, LEE, N, J2, J3, J4, MAXADD, MAXD, MAXR,
  NCONV, NER, NFIRST, NN, NOS,
  J, K, L ) FIXED BIN STATIC;

DECLARE (CARD, DLINE ) CHAR (80) STATIC,
C110  CHAR (10) DEF CARD POS (1),
C1180  CHAR (70) DEF CARD POS (11),
C15  CHAR (5) DEF CARD,
C16  CHAR (2) DEF CARD,
C17  CHAR (7) DEF CARD,
CTAB (80)  CHAR (1) DEF CARD,

(NIC, N2C ) CHAR (5) STATIC,
( ROUTEX, V1(3),
  ROUTE ) LABEL ;

DECLARE (COURSE, VEHS, TASKS ) FILE RECORD DIRECT KEYED ENV (REGIONAL(1));

DECLARE ADDS (500)  CHAR (16) STATIC,
NEW  CHAR (16) STATIC,
NTYP  PICTURE '9' DEF NEW POS (1),
NSTAT  PICTURE '9' DEF NEW POS (2),
NDSTAT  PICTURE '9' DEF NEW POS (3),
NID  CHAR (13) DEF NEW POS (4),
NCIN  CHAR (8) DEF NEW POS (4),
NCDP  CHAR (4) DEF NEW POS (12),
NRATE  CHAR (4) DEF NEW POS (4),
NRANK  CHAR (2) DEF NEW POS (9),
NJTASK  CHAR (6) DEF NEW POS (11),

1 MODS (500) BASED (PM),
  2 MTYPE  CHAR (1),
  2 MSTAT  CHAR (1),
  2 MODSTAT  CHAR (1),
  2 FILL  CHAR (13);

DECLARE 1 RESULT (1200) STATIC,
2 RTYP  PICTURE '9',
2 RSTAT  PICTURE '9',
2 RDSTAT  PICTURE '9',
2 RESID  CHAR (13),

RESREC (1200)  CHAR (16) BASED (PR),
S16  CHAR (16) STATIC,
1 RESC (1200) BASED (PR),
  1 F1, 2 F2, 2 F3  CHAR (1),
  2 RESCIN  CHAR (8),
  2 RESPDP  CHAR (4),
  2 F4  CHAR (1),

1 REST (1200) BASED (PR),
  1 F1, 2 F2, 2 F3  CHAR (1),
  2 RATE  CHAR (5),
  2 RANK  CHAR (2),
SOURCE CARD LISTING FOR
P5B: /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */

DECLARE 1 MAST STATIC;
DECLARE 1 REFS (40) FIXED BIN;

DECLARE 1 TTLCARD STATIC /* PROJECT TITLE DATA BASE */
DECLARE 1 PNAME CHAR (10), /* CKF.TITLE */
DECLARE 1 PTTL CHAR (70);

INPARAM CHAR (100) VARYING,

DECLARE 1 POWER BASED (PP),
DECLARE 1 PARM1 CHAR (4),
DECLARE 1 PARM2 CHAR (4);

DECLARE 1 CINDER (304) STATIC, /* CRS CIN DIRECTORY */
DECLARE 1 CHCIN CHAR (8), /* CKF.CRS.DIRCIN */
DECLARE 1 CHCINPTR FIXED BIN,
DECLARE 1 COVCIN (4) CHAR (760) BASED (PLCIN),
DECLARE 1 COVCINPTR BASED (PTCIN);

DECLARE 1 CDENDOR (1200) STATIC, /* CRS CDP DIRECTORY */
DECLARE 1 CHCDP CHAR (4), /* CKF.CRS.DIRCDP */
DECLARE 1 CHCDPTR FIXED BIN,
DECLARE 1 LOADCDP (1200) BASED (PLCDP),
DECLARE 1 LOADCDP (1200) BASED (PTCDP);

DECLARE 1 ACRS STATIC, /* ABBR. COURSE FILE */
DECLARE 1 ACRS (4) CHAR (4), /* CKF.ACOURSES */
DECLARE 1 SET1 CHAR (4),
DECLARE 1 SET2 CHAR (4),
DECLARE 1 SET3 CHAR (4),
DECLARE 1 PC CHAR (2),
DECLARE 1 RMS CHAR (3),
DECLARE 1 TYCRS CHAR (2),
DECLARE 1 SVC CHAR (1),
DECLARE 1 MI CHAR (1),
DECLARE 1 STCD CHAR (1),
DECLARE 1 STDTE PICTURE "(5)99",
DECLARE 1 TRAPS CHAR (1),
DECLARE 1 TPC CHAR (5),
DECLARE 1 ATTR PICTURE "99999", 
DECLARE 1 STBK PICTURE "99999", 
DECLARE 1 CLEN PICTURE "99999",
DECLARE 1 THRS PICTURE "99999",
DECLARE 1 LRS PICTURE "99999",
DECLARE 1 CAJR PICTURE "(5)99999",
DECLARE 1 CGBO CHAR (1);

DECLARE 1 VEHDIR (200) STATIC, /* VEHICLE DIRECTORY */
DECLARE 1 CHVEH CHAR (13), /* CKF.VEH.DIR */
DECLARE 1 CHVEH (4) CHAR (13),
DECLARE 1 VEHPRTR FIXED BIN,
DECLARE 1 COVVEH (4) CHAR (750) BASED (PTVEH),
DECLARE 1 LOADVEH (750) BASED (PLVEH);

DECLARE 1 AVEHICLES STATIC, /* ABBR. VEH FILE */
DECLARE 1 STICKN CHAR (13), /* CKF.AVEHS */
DECLARE 1 DEVDESIG CHAR (9),

D.6-43
DECLARE 1 TASKDIR (1200) STATIC, /* JOB TASK FILE DIRECTORY */
DECLARE 1 ATASKS STATIC, /* ABR. TASK FILE */

DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
2 CC80 CHAR (1),
2 CC80 CHAR (1),
DECLARE 1 TASKDIR (1200) STATIC,
SOURCE CARD LISTING FOR
P5B: /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */

DISPLAY ( CARD ) ;
DISPLAY ( " " ) ;

/******************************************************************************/
/* FIRST, FETCH THE RESULTS FILE INTO MS */

ON ENDFILE ( RESU ) GO TO LOAD2 ;

LOADR: DO J = 1 TO 1200 ;
READ FILE ( RESU ) INTO ( S16 ) ;
TOTREC = J ;
RESREC ( J ) = S16 ;
END LOADR ;

ER1: NER = 1 ;
DLINE = " MORE THAN 1200 RESULTS RECORDS - CANNOT EDIT" ;
GO TO BUSTED ;

LOAD2: CLOSE FILE ( RESU ) ;

PUT STRING ( DLINE ) EDIT
("*** TOTREC, HAVE BEEN LOADED INTO " ,
"MEMORY FOR EDITING") ( A, F(4), A, A ) ;
DISPLAY ( " " ) ; DISPLAY ( DLINE ) ;

/******************************************************************************/
/* HERE IS CENTRAL PROMPT LOOP AND REPLY-LOOKUP SEQUENCE */

PROMPTA: DISPLAY ( " " ) ;
DISPLAY ( " *** MAKE NEXT ACTION REQUEST" )
REPLY ( CARD ) ;
IF ( CARD = " " ) THEN GO TO PROMPTA ;
IF ( CTAB(1) = " " ) THEN GO TO ELOOK ;
LP1: DO J = 1 TO 20 ;
IF ( CTAB(J) = " " ) THEN GO TO ELPI ;
CARD = SUBSTR ( CARD, J ) ;
GO TO ELOOK ;
ELPI: END LP1 ;

BUMENTRY: DISPLAY ( " " ) ;
/** CANNOT DECODE LAST ENTRY */
" - TRY AGAIN" ) ;
GO TO PROMPTA ;

ELOOK: IF ( C12 = 'D ' ) | ( C17 = 'DELETE ' )
THEN GO TO DELETE ;
IF ( C12 = 'P ' ) | ( C16 = 'PRINT ' )
THEN GO TO PRINT ;
IF ( C15 = 'ADDC ' ) THEN GO TO ADDCOURSE ;
IF ( C15 = 'ADDV ' ) THEN GO TO ADDVEHICLE ;
IF ( C15 = 'ADDT ' )
THEN GO TO ADDJOBJTASK ;
IF ( C15 = 'QUIT ' ) THEN GO TO QUIT ;
GO TO BUMENTRY ;

/******************************************************************************/
/* THIS IS PROCESSOR FOR SINGLE / MULTIPLE DELETES */

DELETE: ROUTE = STEPI ;
GO TO FINDNOS ;
**TAEG REPORT NO. 40**

**SOURCE CARD LISTING FOR**

PS5B: /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */

**PAGE NUMBER :  5**

STEP1: \( K = N1 \);

276 \[ IF ( N2 = 0 ) THEN N2 = N1 ; \]

277

278

279

STEP2: \( \text{ROUTE} = \text{STEP3} ; \)

280 \[ \text{GO TO FINDREC} ; \]

281

282

283

STEP3: \( \text{IF ( K > TOTREC ) } \)

284 \[ \text{THEN DO } ; \text{MTYP} (NRECA) = '0' ; \text{END} ; \]

285 \[ \text{ELSE DO } ; \text{RDSTAT} (NREC) = 3 ; \text{END} ; \]

286

287

288

STEP4: \( \text{DISPLAY ( DLINE ) ;} \)

289

STEP5: \( \text{MOD} = 1 ; \) /* SIGNAL, FILE HAS BEEN CHANGED */

290 \[ \text{GO TO PROMPTA} ; \]

291

292

293

STEP6: \( \text{ROUTE} = \text{STEP6} ; \)

294 \[ \text{GO TO FINDNOS} ; \]

295

296

297

STEP7: \( \text{ROUTE} = \text{STEP8} ; \)

298 \[ \text{GO TO FINDREC} ; \]

299

300

301

STEP8: \( \text{IF ( K > TOTREC ) } \)

302 \[ \text{THEN NEW = ADDS (NRECA) ;} \]

303 \[ \text{ELSE NEW = RESREC (NREC) ;} \]

304

305

306

STEP9: \( \text{YECDP} = \text{NCIN} \);

307 \[ \text{LOOKU = STEP10} ; \]

308 \[ \text{GO TO FINDCDP} ; \]

309

310

311

STEP10: \( \text{IF ( COPREC > 0 ) } \)

312 \[ \text{THEN GO TO STEP10} ; \]

313 \[ \text{SUBSTR (DLINE,45) = 'CDP NOT IN DIRECTORY' ;} \]

314 \[ \text{GO TO OUTLOOP} ; \]

315

316

317

STEP11: \( \text{MAXD} = \text{REFS (2)} ; \) /* MAX SS USED IN DIRECTORY */

318 \[ \text{MAXR} = \text{REFS (1)} ; \) /* MAX COURSE REC NUM */

319 \[ \text{NLINE} = 1 ; \) /* ALWAYS 1 LINE WITH COURSES */

320

321

322

STEP12: \( \text{SUBSTR (DLINE,5) = 'NO MATCHING DATA'} ; \)

323 \[ \text{GO TO OUTLOOP} ; \]

324

325

326

STEP13: \( \text{SUBSTR (DLINE,45) = 'NO MATCHING DATA'} ; \)

327 \[ \text{GO TO OUTLOOP} ; \]

328

329

330

STEP14: \( \text{SUBSTR (DLINE,45) = 'NO MATCHING DATA'} ; \)

331 \[ \text{GO TO OUTLOOP} ; \]

332

333

334

STEP15: \( \text{SUBSTR (DLINE,45) = 'NO MATCHING DATA'} ; \)

335 \[ \text{GO TO OUTLOOP} ; \]

336

337

338

STEP16: \( \text{SUBSTR (DLINE,45) = 'NO MATCHING DATA'} ; \)

339 \[ \text{GO TO OUTLOOP} ; \]

340

341

342

STEP17: \( \text{SUBSTR (DLINE,45) = 'NO MATCHING DATA'} ; \)

343 \[ \text{GO TO OUTLOOP} ; \]

344

D.6-46
TAEG REPORT NO. 40

PAGE NUMBER : 6

SOURCE CARD LISTING FOR
P5B: /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */

STEP10: IF ( CDPREC <= MAXR ) THEN GO TO STEP11;
   ER3: NER = 3;
   PJT STRING (DLINE) DATA (CDPREC, MAXR, K);
       GO TO BUSTED;

STEP11: READ FILE (COURSE) INTO (ACRS) KEY (CDPREC);
   SUBSTR (DLINE,45) = CST;
       GO TO OUTLOOP;

    /* HERE, VEHICLE RECORD PRINTING IS SETUP */

V1(2): MAXD = REFS (8);
    MAXR = REFS (7);
    NLINE = 1;
    NN = 46;

    PUT STRING (DLINE) EDIT ( 'RECN/STOCK/FILE NUM : ' , K,
    ' ' , NID, ' ' ) ( A, P'ZZ,ZZ9', A, A(13), A ) ;

    IF ( NDSTAT = 1 ) THEN GO TO STEP12 ;

STEP12A: SUBSTR (DLINE,NN) = ' (NO MATCHING DATA) ' ;
    GO TO OUTLOOP;

STEP12: YEVEH = NID;
    LOOKU = STEP13;
    GO TO FINDVEH;

STEP13: IF ( VEHREC > 0 ) THEN GO TO STEP14 ;
    SUBSTR (DLINE,NN) = 'VEHICLE NOT IN DIRECTORY' ;
    GO TO OUTLOOP;

STEP14: IF ( VEHREC <= MAXR ) THEN GO TO STEP15 ;
    ER4: NER = 4;
    PJT STRING (DLINE) DATA (VEHREC, MAXR, K);
    GO TO BUSTED;

STEP15: READ FILE (VEHS) INTO (AVEHICLES) KEY (VEHREC);

    SUBSTR (DLINE,45) = DEVDESIG ;
    CARD = ' ' ;
    SUBSTR (CARD,21) = DEVNAME ;

STEP15A: NLINE = 2 ;
    GO TO OUTLOOP;

    /* HERE, FINALLY, JOBTASK PROCESSING ENSUES */

V1(3): MAXD = REFS (12);
    MAXR = REFS (11);
    NLINE = 1;
    NN = 47;

    PUT STRING (DLINE) EDIT ( 'RECN/RATE/RANK/JOB : ' , K,

    IF ( NDSTAT = 2 ) THEN GO TO STP12A ;

STEP16: IF ( TASKREC > 0 ) THEN GO TO STEP17 ;
    SUBSTR (DLINE,47) = 'JOBTASK NOT IN DIRECTORY' ;
    GO TO OUTLOOP;

STEP17: IF ( TASKREC <= MAXR ) THEN GO TO STEP18 ;
ER5: NER = 5 ;
    PUT STRING ( DLINE ) DATA ( TASKREC, MAXR, K ) ;
    GO TO BUSTED ;

STEP18: READ FILE ( TASKS ) INTO ( ATASKS ) KEY ( TASKREC ) ;
    CARD = ' ' ;
    SUBSTR ( CARD, 10 ) = TASKTTL ;
    GO TO STP15A ;

/* HERE IS THE COMMON OUTPUT SEQUENCE FOR COURSES, VEHICLES,
   AND TASKS RECORD PRINT */

OUTLOOP: DISPLAY ( ' ' ) ;
    DISPLAY ( DLINE ) ;
    IF ( NLINE = 2 ) THEN
        DISPLAY ( CARD ) ;
        K = K + 1 ;
        IF ( K <= N2 ) THEN GO TO STEP7 ;
    DISPLAY ( ' ' ) ;
    GO TO PROMPTA ;

/************************************************************************/

/* HERE WE HAVE THE PROCESSING ROUTINES FOR THE ADD-COURSES
   (ADDC) KEYBOARD ENTRY */

ADDCOURSE:
ADDC: IF ( MAXADD > 0 ) THEN GO TO BUMADD ;
    MAXD = REF2 ( 2 ) ;
    MAXR = REF2 ( 1 ) ;
    NFIRST = 1 ;
    NLINE = 1 ;
    ROUTEX = PROMPTA ;
    ROUTE = STEP19 ;
    GO TO FINDASLASH ;

STEP19: YECIN = SUBSTR ( CARD, (J1+1), (J2-J1-1) ) ;
    YECOP = ' ' ;
    IF ( J3 = 0 ) | ( (J3 - J2) = 1 ) THEN GO TO STP19A ;
    YECOP = SUBSTR ( CARD, (J2 + 1), (J3 - J2 - 1) ) ;

STP19A: NINL = 1 ;
    NSTAT = 2 ;
    NID = ' ' ;
    NCDP = YECIN ;
    NCDP = YECOP ;
    NADD = NADD + 1 ;
    IF ( NADD <= 500 ) THEN GO TO STEP20 ;

BUMADD: NADD = NADD - 1 ;
    MAXADD = NADD ;
    DLINE = ' A TOTAL OF ' || NADD || ' NEW RECORDS HAVE ' || 'BEEN ADDED' ;
    DISPLAY ( ' ' ); DISPLAY ( DLINE ) ;
    DLINE = ' ANOTHER RUN OF THIS ROUTINE MUST BE MADE ' || 'TO ENTER FURTHER ADDITIONS' ;
    DISPLAY ( DLINE ) ;
    GO TO PROMPTA ;

STEP20: NREC = TOTREC + NADD ;
SOURCE CARD LISTING FOR
PSB: /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */

DLINE = 'THE FOLLOWING RECORD(S) HAS/HAVE BEEN ADDED...';
DISPLAY (' '); DISPLAY (DLINE);
PUT STRING (DLINE) EDIT ( (10)
  NREC, YECIN, YEC, YECIN.
  A, A, A, A, A(8), A, A(4), A ) ;
IF ( YEC = ' ' ) THEN GO TO STEP23 ;

/* HERE, THE ENTERED CDP WAS NOT BLANK, LOOK IT UP */
LOOKU = STEP21 ;
GO TO FINDCDP ;
STEP21: IF ( CDPREC > 0 ) THEN GO TO STEP22 ;
STEP21A: SUBSTR (DLINE, 55) = '(NO MATCHING DATA)' ;
GO TO ADDOUT ;
STEP22: IF ( CDPREC = MAXR ) THEN GO TO ER3 ;
READ FILE (COURSE) INTO (ACRS) KEY (CDPREC) ;
SUBSTR (DLINE, 55) = CST ;
NDSTAT = 1 ;
GO TO ADDOUT ;
/* HERE, THE GIVEN CDP WAS BLANK ; TRY CDP LOOKUP */
STEP23: LOOKU = STEP24 ;
GO TO FINDCIN ;
STEP24: IF ( CINREC = 0 ) THEN GO TO STEP21A ;
ROUTEX = STEP26 ;
IF ( CINREC = MAXR ) THEN GO TO ER3 ;
STEP25: READ FILE (COURSE) INTO (ACRS) KEY (CINREC) ;
IF ( CIN = YECIN ) THEN GO TO PROMPTA ;
SUBSTR (DLINE, 49) = CDP ;
YEC = CDP ;
YEC = YEC ;
IF ( NFIRST = 1 ) THEN:
DO : NFIRST = 1 ;
ADD = ADD + 1 ;
END :
NADD = NADD + 1 ;
IF ( NADD > 500 ) THEN GO TO BUMADD ;
NKEC = TOTREC + NADD ;
SUBSTR (SUBSTR(DLINE, 33, 6)) EDIT (NREC) ( P'ZZ, ZZ9
  /* HERE IS THE COMMON OUTPUT ROUTINE FOR ADDITIONS */
ADDOUT: DISPLAY (' '); DISPLAY (DLINE) ;
IF ( NLINE = 2 ) THEN:
DISPLAY (CARD) ;
ADD (NADD) = NEW ;
GO TO ROUTEX ; /* NORMAL ROUTE IS TO PROMPTX */
STEP26: /* HERE, FROM ROUTEX, WHEN POSSIBLY ANOTHER CIN/CDP
RECORD TO BE EXTRACTED */
SOURCE CARD LISTING FOR
PSB: /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */

CINREC = CINREC + 1;
IF ( CINREC > MAXR ) THEN GO TO PROMPTA;
ELSE GO TO STEP25;

***************************************************************
/* NOW, THE PROCESSING ENSUES FOR ADDITIONS OF VEHICLE RECORDS */

ADDVEHICLE:

ADDV: IF ( MAXADD > 0 ) THEN GO TO BUMADD;

ROUTEX = PROMPTA;
NLINE = 1;
MAXD = REF$(8);
MAXR = REF$(7);

ROUTE = STEP27;
GO TO FINDASLASH;

STEP27: YEVEH = SUBSTR( CARD, (J1 + 1), (J2 - J1 - 1) ) ;
NTYP = 2; NSTAT = 2; NDSTAT = 2;
NID = YEVEH;
NADD = NADD + 1;
IF ( NADD > 500 ) THEN GO TO BUMADD;
NREC = TOTREC + NADD;
DLINE = 'THE FOLLOWING RECORD HAS BEEN ADDED...';
DISPLAY( DLINE ) ;
DISPLAY( DLINE ) ;
PUT STRING( DLINE ) EDIT( (10) ' ', 'RECN0/STOCK/FILE NUM : ',
N*EC, '/ ', YEVEH, '/' ) ( A, A, P'ZZ, ZZ9', A, A ) ;
LOOKU = STEP28;
GO TO FINDVEH;

STEP28: IF ( VEHREC > 0 ) THEN GO TO STEP29;
SBJSTR( DLINE, 55 ) = 'NO MATCHING DATA';
GO TO ADDOUT;

STEP29: IF ( VEHREC > MAXR ) THEN GO TO ER4;
READ FILE( VEHS ) INTO (AVEHICLES) KEY (VEHREC);
SUBSTR( DLINE, 55 ) = DEVDESIG;
CARD = ' ';
SUBSTR( CARD, 21 ) = DEVNAME;
NDSTAT = 1;
NLINE = 2;
GO TO ADDOUT;

***************************************************************
/* FINALLY, THE TASK ADDITION KEYBOARD ENTRY IS PURSUED */

ADDOBJTASK:

ADDJ: IF ( MAXADD > 0 ) THEN GO TO BUMADD;

ROUTEX = PROMPTA;
NLINE = 1;
MAXD = REF$(12);
MAXR = REF$(11);

ROUTE = STEP30;
GO TO FINDASLASH;

STEP30: IF ( J1 * J2 * J3 * J4 ) = 0 THEN GO TO BUMENTRY;
NTYP = 3 ; NSTAT = 2 ; NDSTAT = 2 ;
NRATE = SUBSTR ( CARD, (J1 + 1), (J2 - J1 - 1) ) ;
NRANK = SUBSTR ( CARD, (J2 + 1), (J3 - J2 - 1) ) ;
NJTASK = SUBSTR ( CARD, (J3 + 1), (J4 - J3 - 1) ) ;
YETASK = NID ;

NADD = NADD + 1 ;
IF ( NADD > 500 ) THEN GO TO BUMADD ;
NREC = TOTREC + NADD ;

DLINE = 'THE FOLLOWING RECORD HAS BEEN ADDED...';
DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;

PUT STRING (DLINE) EDIT
( {10} : 'RECN/RATE/RANK/JOB :', NREC,
'/' : NRATE, '/' : NRANK, '/' : NJTASK )
( A, A, 10Z2, ZZ9, 6 ( A ) ) ;

LOOKU = STEP31 ;
GJ TO FINDTASK ;

STEP31: IF ( TASKREC > 0 ) THEN GO TO STEP32 ;
SJBSTR ( DLINE, 56 ) = 'NO MATCHING DATA';
GO TO ADDOUT ;

STEP32: IF ( TASKREC > MAXR ) THEN GO TO ER5 ;
READ FILE (TASKS) INTO (ATASKS) KEY (TASKREC) ;
CARD = ' ' ;
SUBSTR ( CARD, 20 ) = TASKTTL ;
NOSTAT = 1 ;
NLINE = 2 ;

GO TO ADDOUT ;

FINDASLASH:
J1, J2, J3, J4 = 0 ;
DLINE = CARD ;

J1 = INDEX (CARD, '/');
IF ( J1 = 0 ) THEN GO TO BUMENTRY ;
CTAB (J1) = ' ' ;

J2 = INDEX (CARD, '/');
IF ( J2 <= J1 ) THEN GO TO BUMENTRY ;
CTAB (J2) = ' ' ;

J3 = INDEX (CARD, '/');
IF ( J3 = 0 ) THEN GO TO SLRET ;
IF ( J3 <= J2 ) THEN GO TO BUMENTRY ;
CTAB (J3) = ' ' ;

J4 = INDEX (CARD, '/');
IF ( J4 = 0 ) THEN GO TO SLRET ;
IF ( J4 <= J3 ) THEN GO TO BUMENTRY ;
SLRET: CARD = DLINE ;
GO TO ROUTE ;

/* HERE, IS SUBROUTINE TO DETERMINE LEGALITY OF AN INPUT
RECORD NUMBER.
ON ENTRY - NUMBER TO BE VERIFIED IS IN K
ON EXIT - NREC = SS TO RESULTS STRUCTURE
NRECA = SS TO ADDS STRUCTURE
ERROR EXIT TO BUMENTRY
RETURN VECTOR IS ROUTE */

FINDREC:
IF ( K <= (TOTREC + NADD) ) THEN GO TO FINDRI ;
BUMNUMBER:
DLINE = (5) ;
** RANGE ERROR IN RECORD NUMBER ;
DISPLAY ( ) ;
DISPLAY ( DLINE ) ;
GO TO PROMPTA ;
FINDRI: IF ( K <= TOTREC )
THEN DO ; NREC = K ; NRECA = 0 ; END ;
ELSE DO ; NREC = 0 ; NRECA = K - TOTREC ; END ;
GO TO ROUTE ;

