AD-A073 037 RAND CORP SANTA MONICA CA F/G 5/1

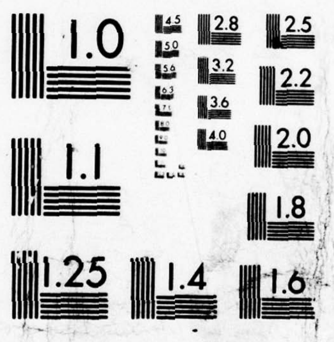
NAVMAN: A MODEL FOR ESTIMATING MAINTENANCF PERSONNFL REQUIREMEN--ETC(U)

JUN 79 B ARMSTRONG, J SCHANK, G BLAIS MDA903-77-C-0107

UNCLASSIFIED RAND/R-2402/2-PA/E NL

1 OF 2

AD
A073037


1.0
4.5
5.0
5.6
6.3
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

R-2402/2-PA&E

June 1979

NAVMAN: A Model for Estimating Maintenance Personnel Requirements for Navy Aircraft: Vol. II, Technical Appendixes

B. Armstrong, J. Schank, C. Blais

A Report prepared for

OFFICE OF THE ASSISTANT SECRETARY OF DEFENSE/ PROGRAM ANALYSIS AND EVALUATION

Rand
SANTA MONICA, CA. 90406

RAND/R-2402/2-PA&E NAVMAN: A MODEL FOR ESTIMATING MAINTENANCE PERSONNEL REQUIREMENTS FOR NAVY AIRCRAFT: VOL. II, TECHNICAL APPENDIXES Armstrong et al.

The research described in this report was sponsored by the Office of the Assistant Secretary of Defense/Program Analysis and Evaluation under Contract No. MDA903-77-C-0107.

Library of Congress Cataloging in Publication Data

Armstrong, Bruce, 1946-
NAVMAN, a model for estimating maintenance personnel requirements for Navy aircraft.

([Report] - Rand Corporation ; R-2402/1-2-PA&E)
Vol. 2 by B. Armstrong, J. Schank, and G. Blais.
CONTENTS: v. 1. Model development and application.--v. 2. Technical appendixes.
1. United States. Navy--Personnel management--Mathematical models. 2. United States. Navy--Personnel management--Data processing. 3. Airplanes, Military--Maintenance and repair--Mathematical models. 4. Airplanes, Military--Maintenance and repair--Data processing. I. Schank, John, 1946- joint author. II. Blais, Gerard, joint author. III. United States. Assistant Secretary of Defense (Program Analysis and Evaluation) IV. Title. V. Series: Rand Corporation. Rand report ; R-2402/1-2-PA&E.
AS36.R3 R-2402/1-2 [VB258] 081s [359.3'015'1]
ISBN 0-8330-0130-2 (v.2) 79-14934

Published by The Rand Corporation

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER R-2402/2-PA&E	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) NAVMAN--A Model for Estimating Maintenance Personnel Requirements for Navy Aircraft: Vol. II, Technical Appendixes		5. TYPE OF REPORT & PERIOD COVERED Interim
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) B. E. Armstrong, J. Schank, G. Blais		8. CONTRACT OR GRANT NUMBER(s) MDA903-77-C-0107
9. PERFORMING ORGANIZATION NAME AND ADDRESS The Rand Corporation 1700 Main Street Santa Monica, CA 90406		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Assistant Secretary of Defense (Program Analysis and Evaluation) Washington, D. C. 20301		12. REPORT DATE June 1979
		13. NUMBER OF PAGES 164
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited

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

No Restrictions

18. SUPPLEMENTARY NOTES

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

NAVMAN
Maintenance Personnel
Aircraft
Computerized Simulation
Naval Personnel
Estimating
Management Planning

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

See Reverse Side

This report is the second of two volumes that describes NAVMAN, a computer model developed for predicting the total organizational and intermediate level maintenance personnel requirements for new U.S. Navy aircraft. This volume presents a detailed description of model operations, program and model variable listings, derivation and definition of model factors, and a reliability/maintainability data bank for current Navy aircraft. See also R-2402/1-PA&E. 164 pp. (Author).

R-2402/2-PA&E
June 1979

NAVMAN: A Model for Estimating Maintenance Personnel Requirements for Navy Aircraft. Vol. II, Technical Appendixes

B. Armstrong, J. Schank, G. Blais

A Report prepared for

OFFICE OF THE ASSISTANT SECRETARY OF DEFENSE/ PROGRAM ANALYSIS AND EVALUATION

Rand
SANTA MONICA, CA. 90406

PREFACE

This two-volume report describes NAVMAN, a computer model for generating estimates of organizational and intermediate-level maintenance personnel requirements for new U.S. Navy aircraft. NAVMAN incorporates into a single framework the diverse methods and factors used by the Navy to estimate below-depot level maintenance personnel requirements. It provides a means that does not now exist in systematic form to estimate these requirements during the early stages of system development--that is, before information about subsystem reliability and maintainability characteristics and other system-peculiar personnel factors is available in detail. Because NAVMAN builds on current Navy methods, it does not provide an independent assessment of what the personnel requirements *should* be. It does provide, however, a reliable approximation of what the detailed Navy methods will eventually generate as requirements.

The development of NAVMAN was sponsored by the Office of the Director of Cost and Economic Analysis, Office of the Assistant Secretary of Defense (Program Analysis and Evaluation). The model is intended primarily for use by Cost and Economic Analysis, and by the Cost Analysis Improvement Group (CAIG) that it chairs, in support of the Defense Systems Acquisition Review Council (DSARC). Among the responsibilities of CAIG and DSARC is critical review of the operating and support cost consequences of the acquisition of new weapon systems. Maintenance personnel requirements are primary contributors to operating and support costs; hence those requirements themselves draw critical review. NAVMAN and a similar model for Air Force tactical aircraft* provide CAIG with an analytic tool for estimating personnel requirements early in the acquisition review process, for assessing the reasonableness of estimates prepared by the military services, and for systematically

---

*See W. S. Furry et al., *MANPOWER: A Model of Tactical Aircraft Maintenance Personnel Requirements: Vol. I, Overview of Model Development and Application; Vol. II, Technical Appendixes*, The Rand Corporation, R-2358/1,2-PA&E, April 1979.

services, and for systematically exploring the effects of changes in certain system and maintenance policy variables on those requirements.

In addition to its use by CAIG, NAVMAN should be helpful to U.S. Navy offices involved in aircraft system personnel determination processes. It should be of special interest to the HARDMAN Project Office, which is concerned with determining the timeliness of Navy personnel requirements. A major conclusion of the HARDMAN study* is that determination of personnel requirements occurs too late in the weapon system acquisition process and fails to address major issues of manpower tradeoffs. HARDMAN recommends developing and implementing analytical tools and models that can define maintenance personnel requirements during the early stages of weapon system development.

Volume I of this report, *Model Development and Application*, provides a complete overview of Navy personnel planning methods and of the features, input requirements, and outputs of NAVMAN. Volume II, *Technical Appendixes*, provides information on detailed model operation, model factors and variables, reliability and maintainability reference information, and a computer program listing.

The methods and factors incorporated in NAVMAN are current as of mid-1978. They are subject to modification, however, for the Navy personnel planning process is a dynamic one and is undergoing important changes. The user of NAVMAN should be aware of the need to update the model periodically.

**Military Manpower versus Hardware Procurement Study (HARDMAN), Final Report*, Chief of Naval Operations, United States Navy, October 1977.

SUMMARY

This volume contains a variety of technical appendixes for those who wish to go beyond a casual use of the model, as described in Volume I.

Appendix A, titled "Detailed Model Operation," describes the logic of the computer program. Also included in this appendix is a description of maintenance responsibilities, personnel requirements determination methods and equations, and a step-by-step explanation of the model's computing routine.

Appendix B presents specific personnel factors that are stored in the program and used for various manning decisions. Factors include those developed to allocate aggregate workloads to specific work centers, minimum manning requirements, and the allocation of total maintenance hours to different types of maintenance functions.

Detailed definition of each NAVMAN variable is provided in Appendix C. Each of the model factors, primarily developed by the Navy, are then explained in Appendix D. The factor descriptions include production delay, administrative support, facilities maintenance, utilities task, personnel roundoff tables, and various personnel allocation functions. Further Navy factors are presented in Appendix E--specifically, the paygrade matrices by work center.

Historical data on the reliability and maintainability of Navy aircraft are provided in Appendix F. This information may be useful for the NAVMAN user as input information where analogous aircraft R&M is appropriate.

Appendix G contains a computer listing of the PL/1 NAVMAN program.

CONTENTS

PREFACE iii

SUMMARY v

Appendix

A. DETAILED MODEL OPERATION 1
   Organizational Maintenance Routine 1
   Intermediate Maintenance Routine 12

B. DEVELOPMENT OF MODEL FACTORS 17
   Percentage Spread of Total Workload to Work Centers 18
   Total Maintenance to Preventive/Corrective Maintenance 18
   Minimum Manning Values for Work Center 230 21
   Total Workload to Production Divisions: Intermediate Level 21
   Intermediate-Level Support Equipment Maintenance Workload 24

C. MODEL SUBROUTINES AND VARIABLES 28

D. MODEL FACTORS 50

E. PAYGRADE MATRICES 60

F. RELIABILITY AND MAINTAINABILITY DATA BANK 76

G. PROGRAM LISTING 91

Appendix A

DETAILED MODEL OPERATION

Section III of Volume I presented a general overview of the features and steps of the NAVMAN computer model. This appendix discusses in detail the logic contained in the computer program. The program variable names are used in the following discussion and are defined in Appendix C of this volume. Most of the names are self-explanatory and should present little problem to the reader. As a further aid, the line numbers of the computer program are referenced; a numbered listing of the program is given in Appendix G. Finally, for clarity, the explanation is presented in two sections--organizational and intermediate maintenance.

ORGANIZATIONAL MAINTENANCE ROUTINE

Personnel requirements for organizational-level maintenance are determined on a work-center basis. Hourly workloads for the scheduled and unscheduled maintenance of the aircraft and its subsystems, for administrative duties, for facilities maintenance, and for other activities are totaled and divided by the appropriate personnel availability to arrive at a work center's personnel requirements. The value for the total number of personnel in a work center is compared to the values in a paygrade matrix to determine the number of men at each skill level or rank.

Administrative Work Centers

Certain work centers have no direct responsibility for the maintenance of the aircraft but rather provide the administrative functions such as supervision, material control, and data analysis. These work centers are either "position" or "directed" manned (i.e., a specific number of billets are required), or are manned on the basis of standards that statistically relate hours to nonreliability and maintainability factors such as flying hours, equipment inventories, or sorties. The administrative work centers, along with the standard equation or directed manning value used in the model, are shown in Table A.1. The equations and directed manning values are from the Squadron Manning Document (SQMD) model. The directed manning values in the SQMD model specify one person for the appropriate work centers (010, 030, 060, 100, 200, and 300; 040 has a requirement for 8 people) based on current Navy squadron sizes of from 4 to 14 aircraft. Because NAVMAN should have the capability of considering larger size squadrons, assumptions were necessary to extrapolate values beyond these historical squadron sizes. Discussions with SQMD analysts suggested the personnel values at various squadron sizes shown in Table A.1 for the directed manned work centers.

Administrative Support (AS) hours for work centers 020 and 050 are determined from statistical equations. Facilities Maintenance (FM) hours (determined as a percentage of AS hours) and Utilities Task (UT) hours (a fixed additive depending on the type of aircraft being considered) are then added. The AS hours for work center 140 are determined from the spread of the total AS hours for the Reliability and Maintainability (R&M) work centers--a process that is described below.

Work center 320, Troubleshooters, is manned only for fixed wing fighter (VF), attack (VA), or antisubmarine (VS) aircraft when at sea. For all other aircraft and for the shore calculations, the hours calculated by the standard equation are added to the workloads of various other work centers.