/* THIS IS THE SUBROUTINE WHICH ISOLATES THE NUMERIC
SINGLE/PAIRS FROM THE DELETE & PRINT KEYBOARD ENTRIES
ON ENTRY - NO REQUIREMENTS
ON EXIT - RETURN VECTOR IS ROUTE
FIRST NUMBER BIN IN N1, CHAR IN NIC
2ND NUM (IF ANY) BIN IN N2, CHAR IN N2C
2ND, IF ABSENT, N2 = 0
- NOS = 1 TO 2, INDICATES SING/PAIR INPUTS
ERROR RETURN TO PROMPTA (BAD NUM) OR BUMENTRY */

FINDNOS:
N1, N2 = 0 ;
NIC, N2C = ;
NCONV = 2 ;
NOS = 1 ;
L1 = INDEX ( CARD, ) ;
* FIND FIRST BLANK */
IF ( L1 = 0 ) | ( L1 > 20 )
| ( SUBSTR(CARD, L1) = )
THEN GO TO BUMENTRY ;
LFN1:
DO L = L1 TO 80 ;
* FIND FIRST DIGIT */
L2 = L ;
IF ( CTAB(L) = )
THEN GO TO FINDN2 ;
END LFN1 ;
GO TO BUMENTRY ;
FINDN2:
DO L = L2 TO 80 ;
L1 = L ;
IF ( CTAB(L) = )
THEN GO TO FINDN3 ;
END FINDN2 ;
GO TO BUMENTRY ;
FINDN3: NIC = SUBSTR ( CARD, L2, (L1-L2) ) ;
GET STRING (NIC) EDIT (N1) ( F5 ) ;
IF ( L1 > 79 ) THEN GO TO BUMENTRY;

LFN2:
DO L = L1 TO 79;
L2 = L;
IF ( CTAB(L) ^= ' ' ) AND ( CTAB(L) ^= ' ' )
THEN GO TO FINDN4;
END LFN2;
GO TO ROUTE;

FINDN4: IF ( L2 > 75 ) THEN L3 = 80;
ELSE L3 = L2 + 5;

LFN3:
DO L = L2 TO L3;
L4 = L;
IF ( CTAB(L) ^= ' ' ) THEN GO TO FINDN5;
END LFN3;

FINDN5: N2C = SUBSTR ( CARD, L2, (L1-L2) );
GET STRING ( N2C ) EDIT ( N2 ) ( F(5) ) ;
IF ( N2 < N1 ) THEN GO TO BUMENTRY;
LFNOUT: NOS = 2;
GO TO ROUTE;

/*****************************/
/ * DIRECTORY LOOKUP SUBROUTINES IN THIS SECTION */
DECLARE ( FLCIN, FLCDP, FLVEH, FLTASK ) BIT (1) STATIC,
YECIN, CHAR (8) STATIC,
YECDP, CHAR (4) STATIC,
YETASK, YEVEH, CHAR (13) STATIC,
CINREC, CPDREC, VEHREC, TASKREC, VEHMAX, TASKMAX,
CINMAX, CPDMAX ) FIXED BIN STATIC,
LOOKU LABEL ;

/* CIN DIRECTORY LOOKUP SUBROUTINE
ENTRY : DESIRED CIN IN YECIN
EXIT : CINREC = CINPTR ( I ) (NO FIND), OR REG NUMBER
RETURN VECTOR IS LOOKU */
FINDCIN: IF ( FLCIN ) THEN GO TO CIN2;
CINMAX = REFS (2);
FLCIN = 'I' B ;
CINLOAD:
D3 J = 1 TO 4;
PLCIN = ADDR ( COVCIN ( J ) ) ;
READ FILE ( DCIN ) INTO ( LOADCIN ) ;
END CINLOAD ;

CIN2:
DO I = 1 TO CINMAX;
IF ( YECIN >= CHCIN(I) ) THEN GO TO ECIN2;
IF ( YECIN ^= CHCIN(I) ) THEN GO TO NFCIN;
CINREC = CINPTR ( I ) ;
GO TO LOOKU ;

ECIN2:
END CIN2 ;

NFCIN: CINREC = 0;
GO TO LOOKU ;

/ * CDP DIRECTORY LOOKUP SUBROUTINE
ENTRY : DESIRED CPD IN YECDP
EXIT : CPDREC = 0 (NO FIND), OR REC NUMBER
RETURN VECTOR IS LOOKU */
TAEG REPORT NO. 40

SOURCE CARD LISTING FOR P5B: /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */

FINDCDP: IF ( FLCDP ) THEN GJ TO CDP2 ;
CDPMAX = REF ( 3 ) ;
FLCDP = '1'B ;

CDPLOAD: DO J = 1 TO 4 ;
PLCDP = ADDR ( COVCDP ( J ) ) ;
READ FILE ( OCPD ) INTO ( LOADCDP ) ;
END CDPLOAD ;

CDP2: DO I = 1 TO CDPMAX ;
IF ( YECDP > CHCDP ( I ) ) THEN GO TO ECNP2 ;
IF ( YEDCP = CHCDP ( I ) ) THEN GO TO NFCDP ;
CDPREG = CDPPTR ( I ) ;
GO TO LOOKU ;

ECNP2: END CDP2 ;

NFCDP: CDPREC = 0 ;
GO TO LOOKU ;

/* VEHICLE ID LOOKUP SUBROUTINE ENTRY: DESIRED VEH ID IN YEVEH EXIT: VEHREC = 0 (NO FIND), OR REC NUMBER RETURN VECTOR IS LOOKU */

FINDVEH: IF ( FLVEH ) THEN GO TO VEH2 ;
VEHMAX = REF ( 8 ) ;
FLVEH = '1'B ;

VEHLOAD: DO J = 1 TO 4 ;
PLVEH = ADDR ( COVVEH ( J ) ) ;
READ FILE ( OVCHEH ) INTO ( LOADVEH ) ;
END VEHLOAD ;

VEH2: DO I = 1 TO VEHMAX ;
IF ( YEVEH > CHVEH ( I ) ) THEN GO TO EVEH2 ;
IF ( YEVEH = CHVEH ( I ) ) THEN GO TO NFVEH ;
VEHREC = VEHPR ( I ) ;
GO TO LOOKU ;

EVEH2: END VEH2 ;

NFVEH: VEHREC = 0 ;
GO TO LOOKU ;

/* TASK ID DIRECTORY LOOKUP SUBROUTINE ENTRY: DESIRED TASK ID (13-CHAR) IN YETASK EXIT: TASKREC = 0 (NO FIND) OR REC NUMBER RETURN VECTOR IS LOOKU */

FINDTASK: IF ( FLTASK ) THEN GO TO TASK2 ;
TASKMAX = REF ( 12 ) ;
FLTASK = '1'B ;

TASKLOAD: DO J = 1 TO 12 ;
PLTASK = ADDR ( COVTASK ( J ) ) ;
READ FILE ( OVTASK ) INTO ( LOADTASK ) ;
END TASKLOAD ;

TASK2: DO I = 1 TO TASKMAX ;
IF ( YETASK > CHTASK ( I ) ) THEN GO TO ETASK2 ;
IF ( YETASK = CHTASK ( I ) ) THEN GO TO NFTASK ;
TASKREC = TASKPR ( I ) ;
GO TO LOOKU ;

ETASK2: END TASK2 ;

NFTASK: TASKREC = 0 ;
**SOURCE CARD LISTING FOR**

**PSB:** /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */

GO TO LOOKU;

//*******************************/

/* TERMINAL, IE, CATASTROPHIC BADNESSES REPORTED HERE */

BUSTED: PUT STRING (CARD) EDIT ( {10} ' ' ', ' ** ERROR NUMBER ' ,
   ' NER, ' HAS OCCURRED ' ' )
   ( A, A, F(12), A ) ;
   DISPLAY ( ' ' ) ; DISPLAY ( CARD ) ;
   DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;

//*******************************/

/* HERE, REE RESULTS PROJECT FILE IS COMPOSED FROM THE INPUT
RESULTS ARRAY AND ANY ADDS ARRAY ENTRIES */

QUIT: OPEN FILE (RESU) RECORD OUTPUT;
FILE (REE) RECORD OUTPUT ;

RELOADR: DO J = 1 TO TOTREC ;
   S16 = RESREC (J) ;
   WRITE FILE (RESU) FROM (S16) ;
   IF ( RDSTAT(J) = 3 ) THEN GO TO ELOADR ;
   WRITE FILE (REE) FROM (S16) ;
   NEE = NEE + 1 ;

ELOADR: END RELOADR ;

IF ( NADD <= 0 ) THEN GO TO EOJ ;

RELOADA: DO J = 1 TO NADD ;
   IF ( MSTAT(J) = '38' | (MTYP(J) = '08' ) ) THEN GO TO ERELOADA ;
   NEW = ADDS (J) ;
   WRITE FILE (RESU) FROM (NEW) ;
   WRITE FILE (REE) FROM (NEW) ;
   NEE = NEE + 1 ;

ERELOADA: END RELOADA ;
   GO TO EOJ ;

//*******************************/

/* HERE WE ARE AT THE LAST ROUNDUPT */

EOJ: OLINE = ' ** ROE SEARCH RESULTS EDITOR IS TERMINATING' ;
   DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;
   PUT STRING (DLINE) EDIT ( ' TOTAL RECORDS ADDED : ' , NADD )
   ( A, F(3) ) ;
   DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;
   PUT STRING (DLINE) EDIT ( ' TOTAL RECORDS IN PROJECT EXTRACT FILE : ' ,
   ' NEE ' ) ( A, P(1,229) ) ;
   DISPLAY ( ' ' ) ; DISPLAY ( DLINE ) ;
   CLOSE FILE (MREF), FILE (ID),
   FILE (COURSE), FILE (DLINE), FILE (DCDP),
   FILE (VEHS), FILE (DVEH),
   FILE (TASKS), FILE (DTASK),
   FILE (RESU), FILE (REE) ;
SOURCE CARD LISTING FOR
P5B: /* INTERACTIVE SEARCH RESULTS EDITOR - WED/09/MAR/77 */
END P5B ;
TAE6 REPORT NO. 40

SOURCE CARD LISTING FOR
P5C: /* PRINT OF ROE SEARCH RESULTS FOR EXTRACT — FRI/11/MAR/77 */

PROC (INPARM) OPTIONS (MAIN):

/*
FRI/11/MAR/77 — VERYFIRST IMPLEMENTATION.

THU/24/MAR/77 — MODS PRIOR TO TRANSMIT TO NCSS.
*/

DECLARE DLINE8 CHAR (8) DEF DLINE POS (8),
DLINE28 CHAR (4) DEF DLINE POS (28),
DLINE33 CHAR (36) DEF DLINE POS (33),
DLINE36 CHAR (36) DEF DLINE POS (36),
DLINE40 CHAR (40) DEF DLINE POS (40),
CARD7 CHAR (70) DEF CARD POS (8),

NODIR CHAR (35) STATIC INIT
( "** DIRECTORY SEARCH WAS NEGATIVE **) ,

NOAB CHAR (36) STATIC INIT
( "(NO ABBREVIATED FILE DATA AVAILABLE)"
),

DECLARE 1 TTLCARD STATIC; /* PROJECT TITLE DATA BASE */
2 PNAME CHAR (10), /* CKF.ID */
2 PTTL CHAR (70),

INPARM CHAR (100) VARYING,
1 POVER BASED (PP),
2 PARM2 CHAR (4),

( DISP, PTR, DPAGE, I, J, K, L, LT, NC, NFIRST,
  NN, NPAGE, NR, NTYPE, NV, NER, NLINE,
  NOF, NJ ) FIXED BIN STATIC,

DTTL CHAR (23) STATIC INIT
( PAGE NUMBER : XXXX ),
DDPGE "PICTURE 'ZZZ9' DEF DTTL POS (20),
MAXLTB FIXED BIN STATIC INIT ( 60 ) ;

DECLARE ( COURSE, VEHS, TASKS ) FILE RECORD DIRECT KEYED ENV ( REGIONAL(1) ) ;

DECLARE ( CARD, DLINE ) CHAR (80) STATIC,
C110 CHAR (10) DEF CARD,
C1180 CHAR (70) DEF CARD POS (11) ;

DECLARE 1 MAST STATIC,
2 REFS (40) FIXED BIN,

1 RESULT BASED (PR),
2 RTYP PICTURE '9',
2 RSTAT PICTURE '9',
2 RDSTAT PICTURE '9',
2 RESID CHAR (13),

1 RESC BASED (PR),
( 2 F1, 2 F2, 2 F3 ) CHAR (1),
2 RESCPN CHAR (8),
2 RESCDP CHAR (4),

1 REST BASED (PR),
( 2 F1, 2 F2, 2 F3 ) CHAR (1),
2 RATE CHAR (5),
2 RANK CHAR (2),
2 JOBTASK CHAR (6),
SOURCE CARD LISTING FOR
P5C: */# PRINT. OF ROE SEARCH RESULTS FOR EXTRACT - FRI/11/MAR/77 */

DECLARE T1  CIN  CDP  CHAR (32) STATIC INIT (  
  'RECONO  CIN  CDP' ),

DECLARE T2  STOCK NUMBER  FILE NUM TRAINING DEVICE* ),

DECLARE T3  CHAR (29) STATIC INIT (  
  'RECONO  RATE  RANK  JOB  TASK*'),

DECLARE WINDUPS (6)  CHAR (44) STATIC INIT (  
  TOTAL NUMBER COURSE RECORDS PROCESSED:,  
  TOTAL NUMBER VEHICLE RECORDS PROCESSED:,  
  TOTAL NUMBER JOBTASK RECORDS PROCESSED:,  
  TOTAL NUMBER ROE RESULT RECORDS READ:,  
  RECORDS WITHOUT MATCHING ABBREVIATED DATA:,  
  TOTAL NUMBER DIRECTORY SEARCH FAILURES: ),

WIND (6)  FIXED BIN STATIC :

DECLARE 2 ACRS  STATIC,  /* ABBR. COURSE FILE */

DECLARE 2 CHCDP  STATIC,  /* CRS COP DIRECTORY */

DECLARE 1 CINDER (304)  STATIC,  /* CRS CIN DIRECTORY */

DECLARE 1 CDPOIR (1200)  STATIC,  /* CRS CDP DIRECTORY */
DECLARE 1 AVEHICLES STATIC, /* ABBR. VEH FILE */
2 STCKCN CHAR (13), /* CKF.AVEHS */
2 DEVDESIG CHAR (9),
2 DEVNAME CHAR (47),
2 DEVCOST PICTURE '89V99',
2 CBGO CHAR (1);
VCOST FIXED DEC (10,2) STATIC,
VORDER CHAR (80) BASED (PV80);
DECLARE 1 VEHDIR (200) STATIC, /* VEHICLE DIRECTORY */
2 CHVEH CHAR (13), /* CKF.VEH.DIR */
2 VEHPRTR FIXED BIN,
COVVEH (4) CHAR (750) BASED (PTVEH),
LOADVEH CHAR (750) BASED (PLVEH);
DECLARE 1 ATASKS STATIC, /* ABBR. TASK FILE */
2 RATING CHAR (7), /* CKF.ATASKS */
2 JOBTASK CHAR (50),
2 TASKTTL CHAR (6),
2 BILCOST PICTURE '89V99',
2 CBGO CHAR (1);
DECLARE 1 TASKDIR (1200) STATIC, /* JOBTASK FILE DIRECTORY */
2 CHTASK CHAR (13), /* CKF.TASKS.DIR */
2 TASKPTR FIXED BIN,
COVTASK (12) CHAR (1500) BASED (PTTASK),
LOADTASK CHAR (1500) BASED (PLTASK);
START: NTYPE, NC, NJ, NV, NR, NN, NDF,
NPAGE, DPAGE, LT, NFIRST,
J, K = 0;
DECL COMMENCE:
DECL OPEN FILE (MCREF) RECORD INPUT,
FILE (10) RECORD INPUT,
FILE (COURSE) [INPUT],
FILE (DCIN) RECORD INPUT,
FILE (DCDP) RECORD INPUT,
FILE (VEHS) INPUT,
FILE (VEHV) RECORD INPUT,
FILE (TASKS) [INPUT],
FILE (DTASK) RECORD INPUT,
FILE (REE) RECORD INPUT;
DECL D.6-59
PAGE NUMBER : 4

SOURCE CARD LISTING FOR
P5C:  /* PRINT OF ROE SEARCH RESULTS FOR EXTRACT - FRI/11/MAR/77 */

ON ENDFILE (REE) GO TO EOJ;
READ FILE (MREF) INTO (MAST);
/* HERE, DEVICE ASSIGNMENTS ARE VERIFIED */

FIRST: DISP, PTR = 1;
IF ( PARM2 = ' BOTH' ) THEN GO TO STEPA;
IF ( PARM2 = ' TERM' )
THEN PTR = 2;
ELSE DISP = 2;
IF ( PARM2 = ' NONE' )
THEN PTR, DISP = 1;
/* HERE, PRINTER DATASET IS INITIALIZED */

STEPA: IF ( PTR = 2 ) THEN GO TO STEPB;
OPEN FILE (SYSPRINT)
LINESIZE (120) PAGESIZE (60):
ON ENDPAGE (SYSPRINT)
BEGIN ;
NPAGE = NPAGE + 1 ;
IF ( NFIRST = 1 ) THEN DO ;
PUT PAGE LINE (5) ; GO TO TSTTTL ; END ;
PUT SKIP (3) ;
NFIRST = 1 ;
TSTTTL: IF ( NTYPE = 3 ) THEN GO TO TASKTTL ;
IF ( NTYPE = 2 ) THEN GO TO VEHTTL ;

CRSTTL: PUT EDIT ( 
*** EXTRACT DATA IS REQUESTED FOR THE FOLLOWING COURSES ***
' PAGE NUMBER : ', NPAGE, 'RECORD', 'NUMBER ORIGIN CIN CDP COURSE TITLE',
'------------------ ------------------- ---------------------'
( X(9), A, SKIP(2), X(9), A, F(3),
SKIP(2), X(4), A, 3 ( SKIP(1), A ) ) ;
GO TO SKP ;

VEHTTL: PUT EDIT ( 
*** EXTRACT DATA IS REQUESTED FOR THE FOLLOWING VEHICLES ***
' PAGE NUMBER : ', NPAGE, 'RECORD', 'DEVICE',
'NUMBER ORIGIN STOCK NUMBER DESIGNATOR',
'------------------ ------------------- ---------------------'
( X(9), A, SKIP(2), X(9), A, F(3),
SKIP(2), X(4), A, COL(35), A, 3 ( SKIP(1), A ) ) ;
GO TO SKP ;

TASKTTL: PJT EDIT ( 
*** EXTRACT DATA IS REQUESTED FOR THE FOLLOWING JOBS / TASKS ***
' PAGE NUMBER : ', NPAGE, 'RECORD',
'NUMBER ORIGIN RATING RANK JOBTASK',
( X(9), A, SKIP(2), X(9), A, F(3),
SKIP(2), X(4), A, 3 ( SKIP(1), A ) ) ;

SKP: PUT SKIP (1) ;
END ;
SOURCE CARD LISTING FOR

/* PRINT OF ROE SEARCH RESULTS FOR EXTRACT - FRI/11/MAR/77 */
/* HERE, INITIALIZATION OF TERMINAL DATASET PROCEEDS */

STEP 3: IF ( DISP = 2 ) THEN GO TO STEPC :

ON CONDITION (NEWDP)
BEGIN
IF ( LT < 61 ) THEN LT = 61 ;
IF ( LT >= 66 ) THEN GO TO LPB ;
LPA:
DO L = LT TO 66 ;
DISPLAY ( ' ' ) ;
END LPA ;

LPB:
DISPLAY ( ' ' ) ;
DPAGE = DPAGE + 1 ;
DPPGE = DPPGE + 1 ;
DISPLAY ( ' ' ) ;
DISPLAY ( ' ' ) ;

/* DISPLAY OUTPUT GOES HERE */
IF ( PTR = 2 ) THEN GO TO RD1 ;
PUT EDIT ( DLINE, CARD ) ( SKIP(3), A, SKIP(1), A ) ;
GO TO RD1 ;

/* NOW, MASTER TITLE STRIP IS OUTPUT */

STEPC: READ FILE (ID) INTO (CARD) ;
DLINE = (10)' ' || ' ' || ' ' || ' ' ||' ' || ' ' ;
CARD = (12)' ' || ' ' ;
DISPLAY ( ' ' ) ;
DISPLAY ( DLINE ) ;
DISPLAY ( ' ' ) ;

/* DISPLAY OUTPUT GOES HERE */
IF ( PTR = 2 ) THEN GO TO RD1 ;
PUT EDIT ( DLINE, CARD ) ( SKIP(3), A, SKIP(1), A ) ;
GO TO RD1 ;

/* HEREIN, IS START OF MAINLINE LOOP */

RD1: READ FILE (REE) INTO (S16) ;
NR = NR + 1 ;
IF ( RTYP = NTYPE ) THEN GO TO STEP1 ;
NTYPE = RTYP ;
IF ( PTR = 1 ) THEN SIGNAL ENDPAGE (SYSPRINT) ;
IF ( DISP = 1 ) THEN SIGNAL CONDITION (NEWDP) ;

STEP 1: GO TO VI (NTYPE) ;

/* HERE, A COURSE RECORD IS PROCESSED */

VI(1): NC = NC + 1 ;
NLINE = 1 ;
DLINE = ' ' ;
PUT STRING (DLINE, EDIT ( NR, RESCIN, RESCDP)
( P ' ' ZZ9', X(12), A, X(2), A ) ) ;
ERL:
IF ( RSTAT = 1 ) || ( RSTAT = 2 ) THEN GO TO STP1 ;
SOURCE CARD LISTING FOR

P5C: /* PRINT OF ROE SEARCH RESULTS FOR EXTRACT - FRI/11/MAR/77 */

PJT STRING (DLINE) DATA ( NR, S16 ) ;
GO TO BUSTED ;

STP1: IF ( RSTAT = 1 ) THEN DLINE8 = 'ORIGINAL' ;
ELSE DLINE8 = 'ADDED' ;

IF ( RESCDP = ' ' ) THEN DLINE28 = 'NONE' ;

IF ( RSTAT = 2 ) | ( RESCDP = ' ' ) THEN

MGDATA: DO ;
NN = NN + 1 ;
DLINECST = NOAB ;
GO TO OUTLOOP ;
END MGDATA ;

YECDP = RESCDP ;
LOOKU = STEP2 ;
GO TO FINDCDP ;

STEP2: IF ( CDPREC = 0 ) THEN

LP1: DO ;
NDF = NDF + 1 ;
DLINECST = NOAB ;
GO TO OUTLOOP ;
END LP1 ;

READ FILE (COURSE) INTO (ACRS) KEY (CDPREC) ;
DLINECST = CST ;
GO TO OUTLOOP ;

/* HERE, VEHICLE TYPE RECORDS ARE PROCESSED */

V1(2): NV = NV + 1 ;
NLINE = 1 ;

DLINE = ' ' ;
PUT STRING (DLINE) EDIT ( NR, RESID )
( P'Z', X(12), A(13) ) ;

IF ( RSTAT = 1 ) | ( RSTAT = 2 ) THEN GO TO STP2 ;
ELSE GO TO ER1 ;

STP2: IF ( RSTAT = 1 ) THEN DLINE8 = 'ORIGINAL' ;
ELSE DLINE8 = 'ADDED' ;

IF ( RSTAT = 2 ) THEN

NOVDATA: DO ;
NN = NN + 1 ;
DLINE33 = NOAB ;
GO TO OUTLOOP ;
END NOVDATA ;

YEVEH = RESID ;
LOOKU = STEP3 ;
GO TO FINDVEH ;

STEP3: IF ( VEHREC = 0 ) THEN

LP2: DO ;
NDF = NDF + 1 ;
DLINE36 = NOAB ;
GO TO OUTLOOP ;
END LP2 ;

READ FILE (VEHS) INTO (AVEHICLES) KEY (VEHREC) ;

DLINE33 = DEVDESIG ;
CARD = ' ' ;
SOURCE CARD LISTING FOR

/* PRINT OF ROE SEARCH RESULTS FOR EXTRACT - FRI/II/MAR/77 */

CARD7 = DEVMNAME ;
NLINE = 2 ;

GO TO OUTLOOP ;

/* NOW, PROCESS A TASK RECORD HERE */

V1(3) : NJ = NJ + 1 ;
NLINE = 1 ;

DLINE = ' ' ;
PUT STRING (DLINE) EDIT ( NR, RATE, RANK, REST JOB TASK )
( P#2,129, X(12), A, X(4), A, X(3), A ) ;

IF ( RSTAT = 1 ) THEN GO TO STP3 ;
IF ( RSTAT = 2 ) THEN GO TO STP4 ;

STP3 : DLINE8 = 'ORIGINAL' ;
STP4 : DLINE8 = 'ADDED' ;

STP5 : IF ( RSTAT = 2 ) THEN

NOTDATA : DO ;
NJ = NJ + 1 ;
DLINE40 = NODIR ;
GO TO OUTLOOP ;
END NOTDATA ;

YETASK = RESID ;
LOOKU = STEP4 ;
GJ TO FINDTASK ;

STEP4 : IF ( TASKREC = 0 ) THEN

LP3 : DO ;
NDF = NDF + 1 ;
DLINE40 = NODIR ;
GO TO OUTLOOP ;
END LP3 ;

READ FILE (TASKS) INTO (ATASKS) KEY (TASKREC) ;