R&M Work Centers

Workloads and therefore personnel requirements for the R&M work centers are determined from the R&M and flying program input data supplied by the user. Since NAVMAN calculates requirements on a work-center basis, the preferred set of factor inputs are Corrective Maintenance (CM) and Preventive Maintenance (PM) factors for each work center. However, during the early stages of system acquisition to which the model is oriented, R&M requirements are specified as design goals and, commonly, at very aggregate levels. The design goals are often based on the performance of current aircraft systems of a similar type (i.e., a fighter aircraft or a fire control radar) taking into

Table A.1

ORGANIZATIONAL MAINTENANCE: DIRECTED AND STANDARD MANNED WORK CENTERS

Work Center	Standard
010 Maintenance Officer 030 Maintenance Administration 060 Data Analysis	UE / Men 0-17 / 1 18-23 / 2 24-29 / 3 30- / 4
040 Quality Assurance	UE / Men 0-17 / 8 18-23 / 10 24-29 / 12 30- / 14
020 Maintenance/Material Control Calculate AS hours; add any FM and UT hours; divide by availability to get personnel; add the number of directed men as a function of the number of shifts.	AS hours = 124.6715 + .3652 (FH) + FM hours + UT hours + directed men Shifts / Directed Men 1 / 2 2 / 3 3 / 4
050 Material Control Calculate AS hours; add any FM and UT hours; divide by availability to get personnel.	AS hours = 57.7481 + .3625 (FH)(RF) + FM hours + UT hours
140 Planned Maintenance	AS hours from percent spread + FM hours
100 Aircraft Division 200 Avionics/Armament Division 300 Line Division	1 man
320 Troubleshooters For all other sea and for all shore squadrons there are no people assigned to this work center. The hours calculated get added to the indicated work centers	Sea: Fixed Wing Fighter, Attack and Antisubmarine Squadrons 5 men Shore and Other Sea Hours = (S)(L)/(X) Hours go to WCs 110, 120 (2 times hours), 210, and 220

NOTES:

Variables

FH*: Flying hours per week
UE*: Number of aircraft per squadron
RF: Requisition factor per flying hour
L: Standby allowance per launch
X: Average number of sorties per launch
S*: Number of sorties per week

*User inputs; values for remaining variables are stored in the model.

consideration any expected R&M improvements due to advances in the state of the art and/or technological change. To provide the maximum user flexibility, NAVMAN accepts a wide range of possible R&M values. The R&M input options include:

- The type of maintenance workload. Data can be entered as PM, CM, or a combination of the two (termed Total Maintenance (TM) in the model). If the detail is available, separate values should be entered for both CM and PM. If the distinction cannot be made, TM data are entered. The model, using percentages based on current Navy aircraft, will break the TM hours into scheduled and unscheduled workloads.
- Work center or Work Unit Code (WUC) data. The user can input R&M data on a work center or a 2-digit WUC basis. Analyses showed that a clean crossover from WUCs to work centers does not exist. Many of the WUCs at the 2-digit level indicate a workload for multiple work centers. Because of this problem, the model accumulates all workload reported in terms of WUCs and spreads the total workload to work centers on the basis of percentages developed from current Navy aircraft.
- Aggregate or disaggregate data. If the user cannot define the data on a work-center or WUC basis, the model will spread an aggregate figure to the individual work centers. The user can enter a combination of disaggregate and aggregate data. For example, if values are known for certain shops because of the use of existing equipment, the user can enter these disaggregate data and then an aggregate figure for the remaining work centers. The model recognizes the disaggregate workloads and adjusts the percentage spreads to allocate the aggregate workload to the remaining work centers.
- The form of the input variables. The model will accept, for CM or TM data, maintenance manhours per flying hour, maintenance manhours per sortie, and mean time between failure/mean time to repair values, or any combination of the factors.

For PM, the model requires maintenance manhours per flying hour, per sortie, per flying day, and per week. If MTBF/MTTR data are used, more than one set of values for a work center can be entered. This would be appropriate for work centers with multiple equipment responsibilities.

- o Whether data do or do not include indirect factors. The direct maintenance workload must be augmented by indirect factors to account for Productive Delay (PD), Productivity Allowance (PA), and Make Ready/Put Away (MR/PA). It is assumed that all PM inputs do not include the indirect hours and therefore must be adjusted to include the indirect workload. CM data will include PA and MR/PA time if the data are taken from the 3M system. However, if contractor data or estimates are used, these indirect hours may not be included in the CM workload. The user can specify if the indirect hours are or are not included and the model will make the proper adjustments. TM data are assumed to include no indirect hours.

The model converts the input data to a work center's direct scheduled and unscheduled aircraft maintenance workload and then adds indirect maintenance hours, AS, FM, and UT hours to arrive at total workload.

Organizational Maintenance Methods and Equations

The steps to determine personnel requirements for organizational-level maintenance are outlined as follows:

1. Read organizational data and determine weekly flying program values.
2. Read any optional override values specified by the user.
3. Read R&M input data.
4. Compute raw PM and raw CM workload for each work center.
5. Add indirect factors to raw workloads to get total PM and CM workloads for each work center.

6. Calculate total AS workload and spread to the individual work centers.
7. Calculate FM workload for each work center.
8. Add any UT hours to sea workloads.
9. Calculate troubleshooter workload for shore squadrons and sea squadrons which are not VF, VA, or VS. Allocate this workload to the appropriate work centers.
10. Convert total hourly workloads for each work center to fractional personnel requirements by dividing by the appropriate availability.
11. Ensure that the minimum number of required personnel are assigned to the weapons work center (WC 230).
12. Convert fractional men to integer requirements using roundoff matrices. Set plus and minus hour bounds on the workloads.
13. Ensure the minimum of 2 plane captains per aircraft for the sea environment.
14. Set personnel and paygrade requirements for directed or standard manned work centers.
15. Set paygrade requirements for R&M work centers.
16. Determine total personnel and paygrade requirements for organizational-level maintenance.
17. Print output reports.
18. Perform any sensitivity analysis specified by the user.

The subsequent paragraphs will explain these steps in terms of the methods and equations used to make the necessary computations in the NAVMAN model.

1. <u>Read organizational data and determine weekly flying program values (lines 461 to 507)</u>

The necessary organizational inputs are listed in Section IV of Volume I. These values are used to determine the following variables in the model (the same equations, with changes in the appropriate variables, are used for sea and shore values):

TOTAL AIRCRAFT = (AIRCRAFT PER SQUADRON) (NUMBER OF SQUADRONS)

SORTIES (PER) WEEK = (SORTIE RATE) (FLYING DAYS WEEK) (AIRCRAFT PER SQUADRON)

FLYING HOURS (PER) WEEK = (SORTIES WEEK) (SORTIE LENGTH)

The type of aircraft affects certain variables in the model that are used in the calculation of various indirect hours. A flag, AIRCRAFT_INDX, is set based on the aircraft type specified by the user and is used to point to the appropriate members of various arrays stored in the model. Page 1 of the output reports is also printed at this time.

2. <u>Read any optional override values specified by the user (lines 508 to 703)</u>

There are a number of non-SQMD factors stored in the model that are used to calculate portions of a total workload or to spread aggregate workloads to work centers or to type of maintenance. These factors were developed by examining the experience of current Navy aircraft. The user has the option of overriding any of these stored values with factors he believes are more representative of the given situation. The values that can be overriden are listed below. The required input format is given in Section IV of Volume I.

Code	Variable[a]
1	CM_PRCT_OTHER or CM_PRCT_VFA
2	WORKCENTER_TM_SPREAD_OTHER or _VFA
3	WORKCENTER_PM_SPREAD_OTHER or _VFA
4	WORKCENTER_CM_SPREAD_OTHER or _VFA
5	I_LEVEL_SPREAD
6	SUPPORT_EQUIPMENT_HOURS_SEA and _SHORE
7	GSE_HOURS_PER_AC_SEA and _SHORE
8	MINIMUM_MEN

[a]All these variables, except code = 7, are arrays.

An additional optional input (code = 9) that can be used is for the variables OTHER_HOURS_SEA and OTHER_HOURS_SHORE. This input will allow the user to preset a prescribed number of hours (workload) into one or more of the work centers. This may be desired if information indicates a workload not covered by normal NAVMAN procedures.

3 & 4. Read R&M input data and calculate raw PM and CM workloads (lines 704 to 873)

As indicated previously, NAVMAN has the capability of accepting a wide range of R&M input data. The end objective is to determine the raw PM and the raw CM workload for each work center. The steps taken by the model to determine these values are a function of the input data supplied by the user.

The first step is to determine the index (member) of the workload arrays that must be loaded for a given R&M data card. For work-center data, this index corresponds to one of the 22 work centers in organizational maintenance (the specific one being supplied as an input field). For WUC data or aggregate data (which has a work center or WUC indicator of 999), the index is set equal to 23, or the last member of all the workload arrays.

Next, the model determines which workload array to load--either the PM, CM, or TM array (the specific one again being supplied as input). Once the specific array and member of that array are determined, the model calculates the workload based on the form of the input data--either maintenance manhours per flying hour (MMH/FH) or per sortie (MMH/S), or MTBF/MTTR. If CM workloads are calculated, the model further checks the input parameters to determine if indirect hours for MR/PA and PA must be added. The equation for adding these indirect hours is:

$$\text{RAW CM} = (\text{RAWER CM})\ (1 + \text{MR/PA} + \text{PA})$$

where RAWER CM is the workload calculated from the input values.

After all the R&M input cards are read, the last element (INDEX = 23) of each workload array is checked to determine if any aggregate or WUC workloads need to be spread to the individual work centers (by AD_SPREAD routine). Finally, any TM workloads are broken into PM and CM workloads (by CWTM_CALC routine) and loaded into the appropriate arrays.* At this point, each R&M work center should have a value in the appropriate member of the raw PM and CM arrays.

5. Add indirect factors to get total PM and CM hours (lines 874 to 893)

After the raw workloads are determined, indirect hours are added to arrive at total CM hours. The following equations are used for this computation:

$$\text{TOTAL CM} = (\text{RAW CM})\ (1 + \text{PD})$$

$$\text{TOTAL PM} = [\text{RAW PM} \times (1 + \text{MR/PA})][1 + (\text{PA} + \text{PD})]$$

6. Calculate total AS hours and spread the total to work centers (lines 894 to 918)

AS workload is determined as:

$$\text{TOTAL AS HOURS} = 306.9048 + .38519(\text{TOTAL RAW PM} + \text{RAW CM})$$

These hours are distributed to the individual work centers on the basis of factors that vary by type of aircraft.

7. Calculate FM workload (lines 919 to 955)

FM hours for each work center are computed as:

$$\text{FM (by work center)} = (\text{AS by work center})\ (\text{work center FM\%})$$

Any FM hours are loaded into OTHER_HOURS. The AS hours for work centers 020 and 050 are also calculated at this point in order to calculate FM hours.

---

*The CM portion of TM hours is assumed to contain no indirect factors and the equation for RAW CM is used.

8. Add any UT hours to sea environment (lines 956 to 974)

UT hours are applicable to sea duty only and vary by work center and aircraft type. UT hours are loaded into OTHER_HOURS_SEA.

9. Calculate troubleshooters workload (lines 975 to 1015)

The procedure used to calculate troubleshooters workload is given in Table A.1. For VF, VA, and VS squadrons at sea, 5 people are assigned to work center 320. For other sea squadrons and for shore squadrons, work center 320 is not manned but the workload due to troubleshooting action is added to the OTHER_HOURS for work center 110, 120 (two times the hours), 210, and 220.

10. Computation of billets (lines 1016 to 1028)

After total workload is calculated for each work center, personnel requirements (billets) are computed by dividing by the appropriate personnel availabilities. The standard work weeks used in this calculation are as follows:

Shore-based: 31.9 productive hours per week of a 40-hour week
VP-deployed: 51.0 productive hours per week of a 57-hour week
Carrier-based: 63.0 productive hours per week of a 70-hour week

SOURCE: OPNAVINST 5330.8.

The equation to compute personnel billets is:

$$\text{Work center billets} = \frac{\text{PM + CM + AS + OTHER}}{\text{productive hours per week}}$$

12. Convert to integer personnel requirements (lines 1044 to 1160)

The fractional personnel are converted to integer requirements using the appropriate roundoff table. Also, a PLUS_HOURS and a MINUS_HOURS value is determined that indicates the number of hours (workload) that can be added to or subtracted from the work-center workload before the billet requirement would change.

11 & 13. Set minimum levels (lines 1029 to 1043 and lines 1161 to 1167)

Because work center 230 requirements are driven by the need to have a certain number and size of load crews available during wartime operations, a minimum value is determined by multiplying the number of aircraft in a squadron by the minimum number of men per aircraft. This minimum requirement is compared to the requirement from the work-center workload and the maximum of the two numbers is used as the personnel requirement for work center 230.