CARD = ' ' ;
CARD7 = TASKTTL ;
NLINE = 2 ;

GO TO OUTLOOP ;

OUTLOOP : IF ( PTR = 1 ) THEN PUT LIST (DLINE) SKIP (2) ;

IF ( PTR = 1 ) & ( NLINE = 2 ) THEN PUT LIST ( CARD ) SKIP (1) ;

IF ( DISP = 2 ) THEN GO TO RD1 ;
IF ( LT > MAXLTB ) THEN SIGNAL CONDITION (NEWDP) ;
DISPLAY ( DLINE ) ;
LT = LT + 1 ;

IF ( NLINE = 2 ) THEN GO TO RD1 ;
DISPLAY ( CARD ) ;
LT = LT + 1 ;
GJ TO RD1 ;
DECLARE ( FLCIN, FLCDP, FLVEH, FLTASK )   BIT (1) STATIC,
    YECIN   CHAR (8) STATIC,
    YECDP   CHAR (4) STATIC,
    YETASK, YEVEH )   CHAR (13) STATIC,
    CINREC, CDPREC, VEHREC, TASKREC, VEHMAX, TASKMAX,
    CINMAX, CDPMAX )   FIXED BIN STATIC,
LOOKU  LABEL ;

/* CIN DIRECTORY LOOKUP SUBROUTINE
ENTRY : DESIRED CIN IN YECIN
EXIT : CINREC = 0 (NO FIND) OR REC NUMBER
RETURN VECTOR IS LOOKU */

FINDCIN: IF ( FLCIN ) THEN GO TO CIN2 ;
        CINMAX = REFS (2) ;
        FLCIN = '1'B ;
        CINLOAD:
            DO J = 1 TO 4 ;
                PLCLCN = ADDR ( COVCIN (J) ) ;
                READ FILE (DCIN) INTO (LOADCIN) ;
            END CINLOAD ;
        CIN2:        DO I = 1 TO CINMAX ;
            IF ( YECIN = CHCIN (I) ) THEN GO TO ECIN2 ;
                CINREC = CIPTRU (I) ;
                    GO TO LOOKU ;
            ECIN2:        END CIN2 ;
        NFCIN:  CINREC = 0 ;
                GO TO LOOKU ;

/* CDP DIRECTORY LOOKUP SUBROUTINE
ENTRY : DESIRED CDP IN YECDP
EXIT : CDPREC = 0 (NO FIND), OR REC NUMBER
RETURN VECTOR IS LOOKU */

FINDCDP: IF ( FLCDP ) THEN GO TO CDP2 ;
        CDPMAX = REFS (3) ;
        FLCDP = '1'B ;
        CDPLOAD:
            DO J = 1 TO 4 ;
                PLCMP = ADDR ( COVCMP (J) ) ;
                READ FILE (DCDP) INTO (LOADCDP) ;
            END CDPLOAD ;
        CDP2:        DO I = 1 TO CDPMAX ;
            IF ( YECDP = CHCDP (I) ) THEN GO TO ECDP2 ;
                CDPREC = CDPTRU (I) ;
                    GO TO LOOKU ;
            ECDP2:        END CDP2 ;
        NFCDP:  CDPREC = 0 ;
                GO TO LOOKU ;

/* VEHICLE ID LOOKUP SUBROUTINE
ENTRY : DESIRED VEH ID IN YEVEH
EXIT : VEHREC = 0 (NO FIND), OR REC NUMBER
RETURN VECTOR IS LOOKU */

FINDVEH: IF ( FLVEH ) THEN GO TO VEH2 ;
        VEHMAX = REFS (8) ;
        FLVEH = '1'B ;
SOURCE CARD LISTING FOR
P5C: /* PRINT OF ROE SEARCH RESULTS FOR EXTRACT - FRI/11/MAR/77 */

VEHLOAD: DO J = 1 TO 4;
             PLVEH = ADDR (Coveh (J) ) ;
             READ FILE (DVEH) INTO (LOADVEH) ;
         END VEHLOAD ;

VEH2:    DO I = 1 TO VEHMAX :
             IF ( YVEH = CHVEH(I) ) THEN GO TO EVEH2 ;
             VEHREC = VEMPTR (I) ;
             GO TO LOOKU ;
         END VEH2 ;

EVEH2:   END VEH2 ;

NFVEH:   VEHREC = 0 ;
         GO TO LOOKU ;

/* TASK ID DIRECTORY LOOKUP SUBROUTINE
ENTRY : DESIRED TASK ID (13-CHAR) IN YETASK
EXIT : TASKREC = 0 (NO FIND) OR REC NUMBER
RETURN VECTOR IS LOOKU */

FINDTASK: IF ( FLTASK ) THEN GO TO TASK2 ;
            TASKMAX = REFS (12) ;
            FLTASK = *1*B ;
             TASKLOAD: DO J = 1 TO 12 ;
                PLTASK = ADDR ( COTASK (J) ) ;
                READ FILE (DTASK) INTO (LOADTASK) ;
            END TASKLOAD ;
            TASK2:    DO I = 1 TO TASKMAX :
                IF ( YETASK = CHTASK(I) ) THEN GO TO ETASK2 ;
                TASKREC = TASKPTR (I) ;
                GO TO LOOKU ;
                ETASK2:   END TASK2 ;
            NFTASK:   TASKREC = 0 ;
                      GO TO LOOKU ;
          /*************/
          /* IF GET HERE, PROGRAM IS TERRIBLE BROKE */
          BUSTED: PUT STRING (CARD) EDIT
                   ( ' * ** ERROR NUMBER * ', NER, ' HAS OCCURRED **' )
                   ( A, F(2), A ) ;
                   IF ( PTR = 1 ) THEN PUT EDIT ( CARD, DLINE )
                       ( SKIP(3), A, SKIP(2) , A ) ;
                      DISPLAY ( ' ' ) ;
                      DISPLAY ( DLINE ) ;
                      DISPLAY ( ' ' ) ;
                      DISPLAY ( CARD ) ;
          /*************/
          /* HERE IS EOF WINDUP PROCESSING */
          EOJ:      WIND(1) = NC ;
                    WIND(2) = NV ;
                    WIND(3) = NJ ;
                    WIND(4) = NR ;
                    WIND(5) = NN ;
                    WIND(6) = NDF ;
            IF ( PTR = 1 ) THEN PUT SKIP (3) ;
            DISPLAY ( ' ' ) ;
            DISPLAY ( ' ' ) ;
            DISPLAY ( ' ' ) ;
            DISPLAY ( ' ' ) ;
            SUMMARY:   DO J = 1 TO 6 ;
SOURCE CARD LISTING FOR

P5C: /* PRINT OF RDE SEARCH RESULTS FOR EXTRACT - FRI/11/MAR/77 */

621 PJT STRING (DLINE) EDIT
622 ( WINDUPS(J), WIND(J) ) ( A(44), P'ZZ,ZZ9' ) ;
624 IF ( PTR = 1 )
625 THEN PUT LIST ( (10)** | | DLINE ) SKIP (1) ;
626 DISPLAY ( DLINE ) ;
628 END SUMMARY ;
630 CLOSE FILE (MREF), FILE (ID), FILE (REE),
631 FILE (COURSE), FILE (DCOP),
633 FILE (VEHS), FILE (DVEH),
634 FILE (TASKS), FILE (DTASK) ;
637 IF ( PTR = 1 ) THEN CLOSE FILE (SYSPRINT) ;
638 DISPLAY ( ' ' ) ;
639 EOD PSC ;
DECLARE INPAM CHAR (100) VARYING,  
PARM CHAR (1) BASED (PP),  
DLINE CHAR (80) STATIC,  
YECIN CHAR (8)  
STATIC INIT ('ABCD') ;  

DECLARE ( N15, N35, N15R, NS, TOTCIN, TOTCDP, T35, LOOP, NT,  
NC, NFP, NER, J ) FIXED BIN STATIC ;  

DECLARE COURSE FILE RECORD INPUT,  
DCIN FILE RECORD OUTPUT,  
DCDP FILE RECORD OUTPUT;  

DECLARE IHESRTO ENTRY  
( CHAR (44), CHAR (27),  
FIXED BIN (31,0), FIXED BIN (31,0),  
ENTRY, ENTRY );  
IHESARC ENTRY  
( FIXED BIN (31,0) );  
E15A ENTRY RETURNS ( CHAR (11) ),  
E35A ENTRY,  
RCODE FIXED BIN (31,0) ;  

DECLARE RSORT CHAR (20) STATIC,  
1 SORTREC BASED (PSR),  
2 STYPE CHAR (1),  
2 DFLD1 CHAR (4),  
2 DFLD2 CHAR (4),  
2 RCPRTR FIXED BIN,  
1 OVERD BASED (POD),  
2 F1 CHAR (8),  
2 F2 FIXED BIN,  
1 OVERC BASED (POC),  
2 F1 CHAR (4),  
2 F2 FIXED BIN ;  

DECLARE 1 ACRS STATIC,  
2 SET1  
3 CDP CHAR (4),  
3 CIN CHAR (8),  
3 CST CHAR (16),  
3 NLC CHAR (4),  
3 NEC CHAR (4),  
2 PC CHAR (2),  
2 SET2  
3 RMS CHAR (3),  
3 TYCRS CHAR (2),  
3 SVC CHAR (1),  
3 MI CHAR (1),  
2 STCD CHAR (1),  
2 STDE PICTURE '99v9',  
2 SET3  
3 TRAPS CHAR (1),  
3 TPC CHAR (9),  
2 ATTR PICTURE '99v9',  
2 STBK PICTURE '99v9';
SOURCE CARD LISTING FOR
P7: /* ABSTRACTED COURSE FILE DIRECTORY CONSTRUCT - 12/14/76 */

DECLARE 1 CINDER (304) STATIC, /* CRS CIN DIRECTORY */
DECLARE 1 COPDIR (1200) STATIC, /* CRS CDP DIRECTORY */
DECLARE 1 MAST STATIC, /* MASTER INITIALIZATION DONE HERE, ONE TIME */

START: PP = ADDR (INPARM);
        PTCP = ADDR (COPDIR);
        PSR = ADDR (RPOS);
        POC = ADDR (DPPR);
        CHCIN = SPACE; CHCDP = SPACE;
        CINPTR = O; CPDPTR = O;
        N15, N35, N15R, NS, TOTCIN, TOTCDP, NT, NER = O;
        NC = -1;

OPEN FILE (MREF) RECORD UPDATE;
READ FILE (MREF) INTO (MAST);
DISPLAY ('** ABBREVIATED COURSE DATABASE CIN/CDP DIRECTORY** ');
DISPLAY ('** CONSTRUCT PROGRAM** ');
DISPLAY ('**');
CALL IHESRTD ( /* BY REC TYPE, BY CIN-CDP */
' SORT FIELDS=(1,1,CH,A,2,8,CH,A),SIZE=E2000 ',
'RECORD TYPE=F,LENGTH(20),50000, RCODE, E15A, E35A');

E15A: /* SORT RECORD MAKEUP PROCEDURE */
PROC RETURNS ( CHAR (11) );
N15 = N15 + 1;
/* THIS IS EOF ROUTINE FOR COURSE INPUT FILE */
ON ENDFILE (COURSE)
BEGIN:
DISPLAY (** EOF ON COURSE INPUT FILE **);
PAGE NUMBER: 3

SOURCE CARD LISTING FOR
P7:  /* ABBREVIATED COURSE FILE DIRECTORY CONSTRUCT - 12/14/76 */
      SORT PHASE START* ) ;
            DISPLAY ( ;' " ' ) ;
            REFS(3), REFS(1) = NC ;
            CALL IHESARC (8) ;  /* TELL SORT NOMORE RECORDS */
            GO TO ENDE15A ;

END ;
/* E15A INITIALIZATION DONE HERE ONLY ONE TIME */
 IF ( N15 > 1 ) THEN GO TO STEPl ;
STEPl:  IF ( N15R > 0 ) THEN GO TO STEP3 ;
/* THIS IS E15A PROC MAINLINE; PROCESS A NEW RECORD */
STEP2:  READ FILE ( COURSE ) INTO ( ACRS ) ;
            NC = NC + 1 ;
            RECPtr = NC ;
            STYPE = '1' ;
            OVERD.F1 = CIN ;
            N15R = 1 ;
            GO TO GIVE ;
/* HERE, SECONDARY ENTRY; GIVE CDP RECORD OUT */
STEP3:  N15R = 0 ;
            STYPE = '2' ;
            OVERD.F1 = * ;
            OVERC.F1 = CDP ;
/* HERE, ANOTHER RECORD GIVEN TO SORT */
GIVE:  NS = NS + 1 ;
            CALL IHESARC (12) ;  /* GOT STILL ANOTHER RECORD */
            RETURN ( RSORT ) ;
ENDE15A:  E0D E15A ;

E35A:  /* HERE IS THE SORT OUTPUT PROCEDURE */
      PROC ( INREC ) ;
DECLARE INREC  CHAR (20) ;
FM1:  FORMAT ( SKIP(1), X(10), 2 ( X(2), F(4) ) ,
       X(2), A(8), 2 ( X(2), F(4) ) ) ;
            N35 = N35 + 1 ;
            RSORT = INREC ;
/* HERE, E35A INITIALIZATION DONE ONE TIME */
 IF ( N35 > 1 ) THEN GO TO ROUTER ;
ROUTER:  DISPLAY ( ' ** SORT PHASE COMPLETE ; STARTING CIN *'
        /* DIRECTORY MAKEUP */ ) ;
            DISPLAY ( ' ) ;
            DISPLAY ( ' CIN NUM CIN TOTAL CDPs REC PTT: ' ) ;
            DISPLAY ( ' ' ) ;
/* ENTRY ROUTING DONE HERE */
ROUTER: IF ( STYPE = '2' ) THEN GO TO TYPE2 ;

/* HERE, STILL PROCESSING CIN ENTRIES */

TYPE1: IF ( OVERD.F1 = YECIN ) THEN

DL1: DO ;

NT = NT + 1 ;

GO TO ANOTHER ;

END DL1 ;

IF ( N35 > 1 ) THEN /* HERE, CINS HAVE CHANGED */

DL2: DO ;

PUT STRING ( DLINE ) EDIT

( TOTCIN, YECIN, NT, NFP )

( X(1), F(4), X(4), A(8), X(6), F(3), X(7), F(3) ) ;

DISPLAY ( DLINE ) ;

END DL2 ;

NT = 1 ;

YECIN = OVERD.F1 ;

NFP = RECPTR ;

TOTCIN = TOTCIN + 1 ;

LOOP = LOOP + 1 ;

IF ( LOOP > 304 ) THEN GO TO BUMCIN ;

CINDER ( LOOP ) = OVERC ;

ANOTHER: CALL IHESSARC ( 4 ) ; /* GIMMIE ANOTHER SORT RECORD */

RETURN ;

/* HERE, STARTING CDP ENTRIES */

TYPE2: IF ( T35 = 1 ) THEN /* MUST FINISH LAST CIN ENTRY */

LP3: DO ;

PUT STRING ( DLINE ) EDIT

( TOTCIN, YECIN, NT, NFP )

( X(1), F(4), X(4), A(8), X(6), F(3), X(7), F(3) ) ;

DISPLAY ( DLINE ) ;

DISPLAY ( ** LAST OF CIN DIRECTORY ENTRIES ENCOUNTERED ** ) ;

DISPLAY ( ** NOW STARTING CDP DIRECTORY ** ) ;

DISPLAY ( ** ) ;

DISPLAY ( 'CDP NUM' 'CDP REC PTR' ) ;

DISPLAY ( '---' '---' ) ;

CINPTR ( LOOP ) = LOOP ;

REFS(2) = LOOP ;

T35 = 2 ;

LOOP = 1 ;

END LP3 ;

/* PROCESS NEXT CDP ENTRY HERE */

TOTCDP = TOTCDP + 1 ;

LOOP = LOOP + 1 ;

IF ( LOOP > 1200 ) THEN GO TO BUMCDP ;

CHCOP( LOOP ) = OVERC.F1 ;

CDPPTR( LOOP ) = OVERC.F2 ;
TAEG REPORT NO. 40

PAGE NUMBER : 5

SOURCE CARD LISTING FOR P7:
/* ABBREVIATED COURSE FILE DIRECTORY CONSTRUCT - 12/14/76 */

PUT STRING (DLINE) EDIT
( TOTCDP, OVERC.F1, OVERC.F2 )
( X(1), F(4), X(4), A(4), X(4), F(4) ) ;
DISPLAY ( DLINE ) ;

GO TO ANOTHER ; /* ONTO NEXT SORT RECORD */

/* HERE, OVERFLOWS TO DIRECTORY MEMORY TABLES */

BUMCIN: DISPLAY ( ' ' ) ;
DISPLAY ( ' *** CIN TABLE OVERFLOW ***' ) ;
NER = 1 ;
GO TO BUM2 ;

BUMCDP: DISPLAY ( ' ' ) ;
DISPLAY ( ' *** CDP TABLE OVERFLOW ***' ) ;
NER = 2 ;

BUM2: DISPLAY ( ' ' ) ;
DISPLAY ( 'DIRECTORY CONSTRUCT PROGRAM IS TERMINATING' ) ;
NER = 2 ;

CALL IHESARC ( 8 ) ; /* STOP SORT RECORDS, EVEN IF MORE */

G3 TO ENDE35A ;

END E35A ;

/*****************************/

/* HERE, BACK (ONE WAY OR ANOTHER) FROM THE SORT;
EITHER THE SORT COMPLETED OK, OR ERROR FORCED
THE TERMINATION */

IF ( RCODE = 16 ) THEN DISPLAY ( ' *** SORT FAILED, CODE = 16 ***' ) ;
ELSE IF ( RCODE = 0 ) THEN DISPLAY ( ' *** SORT COMPLETED OK, CODE = 0 ***' ) ;
ELSE DD ;

PUT STRING (DLINE) EDIT
( ' *** SORT RETURN CODE INVALID, ' ,
' CODE WAS ', RCODE , ' ***' ) ;

( A, A, F(2), A ) ;

DISPLAY ( DLINE ) ;

END

/* NOW, FINISH UP BY WRITING DIRECTORIES OUT TO DISK */

IF ( RCODE = 0 ) | ( NER > 0 ) THEN GO TO STEP99 ;

CPTR (1) = LOOP ;

LP4: DD J = 1 TO 4 ;

PLCIN = ADDR ( COVCIN (J) ) ;
WRITE FILE (DCV1) FROM (LOADCIN) ;
END LP4 ;

DISPLAY ( ' ' ) ;
DISPLAY ( ' *** CIN DIRECTORY HAS BEEN SUCCESSFULLY ' ) ;

LP5: DD J = 1 TO 6 ;

PLCDP = ADDR ( COVCIN (J) ) ;
WRITE FILE (DCDP) FROM (LOADCDP) ;
END LP5 ;

D.6-71
DISPLAY ( ' ' ) ;
DISPLAY ( '*** CDP DIRECTORY HAS BEEN SUCCESSFULLY ' 'REWITTEN' ' ) ;
STEP99: CLOSE FILE (DCIN), FILE (DCDP), FILE (COURSE) ;
REWIRE FILE (MREF) FROM (MAST) ;
CLOSE FILE (MREF) ;
DISPLAY ( ' ' ) ;
DISPLAY ( 'COURSE DIRECTORY CONSTRUCT PROGRAM ' 'IS NOW TERMINATING' ' ) ;
DISPLAY ( ' ' ) ;
END P7 ;
/* COURSE FILE DESCRIPTORS LOAD - WED/15/DEC/76 */

PROC OPTIONS (MAIN);

/* REVISIONS:
WED/15/DEC/76 - FIRST IMPLEMENTATION.
MON/14/MAR/77 - MODS TO PREPARE FOR SHIP TO NCSS. */

DECLARE 1 CINDESC STATIC;
  2 CINDS CHAR (8),
  2 FILL CHAR (5),
  2 DTAB (100) FIXED BIN;

DECLARE 1 CINCARD STATIC;
  2 CINID CHAR (8),
  2 FILL CHAR (1),
  2 CINNUMS CHAR (71),

DLINE CHAR (80) STATIC,
CARD CHAR (80) BASED (PCARD);

DECLARE YECIN CHAR (8)
  STATIC INIT ('ABCDEFGHIJKLMNOPQRSTUVWXYZ'),
  GNUMS (24) FIXED BIN STATIC,
  LCARD CHAR (80) BASED (PLOAD),
  ICOURSE (160) FIXED BIN STATIC,
  ( N15, N35, NT, NC, TOTCIN, T15, NS, J, CINTOT,
  NER ) FIXED BIN STATIC,
RSORT CHAR (213) STATIC;

DECLARE DCARD FILE RECORD INPUT,
DESC FILE RECORD OUTPUT,
DIND FILE RECORD INPUT;

DECLARE 1 CINDER (304) STATIC;
  2 CHCIN CHAR (8),
  2 CINPTR FIXED BIN,
LOADCIN CHAR (760) BASED (PLCIN),
COVCIN (4) CHAR (760) BASED (PTCIN);

DECLARE 1 OINDEX BASED (PIN),
  2 DSTART FIXED BIN,
  2 DEND FIXED BIN,
  2 NCAT FIXED BIN,
  2 NDESC FIXED BIN,
  2 CATEGORIES (15),
  3 PCAT FIXED BIN,
  3 SCAT FIXED BIN,
  3 ECAT FIXED BIN,
  2 PDESC (111) FIXED BIN,
COVER (4) CHAR (80) BASED (PCOV);

DECLARE 1 MAST STATIC;
  2 REFS (40) FIXED BIN;

START: NC, NS, TOTCIN, NEOF, LOOP,
  J, CINTOT, NT = 0 ;

PIN = ADDR (ICOURSE);
PCARD = ADDR (CINCARD);
PTCIN = ADDR (CINDER);
PCOV = ADDR (ICOURSE);
PLOAD = ADDR (COVER(1)) ;

OPEN FILE (DCARD) RECORD INPUT,
SOURCE CARD LISTING FOR
P7A: /* COURSE FILE DESCRIPTORS LOAD - WED/15/DEC/76 */

FILE (DIND) RECORD INPUT;
READ FILE (DIND) IGNORE (1);

LP1: DO J = 1 TO 4;
    PLOAD = ADDR ( COVER (J) ) ;
    READ FILE (DIND) INTO (LCard) ;
    END LP1 ;

DISPLAY ( " " ) ;
DISPLAY ( "COURSE FILE DESCRIPTOR LOAD PROGRAM" ) ;
DISPLAY ( " " ) ;
DISPLAY ( "CIN ASSIGNED DESCRIPTOR(S)" ) ;
DISPLAY ( " " ) ;
CINDESC.FILL = " " ;
OPEN FILE (MREF) RECORD UPDATE ;
READ FILE (MREF) INTO (MAST) ;
ON ENDFILE (DCARD) GO TO EOJ ;
READ FILE (DCARD) INTO (CARD) ;
NC = 1 ;
DISPLAY ( CARD ) ;
GO TO STEP2 ;

/*****************************/
/* NEXT DESCRIPTOR CARD INPUT HERE */

STEP1: READ FILE (DCARD) INTO (CARD) ;
NC = NC + 1 ;
DISPLAY ( CARD ) ;
IF ( CINID = YECIN ) THEN GO TO STEP3 ;
/* HERE, STARTING A BRAND-NEW CIN; FIRST WRITE OLD-CIN
RECORD OUT TO DESCRIPTOR FILE */
WRITE FILE (DESC) FROM (CINDESC) ;
NS = NS + 1 ;
STEP2: YECIN = CINID ;
TOTCIN = TOTCIN + 1 ;

STEP2B: CINDS = CINID ;
/* CINS HERE ARE (=), START */
*/* MAKEUP OF NEW DESC RECORD */

STEP3: GNUMS = 0 ;
GET STRING (CINUMS) EDIT
(GNUMS) ( 24 ( F(2), X(1) ) ) ;
LP3: NT = 0 ;
DO J = 1 TO 24 ;
    NTEST = GNUMS (J) ;
    IF ( NTEST = 0 ) THEN GO TO ENDLP3 ;
    NT = NT + 1 ;
    IF ( PDESC ( NTEST ) = 0 ) THEN GO TO STEP4 ;
NER = NER + 1 ;
DISPLAY ( " " ) ;
PUT STRING (DLINE) EDIT

D.6-74
SOURCE CARD LISTING FOR
P7A: /* COURSE FILE DESCRIPTORS LOAD - WED/15/DEC/76 */

138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194

STEP4: DTAB (NTEST) = PDESC (NTEST) ;
ENDLP3: END LP3 ;
GO TO STEP1 ;

/*********************/
/* HERE, EOF ON INPUT DESCRIPTOR DECK */

EOJ: WRITE FILE (DESC) FROM (CINDESC) ; /* LAST RECORD */

156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193

GO TO ENDLP3 ;

148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193

END P7A ;
TAEG REPORT NO. 40

SOURCE CARD LISTING FOR P7B: /* ABBREVIATED COURSE FILE LISTINGS - MON/17/JAN/77 */

P7B: /* ABBREVIATED COURSE FILE LISTINGS - MON/17/JAN/77 */
PROC (INPARM) OPTIONS (MAIN);

/* REVISIONS:
MON/17/JAN/77 - FIRST IMPLEMENTATION.
SUN/20/MAR/77 - MODS PRIOR TO INSTALLATION AT NCSS.
*/

DECLARE INPARM CHAR (100) VARYING,
  1 POVER BASED (PP),
  2 PARM CHAR (1),
  2 PARM2 CHAR (4);

DECLARE INDESC FILE RECORD DIRECT KEYED ENV (REGIONAL(1)),
COURSE FILE RECORD DIRECT INPUT KEYED ENV (REGIONAL(1));