Work center 310, Plane Captains, also has a minimum requirement of two personnel per aircraft when at sea in order to provide 24-hour coverage of the aircraft.

14. Determine requirements for standard manned work centers (lines 1175 to 1273)

Using the methods described in Table A.1, the requirements for non-R&M work centers are determined. In addition to the total work-center billets, the paygrade matrices are used to determine the appropriate number of personnel of the various skill levels.

15. Determine the quality of the billets for production work centers (lines 1274 to 1373)

Work-center personnel are distributed among paygrades (E-2 to E-9) using an authorization level/paygrade matrix developed from the BUPERS Occupational Classification System, derived paygrades as estimated through operational audit techniques, and the OSD "top-six" guidelines. Separate matrices are used for production work centers, line divisions, and unique work centers.

16 & 17. Determine total requirements and print output reports (lines 1374 to 1397)

The quality and quantity of billets are totaled and the output reports are printed. A description of the output reports is given in Section IV of Volume I.

18. Perform any sensitivity analysis (lines 1398 to 1473)

Any sensitivity analysis desired by the user is accomplished at this point. The model resets certain values according to the value of the sensitivity variable, zeroes out the appropriate requirements arrays, and then branches to the appropriate place in the model to recalculate requirements.

The sensitivity variables accepted as inputs are:

Code	Variable
1	AIRCRAFT_PER_SQUADRON
2	SORTIE_RATE_SEA and _SHORE
3	FLYING_DAYS_WEEK_SEA and _SHORE
4	R&M VALUES

For the first three variables, the user must input the new values to be considered. These values replace the original inputs and any succeeding analysis will use the latest values entered. That is, if the sortie rate is changed by a sensitivity input, any additional sensitivity runs will use the new value in the calculations.

Sensitivities on the R&M data allow the user to apply a factor (input value) to the PM data or to the CM data or to both (input value). Care must be taken that the sensitivity inputs accomplish the results desired by the user. As mentioned, sensitivity values replace stored values in the model. Furthermore, codes 1 and 4 change the raw PM and raw CM values for each work center, while codes 2 and 3 reread the original R&M data to calculate new values. If a sensitivity run on code 1 is made followed by a run for code 4, the latter sensitivity will adjust the workloads calculated by the former sensitivity and not the original workloads.

INTERMEDIATE MAINTENANCE ROUTINE

Each aircraft carrier and every Naval Air Station (NAS) has an Aircraft Intermediate Maintenance Department (AIMD) responsible for the intermediate-level maintenance of all aircraft on the carrier or at the NAS. This centralized facility is composed of a permanent cadre of ship or shore personnel, who perform administrative, supervision, overhead, and support equipment maintenance functions, and Temporary Assigned Duty (TAD) personnel assigned from the aircraft squadrons. These TAD personnel are identified in an aircraft's SQMD and provide the specific repair capabilities required by the AIMD. The personnel requirements for these separate components are estimated on the basis of standard Navy equations (for the permanent cadre) and the R&M of the aircraft (for the TAD personnel). Since the personnel requirements are independent of the flying-hour program, the number of aircraft is the primary variable used to estimate requirements. As such, NAVMAN executes the AIMD routine only for the initial, base case or for sensitivity analysis when the number of aircraft per squadron is changed.

The steps taken by NAVMAN to calculate each of these components of intermediate maintenance personnel are described below.

TAD Calculations (lines 1520 to 1563)

The detailed steps are:

1. Read the intermediate maintenance manhours per aircraft per week and the minimum number of avionics skills required.
2. From the input R&M value and the squadron size, calculate TOTAL_I_LEVEL_MANHOURS per squadron.
3. Using override values specified by the user or factors stored in the model, spread the total manhours to the five production divisions of the AIMD. The percentage spreads (I_LEVEL_SPREAD) are a function of the type of aircraft.
4. For each division, add the support (equipment) maintenance (SM) workload calculated from a factor per aircraft and the number of aircraft. The SM factors vary for each division and for the particular environment (sea or shore). The factors are stored in the model but may be overriden with user inputs.
5. Divide the sum of the aircraft and SM workloads by the appropriate availability to obtain a personnel figure.

6. Calculate AS hours for each division as a linear function of the personnel figure calculated in Step 5. The I_ LEVEL_ AS_COEFF varies for each division.
7. Add AS hours to the aircraft and SM hours to find the total workload for each of the five production divisions.
8. Divide by personnel availability and round to an integer number.
9. Compare the billets calculated for the Avionics Division to the minimum number of avionics skills required to ensure sufficient skill coverage.

The AIMD TAD requirements for a squadron and for the total fleet are printed after the paygrade matrix on output report 3. The individual division personnel requirements are listed after the organizational-level personnel requirements on output report 4.

Permanent Cadre Calculations (lines 1564 to 1700)

The permanent cadre requirements calculated by NAVMAN are based on standard equations contained in ACM-02,* the Navy model for estimating intermediate-level maintenance personnel requirements. The ACM-02 equations incorporated in NAVMAN use number of aircraft as the predicting variable. The remaining positions manned by ACM-02 are independent of any changes caused by the addition of aircraft to a carrier or a NAS. The work centers considered and the appropriate predicting methodology are listed in Table A.2.

NAVMAN calculates the additional permanent personnel required on a carrier and at a NAS because the new aircraft are being added. The steps used in this calculation are:

1. Read the total number of aircraft on a carrier and the number of squadrons of the new aircraft that will be stationed on a carrier.

Work Center Staffing Standards: Aircraft Maintenance--Perform Intermediate Aircraft Maintenance--ACM-02, NAVMMACLANT, January 13, 1978.

Table A.2

PERMANENT CADRE INTERMEDIATE MAINTENANCE PERSONNEL EQUATIONS

Production Control:

Shore: Manhours = $87.666 + .37487X + .0022157X^2$
Sea: Manhours = $4.05029X$

Material Control:

Manhours = $18.575 + .93871X - .0006217X^2$

Data Analysis:

X	Men
0-75	1
76-200	2
201-300	3
301-400	4

Precision Measurement Equipment Shop:

Maintenance Hours = $11.855 + .08987X + .0003166X^2$

$$\text{AS Hours} = \frac{\text{PME Maint. Hours}}{\text{Available Hours}} \times 4.72708$$

Manhours = AS Hours + Maintenance Hours

GSE Production Control:

Manhours = $10.224 + .2386X$

GSE Material Control:

Manhours = $4.86 + .2257X$

GSE Production Work Centers:

Shore: Maint. Hours = $1.02X$
Sea: Maint. Hours = $1.95X$

$$\text{AS Hours (shore)} = \frac{\text{GSE Maint. Hours Shore}}{\text{Available Hours}} \times 5.0861 + 46.25$$

$$\text{AS Hours (sea)} = \frac{\text{GSE Maint. Hours Sea}}{\text{Available Hours}} \times 5.0861 + 46.25$$

Manhours = AS Hours + Maintenance Hours

NOTES: X = number of aircraft; personnel = Manhours/Available Hours.

2. For each NAS, read the number of aircraft at the NAS before the aircraft are added and the number of squadrons of the new aircraft that will be stationed at the NAS.

3. Using the equations in Table A.2, calculate the number of personnel required on a carrier and at each NAS after the new aircraft are added.

4. Repeat Step 3 using the number of aircraft before the new aircraft are added as the predicting variable.

5. Calculate the difference between the personnel in Steps 3 and 4.

The permanent cadre changes due to the new aircraft are shown for a carrier and for each NAS on output report 3.

Appendix B

DEVELOPMENT OF MODEL FACTORS

The Navy's Squadron Manning Document (organizational level) and ACM-02 (intermediate level) maintenance personnel estimating models served as the foundation for the NAVMAN model. Both of these Navy personnel models were designed for use after an aircraft has been in the operational inventory for a period of time and therefore detailed maintenance workloads are typically available from the 3M maintenance data collection system. Also, general and specific factors contained in these models were developed by analysis of historical data. The NAVMAN model, however, has been designed for aircraft that are in the conceptual or development stages of acquisition. During this early time frame, information on operating characteristics and anticipated maintenance workloads is typically not known on a detailed basis.

The Navy factors that were general to all aircraft (e.g., Administrative Support, Facilities Maintenance percentage spreads) were incorporated into the NAVMAN model. However, factors that were specific to certain aircraft (e.g., minimum manning for work center 230, spreads of total I-level workload to specific work centers) had to be developed for the general aircraft categories of NAVMAN. Also, in order to provide various levels of maintenance input options to the user, it was necessary to generate general factors that could spread an aggregate workload to specific work centers or to specific types of maintenance work (preventive or corrective).

The non-Navy factors used in NAVMAN were developed from the maintenance information contained in ACM-02 and current aircrafts' Squadron Manning Documents (SQMDs). These factors are stored in the model but the user has the option, through model inputs, of overriding any of the factors if information is available that suggests that other factors may be more appropriate. The remainder of this appendix outlines the data and analysis used to generate these factors.

PERCENTAGE SPREAD OF TOTAL WORKLOAD TO WORK CENTERS

Many of the factors* contained in NAVMAN vary by work center. Personnel requirements for organizational maintenance are therefore calculated on a work-center basis. However, a user may not have available detailed maintenance workload data but rather only an aggregate maintenance figure such as total maintenance manhours per flying hour. It was therefore necessary to develop percentages that would spread a total maintenance workload to the individual work centers.

Table B.1 shows for a number of aircraft the percentage of the total maintenance workload applicable to specific work centers. These values are based on workloads from the appropriate SQMDs and are differentiated by Preventive Maintenance (PM), Corrective Maintenance (CM), and Total Maintenance (TM). Because sufficient data were not available for all the different types of aircraft considered in NAVMAN, composite values were formed for fighter and attack aircraft versus other types of aircraft. This appeared appropriate because fighter and attack aircraft have much higher workloads in the armament-related work centers (211 and 230), while the other aircraft have much higher avionics-related workloads. These composite percentages are the basis of the total workload to work-center values contained in NAVMAN.† These values are also used to spread workloads entered by Work Unit Codes to the appropriate work centers.

One further point should be mentioned concerning the percentage spreads. No data were available that showed a workload for work center 240, the Photo Shop. Therefore, if an aircraft requires a workload in work center 240, specific input data must be entered for that center.

PERCENTAGE SPREAD OF TOTAL MAINTENANCE TO PREVENTIVE/CORRECTIVE MAINTENANCE

Maintenance inputs to NAVMAN represent direct maintenance workloads. Indirect factors for Make Ready/Put Away (MR/PA), Production Delay (PD), and Productive Allowance (PA) must be added to arrive at

---

*All the factors used in NAVMAN are listed in Appendix D.

†WORKCENTER_TM_SPREAD_VFA, WORKCENTER_CM_SPREAD_VFA, etc. These spreads are shown in Table D.15 of Appendix D.