DECLARE T1 NOBC NEC RMS TPC LEN THRS LHRs ATTR STBX,
  T2 CHAR (74) STATIC INIT ( '----------- ----- --------- ----- ----' ),
  T3 STATIC,
    1 F1 CHAR (15) INIT ( ' PRIORITY : ' ),
    2 PC CHAR (2),
    2 F2 CHAR (16) INIT ( ' TYPE CRS : ' ),
    2 TYCRS CHAR (2),
    2 F3 CHAR (19) INIT ( ' SERVICE CDE : ' ),
    2 SVC CHAR (1),
    2 F4 CHAR (16) INIT ( ' COST/AOB : ' ),
    2 CAOB PICTURE 'I9V.99*
  T30VER CHAR (79) BASED (PT3),
  1 T4 STATIC,
    1 F1 CHAR (15) INIT ( ' TRAPS : ' ),
    2 F2 CHAR (17) INIT ( ' STATUS : ' ),
    2 STCD CHAR (1),
    2 F3 CHAR (20) INIT ( ' STATUS DTE : ' ),
    2 FSTATE PICTURE 'I99'
    2 F4 CHAR (12) INIT ( ' METH-1 : ' ),
    2 MI CHAR (1),
  T40VER CHAR (72) BASED (PT4),
  D2 CHAR (2) DEF DLINE POS (1),
  DLINE CHAR (80) STATIC;

DECLARE ELINE CHAR (38) STATIC INIT ( '*** ERROR XX OCCURRED; NREC WAS XXXXX ***' ),
END CHAR (2) DEF ELINE POS (10),
ENREC PICTURE 'ZZZZ9' DEF ELINE POS (31),
( DISP, PTR, TPAG, J, JKEY, LL, LP, LT, MAX, MAXC,
  MAXCDP, MAXKEY, MAXLD, MAXLP, NREC, NSK,
  PPAGE ) FIXED BIN STATIC,
  D2 CHAR (2) ) LABEL ;

DECLARE CDPDIR (1200) STATIC, /* CRS CDP DIRECTORY */
SOURCE CARD LISTING FOR

PTB: /* ABBREVIATED COURSE FILE LISTINGS - MON/17/JAN/77 */

DECLARE 1 INDEX static, /* OVERALL DESCRIPTOR INDEX */

DECLARE 1 MAST (4) static, /* ABBR. COURSE FILE */

DECLARE 1 ACRS (2)

START:  

COMMENCE:  

MAXCDP = REFS(1) ;
MAXCKEY = REFS(1) + 2 ;
TAEG REPORT NO. 40

PAGE NUMBER : 3

SOURCE CARD LISTING FOR
PTB: /* ABBREVIATED COURSE FILE LISTINGS - MON/17/JAN/77 */

138  MAXC = REFS(4) ;
139
140  IF ( PARM = 'C' ) THEN GO TO CSETUP ;
141
142  OPEN FILE (COURSE) DIRECT INPUT ;
143
144  IF ( PARM = 'A' ) THEN GO TO FIRST ;
145
146  BSETUP: OPEN FILE (DCDP) RECORD INPUT ;
147
148  DO J = 1 TO 6 ;
149  PLCDP = ADDR ( CJVCDP(J) ) ;
150  READ FILE (DCDP) INTO (LOADCDP) ;
151  END CDPLOAD ;
152
153  CSETUP: OPEN FILE (INDESC) INPUT ;
154  DO J = I TO 4 : /* NOTE; SKIPS RECORD 0 */
155  READ FILE (INDESC) INTO (CARD) KEY (J) ;
156  COVER (J) = CARD ;
157  END ;
158
159  NREC = DSTART - 1 ;
160
161  FIRST: DISP, PTR = 1 ;
162  IF ( PARM2 = 'BOTH' ) THEN GO TO STEP1 ;
163  IF ( PARM2 = 'TERM' )
164  THEN PTR = 2 ;
165  ELSE DISP = 2 ;
166
167  STEP1: LT, LP = 100 ;
168  IF ( PTR = 2 ) THEN GO TO STEP6 ;
169  /* HERE, PRINTER DATASET IS INITIALIZED */
170
171  OPEN FILE (MPTR) STREAM PRINT
172  PAGESIZE (58) LINESIZE (120) ;
173
174  ON ENDPAGE (MPTR)
175  BEGIN ;
176  PPAGE = PPAGE + 1 ;
177  IF ( PARM = 'C' ) THEN GO TO STEP2 ;
178
179  PUT FILE (MPTR) EDIT
180  ['PAGE NO :', PPAGE, T1, T2, ' ']
181  ['PAGE, LINE(2), X(10), A, F(4),
182  2 (SKIP(1), X(10), A(74)), SKIP(1), A ] ;
183
184  GO TO SKP ;
185
186  STEP2: PUT FILE (MPTR) EDIT ['PAGE NO :', PPAGE,
187  '*** LISTING OF DESCRIPTOR CATEGORIES AS' ]
188  ['APPLIED TO COURSE DATA ***', ' ']
189  ['PAGE, LINE(5), X(20), A, F(2),
190  SKIP(3), X(20), A, SKIP(1), X(20), A,
191  SKIP(3), A ] ;
192
193  SKP: PUT FILE (MPTR) SKIP (1) ;
194
195  STEP3: EVD ;
196
197  SIGNAL ENDPAGE (MPTR) ;
198
199  STEP6: IF ( PARM = 'C' ) THEN GO TO PLOOP ;
200  /* PRINTING LOOP FOR C-OPTION FOLLOWS */
201
202  CLOOP: NREC = NREC + 1 ;
203  ER2: IF ( NREC > MAXC ) THEN GO TO BUST2 ;
204
205  READ FILE (INDESC) INTO (DLINE) KEY (NREC) ;
206  GO TO ROUTES ;

D.6-78
SOURCE CARD LISTING FOR

/* PRINTING LOOP FOR A-B OPTIONS */

PLOOP:  LL = 1 ; 
IF ( PARM = 'B' ) THEN GO TO BREAD ;

AREAD:  NREC = NREC + 1 ; 
IF ( NREC > MAXCDP ) THEN GO TO EOFAB ;

GETCRS:  READ FILE (COURSE) INTO (ACRS) KEY (NREC) ; 
GO TO ROUTEA ;

BREAD:  JKEY = JKEY + 1 ; 
IF ( JKEY > MAXCKEY ) THEN GO TO EOFAB ;

ER1:  IF ( NREC > MAXCDP ) THEN GO TO BUST1 ; 
GO TO GETCRS ;

/* ROUTING DONE HERE BASED ON PRINT LINE NO 1, 2, OR 3 */

ROUTEA:  GO TO RTA (LL) ;

RTA(1):  PUT STRING (DLINE) EDIT 
( CIN, CDP, CST, NOBC, NEC, RMS, TPC, 
  CLEN, THRS, LHR, ATTR, STBK ) 
( A(8), X(1), A(4), X(1), A(16), X(1), A(4), X(1), 
  A(4), X(1), A(3), X(1), A(5), X(1), P'Z9V,9', 
  2 ( X(2), P'Z9V,9' ), 2 ( X(1), P'Z9V,9' ) ) ; 
  GO TO ROUTE B ;

RTA(2):  T3 = ACRS, BY NAME ; 
T3.TYCRS = ACRS.SET2.TYCRS ; 
T3.SVC = ACRS.SET2.SVC ; 
DLINE = T3OVER ; 
GO TO ROUTEB ;

RTA(3):  T4 = ACRS, BY NAME ; 
T4.TRAPS = ACRS.SET3.TRAPS ; 
T4.MI = ACRS.SET3.MI ; 
DLINE = T4OVER ;

/* NOW, SETUP FOR OUTPUT OF A NEW LINE */

ROUTEB:  GO TO RTB (DISP) ;

RTB(1):  /* HERE TERMINAL IS SERVICED */
LT = LT + 1 ; 
IF ( PARM = 'C' ) THEN GO TO LP2A ;

LP1:  IF ( LL = 1 ) THEN 
DO ; 
LT = LT + 1 ; 
DISPLAY ( ' ' ) ; 
END LP1 ;

LP2A:  IF ( LT > MAXLD ) THEN 
DO ; 
MAX = MAXLD + 4 ;

LP3:  DO J = LT TO MAX ;
DISPLAY ( ' ' ) ; 
END LP3 ;

TPAGE = TPAGE + 1 ;
DISPLAY ( ' ' PAGE :
11 TPAGE ) ;
DISPLAY ( ' ' ) ;
LT = 4 ;

IF ( PARM = 'C' ) THEN GO TO LP3A ;
PAGE NUMBER: 5

SOURCE CARD LISTING FOR
P7B: /* ABBREVIATED COURSE FILE LISTINGS - MON/17/JAN/77 */

DISPLAY ( T1 );
DISPLAY ( T2 );
DISPLAY ( " " );

LP3A:
EVDTLP2;

IF ( PARM = 'C' ) " OR ( D2 = "##" ) " OR ( D2 = "-" ) )
THEN DO ; DISPLAY ( " " ); LT = LT + 1 ; END ;
DISPLAY ( DLINE ) ;

RTB2: /* IF REQUIRED, OFFLINE PTR SERVICED HERE */
GO TO RTC ( PTR ) ;

RTC1: /* HERE, LINE PRINTER IS SERVICED */
NSK = 1 ;
IF ( PARM = 'C' ) THEN GO TO LINEOUT ;
IF ( LL = 1 ) " OR ( LINENO(MPTR) > 54 )
THEN SIGNAL ENDPAGE ( MPTR ) ;
IF ( LL = 1 ) THEN NSK = 2 ;

LINEOUT:
IF ( PARM = 'C' ) " OR ( D2 = "##" ) " OR ( D2 = "-" ) )
THEN NSK = 2 ;

PUT FILE (MPTR) EDIT (DLINE)
( SKIP(NSK), XI10), A(80) ) ;

/* NOW, REGROUP FOR NEXT LINE, IF ANY */

RTC2:
TALLY: IF ( PARM = 'C' ) THEN GO TO STEP5 ;

IF ( NREC >- DEND )
THEN GO TO EOFAB ;
ELSE GO TO CLOOP ;

STEP5:
LL = LL + 1 ;
IF ( LL < 4 ) THEN GO TO ROUTEA ;
ELSE GO TO PLOOP ;

/*****************************/
/* MAX KEY EXCEEDED ERRORS PROCESSED HERE */

BUST1: ENO = '01' ;
GO TO STEP4 ;
BUST2: ENO = '02' ;
STEP4: ENREC = NREC ;

LP4:
IF ( DISP = 1 ) THEN
DO ;
DISPLAY ( ' ' ) ;
DISPLAY ( ELINE ) ;
DISPLAY ( ' ' ) ;
END LP4 ;

IF ( PTR = 1 ) THEN
PUT FILE (MPTR) EDIT (ELINE)
( SKIP(2), X(20), A(38) ) ;

/*****************************/
/* HERE, EOF OCCURED ON INPUT FILE */
SOURCE CARD LISTING FOR
P7B: /* ABBREVIATED COURSE FILE LISTINGS - MON/17/JAN/77 */

345 EOFA8: IF ( PTR = 1 )
346    THEN CLOSE FILE (M PTR) ;
347
348 END P7B ;
PROC OPTIONS (MAIN);

** REVISIONS: 
MON/24/JAN/77 - INITIAL IMPLEMENTATION.
TUE/15/MAR/77 - MODS PRIOR TO SHIPPING TO NCSS.
*

DECLARE CARD CHAR (80) STATIC,
DLINE CHAR (80) STATIC,
CC7079 CHAR (10) DEF CARD POS (70),
FVEHS (200) FIXED BIN STATIC,
MODATA (200) CHAR (13) STATIC,
( DIFLAG, D2FLAG, ND, NV, NT, K, NSK, 
 NTEST, TER, NER, NDD, NFIND, NOD, TOTO, NS, 
 NDIR, J ) FIXED BIN STATIC;

DECLARE 1 INDEX STATIC, /* OVERALL DESCRIPTOR INDEX */
 2 DSTART FIXED BIN, /* CKF.DESC.INDEX */
 2 DVNO FIXED BIN,
 2 NCAT FIXED BIN,
 2 NDSC FIXED BIN,
 2 CATEGORIES (15),
 3 PCAT FIXED BIN,
 3 SCAT FIXED BIN,
 3 ECAT FIXED BIN,
 2 PDESC (111) FIXED BIN,
 2 POVER (4) CHAR (80) BASED (PCOV);

DECLARE 1 MAST STATIC,
 2 REFS (40) FIXED BIN,

DIND FILE RECORD DIRECT 
KEYED ENV ( REGIONAL(1) )

/*****************************/

DECLARE 1 VEHCARD STATIC, /* VEH DESCRIPTOR CARD */
 2 VSV CHAR (13), /* CKF.VDESC */
 2 VLL CHAR (1),
 2 VEHNUMS CHAR (66),

VEH80 CHAR (80) BASED (PVV80),
GVNS (22) FIXED BIN STATIC,
YEVEH CHAR (13) STATIC INIT ( 'ABCDEFGH' ),

1 VEHDESC STATIC, /* VEH DESCRIPTOR FILE */
 2 VEHDS CHAR (13), /* CKF.VEH.DESC */
 2 VTAB (100) FIXED BIN;

DECLARE 1 VEHDIR (200) STATIC, /* VEHICLE DIRECTORY */
 2 CHVEH CHAR (13), /* CKF.VEH.DIR */
 2 VEHPR FIXED BIN,

COVVEH (4) CHAR (750) BASED (PTVEH),
LOADVEH CHAR (750) BASED (PLVEH);

DECLARE 1 AVEHICLES STATIC, /* ABBR. VEH FILE */
 2 STJCKN CHAR (13), /* CKF.AVEHS */
 2 DEVDESIG CHAR (9),
 2 DEVNAME CHAR (47),
 2 DEVCOM PICTURE * (8)999',
 2 CC80 CHAR (1),

VCOST FIXED DEC (10,2) STATIC,
VOVER CHAR (80) BASED (PV80);
SOURCE CARD LISTING FOR
PB: /* VEHICLE DATA/_DESCRIPTOR LOAD, DIRECTORY - MON/24/JAN/77 */

START: ND, NV, TER, NER, NS,
TOD, NFIN, NOD,
J = 0;

PCOV = ADDR (DINDEX);
PV80 = ADDR (VEHCD2) ;
PV80 = ADDR (AVEHICLES);
PTVEH = ADDR (VEHDIR);

NDIR = -1; /* DIRECTORY RECORD POINTER */
NDD = 1; /* DIRECTORY LOAD SS, STARTS AT 2 */
FVEHS = 0; /* VEH NUMBER FLAG TABLE */
CMVEH = 0; /* VEH DIRECTORY */
VEMPTA = 0;

OPEN FILE (MREF) RECORD UPDATE,
FILE (DIND) INPUT,
FILE (VDATA) RECORD INPUT,
FILE (VCDESC) RECORD INPUT,
FILE (VEH) RECORD OUTPUT,
FILE (VDESC) RECORD OUTPUT,
FILE (VHEH) RECORD OUTPUT;

ON ENDFILE (VCDESC) GO TO EOFDESC;
ON ENDFILE (VDATA) GO TO EOFDATA;
READ FILE (MREF) INTO (MAST);

LP1:  DO J = 5 TO B : READ FILE (DIND) INTO (CARD) KEY (J) ;
COVER( J - 4 ) = CARD ;
END LP1;

/*********************/

/* VEHICLE FILE DATA INPUT PROCESSED HERE */

STEP1: READ FILE (VDATA) INTO (CARD) ;
NV = NV + 1;

GET STRiNG (CC7079) EDIT (VCOST) ( F(10,2) ) ;
PUT STRiNG (CC7079) EDIT (VCOST) ( P'(8)9V99" ) ;
VJVER = CARD ;

PUT STRiNG (DLINE) EDIT
( NV, STOCKN, DEVDESIG, DEVNAME )
( F(3), X(2), A, X(2), A, X(2), A ) ;
DISPLAY ( DLINE ) ;

DUPS: IF ( STOCKN = YEVEH ) THEN
DO ;
NV = NV - 1 ;
DISPLAY ( ** DUPLICATE STOCK NUMBER ON PRECEEDING CARD** ) ;
DISPLAY ( ** RECORD WILL BE IGNORED ** ) ;

D.6-83
SOURCE CARD LISTING FOR
P8: /* VEHICLE DATA/DESCRIPTOR LOAD, DIRECTORY - MON/24/JAN/77 */

DISPLAY ( ' ' ' ) ;
GO TO STEP1 ;
END DUPS ;
WRITE FILE (VEHS) FROM (CARD) ;
YEVEH = STOCHN ;
NDD = NDD + 1 ;
NDIR = NDIR + 1 ;
VEHPTR (NDD) = NDIR ;
CHVEH (NDD) = STOCHN ;
GO TO STEP1 ;
/* HERE, EOF ON CARD DATA INPUT DATASET */
EOFDATA: VEHPTR (1) = NDD ; /* LAST SS USED (1-ORIGIN) IN DIR */
DISPLAY ( ' ' ' ) ;
DISPLAY ( ' *** NOW AT EOF ON DATA CARD INPUT *** ' ) ;
PUT STRING (DLINE) EDIT
{ A, PZ, ZZ9, A } ;
DISPLAY ( DLINE ) ;
LP2: DJ J = 1 TO 4 ; /* WRITE OUT DIRECTORY */
PLVEH = ADDR (COVVEH(J)) ;
WRITE FILE (DVEH) FROM (LOADVEH) ;
END LP2 ;
DISPLAY ( ' ' ' ) ;
DISPLAY ( ' *** VEHICLE DIRECTORY HAS BEEN SUCCESSFULLY ')
| 'REWRITTEN' ) ;
REFS (7) = NV - 1 ;
REFS (8) = NDD ;
/**************************************************************************/
/* INITILIZE START OF DESCRIPTOR LOAD HERE */
DISPLAY ( ' ' ' ) ;
DISPLAY ( ' *** LOAD OF VEHICLE DESCRIPTOR FILE FOLLOWS *** ' ) ;
DISPLAY ( ' ' ' ) ;
DISPLAY ( ' STOCK NUMBER DEVICE DESCRIPTORS ' ) ;
DISPLAY ( ' ' ' ) ;
READ FILE (VCDESC) INTO (VEHCARD) ;
ND = 1 ;
GO TO STEP3 ;
/* HERE IS THE DESCRIPTOR LOAD LOOP */
STEP2: READ FILE (VCDESC) INTO (VEHCARD) ;
ND = ND + 1 ;
IF ( VSN = YEVEH ) THEN
DJ J ;
NSK = 1 ;
GO TO STEP4 ;
END LP3 ;
WRITE FILE (VDESC) FROM (VEHDESC) ;
NS = NS + 1 ;

D.6-84
/* VEHICLE DATA/DESCRIPTOR LOAD, DIRECTORY - MON/24/Jan/77 */

STEP3: YVEH = VSN ;
TOTD = TOTD + 1 ;
VTAB = 0 ;
VEHDS = VSN ;
NSK = 2 ;

LP5:  DO J = 1 TO NOD ; /* SEE IF MATCHING DATA AVAILABLE */
      IF ( VSN = CHVEH(J) ) THEN GO TO LP5END ;
      NFIND = NFIND + 1 ;
      FVEH(NJ) = 1 ;
      GO TO STEP4 ;
LP5END:  END LP5 ;

STEP4:  DLINE = VSN || ' ' || VEHNUMS ;
         DISPLAY ( DLINE ) ;
         GVNS = 0 ;
         GET STRING (VEHNUMS) EDIT (GVNS) ( 22 ( F(2), X(1) ) ) ;

LP4:   DO J = 1 TO 22 ;
       NTEST = GVNS (J) ;
       IF ( NTEST = 0 ) THEN GO TO ENLP4 ;
       NT = NT + 1 ;
       IF ( PDESC(NTEST) = ' ' ) THEN GO TO STEP5 ;
       NER = NER + 1 ;
       DISPLAY ( ' ' ) ;
       PUT STRING (DLINE) EDIT
          ( ' ** ERROR IN VEH DESCRIPTOR SUBSCRIPT',
            NTEST, ' DOES NOT EXIST **' ) ;
       DISPLAY ( DLINE ) ;
       IF ( NER > 100 ) THEN GO TO E0J ;
       GO TO ENLP4 ;
ENLP4:  END LP4 ;
GO TO STEP2 ;

/* HERE IS EOF ON DESCRIPTOR CARD INPUT */

EOFDESC: WRITE FILE (VDESC) FROM (VEHDESC) ;
          NS = NS + 1 ;
          REFER (9) = NS ;
          REFER (10) = ND ;
          DISPLAY ( ' ' ) ;
          DISPLAY ( ' ** EOF REACHED ON DESCRIPTOR CARD INPUT FILE ** ' ) ;
          DISPLAY ( ' ' ) ;
          PUT STRING (DLINE) EDIT
             ( ' TOTAL NUMBER DESCRIPTOR CARDS INPUT : ',
               ND ) ( 'P'P', 'ZZ9' ) ;
          DISPLAY ( DLINE ) ;
          PUT STRING (DLINE) EDIT
             ( ' TOTAL NUMBER UNIQUE VEHICLE STOCK NUMBERS : ',
               TOTD ) ( 'A'P', 'ZZ9' ) ;
          DISPLAY ( DLINE ) ;
          PUT STRING (DLINE) EDIT
             ( ' TOTAL MATCHES FOUND IN VEHICLE DATA DIRECTORY : ',
               D.6-85

D.6-85
SOURCE CARD LISTING FOR
P8: /* VEHICLE DATA/DESCRIPTOR LOAD, DIRECTORY - MON/24/JAN/77 */

    NFINID ( A, P'Z', ZZ9' ) ;
    DISPLAY ( DLINE ) ;

/********/
// ANY REQUIRED DIAGNOSTICS OUTPUT IN THIS SECTION */

    IF ( NFINID = NV ) THEN GO TO STEP6 ;

/* HERE DESCRIPTOR / DATA MISMATCHES PRINTED */

    DISPLAY ( ' ' ' ) ;
    DISPLAY ( ' ' ' ) ;
    DISPLAY ( '** THE FOLLOWING DATA FILE VEH NUMBERS' ) ;
    DISPLAY ( ' ' ' ) ;
    DISPLAY ( ' ' ' ) ;

    K = 0 ;
    DO J = 1 TO NDD ;
        IF ( FVEHS(J) = 1 ) THEN GO TO LP6END ;
        K = K + 1 ;
    END ;
    DISPLAY ( 5 ) ' | CHVEH(J) ' ;

    LP6END:  END LP6 ;

/* HERE, DESCRIPTOR VEH NUMS W/O MATCHING DATA FILE ENTRIES
ARE PRINTED - IF ANY */

    IF ( NOD = 0 ) THEN GO TO BITTEREND ;

    DISPLAY ( ' ' ' ) ;
    DISPLAY ( ' ' ' ) ;
    DISPLAY ( '** THE FOLLOWING DESCRIPTOR CARD VEH NUMBERS' ) ;
    DISPLAY ( ' ' ' ) ;
    DISPLAY ( ' ' ' ) ;

    DO J = 1 TO NOD ;
        DISPLAY ( 5 ) ' | NODATA(J) ' ;
    END ;

/* HERE AM THE WINDUP */

    REWRITE FILE ( MREF ) FROM ( MAST ) ;
    CLOSE FILE ( MREF ), FILE ( VDATA ), FILE ( VDESC ) ,
        FILE ( DIND ), FILE ( VDESC ) ;
    DISPLAY ( ' ' ' ) ;
    DISPLAY ( 'VEHICLE DATA/DIRECTORY/DESCRIPTOR LOAD PROGRAM' ) ;
    DISPLAY ( ' ' ' ) ;
    DISPLAY ( 'IS NOW TERMINATING' ) ;
    DISPLAY ( ' ' ' ) ;
    END P8 ;
DECLARE ( DINDEX, VEHDIR ) FILE RECORD DIRECT
KEYED ENV ( REGIONAL(1) ),

INPARM CHAR (100) VARYING,
 1 POVER BASED (PP),
 2 PARM CHAR (1),
 2 PARM2 CHAR (4),
 VCOST FIXED DEC (10,2) STATIC,
 DLINE CHAR (80) STATIC,
 CARD CHAR (80) STATIC,
 *ITEM STOCK-NUMBER CHAR (46) STATIC INIT ( DESIGNATOR VEHICLE COST ),
 T2 CHAR (46) STATIC INIT ( ),
 T3 CHAR (11) STATIC INIT ( ),
 'PAGE ' XXXXX ' ), T3P PICTURE 'ZZZ9' DEF T3 POS (8),
 ELINE CHAR (39) STATIC INIT ( ** ERROR XX OCCURED ; NREC WAS XXXX ** ),
 ENREC PICTURE '99' DEF ELINE POS (10),
 ( DISP, PTR, NPAGE, DPAGE, LT, JKEY, L, LP, NC, NREC,
  MAXC, MAXIND, MAXLT, MAXLTB, MAXSS, MAXVEH,
  J, K ) FIXED BIN STATIC ;

******************************************************************************

DECLARE 1 DINDEX STATIC, /* OVERALL DESCRIPTOR INDEX */
 2 DSTART FIXED BIN, /* CKF.DESC.INDEX */
 2 DEND FIXED BIN,
 2 NCAT FIXED BIN,
 2 NDESC FIXED BIN,
 2 CATEGORIES (15),
 3 PCAT FIXED BIN,
 3 SCAT FIXED BIN,
 3 ECAT FIXED BIN,
 2 POESC (111) FIXED BIN,
 COVER (4) CHAR (80) BASED (PCOV) :