Table B.1

PERCENTAGE SPREAD OF TOTAL WORKLOAD TO WORK CENTERS

	Aircraft	110	120	121	130	131	210	211	220	230	240	310	Total
PM	S-3A	11.3	33.5	0	1.0	10.3	13.3	0	1.0	0	0	29.9	100.3
	EA-6B	15.5	19.8	0	0	11.5	20.0	0	12.5	2.1	0	27.2	98.6
	P-3C	20.4	24.6	0	1.0	1.8	6.0	0	5.0	1.0	0	40.4	100.2
	E-2B	17.8	40.4	0	0	9.9	3.8	0	4.6	0	0	23.2	99.7
	Subcomposite	13.8	29.6	0	0.5	8.4	10.8	0	5.8	0.8	0	30.2	99.9
	A-6E/KA-6D	14.8	22.0	0	0	4.9	6.8	1.0	5.2	2.3	0	43.0	100.0
	F-14A	2.8	30.8	0	0	9.2	0	3.5	2.7	22.1	0	28.0	99.1
	A-7	9.2	9.4	0	1.0	7.1	0	3.3	3.9	30.8	0	35.2	99.9
	F-4	11.3	31.4	0	0	4.8	6.9	4.1	5.6	9.5	0	25.9	99.5
	Subcomposite[a]	9.5	23.4	0	0.3	6.5	3.4	3.0	4.4	16.2	0	33.0	99.7
	Total average	11.6	26.5	0	0.4	7.4	7.1	1.5	5.1	8.5	0	31.6	99.7
CM	S-3A	8.1	10.8	6.5	4.3	5.8	17.9	0	20.5	7.8	0	18.3	100.0
	EA-6B	9.2	11.0	10.5	2.0	3.2	34.4	0	9.4	1.9	0	18.3	99.9
	P-3C	15.5	16.6	7.8	2.7	3.4	24.4	0	15.8	6.6	0	7.1	99.9
	E-2B	12.3	18.6	8.4	1.0	8.0	31.8	0	9.0	0	0	10.9	100.0
	Subcomposite	11.3	14.3	8.3	2.5	5.1	27.1	0	13.7	4.1	0	13.7	100.1
	A-6E/KA-6D	13.6	9.7	6.9	2.2	3.8	9.2	9.2	11.7	10.4	0	23.3	100.0
	F-14A	10.3	21.2	2.0	0	10.5	6.7	6.2	19.1	9.3	0	14.2	99.5
	A-7	5.4	21.6	3.7	1.0	3.3	10.6	10.8	8.9	12.0	0	16.5	93.8[b]
	F-4	7.9	15.2	12.2	1.5	3.4	4.5	8.9	9.4	11.5	0	25.6	100.1
	Subcomposite[a]	9.3	16.9	6.2	1.2	5.3	7.8	8.8	12.3	10.8	0	19.9	98.5
	Total average	10.3	15.6	7.3	1.8	5.2	17.4	4.4	13.0	7.4	0	16.8	99.2

Table B.1--*continued*

	Aircraft	110	120	121	130	131	210	211	220	230	240	310	Total
TM	S-3A	8.3	12.8	5.9	4.0	6.2	17.5	0	18.7	7.1	0	19.3	99.8
	EA-6B	8.6	12.6	8.7	1.7	4.6	31.9	0	9.9	2.0	0	19.9	99.9
	P-3C	16.6	18.5	6.0	2.3	3.0	20.3	0	13.4	5.3	0	14.6	100.0
	E-2B	13.3	22.6	6.8	0.8	8.4	26.7	0	8.2	0	0	13.1	99.9
	Subcomposite	11.7	16.6	6.9	2.2	5.6	24.1	0	12.6	3.6	0	16.7	100.0
	A-6E/KA-6D	14.0	14.4	4.3	1.4	4.2	8.3	6.1	9.2	7.4	0	30.8	100.1
	F-14A	8.4	23.6	1.5	0.3	10.2	5.2	5.5	15.0	12.5	0	17.7	99.9
	A-7	7.2	17.4	2.3	0.8	5.0	6.8	8.1	7.2	20.3	0	25.0	100.1
	F-4	8.9	19.8	8.7	1.2	3.8	5.2	7.5	8.3	10.9	0	25.7	100.0
	Subcomposite[a]	9.6	18.8	4.2	0.9	5.8	6.4	6.8	9.9	12.8	0	24.8	100.0
	Total average	10.7	17.7	5.5	1.6	5.7	15.2	3.4	11.2	8.2	0	20.8	100.0

SOURCE: Appropriate Squadron Manning Documents.

[a]F-14, F-4, A-7, A-6/KA-6.

[b]+6.6% in WC 320.

TM workloads. The application of these factors vary for PM and CM. Therefore, if TM data are used as inputs, factors are required to break the total direct hours into preventive and corrective hours.

The percentage of the total work-center workload that is preventive and corrective maintenance is shown for a number of aircraft in Table B.2. These values are again based on data from the aircrafts' SQMDs. Because of the lack of multiple data points for various types of aircraft, composite figures were formed for fighter and attack aircraft versus other types of aircraft. These composite values are given in the last two columns of Table B.2 and are stored in NAVMAN as arrays titled CM_PRCT_VFA and CM_PRCT_OTHER.*

MINIMUM MANNING VALUES FOR WORK CENTER 230

Personnel requirements in work center 230 for certain types of aircraft are dictated by the need to have an adequate number of load crews available for wartime missions. The number of load crews required is a function of the anticipated activity rate and the number of sorties in a launch, that is, the number of aircraft to be loaded in a given block of time. The size of a load crew is a function of the configuration of the aircraft and the type of munitions. SQMD analysts have determined for the appropriate aircraft the minimum number of personnel required per squadron. These values are shown in Table B.3.

NAVMAN estimates work center 230 personnel requirements based on the input workload and compares the result with a minimum personnel figure to determine actual requirements. The minimum personnel value† varies by aircraft type and is based on the values shown in the last column of Table B.3.

TOTAL WORKLOAD TO PRODUCTION DIVISIONS: INTERMEDIATE LEVEL

One set of factors in the ACM-02 model are Z tables that spread total maintenance workload to individual production work centers. Each

---

*NAVMAN stores only the CM percentages. If the value is nonzero, the PM percentage is one minus the CM value.

†The minimum values for work center 230 are shown in Table D.10 of Appendix D.

Table B.2

PERCENTAGE SPREAD OF TOTAL MAINTENANCE HOURS TO PREVENTIVE AND CORRECTIVE MAINTENANCE

WC	S-3A		A-6E		EA-6B		P-3C		E-2B		F-14A		A-7E		F-4J		Avg All Acft		A/F[a]		Other[b]	
	CM	PM	CM	PM	CM	PM	CM	PM	CM	PM	CM	PM	CM	PM	CM	PM	CM	PM	CM	PM	CM	PM
110	.88	.12	.60	.40	.89	.11	.72	.28	.75	.25	.92	.08	.47	.53	.64	.36	.73	.27	.66	.34	.81	.19
120	.77	.23	.42	.58	.73	.27	.70	.30	.67	.33	.67	.33	.78	.22	.55	.45	.66	.34	.61	.39	.72	.28
121	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
130	.98	.02	1	0	.95	.05	.92	.08	.94	.06	.80	.20	.64	.36	.91	.09	.89	.11	.84	.16	.95	.05
131	.85	.15	.56	.44	.57	.43	.87	.13	.78	.22	.77	.23	.41	.59	.64	.36	.68	.32	.60	.40	.77	.23
210	.93	.07	.69	.31	.89	.11	.93	.07	.97	.03	.98	.02	.97	.03	.62	.38	.87	.13	.82	.18	.93	.07
211	--	--	.94	.06	--	--	--	--	--	--	.84	.16	.83	.17	.84	.16	.86	.14	.86	.14	--	--
220	1	0	.78	.22	.78	.22	.91	.09	.90	.10	.95	.05	.77	.23	.81	.19	.86	.14	.83	.17	.90	.10
230	1	0	.88	.12	.81	.19	.96	.04	--	--	.55	.45	.37	.63	.75	.25	.76	.24	.64	.36	.92	.08
310	.86	.14	.47	.53	.76	.24	.38	.62	.67	.33	.60	.40	.41	.59	.71	.29	.61	.39	.55	.45	.67	.33

SOURCE: Appropriate Navy Aircraft Squadron Manning Documents.

[a]Attack/fighter aircraft.

[b]Other types of aircraft.

Table B.3

SQMD WORK CENTER 230 MINIMUM PERSONNEL REQUIREMENTS

Aircraft	Squadron Size	Minimum Men	Men per AC
F-4J, F-4N	12	19	1.583
F-14A	12	19	1.583
A-7	12	26	2.167
A-6	12	27	2.250
A-6	9	23	2.556
SH-3D	8	4	.500
SH-3H	8	6	.750
P-3	9	5	.556
S-3A	10	10	1.000
EA-6B	4	4	1.000

Naval Air Station (NAS) has a unique Z table that lists the appropriate factors for each type of aircraft located on the base. These percentage spreads were developed by analysis of 3M maintenance data. Since NAVMAN is designed for conceptual aircraft and for all NASs, general factors had to be developed that would allocate the total intermediate maintenance workload to the appropriate shops.

Examination of ACM-02 Z tables for a number of different Aircraft Intermediate Maintenance Departments (AIMDs) suggested that although the work-center percentages for a given type, model, and series of aircraft varied from AIMD to AIMD, the percentage at the department level was a constant. Symbolically, if Zij is the percentage of the total workload (for a given aircraft type) that is attributed to work center i for AIMD j, then

$$Z_{i1} \neq Z_{i2} \neq \ldots \neq Z_{in}$$

However,

$$\Sigma_i Z_{i1} = \Sigma_i Z_{i2} = \ldots = \Sigma_i Z_{in}$$

where the summation is over all work centers in a given department. Therefore, if it were appropriate, general AIMD factors should be developed at the department, rather than the work-center, level.

In developing gross factors, there is the danger of neglecting to account for any specific skill (Navy Enlistment Classification) requirements of the work centers within a division. For example, a 10 hour a week workload for a division would suggest one billet; however, if that 10 hours involved 5 hours apiece for two different skills, the personnel requirements should be two. In reality, this is only a problem for the avionics division where 10 or more specific NECs may be required. For that reason, for the avionics division NAVMAN uses a minimum manning value of the number of specific skills required (user input). Division-level factors are used to spread the total workload to divisions, and personnel requirements are calculated by dividing by the availability. In the avionics division, the resulting figure is compared to minimum skill requirements, and the greater of the two numbers is chosen as the NAVMAN personnel requirement.

The I-level percentage spreads were developed for specific aircraft types from the Z tables of ACM-02. Table B.4 shows the ACM-02 spreads for a number of different aircraft. Where factors were available for more than one aircraft of a given type, the percentages were averaged and then rounded. The actual percentages used in NAVMAN are listed in Table D.11 of Appendix D.

INTERMEDIATE-LEVEL SUPPORT EQUIPMENT MAINTENANCE WORKLOAD

A portion of the total workload for the intermediate-level work centers is due to the preventive and corrective maintenance of aviation support equipment. This equipment includes stands and benches, avionics test equipment, and all yellow flight deck equipment (Ground Support Equipment). ACM-02 states that generally this equipment cannot be directly associated with a specific type and model of aircraft and, therefore, treats support equipment maintenance as an additive to each work center. These additive values vary from AIMD to AIMD.

Although support equipment workload cannot always be associated with a given aircraft, it is reasonable to assume that additional aircraft would require additional support equipment workload. Since the

Table B.4

SPREAD OF TOTAL INTERMEDIATE WORKLOAD TO PRODUCTION DIVISIONS

Aircraft	Power Plants	Airframes	Avionics	Armament	Aviator's Eq.
F-4J	.2436	.1596	.5291	.0402	.0275
F-14A	.2781	.1307	.4996	.0448	.0468
A-7E	.3006	.1363	.5088	.0418	.0125
A-6E	.1835	.1095	.6599	.0186	.0285
S-3A	.2434	.1125	.5749	.0105	.0587
EA-6B	.1469	.1101	.6859	.0046	.0525
E-2B	.1899	.0881	.6567	0	.0653
KA-6D	.2328	.1504	.5752	.0200	.0216
SH-3H	.2304	.1798	.5810	.0026	.0062
SH-3D	.3448	.1401	.5090	.0047	.0014
SH-3G	.3301	.1552	.5103	.0004	.0040
RA-5C	.1822	.0861	.7240	0	.0077
RF-4B	.1196	.3184	.5423	0	.0183
P-3C	.2875	.1626	.5202	.0128	.0219

SOURCE: *Work Center Staffing Standards: Aircraft Maintenance--Perform Intermediate Aircraft Maintenance--ACM-02*, NAVMMACLANT, January 13, 1978.

intent of NAVMAN is to estimate all the maintenance personnel requirements due to an aircraft, an estimate of support equipment maintenance is necessary. To determine appropriate additives, the various support equipment maintenance workloads contained in ACM-02 were examined. These values are shown for a number of different AIMDs in Table B.5. Because NAVMAN estimates intermediate maintenance requirements on a division basis, the workloads are shown for the production divisions.

The values in Table B.5 were subjected to statistical regression to determine if a relationship existed between maintenance hours and number of aircraft. Although a positive correlation did exist, no statistically significant relationship could be developed. As a rough estimate of support equipment maintenance workload, the weekly hours per aircraft were calculated for each AIMD; these values are shown in Table B.6. Since the carrier values were believed to be substantially different from the shore values, two sets of factors were formulated--an average for shore AIMDs and a value for carrier AIMDs.* These values are shown in Table B.6 and stored in NAVMAN. The support

*The higher carrier values reflect the higher aircraft activity rates while deployed at sea.