DECLARE 1 MAST STATIC,
 2 REFS (40) FIXED BIN,
 MRÉF FILE RECORD ;

DECLARE 1 VEHDIR (200) STATIC /* VEHICLE DIRECTORY */
 2 CHEVE CHAR (13), /* CKF.VEH.DIR */
 2 VEMPTR FIXED BIN,
 COVVEH (4) CHAR (750) BASED (PTVEH),
 LOAVEH CHAR (750) BASED (PLVEH) ;

DECLARE 1 AVEHICLES STATIC /* ABBR. VEH FILE */
 2 STOCKN CHAR (13), /* CKF.AVEHS */
 2 DEVDESIG CHAR (9),

D.6-87
SOURCE CARD LISTING FOR
PBA:  /* ABBREVIATED VEHICLE FILE LISTINGS - WED/2/FEB/77 */

2 DEVNAME  CHAR (47)
2 DEVCost  PICTURE '81V99'.
2 CC80  CHAR (1),
VOVER  CHAR (80) BASED (PV80);

START:  NPAGE, DPAGE, NC,
J, K = 0 ;

LT = 100 ;  /* DISPLAY PAGE LINE COUNT */
MAXLT = 60 ;  /* MAX # LINES ON DISPLAY */
JKEY = 1 ;  /* SS IN DIRECTORY FOR EXTRACT -
   NOTE - SKIPS FIRST LOC VEHPT(1) */
PP = ADDR (INPARM);
PV80 = ADDR (AVEHICLES);
PTVEH = ADDR (VEHDIR);
PCOV = ADDR (DINDEX);

COMMENCE:
OPEN FILE (MREF) RECORD INPUT,
FILE (DIND) INPUT,
FILE (VEHS) INPUT,
FILE (DVEH) RECORD INPUT ;
READ FILE (MREF) INTO (MAST);
MAXVEH = REFS (7);
MAXC = REFS (4);

IF ( PARM = 'B' ) THEN GO TO BSETUP ;

ASETUP:  MAXSS = REFS (8) ;  /* HERE, PLAN FOR ABBREVIATED DATA FILE */

DIRLOAD:     DO J * I TO 4 ;
            PLVEH = ADDR ( COVVEH(J) ) ;
            READ FILE (DVEH) INTO (LOADVEH) ;
            END DIRLOAD ;
            MAXLTA = MAXLT - 3 ;
            GO TO FIRST ;

BSETUP:     DO J = 5 TO 8 ;  /* SETUP FOR DESCRIPTOR LIST */
            READ FILE (DIND) INTO (CARD) KEY (J) ;
            COVER (J - 4) = CARD ;
            END BSETUP ;
            MAXIND = REFS (4) ;
            NREC = DSTART - 1 ;
            MAXLTB = MAXLT - 2 ;

FIRST:  DISP, PTR = 1 ;  /* DEVICE ASSIGNMENTS CHECKED HERE */

    IF ( PARM2 = 'BOTH' ) THEN GO TO STEPl ;
    IF ( PARM2 = 'TERM' )
        THEN PTR = 2 ;
        ELSE DISP = 2 ;

STEP1:  LT, LP = 100 ;
        /* HERE, PRINTER DATASET IS INITIALIZED */
        IF ( PTR = 2 ) THEN GO TO STEP4 ;
        OPEN FILE (SYSPRINT)
SOURCE CARD LISTING FOR
P8A: /* ABBREVIATED VEHICLE FILE LISTINGS - WED/2/FEB/77 */

PAGESIZE (58) LINESIZE (120):

ON ENDPAGE (SYSPRINT)
BEGIN;

IF ( PARM = 'B' ) THEN GO TO STEP2;

PJET EDIT ( 'PAGE NO : ' , NPAGE , T1 , T2 , ' ' )
( PAGE , LINE(2) , X(10) , A , F(4) ,
 2 ( SKIP(1) , X(10) , A(46) ) , SKIP(1) , A ) ;
GO TO STEP3;

STEP2: PUT EDIT ( 'PAGE NO : ' , NPAGE,
  '**** LISTING OF DESCRIPTOR CATEGORIES AS',
  'APPLIED TO VEHICLE DATA ****', ' ' )
( PAGE , LINE(5) , X(20) , A , F(2) ,
  SKIP(3) , X(20) , A , SKIP(1) , X(20) , A ,
  SKIP(3) , A ) ;

STEP3: PUT SKIP (1) ;
END ;

SIGNAL ENDPAGE (SYSPRINT);

/* HERE, DISPLAY DATASET SETUP, IF REQUIRED */

STEP4: IF ( DISP = 2 ) THEN GO TO LOOP;

ON CONDITION (NEWDP)
BEGIN;

IF ( LT < 61 ) THEN LT = 61;

LP1:

IF ( LT < 66 ) THEN
  DO L = LT TO 66 ;
  DISPLAY ( ' ' ) ;
  END LP1 ;

DISPLAY ( ' ' ) ;
DPAGE = DPAGE + 1 ;
T3P = DPAGE ;
DISPLAY ( T3 ) ;
DISPLAY ( ' ' ) ;
LT = LT + 3 ;
END TTLA ;

END;

SIGNAL CONDITION (NEWDP);

/*****************************/

/* HERE IS MAIN PROGRAM LOOP */

LOOP: IF ( PARM = 'B' ) THEN GO TO STEP5;

/* HERE, PRINT SETUP FOR B-OPTION - VEHICLE DESIGNATORS */

BLOOP: NREC = NREC + 1 ;

ER1: IF ( NREC > MAXC ) THEN GO TO BUST1 ;

READ FILE (DIND) INTO (DLINE) KEY (NREC) ;
GO TO STEP6 ;
/* ABBREVIATED VEHICLE FILE LISTINGS - WED/2/FEB/77 */

HERE, SETUP FOR A-OPTION, VEHICLE DATA PRINT */

STEP5: JKEY = JKEY + 1;
          IF ( JKEY > MAXSS ) THEN GO TO EOFAB;
          NREC = VEHPTR (JKEY);
ER2: IF ( NREC > MAXVEH ) THEN GO TO BUST2;
READ FILE (VEHS) INTO (AVEHICLES) KEY (NREC);
                      NC = NC + 1;

/* NOW, OUTPUT THE NEW LINE(S); FIRST TRY DISPLAY */

STEP6: IF ( DISP = 2 ) THEN GO TO STEP7;

LP2: IF ( PARM = 'A' ) THEN
          DO :
          IF ( LT > MAXLTA ) THEN SIGNAL CONDITION (NEWDP);
          DISPLAY ('  ');
          PUT STRING (DLINE) EDIT
          ( NC, STOCKN, DEVDESIG, DEVCOST )
          ( F(4), X(2), A(13), X(2), A(9),
            X(2), P$9.9999, $9.9999);
          DISPLAY (DLINE);
          DISPLAY ( (10) ' ' cr ' ' DEVTNAME );
          LT = LT + 3;
          GO TO STEP7;
          END LP2;
          IF ( LT > MAXLTA ) THEN SIGNAL CONDITION (NEWDP);
          IF ( PARM = 'B' ) & ( (D2 = '***') | (D2 = '---') )
          THEN DO ; DISPLAY ('  ');
          LT = LT + 1;
          END;
          LT = LT + 1;
/* HERE, TRY FOR PRINTER OUTPUT */

STEP7: IF ( PTR = 2 ) THEN GO TO TALLY;

LP3: IF ( PARM = 'B' ) THEN
          DO :
          IF ( (D2 = '***') | (D2 = '---') )
            THEN PUT SKIP (2);
          PUT EDIT ( DLINE ) ( SKIP(1), X(10), A(80) ) ;
          GO TO TALLY;
          END LP3;
          PUT EDIT ( NC, STOCKN, DEVDESIG, DEVCOST, DEVTNAME )
          ( SKIP(2), X(10), F(4), X(2), A(13), X(2), A(9),
            X(2), P$9.9999, $9.9999, X(3), A(47) ) ;
/* HERE, RECYCLE FOR NEXT, IF REQUIRED */

TALLY: IF ( PARM = 'A' ) THEN GO TO STEP5;

TALLYB: IF ( NREC >= DEND )
          THEN GO TO EOFAB;
          ELSE GO TO LOOP;

***************
SOURCE CARD LISTING FOR

P8A: /* ABBREVIATED VEHICLE FILE LISTINGS - WED/2/FEB/77 */

/* MAX KEYS EXCEEDED ERRORS PROCESSED HERE */

BUST1: ENO = 1 ;
GO TO ERCOM ;
BUST2: ENO = .2 ;
ERCOM: ENREC = NREC ;
IF ( DISP = 2 ) THEN
LP4:
DO ;
DISPLAY ( " " ) ;
DISPLAY ( " " ) ;
DISPLAY ( ELINE ) ;
DISPLAY ( " " ) ;
END LP4 ;
IF ( PTR = 1 ) THEN
PUT LIST ( ELINE ) SKIP (3) ;

/******* YE OLDE WINDUP HERE */

EOFAB: CLOSE FILE ( MREF ), FILE ( DIND ), FILE ( VEHS ),
FILE ( DVEH ) ;
EVD P8A ;
DECLARE CARD CHAR (80) STATIC,
CC7079 CHAR (10) DEF CARD POS (70),
DLINE CHAR (80) STATIC,
DL4 CHAR (4) DEF DLINE POS (1),
(. DISP, PRTR. NPAGE ) FIXED BIN STATIC,
FTASKS (1200) FIXED BIN STATIC,
NDATA (1200) CHAR (13) STATIC,
( NO, N, DT, NT, K, NSK, TOTD, NS,
NTEST, TER, NER, NOD, NFIND, NOD, NLIST,
NODIR, J ) FIXED BIN STATIC ;

DECLARE 1 DINDEY STATIC ; /* OVERALL DESCRIPTOR INDEX */
2 DSTART FIXED BIN ; /* CKF.DESC.INDEX */
2 DEND FIXED BIN ,
2 MCAT FIXED BIN ,
2 NNAME FIXED BIN ,
2 CATEGORIES (15) ,
3 PCAT FIXED BIN ,
3 SCAT FIXED BIN ,
2 PCAT FIXED BIN ,
2 PDSC (11) FIXED BIN ,
COVER (4) CHAR (80) BASED (PCOV) ;

DECLARE 1 MAST STATIC ,
2 REFS (40) FIXED BIN ,
DIND KEYED ENV ( REGIONAL(1) ) ;

*******************************************************************************/

DECLARE 1 TDATACARD STATIC ,
2 F13 CHAR (13) ,
2 JTTL CHAR (50) ,
2 JI LL CHAR (6) ,
2 BCOST CHAR (10) ,
2 FILL2 CHAR (1) ,
2 TOVER CHAR (80) BASED (PT080) ,
2 TCO ST FIXED DEC (10,2) ;

DECLARE 1 TASKCARD STATIC ; /* TASK DESCRIPTOR CARDS */
2 TASKN CHAR (13) , /* CKF.TDESC */
2 TASKNUMS CHAR (67) ,
1 TASKOVER BASED (PTO) ,
2 RATING CHAR (5) ,
2 RANK CHAR (2) ,
2 JDTASK CHAR (6) ,
2 TASK0 CHAR (80) BASED (PTT80) ,
2 TVNS (22) FIXED BIN STATIC ,
2 YETASK CHAR (13) STATIC INIT ( 'ABCDEF' ) ;

DECLARE 1 TASKDESC STATIC ; /* TASK DESCRIPTOR FILE */
PAGE NUMBER: 2

SOURCE CARD LISTING FOR P9:
/* TASK DATA/DESCR/DIRECTORY LOAD - FRI/28/JAN/77 */

2 TASKDS CHAR (13), /* CKF.TASK.DESC */
2 TTAB (100) FIXED BIN ;

DECLARE 1 TASKDIR (1200) STATIC, /* JOBTASK FILE DIRECTORY */
2 CHTASK CHAR (13), /* CKF.TASKS.DIR */
2 TASKPTR FIXED BIN ;

COVTASK (12) CHAR (1500) BASED (PTTASK),
LOADTASK CHAR (1500) BASED (PTTASK) ;

DECLARE 1 ATASKS STATIC, /* ABR.TASK FILE */
2 RATING CHAR (7), /* CKF.ATASKS */
2 JOBTASK CHAR (6),
2 TASKTTL CHAR (50),
2 FILL CHAR (6),
2 BICOST PICTURE (8)999999,
2 CCB0 CHAR (1) ;

START: ND, NV, TER, NER, NS,
TOTO, NSEND, ND,
J = 0 ;
PCOV = ADDR (DINDEX) ;
PTB0 = ADDR (TASKCARD) ;
PTASK = ADDR (TASKDIR) ;
PTO80 = ADDR (ATASKS) ;
PTO = ADDR (TDATACARD) ;

NDIR = -1 ; /* DIRECTORY RECORD POINTER */
NDD = 1 ; /* DIRECTORY LOAD SS, STARTS AT 2 */
FTASKS = 0 ; /* TASK FLAG TABLE */
CHTASK = 0 ; /* TASK DIRECTORY PRELIMINARY FILL */
NLIST = 1 ; /* PRINTER TITLE CONTROL */

OPEN FILE (MREF) RECORD UPDATE,
FILE (DIND) INPUT,
FILE (TDATA) RECORD INPUT,
FILE (TDESC) RECORD INPUT,
FILE (DTASK) RECORD OUTPUT,
FILE (TASKS) RECORD OUTPUT ;

ON ENDFILE (TDESC) GO TO EOFDESC ;
ON ENDFILE (TDATA) GO TO EOFDATA ;

READ FILE (MREF) INTO (MAST) ;

LP1: DO J = 9 TO 12 ;
READ FILE (DIND) INTO (CARD) KEY (J) ;
COVER ( J - 8 ) = CARD ;
END LP1 ;

DISPLAY ( ' ' ) ;
DISPLAY ( 'TASK DATA/DIRECTORY/DESCRIPTOR LOAD' ) ;
DISPLAY ( 'PROGRAM IS NOW STARTING' ) ;

***************
/* PRINTER/TERM requirements setup in this section */
NPAGE = 0 ;
DISP, PRTR = 1 ;

DISPLAY ( ' ' ) ;
DISPLAY ( 'OUTPUT TO GO TO TERMINAL, OFFLINE PRINTER, ' ) ;
DISPLAY ( 'OR BOTH? ' ) ;

D.6-93
SOURCE CARD LISTING FOR
P9: /* TASK DATA/DESCR/DIRECTORY LOAD - FRI/28/JAN/77 */

TP2: DISPLAY ( 'ENTER TERM, PRTR, BOTH, OR QUIT' ) REPPLY ( DLINE ) ;

IF ( DL4 = 'TERM' ) THEN GO TO STERM ;
IF ( DL4 = 'PRTR' ) THEN GO TO OPRTR ;
IF ( DL4 = 'BOTH' ) THEN GO TO SPRTR ;
IF ( DL4 = 'QUIT' ) THEN GO TO FEND ;

DISPLAY ( ' ' ) ;
DISPLAY ( 'UNABLE TO DETERMINE ENTRY TYPE - TRY AGAIN' ) ;
DISPLAY ( ' ' ) ;
GO TO TP2 ;
OPRTR: DISP = 2 ;

SPRTR: OPEN FILE (SYSPRINT) LINESIZE (120) PAGESIZE (60) ;

TEAM REPORT NO. 40
D.6-94
SOURCE CARD LISTING FOR P9:

/* TASK DATA/DESCR/DIRECTORY LOAD - FRI/28/JAN/77 */

TAASKOVER, JOBASK, JTTL, BCOST
( SKIP(1), X(10), F(4), X(3), A(5), X(2), A(2), X(2), A(6), X(2), A(50), X(3), A(10) );

IF ( DISP = 2 ) THEN GO TO S1A;
PUT STRING (DLINE) EDIT
( N, TASKOVER.RATING, TASKOVER.RANK,
( P'ZZ,ZZ9*, X(2), A, X(1), A, X(1), A, X(2), A ) ;

DISPLAY ( DLINE ) ;

S1A: IF ( F13 = YETASK) THEN
DUPs:
DO:
NV = NV - 1 ;

IF ( PRTR = 1 ) THEN
PJT EDIT ( "** DUPLICATE RATING/RANK/TASK ON THE ",
" PRECEEDING CARD", " RECORD WILL BE IGNORED **" )
( SKIP(2), A, A, SKIP(1), A ) ;

IF ( DISP = 2 ) THEN GO TO STEP1 :
DISPLAY ( "" ) ;
DISPLAY ( "** DUPLICATE RATING/RANK/TASK ON THE ",
" PRECEEDING CARD" ) ;
DISPLAY ( " RECORD WILL BE IGNORED" ) ;
DISPLAY ( "" ) ;
GO TO STEP1 ;
END DUPs :

WRITE FILE (TASKS) FROM (TDATACARD) ;
YETASK = F13 ;
ND = NDD + 1 ;
NDIR = ND + 1 ;
CHTASK (NDD) = F13 ;
TASKPTR (NDD) = NDIR ;
GO TO STEP1 ;

/* HERE, EOF ON CARD DATA INPUT DATASET */

EOFDATA: TASKPTR (1) = NDD ;

LP2: DO J = 1 TO 12 ;
PLTASK = ADDR (COVTASK(J) ) ;
WRITE FILE (DATASK) FROM (LOADTASK) ;
END LP2 ;

DISPLAY ( "" ) ;
DISPLAY ( "** EOF ON DATA INPUT FILE, " ) ;
PUT STRING (DLINE) EDIT
( " CARDS PROCESSED :", NV, " **" )
( A, P'ZZ,ZZ9*, A ) ;
DISPLAY ( DLINE ) ;
DISPLAY ( "" ) ;
DISPLAY ( "** LOAD OF TASK DESCRIPTOR CARDS FOLLOWS **" ) ;
DISPLAY ( "" ) ;
REFS (11) = NV - 1 ;
REFS (12) = NDD ;

/****************************************************/

/* INITILIZE START OF DESCRIPTOR LOAD HERE */

IF ( PRTR = 2 ) THEN GO TO PHAZE2 ;
NLIST = 2 ;
source card listing for
P9: /* task data/descr/directory load - fri/28/jan/77 */

signal endpage (sysprint);

put list (** load of descriptor card data follows ***)

put skip (2);

put skip (3);

phaze2: read file (tcdesc) into (taskcard);

nd = 1;
go to step3;

/* here is the descriptor load loop */

step2: read file (tcdesc) into (taskcard);

nd = nd + 1;

lp3:

if ( taskn = yetask ) then

nsk = 1;
go to step4;

end lp3;

write file (tdesc) from (taskdesc);

ns = ns + 1;

step3: yetask = taskn;

total = total + 1;

ttab = 0;

tasks = taskn;

nsk = 2;

lp5:

d0 j = 1 to nod; /* see if matching data available */

if ( taskn = chtask(j) ) then go to lp5end;

nfind = nfnd + 1;

ftasks (j) = 1;
go to step4;

end lp5;

lp5end:

end lp5;

nod = nod + 1;

nodata (nod) = taskn;

step4: if ( prtr = 1 ) then

put edit; ( nd, taskcard )

( skip(1), x(10), f(4), x(3), a(13), a(67) ) ;

if ( disp = 1 ) then display ( task80 ) ;

tvns = 0;

get string (tasknums) edit (tvns) ( 22 ( f(2), x(1) ) ) ;

lp4:

d0 j = 1 to 22;

ntest = tvns (j);

if ( ntest = 0 ) then go to endlp4;

nt = nt + 1;

if ( pdesc(ntest) = 0 ) then go to step5;

ner = ner + 1;

if ( prtr = 1 ) then

put edit (** error in task number - taskn,

"- descriptor subscript ", ntest,

" does not exist **")

( skip(1), a, a(13), a, f(3), a ) ;

if ( disp = 2 ) then go to bumo;

put string (dline) edit

( ** error in task number - taskn,

descriptor subscript ", ntest,

does not exist ** )

D.6-96
SOURCE CARD LISTING FOR
P9: /* TASK DATA/DESCR/DIRECTORY LOAD - FRI/28/JAN/77 */

{ A; A; F(3); A; };
DISPLAY " " ; DISPLAY ( DLINE ) ;
DISPLAY " " ;

BUND: IF ( NER > 100 ) THEN GO TO EOJ ;
GO TO ENDLP4 ;

STEP5: TTAB ( NTEST ) = PDESC ( NTEST ) ;
ENO LP4 ;
GO TO STEP2 ;
/* HERE IS EOF ON DESCRIPTOR CARD INPUT */

EOFDESC: WRITE FILE ( TDESC ) FROM ( TASKDESC ) ;
NS = NS + 1 ;
REFS (13) = NS ;
REFS (14) = ND ;
DISPLAY " " ; DISPLAY ( "** EOF REACHED ON TASK DESCRIPTOR INPUT FILE,**" ) ;
PUT STRING ( DLINE ) EDIT
( " CARDS PROCESSED : ", ND, " **" ) ;
( A; "ZZZZZZZZZ", A; ) ;
DISPLAY ( DLINE ) ;

***************
/* ANY REQUIRED DIAGNOSTICS OUTPUT IN THIS SECTION */

IF ( NFDIV = NV ) THEN GO TO STEP6 ;
/* HERE DESCRIPTOR / DATA MISMATCHES PRINTED */

IF ( PRTR = 2 ) THEN GO TO LP6A ;
PUT EDIT ( "** THE FOLLOWING DATA FILE TASK NUMBERS**, HAVE NO MATCH IN THE DESCRIPTOR CARD FILE **" ) ;
( SKIP(3); A; SKIP(1); A; ) ;
PUT SKIP (2) ;
LP6A:
DISPLAY " " ; /* NOTE - ALWAYS DISPLAY */
DISPLAY ( " THE FOLLOWING DATA FILE TASK NUMBERS**, HAVE NO MATCH IN THE DESCRIPTOR CARD FILE **" ) ;
DISPLAY " " ;

K = 0 ;
LP6:
DO J = 1 TO NOD ;
IF ( FTASKS(J) = 1 ) THEN GO TO LP6END ;
K = K + 1 ;
PJRT STRING ( DLINE ) EDIT ( K; CHTASK(J) ) ;
( F(3); X(2); A; ) ;
DISPLAY ( DLINE ) ;
IF ( PRTR = 1 ) THEN
PUT EDIT ( K; CHTASK(J) ) ;
( SKIP(1); X(10); F(3); X(3); A; ) ;
LP6END:
END LP6 ;
/* HERE, DESCRIPTOR TASK NUMBERS W/O MATCHING DATA FILE ENTRIES ARE PRINTED - IF ANY */

STEP6: IF ( NOD = 0 ) THEN GO TO BITTEREND ;

IF ( PRTR = 2 ) THEN GO TO LP7A ;
PUT EDIT ( "** THE FOLLOWING DESCRIPTOR CARD TASK NUMBERS**, HAVE NO MATCH IN THE DATA FILE **" ) ;
( SKIP(3); A; SKIP(1); A; ) ;
SOURCE CARD LISTING FOR
P9: /* TASK DATA/DESCR/DIRECTORY LOAD - FRI/28/JAN/77 */

414 PUT SKIP(2) ;
415 LP7A: DISPLAY ( ' ' ' ' ) ; DISPLAY ( ' ' ' ' ) ;
417 DISPLAY ( '*** THE FOLLOWING DESCRIPTOR CARD TASK NUMBERS ' ) ;
418 DISPLAY ( ' HAVE NO MATCH IN THE DATA FILE **' ) ;
419 DISPLAY ( ' ' ' ' ) ;

420 LP7: DO J = 1 TO NOD ;
421 IF ( PRTR = 1 ) THEN
422 PUT EDIT ( J, NODATA(J) ) ( SKIP(1), X(10), F(3),
423 X(3), A(13) ) ;
424 PUT STRING (DLINE) EDIT ( J, NODATA(J) )
425 ( F(3), X(2), A ) ;
426 DISPLAY ( DLINE ) ;
427 END LP7 ;

432 /***********************************************************************************/
433 /* HERE AM THE WINDUP */
434 BITTEREND:
435 REWRITE FILE (MREF) FROM (MAST) ;
436 FEND:
437 E0J: CLOSE FILE (MREF), FILE (DIND), FILE (TDATA),
438 FILE (TDESC), FILE (DTASK), FILE (TDESC),
439 FILE (TASKS) ;
440 DISPLAY ( ' ' ' ' ) ;
442 DISPLAY ( ' TASK FILE DATA/DIRECTORY/DESCRIPTOR LOAD PROGRAM' )
443 ( ' IS NOW TERMINATING' ) ;
444 END P9 ;
SOURCE CARD LISTING FOR
P9A: /* ABBREVIATED TASK FILE LISTINGS - FRI/4/FEB/77 */

P9A: /* ABBREVIATED TASK FILE LISTINGS - FRI/4/FEB/77 */
PROC (INPARM) OPTIONS (MAIN);

/* REVISIONS:
FRI/04/FEB/77 - FIRST IMPLEMENTATION.
SUN/20/MAR/77 - MODS PRIOR TO INSTALLATION AT NCSS. */

DECLARE (DISP, PTR, NPAGE, DPAGE, LT, JKEY, LP, L, MAXC,
MAXIND, MAXLT, MAXLTA, MAXTLB, MAXSS,
MAXTASK, NC, NREC,
J, K) FIXED BIN STATIC;

DECLARE (DIND, TASKS) FILE RECORD DIRECT
KEYED ENV (REGIONAL(1)),

INPARM CHAR (100) VARYING,
1 POWER BASED (PP),
2 PARM CHAR (1),
2 PARM2 CHAR (4),

TCOST FIXED DEC (10, 2) STATIC,
(DLINE, CARD ) CHAR (80) STATIC,
D2 CHAR (2) DEF DLINE POS (1),
BLANK CHAR (4) STATIC INIT (' ' ),

'T1 RATING RANK CHAR (45) STATIC INIT (',' ),
------
'T2 ------ ------ -----------

'ELENE CHAR (39) STATIC INIT (' ' ),
END CHAR (2) DEF ELINE POS (10),
ENREC PICTURE 'ZZZ9' DEF ELINE POS (32),

'T3 CHAR (11) STATIC INIT (',' ),

'PAGE : XXXX ' ),
T3P PICTURE 'ZZZ9' DEF T3 POS (8);

*******************************************************************************/

DECLARE 1 DINDEX STATIC, /* OVERALL DESCRIPTOR INDEX */
2 DSTART FIXED BIN, /* CKF.DESC.INDEX */
2 DEND FIXED BIN,
2 NDESC FIXED BIN,
2 CATEGORIES (15),
3 PCAT FIXED BIN,
3 SCAT FIXED BIN,
3 ECAT FIXED BIN,
2 PDESC (111) FIXED BIN,
COVER (4) CHAR (80) BASED (PCOV);

DECLARE 1 MAST STATIC,
2 REFS (40) FIXED BIN;

DECLARE 1 ATASKS STATIC, /* ABBR. TASK FILE */
2 RATING CHAR (7), /* CKF. ATASKS */
2 JOBTASK CHAR (6),
2 TASKTTL CHAR (50),
2 FILL CHAR (6),
2 BILCOST PICTURE '(8)9999',
2 CC80 CHAR (1);

DECLARE 1 TASKOVER BASED (PTO),
SOURCE CARD LISTING FOR P9A: /* ABBREVIATED TASK FILE LISTINGS - FRI/4/FEB/77 */

DECLARE 1 TASKDIR (1200) STATIC, /* JOBTASK FILE DIRECTORY */
2 CHTASK CHAR (13), /* CKF.TASKS.DIR */
2 TASKPTR FIXED BIN,
COVTASK (12) CHAR (1500) BASED (PTTASK),
LOADTASK CHAR (1500) BASED (PLTASK);

START:
PP = ADDR (INPARM);
PTO = ADDR (ATASKS);
PCOV = ADDR (DINDEX);
PTTASK = ADDR (TASKDIR);

NPAGE, DPAGE, NC,
J, K = 0 ;
LT = 100 ; /* DISPLAY PAGE LINE COUNTER */
MAXLT = 60 ; /* MAX # LINES ON DISPLAY */
JKEY « 1 ; /* DIRECTORY SS, SKIPS DIR(l) */

COMMENCE: OPEN FILE (MREF) RECORD INPUT,
FILE (DIND) INPUT,
FILE (TASKS) INPUT,
FILE (DTASK) RECORD INPUT ;
READ FILE (MREF) INTO (MAST);
MAXTASK = REFS (11); MAXC = REFS (4); MAXSS = REFS (12);
IF ( PARM = 'B' ) THEN GO TO BSETUP ; /* A-OPTION, PRINT DATA FILE SETUP HERE */
MAXLTA = MAXLT - 3 ;

DIRLOAD: DO J = 1 TO 12 ;
PLTASK = ADDR (COVTASK(J));
READ FILE (DTASK) INTO (LOADTASK); END DIRLOAD ;
GO TO FIRST ; /* B-OPTION ROUTE INITIALIZED HERE */

BSETUP: DO J = 9 TO 12 ;
READ FILE (DIND) INTO (CARD) KEY (J);
COVER (.J - 8) = CARD ;
END BSETUP ;
MAXIND = REFS (4); NREC = DSTART - 1 ;
MAXLTB = MAXLT - 1 ;

/* NOW INITILIZE DEVICE ASSIGNMENTS */

FIRST: DISP, PTR = 1 ;
IF { PARM2 = 'BOTH' } THEN GO TO STEP1 ; IF { PARM2 = 'TERM' } THEN PTR = 2 ; ELSE DISP = 2 ;
SOURCE CARD LISTING FOR P9A: /* ABBREVIATED TASK FILE LISTINGS - FRI/4/FEB/77 */

STEP1: LT, LP = 100;
/* HERE, PRINTER DATASET IS INITIALIZED */
IF ( PTR = 2 ) THEN GO TO STEP4;
OPEN FILE (SYSPRINT)
LINESIZE (120) PAGESIZE (58);
ON ENDPAGE (SYSPRINT)
BEGIN;
   NPAGE = NPAGE + 1;
   IF ( PARM = 'B' ) THEN GO TO STEP2;
   PJT EDIT ( 'PAGE NO : ', NPAGE, T1, T2, ' ' )
       ( PAGE, LINE(2), X(10), A, F(4) ),
       2 ( SKIP(1), X(10), A(45) ), SKIP(1), A );
   GO TO STEP3;
STEP2: PUT EDIT ( 'PAGE NO : ', NPAGE,
       '*** LISTING OF DESCRIPTOR CATEGORIES AS',
       APPLIED TO JOB TASK DATA '***', ' ' )
       ( PAGE, LINE(5), X(20), A, F(2) ),
       SKIP(3), X(20), A, SKIP(1), X(20), A,
       SKIP(3), A );
STEP3: PUT SKIP (1);
END;
SIGNAL ENDPAGE (SYSPRINT);
/* HERE, SETUP DISPLAY DATASET, IF REQUIRED */
STEP4: IF ( DISP = 2 ) THEN GO TO LOOP;
ON CONDITION (NEWDP)
BEGIN;
   IF ( LT < 66 ) THEN
      LP1:
          DO L = LT TO 66;
          DISPLAY ( BLANK ) ;
      END LP1;
      DISPLAY ( BLANK );
      DPAGE = DPAGE + 1;
      T3P = DPAGE ;
      DISPLAY ( T3 ) ;
      DISPLAY ( BLANK );
      LT = 4;
      ATTL:
          IF ( PARM = 'A' ) THEN
              DISPLAY ( 'ITEM RATE RANK TASK COST-_PER-BILLET' ) ;
              DISPLAY ( '----- ----- ----- ----- ' ) ;
              DISPLAY ( ' ' ) ;
              LT = LT + 3 ;
          END ATTL:
      END;
      SIGNAL CONDITION (NEWDP);
/* MAINLINE LOOP FOLLOWS */
LOOP: IF ( PARM = 'B' ) THEN GO TO STEP5;
/* HERE IS B-SECTION PRINT SETUP */

8LOOP: NREC = NREC + 1;
ER1: IF ( NREC > MAXC ) THEN GO TO BUST1;
READ FILE (DIND) INTO (DLINE) KEY (NREC);
GO TO STEP6;

/* NOW, FOR A-OPTION SETUP */

STEP5: JKEY = JKEY + 1;
IF ( JKEY > MAXSS ) THEN GO TO EOFAB;
NREC = TASKPTR (JKEY);
ER2: IF ( NREC > MAXTASK ) THEN GO TO BUST2;
READ FILE (TASKS) INTO (ATASKS) KEY (NREC);
NC = NC + 1;

/* NOW, FOR NEW LINE(S) OUTPUT ; FIRST TRY TERMINAL */

STEP6: IF ( DISP = 2 ) THEN GO TO STEP7;
LP2: DJ;
IF ( LT > MAXLTA ) THEN SIGNAL CONDITION (NEWDP);
DISPLAY ( BLANK ) ;
PUT STRING (DLINE) EDIT
( NC, TASKOVER.RATING, TASKOVER.RANK, TASKOVER.JOBTASK, BILCOST )
( F(4), X(3), A(5), X(2), A(2), X(2),
A(6), X(2), P'$$,$$$,$$$,$$$,V999 ) ;
DISPLAY (DLINE) ;
DISPLAY (1101" II TASKTTL ) ;
LT = LT + 3 ;
GO TO STEP7 ;
END LP2 ;

IF ( LT > MAXLTB ) THEN SIGNAL CONDITION (NEWDP) ;
IF ( (D2 = "**") | (D2 = "---") ) THEN
DJ ; DISPLAY ( ) ; LT = LT + 1 ; END ;
DISPLAY ( DLINE ) ;
LT = LT + 1 ;

/* HERE, TRY FOR PRINTER OUTPUT */

STEP7: IF ( PTR = 2 ) THEN GO TO TALLY ;
LP3: DJ ;
IF ( PARM = "B" ) THEN
IF ( (D2 = "**") | (D2 = "---") ) THEN PUT SKIP (1) ;
PJT EDIT ( DLINE ) ( SKIP(1), X(10), A(80) ) ;
GO TO TALLY ;
END LP3 ;
PUT EDIT ( NC, TASKOVER.RATING, TASKOVER.RANK, TASKOVER.JOBTASK, BILCOST, TASKTTL )
( SKIP(2), X(10), P'$$,Z999, X(3), A(5), X(4), A(2),
X(3), A(6), X(3), P'$$,$$$,$$$,$$$,V999, X(3), A(50) ) ;

/* HERE, RECYCLE FOR NEXT RECORD, IF ANY */
TAEG REPORT NO. 40

SOURCE CARD LISTING FOR P9A: /* ABBREVIATED TASK FILE LISTINGS - FR1/4/FEB/77 */

TALLY: IF ( PARM = 'A' ) THEN GO TO STEP5 ;
TALLYB: IF ( NREC >= DEND ) THEN GO TO EOFAB ; ELSE GO TO LOOP ;

="/MAX KEY EXCEEDED ERRORS PROCESSED IN THIS SECTION="/ 

BUST1: ENQ = '01' ;
BUST2: ENQ = '02' ;
ERCOM: ENREC = NREC ;

IF ( DISP = 2 ) THEN
   LP*: DO ;
      DISPLAY ( BLANK ) ;
      DISPLAY ( BLANK ) ;
      DISPLAY ( ELINE ) ;
      DISPLAY ( BLANK ) ;
END LP* ;

IF ( PTR = 1 ) THEN
   PJT LIST ( ELINE ) SKIP (3) ;

="/HERE IS PROGRAM WINDUP="/ 

EOFAB: CLOSE FILE ( DIND ), FILE ( MREF ), FILE ( TASKS ),
       FILE ( DTASK ) ;

END P9A ;
DECLARE 1 RNIT, /* ORIGINAL NITRAS TAPE RECORD */
  2 F1 CHAR (36),
  2 FILL1 CHAR (6),
  2 PC CHAR (2),
  2 FILL2 CHAR (1),
  2 F2 CHAR (7),
  2 FILL3 CHAR (9),
  2 STCD CHAR (1),
  2 STDE FIXED DEC (5),
  2 FILL4 CHAR (8),
  2 ATTR FIXED DEC (3,1),
  2 STBK FIXED DEC (3,1),
  2 THR1S FIXED DEC (5),
  2 LHR1S FIXED DEC (5),
  2 FILL5 CHAR (3),
  2 F3 CHAR (6),
  2 FILL6 CHAR (42),
  2 CLEN FIXED DEC (5),
  2 FILL7 CHAR (663);

DECLARE 1 RACRS, /* ABBREVIATED COURSE RECORD */
  /* (CARD-IMAGE) */
  2 F1 CHAR (36),
  2 FILL1 CHAR (6),
  2 PC CHAR (2),
  2 F2 CHAR (7),
  2 STCD CHAR (1),
  2 STDE PICTURE '59',
  2 F3 CHAR (6),
  2 ATTR PICTURE '99V99',
  2 STBK PICTURE '99V99',
  2 CLEN PICTURE '999',
  2 THR1S PICTURE '999',
  2 LHR1S PICTURE '999',
  2 CDBB PICTURE '59V99',
  2 CC80 CHAR (1);

DECLARE 1 OVERLAY BASED (POV),
  2 TAPCDP CHAR (4),
  2 TAPCIN CHAR (8),
  2 TAPCST CHAR (16);

DECLARE ( TC, CC, NTOT, NDONE, DDONE, NSKP, TOTCIN, CINER,
  ( 'NITRAS COURSE SHORT TITLE' ));

DECLARE TI CHAR (44) STATIC INIT ('DESCRIPTOR CIN NITRAS COURSE' ),
  T2 CHAR (64) STATIC INIT ('CARD NUM COUNT CIN REC NUM CDP COURSE SHORT TITLE' );

DECLARE ROATE CHAR (6) STATIC,
  GDATE CHAR (8) STATIC,
  RTIME CHAR (9) STATIC,
SOURCE CARD LISTING FOR
P14: /* NITRAS CDP EXTRACT VIA DESCRIPTOR CIN - MON/13/DEC/76 */

GTIME CHAR (4) STATIC;

FMT1: FORMAT ( X(9), P'Z',229', A ) ;

***************************************************************************

START: CCBO = 1 ;
CAOB = 0 ;
POV = ADDR (RNIT) ;
NC, NDONE, DDONE, TOTCIN, CINER = 0 ;
RDATE = DATE ;
GDATE = SUBSTR (RDATE,3,2) ;
SUBSTR (RDATE,5,2) ;
SUBSTR (RDATE,1,2) ;
RTIME = TIME ;
GTIME = SUBSTR (RTIME,1,4) ;
OPEN FILE (NITRAS) RECORD INPUT,
FILE (DESCR) RECORD INPUT,
FILE (COURSE) RECORD OUTPUT ;
ON ENDFILE (NITRAS) GO TO EOFN ;
ON ENDFILE (DESCR) GO TO EOFD ;
***************************************************************************

DISPLAY ( ' ' ) ;
DISPLAY ( 'ABBREVIATED COURSE DATA BASE EXTRACT PROGRAM' ) ;
DISPLAY ( ' ' ) ;
DISPLAY ( ' ' ) ;
DISPLAY ( T1 ) ;
DISPLAY ( T2 ) ;
***************************************************************************

READ FILE (NITRAS) INTO (RNIT) ;
TCIN = TAPCIN ;
TC, CC = 1 ;

READ FILE (DESCR) INTO (DCARD) ;
CCIN, PFIRST = CRDCIN ;

STEP1: IF ( TCIN = CCIN ) THEN GO TO TEST2 ;
YES: RACRS = RNIT, BY NAME ;
WRITE FILE (COURSE) FROM (RACRS) ;
NC = NC + 1 ;
IF ( PFIRST = (8)' ' ) THEN GO TO STEP2 ;
TOTALCIN = TOTCIN + 1 ;

GET STRING (DLINE) EDIT
( CC, TOTCIN, PFIRST, TC, TAPCDP, TAPCST )
( X(11), P'Z',229', X(5), P'Z',229', X(2), A(8),
X(3), P'Z',229', X(4), A(4), X(4), A(16) ) ;
DISPLAY ( ' ' ) ;
***************************************************************************

STEP2: PUT STRING (DLINE) EDIT
( TC, TAPCDP, TAPCST )
( X(39), P'Z',229', X(4), A(4), X(4), A(16) ) ;
DISPLAY ( DLINE ) ;
SOURCE CARD LISTING FOR P14: /* NITRAS COP EXTRACT VIA DESCRIPTOR CIN - MON/13/DEC/76 */

STEP3: READ FILE (NITRAS) INTO (RNIT);
TC = TC + 1;
IF ( TAPCIN = TCIN ) THEN GO TO YES;
TCIN = TAPCIN;
/* HERE, TAPE CIN HAS ADVANCED */

STEP4: READ FILE (DESCR) INTO (DCARD);
CC = CC + 1;
IF ( CRDCIN = CCIN ) THEN GO TO STEP4;
GO TO STEP1;
/* HERE, CINS NO COMPARE; TRY CCIN BEHIND TCIN */

TEST2: IF ( CCIN > TCIN ) THEN GO TO STEP5:
SRET = STEP4;
GO TO CINGONE;
/* HERE, TAPE CIN IS BEHIND CARD CIN */

STEP5: READ FILE (NITRAS) INTO (RNIT);
TC = TC + 1;
TCIN = TAPCIN;
GO TO TEST1;

EOFN: DDONE = TC;
SRET = STEP6;
DISPLAY (' ');
DISPLAY ('** EOF ON NITRAS-TAPE INPUT **');
IF ( DDONE > 0 ) THEN GO TO QUIT;
IF ( PFIRST = (8) ) THEN GO TO CINGONE;

STEP6: READ FILE (DESCR) INTO (DCARD);
CC = CC + 1;
/* NOTE - FALLTHRU TO CINGONE SUBROUTINE, BELOW */

CINGONE: TOTCIN = TOTCIN + 1;
PUT STRING (DLINE) EDIT
('**', CC, TOTCIN, CCIN, ' - NO CIN MATCH IN NITRAS - ')
(X(4), A, X(3), P'Z', ZZ9', X(5), P'Z', ZZ9',
X(2), A(8), X(2), A);
DISPLAY (' ');
DISPLAY (DLINE);
CINER = CINER + 1;
GO TO SRET;

/* HERE, EOF ON CARD-DESCRIPTOR INPUT */
SOURCE CARD LISTING FOR Pi4: /* NITRAS CDP EXTRACT VIA DESCRIPTOR CIN - MON/13/DEC/76 */

EOFD: DDONE = CC;

DISPLAY ('** EOF ON CARD-DESCRIPTOR INPUT **');
G0 TO QUIT;

QUIT: DISPLAY (''); DISPLAY ('');
PUT STRING (DLINE) EDIT
(CC, ': TOTAL DESCRIPTOR CARDS READ') (R(FMT1));
DISPLAY (DLINE);
PUT STRING (DLINE) EDIT
(TOTCIN, ': TOTAL CIN NUMBERS INPUT') (R(FMT1));
DISPLAY (DLINE);
PUT STRING (DLINE) EDIT
(CINER, ': TOTAL CIN ERRORS (NO MATCH IN NITRAS)')
(R(FMT1));
DISPLAY (DLINE);
PUT STRING (DLINE) EDIT
(NC, ': TOTAL COURSES EXTRACTED') (R(FMT1));
DISPLAY (DLINE);
PUT STRING (DLINE) EDIT
(TC, ': LAST RECORD READ IN NITRAS') (R(FMT1));
DISPLAY (DLINE);
DISPLAY ('');
CLOSE FILE (NITRAS), FILE (DESCR), FILE (COURSE);
DISPLAY ('');
DISPLAY ('COURSE DATABASE CONSTRUCT PROGRAM IS NOW TERMINATING');
DISPLAY ('');
END Pi4;
TAEG REPORT NO. 40

PAGE NUMBER : 1

SOURCE CARD LISTING FOR
P16: /* DESCRIPTOR MASTER INDEX MAKEUP - FRI/14/JAN/77 */

P16: /* DESCRIPTOR MASTER INDEX MAKEUP - FRI/14/JAN/77 */
PROC OPTIONS (MAIN);
/* REVISIONS :
FRI/14/JAN/77 - FIRST VERSION IMPLEMENTED.
MON/14/MAR/77 - MODS TO PREPARE FOR SHIP TO NCSS. */

DECLARE 1 DINDEX BASED (PIN);
2 DSTART FIXED BIN,
2 DEND FIXED BIN,
2 NCAT FIXED BIN,
2 NCAT FIXED BIN,
2 NODESC FIXED BIN,
2 CATEGORIES (15),
3 PCAT FIXED BIN,
3 SCAT FIXED BIN,
2 PCAT FIXED BIN,
2 PCAT FIXED BIN,
2 PDESC (111) FIXED BIN,
( ICOURSE(160), IVEH(160), ITASK(160) )
COVER (4) CHAR (80) BASED (PCOV);
DECLARE CARD CF2 CHAR (80) STATIC,
2 CTYP CHAR (2) DEF CARD POS (1),
2 CTYP CHAR (2) DEF CARD POS (4),
2 TREC PICTURE '9999' DEF CARD,
2 FLAG (3) BIT (1) STATIC,
( NREC, NER, NOUT, NC, NC, NPAGE,
J, K, L ) FIXED BIN STATIC;
DECLARE 1 MAST STATIC,
2 REF (40) FIXED BIN ;
DECLARE TTESTS (3) CHAR (8) STATIC INIT
' COURS', ' VEHICLES', ' TASKS' },,
BLANK CHAR (4) STATIC INIT ( ' ' ),
DLINE CHAR (80) STATIC,
POINT (3) POINTER,
( ROUTE, RPRINT ) LABEL ;
DECLARE 1 CARBIN STATIC,
2 CNJMS (40) FIXED BIN ;
DECLARE EMSG (7) CHAR (38) STATIC INIT ( 'FIRST CARD NOT (--)-TYPE CARD',
'DUPLICATE (--)-TYPE CARD',
'TYPE OF (--)-CARD INVALID',
'TYPE (**) CARD MUST FOLLOW TYPE (--)-',
'CATEGORY HAS NO DESCRIPTORS',
'DESCRIPTOR NUMBER CONVERSION ERROR',
'CATEGORY TABLE SIZE EXCEEDED, MAX = 15' ) ;

/*****************************/

START: ON ERR3R GO TO QUIT ;
OPEN FILE (CARDS) RECORD INPUT ;
ON ENDFILE (CARDS) GO TO EOFCARDS ;
NREC = 13 ;
NER, NC,
NPAGE = 0 ;
ICOURSE; IVEH; ITASK = 0 ;
FLAG = '0' B ;
POINT (1) = ADDR (ICOURSE) ;

D.6-108
SOURCE CARD LISTING FOR
P16: /* DESCRIPTOR MASTER INDEX MAKEUP - FRI/14/JAN/77 */

POINT (2) = ADDR (lVEHL);  
POINT (3) = ADDR (lTASK);  

DISPLAY (' ' );  
DISPLAY (' ', 'MASTER DESCRIPTOR INDEX CONSTRUCT', 'PROGRAM IS NOW STARTING' );

*******************************************************************************/

/* VERY FIRST CARD PROCESSED HERE */
READ FILE (CARDS) INTO (CARD);

ER1: NER = 1 ;  IF ( CF2 = '---' ) THEN GO TO QUIT ;
/* HERE, CONTENTS OF (--) CARD VERIFIED */

STEP1: DO K = 1 TO 3 ;  
IF ( CTYP = 'TESTS(K)' ) THEN GO TO ENDS1 ;

J = K ;  
ER2: NER = 2 ;  IF ( FLAG(J) ) THEN GO TO QUIT ;

PIN = POINT(J) ;
FLAG(J) = 'B';
GO TO Pass1 ;

ENDS1: END STEP1 ;

ER3: NER = 3 ;  GO TO QUIT ;
/* HERE, START PROCESSING NEW (--) TYPE */

PASS1: DSTART = NREC ;

R1: READ FILE (CARDS) INTO (CARD) ;
NREC = NREC + 1 ;  NC = NC + 1 ;  IF ( CF2 = '---' ) THEN GO TO R1 ;

ER4: NER = 4 ;  IF ( CF2 = '***' ) THEN GO TO QUIT ;
/* HERE, STARTING TO PROCESS NEW (**) CATEGORY */

STEP2: NCAT = NCAT + 1 ;
PCAT (NCAT) = NREC ;

R2: READ FILE (CARDS) INTO (CARD) ;
NREC = NREC + 1 ;  NC = NC + 1 ;  IF ( CF2 = '---' ) THEN GO TO R2 ;

ER5: NER = 5 ;  IF ( CF2 = '***' ) | ( CF2 = '---' ) THEN GO TO QUIT ;
/* NOW, DESCRIPTORS WITHIN A CATEGORY START */

STEP3: NDESC = NDESC + 1 ;
SCAT (NCAT) = N ;

R3: READ FILE (CARDS) INTO (CARD) ;
NREC = NREC + 1 ;  NC = NC + 1 ;  IF ( CF2 = '---' ) THEN GO TO R3 ;

IF ( CF2 = '***' ) THEN GO TO STEP4 ;
SOURCE CARD LISTING FOR
P16: /* DESCRIPTOR MASTER INDEX MAKEUP - FRI/14/JAN/77 */

IF ( CF2 = '---' ) THEN GO TO STEP5 ;

ER6A: NER = 6 ;
GET STRING (CF2) EDIT (N) ( F(2) ) ;
GO TO STEP3 ;

/* HERE, DONE WITH ONE CATEGORY - RECYCLING TO NEXT */

STEP4: NER = 7 ;
ER7: IF ( NCAT > 14 ) THEY GO TO QUIT ;
ECAT (NCAT) = N ;
GO TO STEP2 ;

/* NOW, DONE ONE (--) TYPE - RECYCLING TO NEXT */

STEP5: ECAT (NCAT) = N ;
DEND = NREC - 1 ;
GO TO STEP1 ;

EOFCARDS: /* NOW, EOF ON INPUT DESCRIPTOR CARDS */
MAXREC = NREC ;
DEND = NREC ;
ECAT (NCAT) = N ;
CLOSE FILE (CARDS) ;
OPEN FILE (CARDS) RECORD INPUT ;
FILE (MREF) RECORD UPDATE ;
FILE (DIND) RECORD OUTPUT ;
ON ENDFILE (CARDS) GO TO EOJ ;
READ FILE (MREF) INTO (MAST) ;
REFS (4) = NREC ;
REWRITE FILE (MREF) FROM (MAST) ;

/* FIRST OUTPUT HEADER RECORD */
CNUMS = 0 ;
CNUMS(1) = MAXREC ;
WRITE FILE (DIND) FROM (CARBIN) ;
NOUT = 0 ;

/* NOW, FOR THE COURSE INDEX */
PCOV = ADDR (ICOURSE) ;
PIN = ADDR (ICOURSE) ;
LP1: DO J = 1 TO 4 ;
CARD = COVER (J) ;
WRITE FILE (DIND) FROM (CARD) ;
END LP1 ;
NOUT = NOUT + 4 ;