Table B.5

INTERMEDIATE-LEVEL SUPPORT EQUIPMENT MAINTENANCE WORKLOAD

(Hours per Week)

AIMD	Number of Aircraft	Power Plants Division	Airframes Division	Avionics Division	Armament Division	Aviator Equip. Division	GSE
Alemada	85	8.85	24.95	80.07	2.25	41.87	137.58
Cecil Field	289	28.66	71.68	246.11	30.06	56.27	304.10
Jacksonville	191	25.22	34.40	203.62	19.20	68.05	219.03
Lemoore	247	35.93	60.37	149.25	42.71	49.62	159.49
Miramar	360	83.66	41.69	422.43	13.47	47.40	281.21
Norfolk	190	24.18	15.46	151.26	-	45.54	126.48
North Island	215	36.90	31.37	308.49	22.32	120.99	291.96
Oceana	283	51.11	46.97	224.77	33.72	45.42	242.88
CV (Carrier)	90	46.56	37.51	173.37	15.57	34.86	175.59

SOURCE: *Work Center Staffing Standards: Aircraft Maintenance--Perform Intermediate Aircraft Maintenance--ACM-02*, NAVMMACLANT, January 13, 1978.

equipment workload for each I-level division is therefore calculated as the product of the number of (the new) aircraft and the appropriate hours per aircraft.

Table B.6

INTERMEDIATE-LEVEL SUPPORT EQUIPMENT MAINTENANCE WORKLOAD

(Hours per Aircraft per Week)

AIMD	Power Plants	Airframes	Avionics	Armament	Aviators' Equip.	GSE
Alemada	.10	.29	.94	.03	.49	1.62
Cecil Field	.10	.25	.85	.10	.19	1.05
Jacksonville	.13	.18	1.07	.10	.36	1.15
Lemoore	.15	.24	.60	.17	.20	.65
Miramar	.23	.12	1.17	.04	.13	.78
Norfolk	.13	.08	.80	–	.24	.67
North Island	.17	.15	1.43	.10	.56	1.36
Oceana	.18	.17	.79	.12	.16	.86
Total	1.19	1.48	7.65	.66	2.33	8.14
Average	.15	.19	.95	.09	.29	1.02
CV (Carrier)	.52	.42	1.93	.17	.39	1.95

Appendix C

MODEL SUBROUTINES AND VARIABLES

This appendix contains an alphabetic listing of the subroutines and variables used in NAVMAN. Whether a variable is a stored value or a user input is indicated in the variable definition.

SUBROUTINE NAME	FUNCTION
AD_SPREAD	Distributes an aggregate workload to the individual work centers.
AIMD_CALCULATIONS	Determines temporary and permanent Intermediate maintenance billets for carriers and Naval Air Stations.
AIMD_FIXED	Determines a portion of the permanent Intermediate maintenance billets for carriers and Naval Air Stations. Called by AIMD_CALCULATIONS.
CWTM_CALC	Distributes a total maintenance workload to preventive and corrective maintenance for a given work center.
INPUT_ERROR_EXIT	Indicates an illegal R&M workload type for AA_TYPE.
INPUT_ERROR_EXIT2	Indicates an illegal work center code for XXX_CODE.
INPUT_ERROR_EXIT3	Indicates an illegal aircraft code for AIRCRAFT_TYPE.
INTEGER	Converts fractional men to integer requirements for I level maintenance.
PAGEONE_REPORT	Prints page one of output report.
PAGETWO_REPORT	Prints headings for page two of output report.

PAGETWO_DETAIL_REPORT	Prints input R&M data on page two of output report.
PAGETWO_SPREAD_REPORT	Prints values that spread an aggregate workload to work centers on page two of output report.
PAGETHREE_REPORT	Prints page three of output report.
PAGEFOUR_REPORT	Prints page four of output report.
PAGEFIVE_REPORT	Prints page five of output report.
RESET	Recalculates model values based on sensitivity inputs.

VARIABLE NAME	DEFINITION
AA_TYPE	Input value that identifies the type of maintenance workload: PM = Preventive Maintenance; CM = Corrective Maintenance; TM = Total Maintenance.
ADMIN_SUPPORT_SPREADS (22,10)	Stored array that gives the percentage of the total administrative support workload that goes to work center I (22) for aircraft type J (10); see Appendix D.
AFTER_SEA_X	The total number of aircraft onboard an aircraft carrier; = NUMBER_AC_ON_SEA.
AFTER_SHORE_X (5)	Array containing the total number of all types of aircraft based at Naval Air Station IA (5) after the new aircraft is added; = SHORE_AC_BEFORE + SHORE_SQ_ADDED * AIRCRAFT_PER_SQUADRON.
AIMD_CADRE_ADDED_SEA	The number of permanent I level personnel required on a carrier because the new aircraft is added; = SEA_MEN_XA - SEA_MEN_XB.
AIMD_CADRE_ADDED_SHORE (5)	The number of permanent I level personnel required at Naval Air Station IA (5) because the new aircraft are added; = TOTAL_SHORE_XA - TOTAL_SHORE_XB.
AIMD_FLAG	Counter set equal to 1 when AIMD_CALCULATIONS is called; used as a print option for AIMD data.

AIMD_MEN (7)	Array in the AIMD_FIXED subroutine that holds billet requirements for the permanent portion of the Intermediate maintenance facility.
AIMD_TOTAL_CADRE_ADDED	Output variable indicating total number of permanent I level billets required.
AIRCRAFT	Variable in the AIMD_FIXED subroutine that indicates the number of aircraft used to calculate AIMD_MEN.
AIRCRAFT_CODES (10,2)	Stored array containing the aircraft identifiers accepted by the model; see Appendix D for appropriate codes.
AIRCRAFT_INDX	Index based on the TYPE_OF_AIRCRAFT input to the model; relates to a member of AIRCRAFT_CODES.
AIRCRAFT_PER_SQUADRON	Input value indicating the number of aircraft in each squadron.
AS2_COEFF1	Stored coefficient for the fixed portion of the AS hour equation for work center 020; declared in the model as 124.6715.
AS2_COEFF2	Stored coefficient for the variable portion of the AS hour equation for work center 020; declared in the model as .3652.
AS5_COEFF1	Stored coefficient for the fixed portion of the AS hour equation for work center 050; declared in the model as 57.7481.

AS5_COEFF2	Stored coefficient for the variable portion of the AS hour equation for work center 050; declared in the model as .3625.
AS_COEFF1	Stored coefficient for the fixed portion of the AS hour equation for the production work centers; declared in the model as 306.9048.
AS_COEFF2	Stored coefficient for the variable portion of the AS hour equation for the production work centers; declared in the model as .38519.
AS_HOURS_SEA (23) AS_HOURS_SHORE (23)	The weekly Administrative Support hours per squadron for the individual work centers.
AVAIL	Value used in AIMD_FIXED to indicate weekly personnel availability.
AVAILABILITY_SEA	Stored value indicating the Organizational personnel weekly availability when at sea; declared in the model as 63.0.
AVAILABILITY_SHORE	Stored value indicating the Organizational personnel weekly availability when on shore; declared in the model as 31.9.
AVAILABILITY_VP	Stored value indicating the Organizational personnel weekly availability for a VP squadron; declared in the model as 51.0.

BEFORE_SEA_X	The number of aircraft on a carrier without the new aircraft; = NUMBER_AC_ON_SEA - (NUMBER_SQ_ON_SEA * AIRCRAFT_PER_SQUADRON).
BEFORE_SHORE_X (5)	The number of aircraft based at Naval Air Station IA (5) before the new aircraft are added; = SHORE_AS_BEFORE(IA).
CM_PERCENT	The percentage of a work center's total maintenance workload that is corrective maintenance.
CM_PRCT_OTHER (23)	Stored array containing work center CM_PERCENT for aircraft with codes other than fighter or attack; see Appendix D.
CM_PRCT_VFA (23)	Stored array containing work center CM_PERCENT for fighter and attack aircraft; see Appendix D.
DEFAULT_CODE	Input code indicating the variable for which the user wishes to override the stored value; see Volume I, Table 9 for appropriate codes.
FACILITIES_MAINTENANCE_ FACTORS (23)	Stored array used to calculate a work center's Facility Maintenance hours; see Appendix D.
FACTOR1	Value used by RESET routine to set new raw workloads.
FACTOR2	Value used by RESET routine to set new sorties per week.
FACTOR3	Value used by RESET routine to set new flying hours per week.

FACTOR_OTHER (23)	Array used in AD_SPREAD routine that spreads an aggregate workload to the individual work centers for aircraft with codes other than fighter or attack.
FACTOR_VFA (23)	Array used in AD_SPREAD routine that spreads an aggregate workload to the individual work centers for fighter or attack aircraft.
FLYING_DAYS_WEEK_SEA FLYING_DAYS_WEEK_SHORE	Input values indicating the number of flying days per week.
FLYING_HOURS_AWEEK_SEA FLYING_HOURS_AWEEK_SHORE	The weekly flying hours per aircraft; = FLYING_HOURS_WEEK / AIRCRAFT_PER_ SQUADRON.
FLYING_HOURS_WEEK_SEA FLYING_HOURS_WEEK_SHORE	The weekly flying hours per squadron; = SORTIES_WEEK * SORTIE_LENGTH.
GRADE_LEVEL_SEA (23,10) GRADE_LEVEL_SHORE (23,10)	Array containing required billets for work center I (23) and paygrade J (10); GRADE_LEVEL (I,1) is not used and GRADE_LEVEL (I,10) has total billet requirements for work center I.
GSE_HOURS_PER_AC_SEA	Stored value indicating the weekly hours per aircraft required for I level maintenance of ground support equipment at sea; declared in the model as 1.95.
GSE_HOURS_PER_AC_SHORE	Stored value indicating the weekly hours per aircraft required for I level maintenance of ground support equipment on shore; declared in the model as 1.02.

HOURS_SEA HOURS_SHORE	The weekly workload for work center 320 that is counted in work centers 110, 120, 210, and 220.
I_LEVEL_AS_COEFF (5)	Stored coefficient used to calculate weekly Administrative Support hours for the five production divisions of an AIMD; see Appendix D.
I_LEVEL_AVAILABILITY_SEA	Stored value for the weekly availability for I level personnel when at sea; declared in the model as 60.0.
I_LEVEL_AVAILABILITY_SHORE	Stored value for weekly availability for I level personnel when on shore; declared in the model as 31.9.
I_LEVEL_MANHOURS_WEEK	Input value indicating the weekly I level maintenance manhours per aircraft.
I_LEVEL_MANPOWER_SEA (5) I_LEVEL_MANPOWER_SHORE (5)	The temporary (TAD) I level personnel requirements for AIMD production division I (5).
I_LEVEL_ROUNDOFF (7)	Stored array used to convert fractional people to integer requirements for Intermediate Maintenance; see Appendix D.
I_LEVEL_SPREAD (5,10)	Stored array that distributes total weekly I level maintenance hours to division I (5) for aircraft type J (10); see Appendix D.
I_TYPE	Input value that indicates if corrective maintenance workload has indirect factors included (= 0) or not (= 1).

II	Pointer used in INTEGERIZE routine to index the roundoff table.
IJ	Variable used in INTEGERIZE routine to insure M_SEA and M_SHORE values are non-zero.
INDX	Pointer corresponding to a particular work center in an array.
ISNR_SEA ISNR_SHORE	Variable used to determine paygrade requirements for personnel in work centers 100, 200, and 300.
J_TYPE	Input value that indicates the form of the R&M CM and TM input data; 1 = MMH/FH, 2 = MMH/S, 3 = MTBF/MTTR.
K_TYPE	Input value that indicates if R&M data are by Work Unit Code (= 1) or Work Center (= 0).
LINE_DIVISION_MATRIX (9,120)	Stored array that indicates the paygrade breakout (E-1 through E-9) for J people (120) in work center 310; see Appendix E.
LOAD_FACTOR_OTHER	Variable used in AD_SPREAD routine to distribute an aggregate workload to individual work centers for aircraft types other than fighter and attack.
LOAD_FACTOR_VFA	Variable used in AD_SPREAD routine to distribute an aggregate workload to individual work centers for fighter and attack aircraft.

M_SEA (23) M_SHORE (23)	Arrays that hold the maintenance manpower requirements for work center I (23).
MAINTENNACE_TOTAL_SEA MAINTENANCE_TOTAL_SHORE	Variables used in PAGEFOUR_REPORT to accumulate total organizational maintenance manhours.
MAINTENANCE_TOTAL_MSEA MAINTENANCE_TOTAL_MSHORE	Variables used in PAGEFOUR_REPORT to accumulate total organizational manpower requirements.
MAKE_READY_PUTAWAY_FACTOR	Stored factor used to convert raw maintenance hours to total maintenance hours; declared in the model as .30.
MEN	Variable used in INTEGER that holds fractional manpower requirements.
MINIMUM_MEN (10)	Stored array that indicates the minimum men per aircraft for work center 230 by aircraft type J (10); see Appendix D.
MINUS_HOURS_SEA (23) MINUS_HOURS_SHORE (23)	Array that indicates for work center I (23) the number of hours that could be subtracted from the work center workload before the manpower requirement would be reduced.
NUMBER_AC_ON_SEA	Input value indicating the total number of aircraft onboard a carrier; used in AIMD_CALCULATIONS to determine changes in permanent portion of the AIMD.
NUMBER_DEFAULT_INPUTS	Input value indicating the number of override cards that follow.

NUMBER_OF_AVIONICS_SKILLS_REQ	Input value indicating the number of different skills (NECs) required in the Avionics Division of Intermediate maintenance.
NUMBER_OF_NAS_DEPLOYED	Input value indicating the number of Naval Air Stations the aircraft will be assigned to.
NUMBER_OF_SHIFTS	Input value indicating the number of work shifts in organizational maintenace.
NUMBER_OF_SQUADRONS	Input value for the total number of squadrons in the fleet.
NUMBER_SQ_ON_SEA	Input value for the total number of squadrons of the new aircraft that will be placed on a carrier.
OTHER_HOURS_SEA (23) OTHER_HOURS_SHORE (23)	Arrays that hold non-maintenance and non-AS hours for work center I (23).
PAYGRADE_MATRIX020 (9,20)	Stored array that indicates the paygrade breakout for J people (20) in work center 020; see Appendix E.
PAYGRADE_MATRIX050 (9,20)	Stored array that indicates the paygrade distribution for J people (20) in work center 050; see Appendix E.
PAYGRADE_MATRIX230 (9,40)	Stored array that indicates the paygrade distribution for J people (40) in work center 230; see Appendix E.

PLUS_HOURS_SEA (23) PLUS_HOURS_SHORE (23)	Array that indicates for work center I (23) the number of hours that could be added to the work center workload before the manpower requirements would be increased.
PM_PERCENT	Variable used in CWTM_CALC routine that indicates the percent of a total maintenance workload that is preventive maintenance.
PROD_DELAY_FACTOR_SEA (23) PROD_DELAY_FACTOR_SHORE (23)	Stored arrays indicating the indirect factor for production delay for work center I (23); see Appendix D.
PRODUCTION_MATRIX (9,80)	Stored array that indicates the paygrade distribution for J people (80) in the production work centers; see Appendix E.
PRODUCTIVITY_ALLOWANCE_ FACTOR	Stored value for the indirect hours due to productive allowance; declared in the model as .20.
RAW_CM_WORKLOAD_SEA (23) RAW_CM_WORKLOAD_SHORE (23)	Arrays containing the corrective maintenance workloads for work center I (23) before the indirect hours for Production Delay are added.
RAW_PM_WORKLOAD_SEA (23) RAW_PM_WORKLOAD_SHORE (23)	Arrays containing the preventive maintenance workloads for work center I (23) before the indirect hours for make ready/put away, productive allowance, and production delay are added.

RAW_TM_WORKLOAD_SEA (23) RAW_TM_WORKLOAD_SHORE (23)	Arrays used for TM inputs that contain the maintenance workloads for work center I (23) before any indirect factors are added; theoretically = RAW_PM_WORKLOAD + RAWER_CM_WORKLOAD.
RAWER_CM_WORKLOAD_SEA (23) RAWER_CM_WORKLOAD_SHORE (23)	Arrays used to store the corrective maintenance workloads for work center I (23); if I_TYPE = 0, then RAW_CM = RAWER_CM; if I_TYPE = 1, then RAW_CM = RAWER_CM * (1 + MAKE_READY_PUTAWAY_FACTOR + PRODUCTIVE_ALLOWANCE_FACTOR).
REQUISITION_FACTOR (10)	Stored array containing the number of requisitions per flying hour for aircraft type I (10); used in work center 050 calculations; see Appendix D.
RMEN	Variable used in INTEGER routine to hold integer number of people.
ROUNDOFF_TABLE_SEA (10) ROUNDOFF_TABLE_SHORE (0)	Stored arrays indicating the cutoff values when determining integer number of people; see Appendix D.
SEA_HOURS_XA	Value used in AIMD_CALCULATIONS routine for hours at sea for fixed portions of Intermediate Maintenance after the new aircraft are added to the carrier.
SEA_HOURS_XB	Value used in AIMD_CALCULATIONS routine for hours at sea for fixed portion of Intermediate Maintenance before the new aircraft are added to the carrier.

SEA_MEN_XA (7)	Array used in AIMD_CALCULATIONS for fixed AIMD integer personnel at sea after the new aircraft are added to the carrier; = SEA_HOURS_XA / I_LEVEL_AVAILABILITY_SEA (rounded off by INTEGER routine).
SEA_MEN_XB (7)	Array used in AIMD_CALCULATIONS for fixed AIMD integer personnel at sea before the new aircraft are added to the carrier; = SEA_HOURS_XB / I_LEVEL_AVAILABILITY_SEA (rounded off by INTEGER routine).
SENSITIVITY CODE	Input value indicating the sensitivity variable.
SENSITIVITY FLAG	Counter set when sensitivity values are read.
SENSITIVITY_VALUE1 SENSITIVITY_VALUE2	Input values for the sensitivity variable.
SHORE_AC_BEFORE (5)	Input values for the number of all types of aircraft at Naval Air Station I (5) before the new aircraft are added.
SHORE_HOURS_XA (7)	Array used in AIMD_CALCULATIONS routine for hours for fixed portion of Intermediate Maintenance on shore after the new aircraft are added to a Naval Air Station.
SHORE_HOURS_XB (7)	Array used in AIMD_CALCULATIONS routine for hours for fixed portion of Intermediate Maintenance on shore before the new aircraft are added to a Naval Air Station.

SHORE_MEN_XA (7)	Array used in AIMD_CALCULATIONS routine for fixed Intermediate Maintenance personnel on shore after the new aircraft are added to a Naval Air Station; = SHORE_HOURS_XA / I_LEVEL_AVAILABILITY_SHORE; (rounded off by INTEGER routine).
SHORE_MEN_XB (7)	Array used in AIMD_CALCULATIONS routine for fixed Intermediate Maintenance personnel on shore before the new aircraft are added to a Naval Air Station; = SHORE_HOURS_XB / I_LEVEL_AVAILABILITY_SHORE; (rounded off by INTEGER routine).
SHORE_SQ_ADDED (5)	Input array indicating the number of squadrons of the new aircraft added to Naval Air Station I (5).
SORTIE_LENGTH_SEA SORTIE_LENGTH_SHORE	Input values for the average sortie length in hours.
SORTIE_RATE_SEA SORTIE_RATE_SHORE	Input values for the average number of sorties per aircraft per flying day.
SORTIES_WEEK_SEA SORTIES_WEEK_SHORE	The number of sorties per squadron per week; = SORTIE_RATE * FLYING_DAYS * AIRCRAFT_PER_SQUADRON.
STORE_CM_MMH_FH (21) STORE_CM_MMH_S (21) STORE_CM_MTBF (21) STORE_CM_MTTR (21) STORE_PM_MMH_DAY (21) STORE_PM_MMH_FH (21) STORE_PM_MMH_S (21) STORE_PM_MMH_WEEK (21)	Arrays used to store the input data for Preventive and Corrective Maintenance for work center I (21); used to print page two of the output report.

STORE_TITLE	Input string used as a title on each page of output report.
SUBTOTAL_FOUR_HSEA SUBTOTAL_FOUR_NSHORE	Used in PAGEFOUR_REPORT routine to hold the total hours for work centers 010, 020, 030, 040, 050, and 060.
SUBTOTAL_FOUR_MSEA SUBTOTAL_FOUR_MSHORE	Used in PAGEFOUR_REPORT routine to hold the total manpower requirement for work centers 010, 020, 030, 040, 050, and 060.
SUBTOTAL_ONE_HSEA SUBTOTAL_ONE_HSHORE	Used in PAGEFOUR_REPORT routine to hold total hours for work centers 100, 110, 120, 121, 130, 131, and 140.
SUBTOTAL_ONE_MSEA SUBTOTAL_ONE_MSHORE	Used in PAGEFOUR_REPORT routine to hold total manpower requirements for work centers 100, 110, 120, 121, 130, 131, and 140.
SUBTOTAL_THREE_HSEA SUBTOTAL_THREE_HSHORE	Used in PAGEFOUR_REPORT routine to hold total hours for work centers 300, 310, and 320.
SUBTOTAL_THREE_MSEA SUBTOTAL_THREE_MSHORE	Used in PAGEFOUR_REPORT routine to hold manpower requirements for work centers 300, 310, and 320.
SUBTOTAL_TWO_HSEA SUBTOTAL_TWO_HSHORE	Used in PAGEFOUR_REPORT routine to hold total hours for work centers 200, 210, 211, 220, 230, and 240.
SUBTOTAL_TWO_MSEA SUBTOTAL_TWO_MSHORE	Used in PAGEFOUR_REPORT routine to hold total manpower requirements for work centers 200, 210, 211, 220, 230, and 240.

SUPPORT_EQUIPMENT_HOURS_ SEA (5) SUPPORT_EQUIPMENT_HOURS_ SHORE (5)	Stored arrays containing the support equipment maintenance hours per aircraft per week for Intermediate Maintenance Division I (5); see Appendix D.
TEMPMEN	Variable used in AIMD_CALCULATIONS routine to hold integer number of personnel requirements for TAD portion of AIMD.
TOT_SHORE_XA (7)	Array used in AIMD_CALCULATIONS routine to store the total number of permanent AIMD personnel required at Naval Air Station I (7) after the new aircraft are added.
TOT_SHORE_XB (7)	Array used in AIMD_CALCULATIONS routine to store the total number of permanent AIMD personnel required at Naval Air Station I (7) before the new aircraft are added.
TOTAL_AIRCRAFT	Total number of aircraft in the fleet; = AIRCRAFT_PER_SQUADRON * NUMBER_OF_ SQUADRONS.
TOTAL_AS_HOURS_SEA TOTAL_AS_HOURS_SHORE	The total number of Administrative Support hours for the production work centers.
TOTAL_CM_WORKLOAD_SEA (23) TOTAL_CM_WORKLOAD_SHORE (23)	Arrays containing the total (direct plus indirect) Corrective Maintenance workloads for work center I (23).
TOTAL_FLEET_I_LEVEL_SEA (5) TOTAL_FLEET_I_LEVEL_SHORE (5)	The total fleet TAD requirements for AIMD production division I (5); = I_LEVEL_ MANPOWER (I) * NUMBER_OF_SQUADRONS.

TOTAL_FLEET_SEA TOTAL_FLEET_SHORE	Variables used in the PAGETHREE_REPORT routine containing the total number of organizational and intermediate personnel required for the entire fleet.
TOTAL_I_LEVEL_MANHOURS	The total weekly Intermediate Maintenance workload for a squadron; = I_LEVEL_MANHOURS_WEEK * AIRCRAFT_PER_SQUADRON.
TOTAL_I_LEVEL_SEA TOTAL_I_LEVEL_SHORE	Total TAD AIMD requirements for a squadron; equal to the sum of I_LEVEL_MANPOWER over the 5 production divisions.
TOTAL_PERSONNEL_SEA TOTAL_PERSONNEL_SHORE	The total organizational and intermediate maintenance personnel for a squadron.
TOTAL_PM_WORKLOAD_SEA (23) TOTAL_PM_WORKLOAD_SHORE (23)	Arrays containing the total (direct plus indirect) weekly Preventive Maintenance workload for work center I (23).
TOTAL_RAW_PM_PLUS_CM_SEA TOTAL_RAW_PM_PLUS_CM_SHORE	The total weekly direct Preventive and Corrective Maintenance workload for a squadron; used to determine TOTAL_AS_HOURS.
TOTAL_TM_WORKLOAD_SEA (23) TOTAL_TM_WORKLOAD_SHORE (23)	The total weekly workload for a squadron in work center I (23). The sum of CM + PM + AS + OTHER hours.
TYPE_OF_AIRCRAFT	Input string of characters indicating the type of aircraft to be considered.
UTILITY_TASK_HOURS1 (23)	Stored array containing Utility Task hours per squadron for work center I (23) for VA, VF, and HS aircraft; see Appendix D.