/* VEHICLE INDEX IS OUTPUT HERE */

LP2: DO J = 1 TO 4 ;
CARD = COVER (J) ;
WRITE FILE (DIND) FROM (CARD) ;
END LP2 ;
SOURCE CARD LISTING FOR P16: /* DESCRIPTOR MASTER INDEX MAKEUP - FRI/14/JAN/77 */

NOUT = NOUT + 4;

/* NOW FOR THE TASK INDEX OUTPUT */

PCOV = ADDR (ITASK);
PIN = ADDR (ITASK);

LP3: DO J = 1 TO 4;
CARD = COVER (J); WRITE FILE (DIND) FROM (CARD);
END LP3;
NOUT = NOUT + 4;

/* NOW, OUTPUT THE DESCRIPTOR CARD FILE */

DISPLAY ( "DESCRIPTOR CARD FILE LISTING FOLLOWS" );
DISPLAY ( "DESCRIPTOR CARD\n" );
NCC = 0;

RALL: READ FILE (CARDS) INTO (CARD);
NCC = NCC + 1;
WRITE FILE (DIND) FROM (CARD);
NOUT = NOUT + 1;
IF ( CF2 = "---" ) THEN DJ; DISPLAY ( "" ); DISPLAY ( "" ); END;
IF ( CF2 = "***" ) THEN DISPLAY ( "" );
DISPLAY ( CARD );
GO TO RALL;

QUIT: /* THIS IS THE FORCED-TERMINATION ROUTINE */

PUT STRING (DLINE) EDIT
("** ERROR NUMBER , NER,  HAS OCCURED **")
(A, F(2), A);
DISPLAY ( "" ); DISPLAY ( DLINE );
DLINE = EMSG ( NER );
DISPLAY ( DLINE );
DISPLAY ( "" );
GO TO EOJ;

EOJ: CLOSE FILE (CARDS), FILE (DIND),
FILE (MREF);
DISPLAY ( "" );
PUT STRING (DLINE) EDIT

D.6-111
SOURCE CARD LISTING FOR P16:

```
/* DESCRIPTOR MASTER INDEX MAKEUP - FRI/14/JAN/77 */

('** TOTAL DESCRIPTOR INDEX CARDS READ : ',
   NCC, ' **' ) ( A, 'P' Z, 'Z9', A ) ;

DISPLAY ( DLINE ) ;

DISPLAY ( ' ' ) ;

DISPLAY ( 'MASTER DESCRIPTOR INDEX PROCESSING ' ) ;

DISPLAY ( ' IS NOW TERMINATING ' ) ;

DISPLAY ( ' ' ) ;

END P16 ;
```
P17: /* PROJECT FILE INITIALIZATION - WED/23/FEB/77 */

DECLARE INPARM CHAR (100) VARYING;
DECLARE PARM2 CHAR (4) BASED (PP);
DECLARE DISP, PTR, NPAGE, DPAGE, L, LT, NC, J, K ) FIXED BIN STATIC;
DECLARE DTYPE (20) CHAR (4) STATIC INIT (/* FIRST 13 ARE CARD */
"ID", "REC", "REJ", "REV", /* IMAGE TYPE */
"EXD", "BPT", "RKP", "RRPJ", "ALTP",
"REE", "EXC", "EXJ", "EXV", /* NOS 14 THRU 20, */
"CMR", "TRER", "FINR" );
DECLARE ( FLAGS(20), IFLAG ) BIT (1) STATIC;
DECLARE ( CARD, DLINE ) CHAR (80) STATIC;
DECLARE T1 CHAR (18) STATIC INIT ("ID - PROJECT DESCRIPTION",
"REC - COURSES ROE SEARCH ARGUMENTS",
"REJ - TASKS ROE SEARCH ARGUMENTS",
"REV - VEHICLES ROE SEARCH ARGUMENTS",
"EXD - EXTRACT DEFAULTS",
"BPT - BENEFIT PATTERN",
"RKP - RISK PROFILE",
"RRPJ - RISK REDUCTION PROJECTS",
"VQAL - VARIABLES QUALITATIVE",
"VARF - VARIABLE REFERENCES",
"SCEN - SCENARIOS",
"ALTP - ALTERNATE PROJECTS",
"REE - ROE SEARCH RESULTS",
"EXC - EXTRACT DB - COURSES",
"EXJ - EXTRACT DB - JOBTASKS",
"EXV - EXTRACT DB - VEHICLES",
"CMR - COST MODEL RESULTS",
"TRER - TREE RESULTS",
"FINR - FINANCIAL RESULTS" );
DECLARE T1 CHAR (18) STATIC INIT ("PAGE NUMBER : XXXX" );
DECLARE T2 CHAR (80) STATIC INIT ("LOAD OF PROJECT FILETYPE : XXXX" );
DECLARE T2TTL CHAR (53) DEF T2 POS (28);
SOURCE CARD LISTING FOR
P17: /* PROJECT FILE INITIALIZATION - WED/23/FEB/77 */

START:
  FLAGS = '0'B ;
  NC, NPAGE, DPAGE,
  J, K = 0 ;
  PP = ADDR (INPARAM) ;
  DISPLAY ( ' ' ) ;
  DISPLAY ( 'PROJECT FILE INITIALIZATION PROGRAM IS STARTING' ) ;
  DISPLAY ( ' ' ) ;
  /* HERE, DEVICE ASSIGNMENTS ARE VERIFIED */
  FIRST:
    DISP, PTR = 1 ;
    IF ( PARM2 = 'BOTH' ) THEN GO TO STEPA ;
    IF ( PARM2 = 'TERM' )
      THEN PTR = 2 ; ELSE DISP = 2 ;
    /* HERE, PRINTER DATASET IS INITIALIZED */
    STEPA:
      IF ( PTR = 2 ) THEN GO TO STEPB :
      OPEN FILE (SYSPRINT)
        LINESIZE (120) PAGESIZE (58) ;
      ON ENDPAGE (SYSPRINT)
        BEGIN :
        NPAGE = NPAGE + 1 ;
        PJT EDIT ( 'PAGE NUMBER : ', NPAGE, T2, ' ' )
          ( PAGE, X(10), A, F(4),
            SKIP(1), X(10), A, SKIP(1), A ) ;
        PJT SKIP (1) :
        END ;
        /* HERE, INITIALIZATION OF TERMINAL DATASET PROCEEDS */
        STEPB:
          IF ( DISP = 2 ) THEN GO TO COMMENCE :
          ON CONDITION (NEWDP)
            BEGIN :
            IF ( LT < 61 ) THEN LT = 61 ;
            IF ( LT < 66 ) THEN
              DO L = LT TO 66 ;
              DISPLAY ( ' ' ) ;
              END LPA :
              DISPLAY ( ' ' ) ;
              DPAGE = DPAGE + 1 ;
              TIPGE = DPAGE ;
              DISPLAY ( T1 ) ;
              DISPLAY ( T2 ) ;
              DISPLAY ( ' ' ) ;
              LT = 4 ;
              END ;
              /************************ */
              COMMENCE:
                OPEN FILE (CARDS) RECORD INPUT:
          ON ENDFILE (CARDS) GO TO PASS2 :
                READ FILE (CARDS) INTO (CARD) :
                NC = NC + 1 ;
                IFLAG = '0'B ;
IF ( F4 = 'INIT' ) THEN

DO ;
IFLAG = '1'B ;
READ FILE (CARDS) INTO (CARD) ;
NC = NC + 1 ;
END LP1 ;

IF ( F2 = '***' ) THEN GO TO STEP2 ;

/* HERE, FIRST CARD READ WAS NOT '***' CARD */
T2TTL = '*** FIRST TITLE CARD IS MISSING ***' ;

STEP1: IF ( PTR = 1 ) THEN SIGNAL ENDPAGE (SYSPRINT) ;

IF ( DISP = 1 ) THEN SIGNAL CONDITION (NEWDP) ;
ELSE DISPLAY ( T2 ) ;

R1: READ FILE (CARDS) INTO (CARD) ;
NC = NC + 1 ;
IF ( F2 = '***' ) THEN GO TO STEP2 ;
IF ( PTR = 1 ) THEN PUT EDIT (CARD) [ SKIP(1), X(10), A(80) ] ;
IF ( DISP = 1 ) THEN DISPLAY ( CARD ) ;
GO TO R1 ;

/* HERE, A TITLE CARD HAS BEEN FOUND - VERIFY IT */

STEP2: DO J = 1 TO 13 ;
K = J ;
IF ( FN = DTYPE(J) ) THEN GO TO STEP3 ;
END STEP2 ;

T2TTL = '*** TITLE CARD FILENAME = ' || FN || ' - IS INVALID ***' ;
GO TO STEP1 ;

/* HERE, NEW TITLE (FILE) HEADER FOUND; VERIFY NON-DUPLICATE */

STEP3: IF ( FLAGS(K) ) THEN
DUPFN: DO ;
T2TTL = '*** FILENAME = ' || FN || ' - IS DUPLICATED ***' ;
GO TO STEP1 ;
END DUPFN ;

FLAGS ( K ) = '1'B ;
T2TTL = TTLS ( K ) ;

FNME = DTYPE ( K ) ;
OPEN FILE (PROJ) RECORD OUTPUT TITLE ( FNME ) ;

IF ( PTR = 1 ) THEN SIGNAL ENDPAGE (SYSPRINT) ;
IF ( DISP = 1 ) THEN SIGNAL CONDITION (NEWDP) ;
ELSE DISPLAY ( T2 ) ;

/* HERE IS TEXT LINE OUTPJT LOOP */

R2: READ FILE (CARDS) INTO (CARD) ;
NC = NC + 1 ;
TAEG REPORT NO. 40

PAGE NUMBER : 4

SOURCE CARD LISTING FOR
P17: /* PROJECT FILE INITIALIZATION - WED/23/FEB/77 */

IF ( F2 = "***" ) THEN
207  LP2:
208 DO;
209  CLOSE_FILE (PROJ);
210  GO TO STEP2;
211  END LP2;
212 IF ( PTR = 1 )
213  THEN PUT EDIT (CARD) ( SKIP(1), X(10), A(80) ) ;
214 IF ( DISP = 1 )
215  THEN DISPLAY ( CARD ) ;
216 WRITE_FILE (PROJ) FROM (CARD) ;
217 GO TO R2 ;
218
219 /* HERE, EOF ON INPUT DATASET */
220 PASS2:  CLOSE_FILE (PROJ), FILE (CARDS) ;
221 IF ( ~IFLAG ) THEN GO TO EOJ ;
222 CARD = "" ;    /* DUMMY OUTPUT RECORD FOR FILE INIT */
223 T2 = "INITIALIZATION OF REMAINING FILES (IF ANY) WITH A '" SINGLE BLANK RECORD" ;
224 IF ( PTR = 1 ) THEN SIGNAL ENDPAGE (SYSPRINT) ;
225 IF ( DISP = 1 ) THEN SIGNAL CONDITION (NEWDP) ;
226 LP3:
227 DO J = 1 TO 13 ;    /* NOTE - ONLY INITIALIZING CARD IMAGE FILES UP THRU ALTP, INCL. */
228 IF ( FLAGS(J) ) THEN GO TO LP3END ;
229 OPEN_FILE (PROJ) RECORD OUTPUT TITLE ( Dtype(J) ) ;
230 IF ( J <= 13 ) THEN GO TO WRITE80 ;
231 ELSE GO TO WAY (J) ;
232 WAY(14):  GO TO STEP4 ;
233 WAY(15):  GO TO STEP4 ;
234 WAY(16):  GO TO STEP4 ;
235 WAY(17):  GO TO STEP4 ;
236 WAY(18):  GO TO STEP4 ;
237 WAY(19):  GO TO STEP4 ;
238 WAY(20):  GO TO STEP4 ;
239 WAY(21):  GO TO STEP4 ;
240 WAY(22):  GO TO STEP4 ;
241 WAY(23):  GO TO STEP4 ;
242 WAY(24):  GO TO STEP4 ;
243 WAY(25):  GO TO STEP4 ;
244 WAY(26):  GO TO STEP4 ;
245 WAY(27):  GO TO STEP4 ;
246 WAY(28):  GO TO STEP4 ;
247 WAY(29):  GO TO STEP4 ;
248 WAY(30):  GO TO STEP4 ;
249 WAY(31):  GO TO STEP4 ;
250 WAY(32):  GO TO STEP4 ;
251 WAY(33):  GO TO STEP4 ;
252 WAY(34):  GO TO STEP4 ;
253 WAY(35):  GO TO STEP4 ;
254 WAY(36):  GO TO STEP4 ;
255 WAY(37):  GO TO STEP4 ;
256 WAY(38):  GO TO STEP4 ;
257 WAY(39):  GO TO STEP4 ;
258 WAY(40):  GO TO STEP4 ;
259 WRITE80:  WRITE_FILE (PROJ) FROM (CARD) ;
260 STEP4:
261  CLOSE_FILE (PROJ) ;
262  DLINE = "FILE : " Dtype(J) " HAS BEEN INITIALIZED WITH ONE (1) BLANK RECORD" ;
263  IF ( PTR = 1 ) THEN PUT EDIT (DLINE) ( SKIP(2), X(10), A ) ;
264  DISPLAY ( ) ; DISPLAY ( DLINE ) ;
265 LP3END:  END LP3 ;
266
267 /********************************************************************************/
SOURCE CARD LISTING FOR P17: /* PROJECT FILE INITIALIZATION - WED/23/FEB/77 */

/* HERE IS THE END PROCESSING */

EOJ: IF ( IFLAG ) THEN

LP4: DO ;

IF ( PTR = 1 ) THEN PUT EDIT

( "** ALL DATASETS INITIALIZED **" )
( SKIP(3), X(20), A ) ;

DISPLAY ( ' ' ) ;
DISPLAY ( "** ALL DATASETS INITIALIZED **" ) ;

END LP4 ;

IF ( PTR = 1 ) THEN PUT EDIT

( "** TOTAL NUMBER INPUT CARDS READ : ' , NC, ' **" )
( SKIP(3), X(20), A, F(4), A ) ;

PUT STRING (DLINE) EDIT

( "** TOTAL NUMBER INPUT CARDS READ : ' , NC, ' **" )
( A, F(4), A ) ;

DISPLAY ( ' ' ) ;
DISPLAY ( DLINE ) ;

DISPLAY ( ' ' ) ;
DISPLAY ( "PROJECT FILE INITIALIZATION PROGRAM IS NOW TERMINATING" ) ;

DISPLAY ( ' ' ) ;

END P17 ;
SOURCE CARD LISTING FOR

PI8: /* INITIALIZE MASTER REF FILE TO ZERO - WED/12/JAN/77 */

PROC OPTIONS (MAIN);

/* REVISIONS:
WED/12/JAN/77 - FIRST IMPLEMENTED VERSION.
MON/14/MAR/77 - MODS TO PREPARE FOR SHIP TO NCSS.
*/

DECLARE 1 MAST STATIC;
2 REF (40) FIXED BIN;

OPEN FILE (MREF) RECORD OUTPUT;
REFS = 0;
WRITE FILE (MREF) FROM (MAST);
CLOSE FILE (MREF);
END PI8;
PAGE NUMBER: 1

SOURCE CARD LISTING FOR
P20: /* PRINT ASSIGNED DESIGNATORS - WED/19/JAN/77 */

P20: /* PRINT ASSIGNED DESIGNATORS - WED/19/JAN/77 */
PROC (INPARAM) OPTIONS (MAIN);

/* REVISIONS:
WED/18/JAN/77 - STARTED WITH COURSE DESCRIPTORS ONLY.
MON/31/JAN/77 - ADDING VEHICLE AND TASK VARIATIONS.
SUN/20/MAR/77 - MODS PRIOR TO INSTALLATION AT NCSS.
*/

DECLARE INPARM CHAR (100) VARYING,
1 POVER BASED (PP),
 2 PARM CHAR (1),
 2 PARM2 CHAR (4),
C1 CHAR (1) STATIC,
C4 CHAR (4) STATIC,
( DIND, VEHs, TASKs ) FILE RECORD DIRECT
KEYED ENV ( REGIONAL(1) ) ;

DECLARE DLINE CHAR (80) STATIC,
CFILL CHAR (74) DEF DLINE POS (7),
DFILL CHAR (68) DEF DLINE POS (13),
CARD CHAR (80) STATIC,
TTL1 CHAR (42) STATIC INIT (PAGE XXXX: VEHICLE DESCRIPTOR ASSIGNMENTS),
NPGE PICTURE 'ZZZ9' DEF TTL1 POS (6),
TYPE CHAR (7) DEF TTL1 POS (13),
( DISP, PTR, LL, ND, NC, NPAGE, DPAGE, LT, NREC,
  MAXD, KMAX, J, CFLAG, KD, MAXD2,
  JD, L, LNOWN, NER,
  K ) FIXED BIN STATIC,
SRET LABEL ;

SHARE LABEL:

*******************************************************************************/
DECLARE 1 DINDEX STATIC, /* OVERALL DESCRIPTOR INDEX */
 2 DSTART FIXED BIN, /* CKF.DESC.INDEX */
 2 DEND FIXED BIN,
 2 NCAT FIXED BIN,
 2 NDESC FIXED BIN,
 2 CATEGORIES (15),
 3 PCAT FIXED BIN,
 3 SCAT FIXED BIN,
 3 ECAT FIXED BIN,
 2 PODESC (111) FIXED BIN,
COVER (4) CHAR (80) BASED (PCOV) ;

DECLARE 1 MAST STATIC,
 2 REFS (40) FIXED BIN,
MREF FILE RECORD ;

DECLARE 1 CINDESC BASED (PTACIN), /* COURSE DESCRIPTORS */
 2 CINDS CHAR (8), /* CKF.CRS.DESC */
 2 FILL CHAR (5),
 2 DTAB (100) FIXED BIN,
CIN213 CHAR (213) STATIC ;

DECLARE 1 CINDER (304) STATIC, /* CRS CIN DIRECTORY */
 2 CHCIN CHAR (8), /* CKF.CRS.DIRCIN */
 2 CIPRTR FIXED BIN,
LOADCIN CHAR (760) BASED (PLCIN),
SOURCE CARD LISTING FOR
P20: /* PRINT ASSIGNED DESIGNATORS - WED/19/JAN/77 */

DECLARE 1 VEHDESC STATIC, /* VEH DESCRIPTOR FILE */
  2 VEHDS CHAR (13), /* CKF.VEH.DESC */
  2 VTAB (100) FIXED BIN;

DECLARE 1 VEHDIR (200) STATIC, /* VEHICLE DIRECTORY */
  2 CHVEH CHAR (13), /* CKF.VEH.DIR */
  2 VEHPTR FIXED BIN,
  COVVEH (4) CHAR (750) BASED (PTVEH),
  LOADVEM CHAR (750) BASED (PLVEH);

DECLARE 1 TASKDESC STATIC, /* TASK DESCRIPTOR FILE */
  2 VEHDS CHAR (13), /* CKF.TASK.DESC */
  2 TTAB (100) FIXED BIN,
  1 TASKOVER BASED (PTO),
  2 RATING CHAR (5),
  2 RANK CHAR (2),
  2 JOBTASK CHAR (6);

DECLARE 1 TASKDIR (1200) STATIC, /* JOBTASK FILE DIRECTORY */
  2 CHTASK CHAR (13), /* CKF.TASK.DIR */
  2 TASKPTR FIXED BIN,
  COVTASK (12) CHAR (1500) BASED (PTTASK),
  LOADTASK CHAR (1500) BASED (PLTASK);

DECLARE 1 ATASKS STATIC, /* ABBR. TASK FILE */
  2 RATING CHAR (7), /* CKF.ATASKS */
  2 JOBTASK CHAR (6),
  2 TASKTTL CHAR (50),
  2 FILL CHAR (6),
  2 BILCOST PICTURE '9V999',
  2 CC80 CHAR (1);

DECLARE 1 AVEHICLES STATIC, /* ABBR. VEH FILE */
  2 STOCKN CHAR (13), /* CKF.AVEHS */
  2 DEVDDESIG CHAR (9),
  2 DEVNAME CHAR (47),
  2 DEVCOST PICTURE '9V999',
  2 CC80 CHAR (1),
  VCOST FIXED DEC (10,2) STATIC,
  VOVER CHAR (80) BASED (PV80);

*******************************************************************************/

START: PP = ADDR ( INPARM );
PTCIN = ADDR (CINDER);
PCOV = ADDR (DINDEX);
PTO = ADDR (TASKDESC);
PY80 = ADDR (AVEHICLES);
PTVEH = ADDR (VEHDIR);
PTTASK = ADDR (TASKDIR);
PTRCIN = ADDR (CIN213);
NPAGE, DPAGE, NREC,
K = 0;
LT = 100;

COMMENCE:
OPEN FILE (MREF) RECORD INPUT,
FILE (DIND) INPJST;
READ FILE (MREF) INTO (MAST);
KMAX = REFS (4);
MAXD = REFS (2);
SOURCE CARD LISTING FOR P20: /* PRINT ASSIGNED DESIGNATORS - WED/19/JAN/77 */

/******************** */
/* HERE, COURSE DESCRIPTORS ARE SETUP */
TRYC: IF (PARM = 'C') THEN GO TO TRYV;
    TYPE = 'COURSE';
    LP1: DO J = 1 TO 4;
        READ FILE (DIND) INTO (CARD) KEY (J);
        COVER (J) = CARD;
    END LP1;
    OPEN FILE (DESC) RECORD INPUT,
        FILE (DCIN) RECORD INPUT;
    ON ENDFILE (DESC) GO TO EOJ;
    LP2: DO J = 1 TO 4;
        PLCIN = ADDR (COVCINU));
        READ FILE (DCIN) INTO (LOADCIN);
    END LP2;
    GO TO CHECKDEV;
/* HERE, VEHIICLE DESCRIPTORS ARE INITIILIZED */
TRYV: IF (PARM = 'P') THEN GO TO TRYT;
    TYPE = 'VEHICLE';
    MAXO = REFS (8);
    PTRCIN = ADDR (VEHDESC);
    OPEN FILE (VDESC) RECORD INPUT,
        FILE (VEHS) INPUT;
    ON ENDFILE (VDESC) GO TO EOJ;
    LP3: DO J = 5 TO 8;
        READ FILE (DIND) INTO (CARD) KEY (J);
        COVER (J - 4) = CARD;
    END LP3;
    LP4: DO J = 1 TO 4;
        PLVEH = ADDR ( 'COVVEH(J) ');
        READ FILE (DVEH) INTO (LOADVEH);
    END LP4;
    GO TO CHECKDEV;
/* HERE, SETUP OF TASK DESCRIPTORS */
TRYT: IF (PARM = 'T') THEN GO TO LP5;
   性别 = 1;
    CI = PARM; C4 = PARM2;
    PUT STRING (DLINE) DATA (CI, C4);
    GO TO BUSTED;
    LP5: DO J = 9 TO 12;
        READ FILE (DIND) INTO (CARD) KEY (J);
        COVER (J - 8) = CARD;
    END LP5;
    TYPE = 'JOBTASK';
    MAXO = REFS (12);
    MAXO2 = MAXO / 2;
    PTRCIN = ADDR (TASKDESC);
    OPEN FILE (TDESC) RECORD INPUT,
FILE (TASKS) INPUT,
FILE (DTASK) RECORD INPUT;
ENDFILE (TDESC) GO TO EOJ;

DO J = 1 TO 12;
PLTASK = ADDR (COVTASK(J)));
READ FILE (DTASK) INTO (LOADTASK);
END LP6;

/* HERE, OUTPUT DEVICE ALLOCATION IS CHECKED */

CHECKDEV:
   DISP, PTR = 1;
   IF ( PARM2 = 'BOTH' ) THEN GO TO STEP1;
   IF ( PARM2 = 'TERM' ) THEN PTR = 2;
   ELSE DISP = 2;

/* HERE, PRINTER DATASET INITIALIZED IF REQUIRED */

STEPl: IF ( PTR = 2 ) THEN GO TO STEP2;
OPEN FILE (SYSPRINT) STREAM PRINT
   LINESIZE (120) PAGESIZE (60);
ON ENDPAGE (SYSPRINT)
   BEGIN;
   NPAGE = NPAGE + 1;
   NPGE = NPAGE;
   PUT EDIT (TTL1, ' ', ' ');
   PAGE, LINE(5), X(10), A(42), SKIP(1), A);
   PUT SKIP (1);
END;
SIGNAL ENDPAGE (SYSPRINT);

/* HERE IS THE DISPLAY ENDPAGE ROUTINE, IF REQUIRED */

STEP2: IF ( DISP = 2 ) THEN GO TO DLOOP;
ON CONDITION (NEWDP)
   BEGIN;
   LT = LNOW + 6;
   IF ( LT < 61 ) THEN LT = 61;
   IF ( LT < 66 ) THEN DO L = LNOW TO 66;
   DISPLAY ( ' ');
   END LP7;
   DISPLAY ( ' ');
   DPAGE = DPAGE + 1;
   NPGE = NPGE;
   DISPLAY (TTL1);
   DISPLAY ( ' ');
   LT = 4;
   IF ( PARM = 'C' ) THEN DISPLAY
      ('DESCRIPTORS ASSIGNED TO COURSES');
   IF ( PARM = 'V' ) THEN DISPLAY
      ('DESCRIPTORS ASSIGNED TO VEHICLES');
   IF ( PARM = 'J' ) THEN DISPLAY
      ('DESCRIPTORS ASSIGNED TO TASKS');
   DISPLAY ( ' ');
   LT = LT + 2;
END;
SOURCE CARD LISTING FOR
P20: /* PRINT ASSIGNED DESIGNATORS - WED/19/JAN/77 */