UTILITY_TASK_HOURS2 (23)	Stored array containing the weekly Utility Task hours per squadron for work center I (23) for VS, VAQ, VAW, and RVAH aircraft; see Appendix D.
UTILITY_TASK_HOURS3 (23)	Stored array containing the weekly Utility Task hours per squadron for work center I (23) for VP, VQ, and HM aircraft; see Appendix D.
V1, V2, V3, V4	Input values for the Reliability and Maintainability parameters.
VAR_L	Variable used in the calculation of work center 320 workload indicating the pre-launch standby time per launch.
VAR_X	Variable used in the calculation of work center 320 workload indicating the average number of sorties per launch.
WC_CODE	Input value indicating the work center for which WC_OTHER_HOURS apply.
WC_OTHER_HOURS	Input (optional) value indicating direct manning workload that should be added to work center WC_CODE.
WORK_CENTER_CODES (23)	Stored array containing the numeric work center codes.
WORK_CENTER_NAMES (23)	Stored array containing the alphanumeric titles of the work centers.

WORKCENTER_CM_SPREAD_ OTHER (23)	Stored array indicating the percentage of a total CM workload that is placed in work center I (23) for aircraft types other than fighter or attack; see Appendix D.
WORKCENTER_CM_SPREAD_ VFA (23)	Stored array indicating the percentage of a total CM workload that is placed in work center I (23) for fighter and attack aircraft; see Appendix D.
WORKCENTER_PM_SPREAD_ OTHER (23)	Stored array indicating the percentage of a total PM workload that is placed in work center I (23) for aircraft types other than fighter or attack; see Appendix D.
WORKCENTER_PM_SPREAD_ VFA (23)	Stored array indicating the percentage of a total PM workload that is placed in work center I (23) for fighter or attack aircraft; see Appendix D.
WORKCENTER_TM_SPREAD_ OTHER (23)	Stored array indicating the percentage of a total TM (= PM + CM) workload that is placed in work center I (23) for aircraft types other than fighter or attack; see Appendix D.
WORKCENTER_TM_SPREAD_ VFA (23)	Stored array indicating the percentage of a total TM (= PM + CM) workload that is placed in work center I (23) for fighter or attack aircraft; see Appendix D.
WORKLOAD_SEA (23) WORKLOAD_SHORE (23)	Arrays used in AD_SPREAD procedures to hold the work center workloads that result from spreading a total workload.

WUC_PTR	Value that indicates the number of R&M input records that contain work unit code data.
WUC_XXX (25) WUC_J_TYPE (25) WUC_V1 (25) WUC_V2 (25)	Arrays that contain the R&M input data for work unit code records. The members of the arrays correspond to the order of input.
XXX_CODE	Input code indicating the work center (or if = 999, the total value) for which the R&M input values apply.

Appendix D

MODEL FACTORS

This appendix contains listings of all the factors contained in NAVMAN. The majority of the factors (Table D.1 through D.9) were developed by the Navy and are contained in the Squadron Manning Document (SQMD) model. The remaining model factors (Tables D.10 through D.15) were developed by analyzing historic Navy aircraft data (see Appendix B). The various tables are:

Table	Title	Page
D.1	Aircraft Codes	51
D.2	Requisition Factors	51
D.3	Production Delay Factors	52
D.4	Administrative Support Spreads	53
D.5	Facilities Maintenance Factors	54
D.6	Utility Task Hours	55
D.7	Intermediate Level Roundoff Table	56
D.8	Organizational Level Roundoff Table: Shore	56
D.9	Organizational Level Roundoff Table: Sea	56
D.10	Minimum Value for Work Center 230	57
D.11	I Level Spread of Total Hours to Production Divisions	57
D.12	I Level Support Equipment Hours per Aircraft	57
D.13	I Level Administrative Support Coefficients	58
D.14	CM Percent of TM Hours	58
D.15	Aggregate Workload to Work Center Spreads	59

Table D.1

AIRCRAFT CODES

VA, Attack (A-6, A-7)
VF, Fighter (F-14, F-4)
VP, Patrol (P-3)
VS, Anti Submarine (S-3)
VAW, Early Warning (E-2)
VAQ, ECM (EA-6)
VQ, Intelligence (EA-3)
RVAH, Photo (RF-8, RA-5C)
HM, Mine Sweeping
HS, Anti Sub Helio (SH-3)

NOTE: The model will accept either the mnemonic or the name.

Table D.2

REQUISITION FACTORS

Aircraft Type	Requisition Factor
VA, Attack	1.2723
VF, Fighter	1.9962
VP, Patrol	2.3956
VS, Anti Submarine	1.5376
VAW, Early Warning	3.2059
VAQ, ECM	1.9962
VQ, Intelligence	3.1333
NM, Mine Sweeping	1.5376
NS, Anti Submarine	1.5376

NOTE: The requisition factor is used in the standard equation for work center 050.

Table D.3

PRODUCTION DELAY FACTORS

Work Center	Production Delay Factors: Carrier Squadrons	Shore Squadrons
010 Maintenance Officer	0	0
020 Maintenance/Material Control	0	0
030 Maintenance Administration	0	0
040 Quality Assurance	0	0
050 Material Control	0	0
060 Data Analysis	0	0
100 Aircraft Division	0	0
110 Power Plants Branch	.30	.10
120 Airframes Branch	.20	.15
121 Corrosion Control	.10	.10
130 Aviator Equipment	.05	.05
131 Safety Equipment	.10	.10
140 Planned Maintenance	0	0
200 Avionics/Armament Division	0	0
210 Electrical Branch	.35	.30
211 Electronic Fire Control Branch	.35	.30
220 Electrical/Instruments Branch	.35	.20
230 Weapons Branch	.30	.10
240 Photo Shop	.35	.30
300 Line Division	0	0
310 Plane Captains	.20	.10
320 Troubleshooters	0	0

NOTES: Make ready/put away factor = .30; productive allowance factor = .20.

Table D.4

ADMINISTRATIVE SUPPORT SPREADS

Work Center	VA	VF	VP	VS	VAW	VAQ	VQ	RVAH	HS	HM
010	0	0	0	0	0	0	0	0	0	0
020[a]	0	0	0	0	0	0	0	0	0	0
030	0	0	0	0	0	0	0	0	0	0
040	0	0	0	0	0	0	0	0	0	0
050[a]	0	0	0	0	0	0	0	0	0	0
060	0	0	0	0	0	0	0	0	0	0
100	0	0	0	0	0	0	0	0	0	0
110	.088	.100	.125	.095	.130	.127	.125	.110	.095	.095
120	.090	.105	.145	.058	.193	.163	.145	.091	.058	.108
121	.066	.054	.088	.088	.088	.053	.088	.042	.126	.126
130	.065	.050	.066	.085	.040	.044	.066	.046	.128	.128
131	.079	.084	.084	.091	.087	.091	.084	.063	0	0
140	.067	.045	.097	.024	0	.059	.097	0	.024	.024
200	0	0	0	0	0	0	0	0	0	0
210	.086	.082	.113	.174	.138	.141	.187	.076	.174	.144
211	.108	.118	0	0	0	0	0	0	0	0
220	.106	.108	.095	.106	.142	.113	.115	.087	.106	.136
230	.123	.078	.094	.092	0	.070	0	.074	.092	.132
240	0	0	0	0	0	0	0	.25	0	0
300	0	0	0	0	0	0	0	0	0	0
310	.122	.174	.093	.187	.182	.139	.093	.161	.197	.107
320	0	0	0	0	0	0	0	0	0	0

[a]AS hours for these work centers are computed from standard equations.

Table D.5

FACILITIES MAINTENANCE FACTORS

Work Center	FM Factor
010	0
020	.0630
030	0
040	0
050	.0653
060	0
100	0
110	.0956
120	.0998
121	.0621
130	.0590
131	.0696
140	.0923
200	0
210	.0769
211	.1060
220	.0578
230	.0891
240	.0408
300	0
310	.3182
320	0

Table D.6

UTILITY TASK HOURS[a]

Work Center	VA,VR,HS	VS,VAW, VAQ,RVAH	VP,VQ,HM
010	0	0	0
020	10.4	10.4	10.4
030	0	0	0
040	0	0	0
050	10.4	10.4	10.4
060	0	0	0
100	0	0	0
110	41.5	20.7	0
120	41.5	20.7	0
121	0	0	0
130	10.4	10.4	10.4
131	10.4	10.4	10.4
140	0	0	0
200	0	0	0
210	62.2	20.7	0
211	62.2	0	0
220	41.5	20.7	0
230	41.5	20.7	0
240	0	0	0
300	0	0	0
310	103.7	62.2	0
320	0	0	0

[a]UT hours apply to carrier-based squadrons only. There are no UT hours for shore activities.

Table D.7

INTERMEDIATE LEVEL ROUNDOFF TABLE

If Fractional Men Are Greater Than	Integer Requirement
1.076	2
2.151	3
3.227	4
4.302	5
5.378	6
6.453	7
7.5	8
X.5	X+1

Table D.8

ORGANIZATIONAL LEVEL ROUNDOFF TABLE: SHORE

If Fractional Men Are Greater Than	Integer Requirement
1.078	2
2.156	3
3.234	4
4.312	5
5.391	6
6.469	7
7.5	8
X.5	X+1

Table D.9

ORGANIZATIONAL LEVEL ROUNDOFF TABLE: SEA

If Fractional Men Are Greater Than	Integer Requirement
1.05	2
2.10	3
3.15	4
4.20	5
5.25	6
6.30	7
7.35	8
8.40	9
9.45	10
10.5	11
X.5	X+1

Table D.10

MINIMUM VALUE FOR WORK CENTER 230

Type of Aircraft	Minimum Men per Aircraft
VA	2.200
VF	1.583
VP	.556
VS	1.000
VAW	0
VAQ	1.000
VQ	0
RVAH	0
HM	.500
HS	.750

Table D.11

I LEVEL SPREAD OF TOTAL HOURS TO PRODUCTION DIVISIONS

	Aircraft Type									
Division	VA	VF	VP	VS	VAW	VAQ	VQ	RVAH	HM	HS
Power Plants	.20	.25	.28	.25	.20	.15	.35	.20	.33	.33
Airframe	.12	.15	.16	.11	.10	.11	.18	.23	.15	.15
Avionics	.60	.50	.52	.57	.65	.69	.50	.55	.50	.50
Armament	.05	.05	.01	.01	0	0	0	0	.01	.01
Aviator's Eq.	.03	.05	.03	.06	.05	.05	.02	.02	.01	.01

Table D.12

I LEVEL SUPPORT EQUIPMENT HOURS PER AIRCRAFT

Division	Sea	Shore
Power Plants	.52	.15
Airframe	.42	.19
Avionics	1.93	.95
Armament	.17	.09
Aviator's Eq.	.39	.29

Table D.13

I LEVEL ADMINISTRATIVE SUPPORT COEFFICIENTS

Division	Coefficients
Power Plants	2.3500
Airframe	4.5139
Avionics	4.7271
Armament	5.2731
Aviator's Eq.	6.1751

Table D.14

CM PERCENT OF TM HOURS

Work Center	VF, VA	All Other
010	0[a]	0
020	0	0
030	0	0
040	0	0
050	0	0
060	0	0
100	0	0
110	.66	.81
120	.61	.72
121	1.00	1.00
130	.84	.95
131	.60	.77
140	0	0
200	0	0
210	.82	.93
211	.86	0
220	.83	.90
230	.64	.92
240	0	.90
300	0	0
310	.55	.67
320	0	0

[a]0 indicates no CM or PM workload for that work center.