/*****************************/
/* THIS IS THE PROGRAM MAINLINE */

DLOOP: IF ( PARM = 'T' ) THEN GO TO TLOOP ;
       IF ( PARM = 'V' ) THEN GO TO VLOOP ;

CLOOP: READ FILE (DESC) INTO (CINDESC) ;
        NREC = NREC + 1 ;
        DLINE = 'CIN : ' || CINDS ;

LOOKCIN: DO J = 1 TO MAXD ;
         K = J ;
         IF ( CINDS = CHCIN(J) ) THEN GO TO STEP3 ;
         END LOOKCIN ;

SUBSTR ( DLINE, 17 ) = '(NO MATCHING NITRAS DATA AVAILABLE)' ;
GO TO STEP3 ;

/* FOLLOWING IS THE VEHICLE FIRST LINE SETUP */

VLOOP: READ FILE (VDESC) INTO (VEHDESC) ;
      NREC = NREC + 1 ;
      DLINE = VEHDS ;

LOOKV:  DO J = 1 TO MAXD ;
       K = J ;
       IF ( VEHDS = CHVEH(J) ) THEN GO TO FINDV ;
       END LOOKV ;

SUBSTR ( DLINE, 15 ) = 'NO MATCHING DATA IN VEHICLE FILE' ;
GO TO STEP3 ;

FINDV:  READ FILE (VEHS) INTO (AVEHICLES) KEY (VEHPTR(K)) ;
        SUBSTR ( DLINE, 15 ) = '/ ' || DEVDESIG
defname ;
        GO TO STEP3 ;

/* FOLLOWING IS SETUP OF FIRST LINE FOR TASKS */

TLOOP: READ FILE (TDESC) INTO (TASKDESC) ;
      NREC = NREC + 1 ;
      DLINE = TASKOVER.RATING || ' / '
taskover.rank || ' / '
taskover.jobtask ;

IF ( TASKDS <= CHTASK(MAXD2) )
THEN DO J = 1 TO MAXD2 ;
       K = J ;
       IF ( TASKDS = CHTASK(J) ) THEN GO TO FINDT ;
       END ;

ELSE DO J = MAXD2 TO MAXD ;
       K = J ;
       IF ( TASKDS = CHTASK(J) ) THEN GO TO FINDT ;
       END ;

SUBSTR ( DLINE, 23 ) =
'NO MATCHING DATA IN TASK FILE' ;
GO TO STEP3 ;
TAEG REPORT NO. 40

SOURCE CARD LISTING FOR
P20: /* PRINT ASSIGNED DESIGNATORS - WED/19/JAN/77 */

FINDT: READ FILE (TASKS) INTO (ATASKS) KEY (TASKPTR(K)) ;
         SUBSTR ( DLINE, 23 ) = TASKTTL ;
         GO TO STEP3 ;

/* HERE, FIRST LINE IS OUTPUT */

STEP3: LL = 1 ;
         SRET = STEP4 ;
         GO TO OUTLOOP ;

/* NOW, BACK FROM FIRST LINE OUTPUT */

STEP4: ND = 0 ;
         CFLAG = 0 ;
         NC = 1 ;
/* CATEGORIES SS */

STEP5: IF ( ND >= 100 ) THEN GO TO DLOOP ;
         ND = ND + 1 ;
         IF ( DTAB(ND) = 0 ) THEN GO TO STEP5 ;
         JD = DTAB(ND) ;
         KD = PDESC(ND) ;
/* PTR TO DESCRIPTOR CARD */
         IF ( KD = JD ) THEN GO TO STEP6 ;
         NER = 2 ;
         ER2: PUT STRING (DLINE) DATA ( ND, JD, KD, KMAX ) ;
         GO TO BUSTED ;

STEP6: IF ( KD <= KMAX ) THEN GO TO STEP7 ;
         NER = 3 ;
         ER3: GO TO ER2A ;

STEP7: IF ( ND <= SCAT(NC) ) THEN GO TO STEP9 ;

STEP8: NC = NC + 1 ;
         CFLAG = 0 ;
         IF ( NC <= 15 ) THEN GO TO STEP7 ;
         NER = 4 ;
         GO TO ER2A ;

STEP9: IF ( ND <= SCAT(NC) ) THEN GO TO STEP10 ;

STEP10: IF ( CFLAG = 0 ) THEN GO TO DETOUT ;
/* HERE, CATEGORY LINE IS OUTPUT */

CATOUT: READ FILE (DIND) INTO (CARD) KEY (PCAT(NC)) ;
         DLINE = ' ' ;
         CFILL = CARD ;
         LL = 2 ;
         CFLAG = 1 ;
         SRET = DETOUT ;
         GO TO OUTLOOP ;

/* HERE, DETAIL LINE IS OUTPUT */

DETOUT: READ FILE (DIND) INTO (CARD) KEY (KD) ;
         DLINE = ' ' ;
         DFILL = CARD ;
         LL = 3 ;
         SRET = STEPS ;
         GO TO OUTLOOP ;

*******************************************************************************/
OUTLOOP: IF ( DISP = 2 ) /* FIRST, TRY DISPLAY */
    THEN GO TO TRYPTR ;
    IF ( LL = 1 ) THEN
        IF ( LT > 60 )
            THEN DO :
                SIGNAL CONDITION ( NEWDP ) :
                END ;
            ELSE DO :
                LT = LT + 3 ;
                DISPLAY ( • ) : 
                END ;
        GO TO DCCOUT ;
        END LINE1 ;
    IF ( LL = 2 ) THEN
        IF ( LT > 61 )
            THEN DO :
                SIGNAL CONDITION ( NEWDP ) :
                GO TO DCCOUT ;
                END ;
            ELSE DO :
                LT = LT + 2 ;
                GO TO DB2 ;
                END ;
        END LINE2 ;
    IF ( LT > 62 ) THEN SIGNAL CONDITION ( NEWDP ) ;
    ELSE LT = LT + 1 ;

DCCOUT: DISPLAY ( DLINE ) ;

TRYPTR: IF ( PTR = 2 ) /* PRINTER SERVICING ATTEMPTED HERE */
    THEN GO TO SRET ; /* OTHERWISE, RETURN TO CALLER */
    IF ( LL < 3 )
        THEN IF ( LL = 1 ) THEN PUT SKIP ( 2 ) :
            ELSE PUT SKIP ( 1 ) :
        PTROUT: PUT EDIT ( DLINE ) ( SKIP(1), A(80) ) :
        GO TO SRET ; /* RETURN TO SUBROUTINE CALLER */

/*****************/
/* THIS IS THE ERROR PROCESSOR */

BUSTED: PUT STRING ( CARD ) LIST
       ( " ** ERROR NUMBER " II NER II " HAS OCCURED **" ) ;

DCCERR: DO
    DISPLAY ( • ) ;
    DISPLAY ( • ) ;
    DISPLAY ( CARD ) ;
    DISPLAY ( • ) ;
    DISPLAY ( DLINE ) ;
    DISPLAY ( • ) ;
    END DCCERR ;
SOURCE CARD LISTING FOR P20: /* PRINT ASSIGNED DESIGNATORS - WED/19/JAN/77 */

483 IF ( PTR < 1 ) THEN
484 PUT EDIT ( CARD, DLINE )
485 ( SKIP (3), A(80), SKIP(2), A(80) ) ;
486
487 /***********************************************************
488 /* THIS IS THE ENDFILE PROCESSOR */
489 ***********************************************************/

490 EDJ: PUT STRING (DLINE) EDIT
491 { ' ** RECORDS PROCESSED = ', NREC, ' **' }
492 ( A, F(4), A ) ;
493 IF ( DISP = 2 ) THEN GO TO STEP11 ;
494
495          DISPLAY ( ' ' ) ;
496          DISPLAY ( DLINE ) ;
497          DISPLAY ( ' ' ) ;
498          DISPLAY ( ' ' ) ;
499          DISPLAY ( ' ' ) ;
500
501 STEP11: IF ( PTR = 1 ) THEN
502 PUT EDIT ( DLINE ) ( SKIP(3), A(80) ) ;
503
504           CLOSE FILE (DIND), FILE (MREF),
505           FILE (SYSPRINT) ;
506
507 END P20 ;
This technique for probability encoding was reproduced from a report titled PROBABILITY ENCODING IN DECISION ANALYSIS* with the permission of one of its authors, Dr. Carl S. Spetzler.

Appendix B

AN INTERACTIVE COMPUTER INTERVIEW

B.1 Introduction

This appendix contains a sample session with the PEP program (Probability Encoding Program) developed by the Decision Analysis Group at Stanford Research Institute. This program is an interactive computer interview, which is based on the interval method for encoding. The subject is always given two or three intervals on which to bet and is asked to order the intervals in terms of his preferences. The program relies on an algorithm that finds the indifference point(s) by an iterative procedure. An opportunity always exists for the subject to bypass the iteration if he has decided on the indifference point. The program asks seven sets of questions, and the fractiles corresponding to the probability levels 1/6, 1/3, 1/2, 2/3, and 5/6 can be inferred from the answers. In fact, each fractile is encoded twice to provide a consistency check. The distribution is displayed in graphical and tabular form, and the subject is then made aware of any inconsistencies.

B.2 A Sample Session

The easiest way to demonstrate the program is by means of a sample session. The following example required roughly 25 minutes and cost around $9.00. The instructions part took up some of the time; an experienced user would find the length of the session to be between 10 and 15 minutes. All inputs by the subjects are underlined.
PLEASE TYPE YOUR NAME—CARL-AXEL STAHL VON HOLSTEIN
DO YOU NEED INSTRUCTIONS?  YES

THIS PROGRAM MEASURES YOUR PERSONAL JUDGEMENT ABOUT
AN UNCERTAIN QUANTITY. YOUR JUDGEMENT IS THEN INTERPRETED
AS A PROBABILITY DISTRIBUTION. KNOWLEDGE ABOUT PROBABILITY
WILL NOT NECESSARILY HELP YOU. THERE ARE ALSO NO RIGHT
OR WRONG ANSWERS BECAUSE YOUR JUDGEMENT IS UNIQUE.

SO RELAX AND TRY TO RESPOND WITH YOUR JUDGEMENT.

MOST OF THE QUESTIONS WILL BE IN A FORM WHERE YOU NEED
TO MAKE DECISIONS. YOU WILL BE GIVEN EITHER TWO OR
THREE NUMBERED RANGES. YOU WILL BE ASKED TO DECIDE
WHICH OF THE RANGES ARE THE MOST LIKELY.

ONE WAY TO ANSWER THE QUESTIONS IS TO ASK YOURSELF:
"WHICH OF THESE RANGES WOULD I PREFER TO BET ON?"

YOU ARE EXPECTED TO ORDER THE RANGES IN TERMS OF YOUR
PREFERENCES. FOR EXAMPLE, IF ONLY TWO RANGES ARE GIVEN,
YOU SIMPLY TYPE '12' OR '21'. IF YOU ARE GIVEN THREE
RANGES AND IF YOU PREFER 3 TO 2 AND 2 TO 1, YOU TYPE
'321'. IF YOU THINK 3 AND 2 ARE ABOUT EQUALLY LIKELY
BUT PREFER 3 AND 2 TO RANGE 1, TYPE '3=21'.

THE PROGRAM WILL ATTEMPT TO FIND RANGES THAT ARE EQUALLY
LIKELY IN YOUR JUDGEMENT. WHEN YOU HAVE REACHED THIS
POINT TYPE 'EQ' OR '1=2' OR '1=2=3'.

YOU ARE ALSO ALLOWED A NUMBER OF OTHER RESPONSES.

DO YOU WISH TO SEE THE ENTIRE LIST?  YES

RESPONSES  MEANING
-----------  -------
12,21        YOU ARE COMPARING TWO NUMBERED RANGES.
             YOU PREFER THE FIRST NUMBERED RANGE
             TO THE SECOND.
123,321,ETC... YOU ARE COMPARING THREE NUMBERED
             RANGES. YOU PREFER THE FIRST
             NUMBERED RANGE TO THE SECOND AND
             THE SECOND TO THE THIRD, ETC...
1=23,2=31,ETC... YOU ARE AGAIN COMPARING THREE NUM-
             BERED RANGES. THE EQUAL SIGN INDI-
             CATES EQUALLY LIKELY RANGES.
             THE RANGES ARE STILL ORDERED BY
             DESIRABILITY OR PREFERENCE.
EQ,EQUAL    IN YOUR JUDGEMENT THE RANGES ARE EQUALLY
             LIKELY. '1=2=3' AND '1=2' ARE EQUIVALENT
             TO TYPING 'EQ'.
BAC,BACK UP  YOU WISH TO CHANGE A PREVIOUS RESPONSE.
             YOU MAY BACK UP A MAXIMUM OF FIVE LINES
             BY REFERENCING A LINE NUMBER. YOU MAY
             ALSO RETURN TO THE START OF A SECTION OF
             QUESTIONS BY TYPING THE SECTION LETTER.

E-4
TER, TERMINATE
YOU WANT TO JUMP OVER THE REST OF THE QUESTIONS AND SEE THE OUTPUTS.

SKI, SKIP
YOU KNOW WHAT VALUES DIVIDE THE RANGES.
THE PROGRAM WILL REQUEST THAT YOU TYPE THEM IN DIRECTLY. SIMPLY TYPE IN YOUR VALUES WHEN THE '?' APPEARS.

DIF, DIFFERENCE
THE PROGRAM PRINTS THE DIFFERENCES BETWEEN THE RANGE BOUNDARIES.

SUP, SUPPRESS
THE PROGRAM STOPS PRINTING DIFFERENCES.

QUI, QUIT
QUIT ENTIRELY. YOU WANT TO JUMP OVER THE REST OF THE QUESTIONS AND THE OUTPUT

WITH THIS INFORMATION, GO AHEAD AND TRY THE PROGRAM.
ALL OF YOUR RESPONSES MUST BE TERMINATED BY PRESSING THE RETURN KEY. FEEL FREE TO ASK FOR MORE INFORMATION AT ANY TIME. JUST TYPE THE WORD 'HELP'.

WHAT VARIABLE IS TO BE INVESTIGATED? PROJECTED SALES

DOES PROJECTED SALES HAVE UPPER AND LOWER LIMITS? YES
WHAT IS THE UPPER LIMIT? 500
WHAT IS THE LOWER LIMIT? 100

[The program sets a limit to infinity if the limit is said not to exist.]

-------------------------------
<table>
<thead>
<tr>
<th>LINE#</th>
<th>RANGE1</th>
<th>RANGE2</th>
<th>T()</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100.</td>
<td>300.</td>
<td>500.</td>
</tr>
<tr>
<td>2</td>
<td>100.</td>
<td>195.</td>
<td>500.</td>
</tr>
<tr>
<td>3</td>
<td>100.</td>
<td>147.</td>
<td>500.</td>
</tr>
<tr>
<td>4</td>
<td>100.</td>
<td>170.</td>
<td>500.</td>
</tr>
<tr>
<td>5</td>
<td>100.</td>
<td>183.</td>
<td>500.</td>
</tr>
<tr>
<td>6</td>
<td>100.</td>
<td>176.</td>
<td>500.</td>
</tr>
<tr>
<td>7</td>
<td>100.</td>
<td>180.</td>
<td>500.</td>
</tr>
</tbody>
</table>

[180 is the 1/2 - fractile.]
THREE RANGES WERE JUST PRINTED OUT. YOU ARE EXPECTED TO ORDER THESE THREE RANGES IN ACCORDANCE WITH THEIR DESIRABILITY.

FOR EXAMPLE, IF YOU PREFER THE SECOND RANGE TO THE FIRST RANGE AND THE FIRST RANGE TO THE THIRD, YOU SHOULD RESPOND WITH '213'. YOU ARE ALLOWED TO USE AN EQUAL SIGN TO INDICATE EQUALLY LIKELY RANGES, SUCH AS '12=3'. THIS WOULD MEAN YOU CONSIDER THE SECOND AND THIRD RANGES TO BE EQUALLY LIKELY, BUT PREFER THE FIRST RANGE TO BOTH.

<table>
<thead>
<tr>
<th>RANGE1</th>
<th>RANGE2</th>
<th>RANGE3</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM</td>
<td>TO/FROM</td>
<td>TO/FROM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>100.</td>
<td>139.</td>
</tr>
<tr>
<td>2</td>
<td>100.</td>
<td>139.</td>
</tr>
<tr>
<td>3</td>
<td>100.</td>
<td>139.</td>
</tr>
<tr>
<td>3</td>
<td>100.</td>
<td>(39.2)</td>
</tr>
<tr>
<td>4</td>
<td>100.</td>
<td>(59.2)</td>
</tr>
<tr>
<td>5</td>
<td>100.</td>
<td>(49.1)</td>
</tr>
<tr>
<td>6</td>
<td>100.</td>
<td>(54.1)</td>
</tr>
<tr>
<td>7</td>
<td>100.</td>
<td>(50.6)</td>
</tr>
</tbody>
</table>

[157 and 192 are the 1/3- and 2/3-fractiles, respectively.]

<table>
<thead>
<tr>
<th>RANGE1</th>
<th>RANGE2</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM</td>
<td>TO/FROM</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>100.</td>
</tr>
<tr>
<td>2</td>
<td>100.</td>
</tr>
<tr>
<td>3</td>
<td>100.</td>
</tr>
<tr>
<td>4</td>
<td>100.</td>
</tr>
</tbody>
</table>

[125 is the 1/6-fractile since it divides an interval with probability 1/3 into two equally likely intervals.]
[The subject realized that his indifference point should be 127 for the range in Set C.]

---**D**---

<table>
<thead>
<tr>
<th>RANGE1</th>
<th>RANGE2</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM</td>
<td>TO</td>
</tr>
<tr>
<td>192.</td>
<td>346.</td>
</tr>
<tr>
<td>500.</td>
<td>BACK</td>
</tr>
<tr>
<td>WHERE</td>
<td>TO</td>
</tr>
<tr>
<td>192.</td>
<td>346.</td>
</tr>
<tr>
<td>500.</td>
<td>BACK</td>
</tr>
</tbody>
</table>

---**C**---

<table>
<thead>
<tr>
<th>RANGE1</th>
<th>RANGE2</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM</td>
<td>TO</td>
</tr>
<tr>
<td>100.</td>
<td>128.</td>
</tr>
<tr>
<td>157.</td>
<td>SKIP</td>
</tr>
<tr>
<td>100.</td>
<td>?</td>
</tr>
<tr>
<td>127</td>
<td>END</td>
</tr>
</tbody>
</table>

---**D**---

<table>
<thead>
<tr>
<th>RANGE1</th>
<th>RANGE2</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROM</td>
<td>TO</td>
</tr>
<tr>
<td>192.</td>
<td>346.</td>
</tr>
<tr>
<td>500.</td>
<td>12</td>
</tr>
<tr>
<td>265.</td>
<td>500.</td>
</tr>
<tr>
<td>230.</td>
<td>12</td>
</tr>
</tbody>
</table>

THIS ENDS THE FIRST QUESTION SET, A SHORTER SET FOLLOWS—PLEASE CONTINUE.

[230 is the 5/6 fractile. The first four sets of questions have established one value for each of the five fractiles. The next three sets are used for a consistency check. The subject can terminate the interview at this point (and anywhere else, too) and go directly to the outputs.]
### TAEG REPORT NO. 40

<table>
<thead>
<tr>
<th>LINE#</th>
<th>RANGE1</th>
<th>RANGE2</th>
<th>RANGE3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100.</td>
<td>121.</td>
<td>161.</td>
</tr>
<tr>
<td>2</td>
<td>100.</td>
<td>121.</td>
<td>151.</td>
</tr>
<tr>
<td>3</td>
<td>100.</td>
<td>137.</td>
<td>161.</td>
</tr>
<tr>
<td>4</td>
<td>100.</td>
<td>134.</td>
<td>151.</td>
</tr>
<tr>
<td>5</td>
<td>100.</td>
<td>135.</td>
<td>156.</td>
</tr>
</tbody>
</table>

(The whole range was assigned probability 1/2 in Set A. 135 and 156 are therefore the 1/6- and 1/3-fractiles.)

<table>
<thead>
<tr>
<th>LINE#</th>
<th>RANGE1</th>
<th>RANGE2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>157.</td>
<td>174.</td>
</tr>
<tr>
<td>2</td>
<td>157.</td>
<td>183.</td>
</tr>
<tr>
<td>3</td>
<td>157.</td>
<td>179.</td>
</tr>
</tbody>
</table>

(179 is a new value for the median (the 1/2-fractile) since the limits of the range corresponded to the probability levels 1/3 and 2/3 in Set B.)

Don't give up—This is the last one

<table>
<thead>
<tr>
<th>LINE#</th>
<th>RANGE1</th>
<th>RANGE2</th>
<th>RANGE3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>180.</td>
<td>195.</td>
<td>341.</td>
</tr>
<tr>
<td>2</td>
<td>180.</td>
<td>210.</td>
<td>290.</td>
</tr>
</tbody>
</table>

(210 and 290 are new values for the 2/3- and 5/6-fractiles.)

E-8
PLEASE ROLL THE PAPER FORWARD AND HIT CARRIAGE RETURN

---------------------------------------------------------------------------
* PEP : OUTPUT
---------------------------------------------------------------------------
VARIABLE NAME: PROJECTED SALES

PREPARED BY CARL-AXEL STAEL VON HOLSTEIN

<table>
<thead>
<tr>
<th>PROBABILITY</th>
<th>THAT X IS LESS THAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>(****) 500.</td>
</tr>
<tr>
<td>.833</td>
<td>(****) 230.</td>
</tr>
<tr>
<td>.667</td>
<td>(****) 192.</td>
</tr>
<tr>
<td>.500</td>
<td>(****) 180.</td>
</tr>
<tr>
<td>.333</td>
<td>(****) 157.</td>
</tr>
<tr>
<td>.167</td>
<td>(****) 127.</td>
</tr>
<tr>
<td>.000</td>
<td>(****) 100.</td>
</tr>
</tbody>
</table>

[The subject can now see some inconsistencies, especially at the high end. The 5/6-fractile was first inferred to be 230 from Set D and later to be 290 from Set F. The subject now has to reconcile the inconsistencies, but that is done outside of the program.]


Delphi Techniques and Cross-Impact Analysis, a Paper. Author unknown.


DISTRIBUTION LIST

Air Force

Headquarters, Air Training Command (XPTD, Dr. D. E. Meyer)
Headquarters, Air Training Command (XPTIA, Mr. Goldman)
Air Force Human Resources Laboratory, Brooks Air Force Base
Air Force Human Resources Laboratory (Library), Lowry Air Force Base
Air Force Office of Scientific Research/AR (Dr. A. R. Fregly)

Army

Commandant, TRADOC (Technical Library)
Army Research Institute (Dr. Ralph R. Canter, 316C; Dr. Edgar Johnson)

Coast Guard

Commandant, U.S. Coast Guard Headquarters (G-P-1/62; G-RT/81)

Marine Corps

CMC (Code OT)
CGMCDEC (Mr. Greenup)
Director, Marine Corps Institute

Navy

Assistant Secretary of the Navy (R&D) (4E741, Dr. S. Koslov)
CNO (OP-987P7, CAPT Connery; OP-991B, M. Malehorn; OP-987P10, Dr. R. Smith; OP-987, H. Stone)
NAVCOMPT (Code NCD-7)
ONR (Code 458, 455)
ONRBO Boston (J. Lester)
ONRBO Chicago
ONRBO Pasadena (E. E. Gloye)
CMN (MAT-03424, Mr. A. L. Rubinstein)
CNET (01, 00A, N-5 (6 copies))
CNAVRES (Code 02)
COMNAVSEASYSCOM (03, 047C1, 047C12)
COMNAVAIRSYSCOM (03, 340F)
CNET SUPPORT (00, 01A)
CNTECHTRA (0161, Dr. Kerr (5 copies))
CNAVTRA (F. Schufletowski)
COMTRALANT
COMTRALANT (Educational Advisor)
COMTRAPAC
DISTRIBUTION LIST (continued)

Navy (continued)

CO NAVPERSRANDCEN (Code 02, Dr. Regan; Dr. Earl Jones; Library)
NAVPERSRANDCEN Liaison (Code 01H)
Superintendent NAVPGSCOL (Code 2124)
Superintendent Naval Academy Annapolis (Chairman, Behavioral Science Dept.)
CO NAVEDTRAPRODEVcen (AH3)
CO NAVEDTRASUPPCEN NORVA
CO NAVEDTRASUPPCENPAC (5 copies)
CO NAVAEROMEDRSCHLAB (Chief Aviation Psych. Div.)
CO FLTCOMDIRSYSTRACENPAC
CO NAMTRAGRU
CISO, NTTC Corry Station
CO NAVTRAEQIPCECN (N-21, N-215, N-131 (2 copies), N-2211, N-00AF, N-00M, PM TRADE)
Center for Naval Analyses (2 copies)
U.S. Naval Institute (CDR Bowler)

Other DOD

Military Assistant for Human Resources, OAD (E&LS) ODDR&E (COL Henry Taylor)
OASD (I&L) - WR (LT COL Grossel)
Director, Human Resources Office, Defense Advanced Research Projects Agency
(R. Young)
Defense Advanced Research Projects Agency (CTO, Dr. H. F. O'Neil, Jr.)
Institute for Defense Analyses (Dr. Jesse Orlansky)

Non-DOD

Essex Corporation (Dr. John Collins)

Information Exchanges

DDC (12 copies)
DLSIE (James Dowling)
Scientific Technical Information Office, NASA
Executive Editor, Psychological Abstracts, American Psychological Association
ERIC Processing and Reference Facility, Bethesda, MD (2 copies)