Table D.15

AGGREGATE WORKLOAD TO WORK CENTER SPREADS

Work Center	TM VF, VA	TM Other	CM VF, VA	CM Other	PM VF, VA	PM Other
010	0	0	0	0	0	0
020	0	0	0	0	0	0
030	0	0	0	0	0	0
040	0	0	0	0	0	0
050	0	0	0	0	0	0
060	0	0	0	0	0	0
100	0	0	0	0	0	0
110	.096	.117	.095	.112	.096	.139
120	.188	.166	.173	.143	.235	.296
121	.042	.069	.063	.083	0	0
130	.009	.022	.011	.025	.003	.005
131	.058	.056	.053	.051	.065	.084
140	0	0	0	0	0	0
200	0	0	0	0	0	0
210	.064	.241	.079	.271	.034	.108
211	.068	0	.090	0	.030	0
220	.099	.126	.124	.137	.044	.058
230	.128	.036	.110	.041	.162	.008
240	0	0	0	0	0	0
300	0	0	0	0	0	0
310	.248	.167	.202	.137	.331	.302
320	0	0	0	0	0	0

Appendix E

PAYGRADE MATRICES

This appendix contains listings of the paygrade matrices used by NAVMAN. Each matrix shows the required number of personnel of each rank or skill level when a work center is to be manned with a given number of billets. The matrices were developed by the Navy and are used in the Squadron Manning Document (SQMD) model.

Title	Page
Paygrade Matrices Index	61
050 Matrix: Material Control	62
020 Matrix: Material Control/Maintenance	63
Production Matrix	64
Ordnance: Work Center 230	68
Line Division	70

PAYGRADE MATRICES INDEX

REFERENCE TABLE

Work Center Number	Work Center Description	Paygrade or Reference
010	Maintenance Officer	Lt. Cmdr.
020	Material Control/Maint.	See 020 matrix
030	Maintenance Admin.	E-5
040	Quality Assurance	E-8(1), E-4(1), rest are E-6s
050	Material Control	See 050 matrix
060	Data Analysis	E-6
100	Aircraft Division	Senior to supervisor in 1XX, E-8 at most
110	Power Plant	See production matrix
120	Airframes	See production matrix
121	Corrosion Control	See production matrix
130	Aviator Equipment	See production matrix
131	Safety Equipment	See production matrix
140	Planned Maintenance	E-6
200	Avionics/Armament	Senior to supervisor in 2XX, E-8 at most
210	Electrical	See production matrix
211	Electronic Fire Control	See production matrix
220	Electrical/Instrument	See production matrix
230	Weapons	See 230 matrix
240	Photo	See production matrix
300	Line Division	Senior to supervisor in 3XX, E-8 at most
310	Plane Captains	See line div. matrix
320	Troubleshooters	See line div. matrix

050 MATRIX: MATERIAL CONTROL

Grades	Billets 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
E-9																				
E-8																				
E-7								1	1	1	1	1	1	1	1					
E-6			1	1	1	1	1	1	1	1	1	2	2	2	2					
E-5	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2					
E-4				1	1	2	2	2	2	3	3	3	4	4	4					
E-3		1	1	1	2	2	2	2	3	3	4	4	4	5	6					
E-2																				

020 MATRIX: MATERIAL CONTROL/MAINTENANCE

	Billets																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Grades																				
E-9																				
E-8																				
E-7								1	1	1	1	1	1	1	1	1	1	1	1	1
E-6					1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
E-5	1	1	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	3	3
E-4			1	1	1	2	2	2	2	3	3	3	4	4	4	4	5	5	5	5
E-3		1	1	2	2	2	3	3	4	4	4	5	5	6	6	7	7	8	9	10
E-2																				

PRODUCTION MATRIX

	Billets																			
Grades	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
E-9																				
E-8																				
E-7											1	1	1	1	1	1	1	1	1	1
E-6					1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2
E-5	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4
E-4			1	1	1	2	2	3	3	3	3	4	4	4	4	4	4	4	5	5
E-3		1	1	2	2	2	3	3	3	4	4	4	4	5	5	6	7	7	7	8
E-2																				

PRODUCTION MATRIX (cont.)

	Billets																			
Grades	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
E-9																				
E-8																				1
E-7	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	1
E-6	2	2	2	2	2	3	3	3	3	3	3	3	3	3	3	3	4	4	4	4
E-5	4	4	4	5	5	5	5	5	5	5	6	6	6	6	6	7	7	7	7	8
E-4	5	5	6	6	6	6	6	7	7	7	7	8	8	9	9	9	9	9	10	10
E-3	9	10	10	10	11	11	12	12	13	13	13	13	14	14	15	15	15	16	16	16
E-2																				

PRODUCTION MATRIX (cont.)

	Billets																			
	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
Grades																				
E-9																				
E-8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
E-7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2
E-6	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5	5	6	6	6	6
E-5	8	8	8	9	9	9	10	10	10	10	11	11	11	11	12	12	12	12	12	12
E-4	10	10	11	11	11	12	12	12	12	13	13	13	14	14	14	15	15	15	15	15
E-3	17	18	18	18	19	19	19	19	20	20	20	21	21	22	22	22	22	23	24	24
E-2																				

PRODUCTION MATRIX (cont.)

	Billets																			
	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
Grades																				
E-9																				
E-8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1					
E-7	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2					
E-6	6	6	6	6	6	6	6	6	6	6	6	6	6	6	6					
E-5	13	13	13	13	14	14	14	14	15	15	15	15	16	16	16					
E-4	15	16	16	16	16	17	17	17	17	18	18	18	18	19	19					
E-3	24	24	25	26	26	26	27	28	28	28	29	30	30	30	31					
E-2																				

ORDNANCE: WORK CENTER 230

	Total Billets Required																			
Grades	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
E-9																				
E-8																				
E-7											1	1	1	1	1	1	1	1	1	1
E-6					1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2
E-5	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	3	4	4	4	4
E-4			1	1	1	2	2	2	2	2	2	2	3	3	4	4	4	5	5	6
E-3		1	1	2	2	2	3	3	3	4	4	4	4	5	5	6	6	6	7	7
E-2																				

ORDNANCE: WORK CENTER 230 (cont.)

	Total Billets Required																			
	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
Grades																				
E-9																				
E-8																				
E-7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
E-6	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
E-5	4	5	5	5	6	6	6	6	7	7	7	8	8	8	8	8	9	9	9	9
E-4	6	6	6	6	6	7	7	8	8	8	9	9	9	10	10	10	10	11	11	12
E-3	7	7	8	9	9	9	10	10	10	11	11	11	12	12	13	14	14	14	15	15
E-2																				

LINE DIVISION

Grades	Total Billets Required																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
E-9																				
E-8																				
E-7											1	1	1	1	1	1	1	1	1	1
E-6							1	1	1	1	1	1	1	1	1	1	1	1	1	1
E-5	1	1	1	1	1	1														1
E-4							1	1	1	1	1	1	2	2	2	2	2	2	2	2
E-3		1	2	3	4	5	5	6	7	8	8	9	9	10	11	12	13	14	15	15
E-2																				

LINE DIVISION (cont.)

	Total Billets Required																			
Grades	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
E-9																				
E-8																				
E-7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2
E-6	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2
E-5	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-4	3	3	3	3	3	3	3	4	4	4	4	4	4	4	4	5	5	5	5	5
E-3	15	16	17	18	18	19	20	20	21	21	22	23	24	25	26	26	27	28	29	29
E-2																				

LINE DIVISION (cont.)

	Total Billets Required																			
Grades	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
E-9																				
E-8																				
E-7	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-6	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-5	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-4	5	5	5	6	6	6	6	6	6	6	6	7	7	7	7	7	7	7	7	8
E-3	30	31	32	32	33	34	35	36	37	38	39	39	40	41	42	43	44	45	46	46
E-2																				

LINE DIVISION (cont.)

	Total Billets Required																			
Grades	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
E-9																				
E-8																				
E-7	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-6	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-5	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-4	8	8	8	8	8	8	8	9	9	9	9	9	9	9	9	10	10	10	10	10
E-3	47	48	49	50	51	52	53	53	54	55	56	57	58	59	60	60	61	62	63	64
E-2																				

LINE DIVISION (cont.)

	Total Billets Required																			
Grades	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
E-9																				
E-8																				
E-7	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-6	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-5	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-4	10	10	10	11	11	11	11	11	11	11	11	12	12	12	12	12	12	12	12	13
E-3	65	66	67	67	68	69	70	71	72	73	74	74	75	76	77	78	79	80	81	81
E-2																				

LINE DIVISION (cont.)

Grades	Total Billets Required																			
	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120
E-9																				
E-8																				
E-7	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-6	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-5	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
E-4	13	13	13	13	13	13	13	14	14	14	14	14	14	14	14	15	15	15	15	15
E-3	82	83	84	85	86	87	88	88	89	90	91	92	93	94	95	95	96	97	98	99
E-2																				

Appendix F

RELIABILITY AND MAINTAINABILITY DATA BANK

This appendix contains historical data on the reliability and maintainability of Navy aircraft. During the early stages of an aircraft's development process, data for analogous aircraft subsystems are often used as benchmarks for a new aircraft's reliability; in this regard, the data in this appendix may prove useful to a user of the NAVMAN model. The tables and data are described below.

Table F.1: Readiness/Utilization History

This table contains organizational and intermediate level maintenance data for various Navy aircraft. Included are:

- Total Direct (Organizational plus Intermediate) Maintenance Manhours per Flying Hour (DMMH/FH)
- Organizational Level Unscheduled (Maintenance Action Forms and Support Action Forms) Maintenance Manhours per Flying Hour
- Total monthly flying hours for the aircraft fleet
- Total monthly flying hours per aircraft

Table F.2: PM/CM Data Bank (October 1976)

This table presents organizational level preventive maintenance manhours and corrective maintenance percentages for the various work centers for a number of Navy aircraft. Included are:

- Work Center (W/C) and Rating (RTG)
- Preventive Maintenance Manhours (PMMH) per
  - Sortie
  - Week per aircraft (WK/AC)
  - Flying hour (FH)
  - Daily inspection (DLY IN)
- The percentage of the total corrective maintenance (CM) MAF and SAF workloads attributable to each organizational level work center.

Table F.3: PM/CM Data Bank (July 1977)

See previous discussion.

Table F.4: Intermediate Level Maintenance Manhours per Aircraft per Month (b value)

This table contains the values used in the ACM-02 model to generate intermediate level monthly workloads.

For organizational level maintenance, the preventive maintenance workloads by work center can be found in Tables F.2 and F.3. The corrective maintenance workloads can be calculated by using the values in Table F.1 and the percentage spreads in Tables F.2 or F.3. MAF and SAF workloads are calculated separately and then summed to determine the total CM workload per work center.

The data in Tables F.1 through F.3 were received from the Navy Manpower and Material Analysis Center, Atlantic (NAVMMACLANT). They were extracted by NAVMMACLANT analysts from the 3M maintenance data collection system and summarized in the format shown. The intermediate maintenance values in Table F.4 were extracted from ACM-02.

Table F.1

READINESS/UTILIZATION HISTORY

	1976	1977	1978
A-4C			
Total DMMH/FH	22.6	75.6	73.7
Org unsch-MAF	7.2	17.5	7.1
Org unsch-SAF	7.3	32.4	43.1
Flight hrs	4,422.0	149.0	(NA)
Flight hrs/ac	18.4	7.9	(NA)
A-4E			
Total DMMH/FH	17.2	21.2	22.9
Org unsch-MAF	5.6	6.0	6.3
Org unsch-SAF	5.7	6.8	8.1
Flight hrs	17,075.0	14,762.0	14,291.0
Flight hrs/ac	24.1	20.8	21.9
A-6A			
Total DMMH/FH	58.8	54.4	46.0
Org unsch-MAF	18.8	16.7	14.5
Org unsch-SAF	14.5	13.5	9.5
Flight hrs	14,379.0	7,191.0	2,164.0
Flight hrs/ac	19.9	20.7	15.7
A-6E			
Total DMMH/FH	33.7	41.3	41.8
Org unsch-MAF	9.4	11.7	12.1
Org unsch-SAF	10.3	12.2	11.3
Flight hrs	57,826.0	65,214.0	74,824.0
Flight hrs/ac	30.7	27.3	28.6
A-7A			
Total DMMH/FH	24.3	21.1	20.5
Org unsch-MAF	7.8	5.9	5.5
Org unsch-SAF	8.1	6.7	6.7
Flight hrs	29,136.0	27,844.0	13,666.0
Flight hrs/ac	30.2	31.5	29.0
A-7E			
Total DMMH/FH	23.8	28.0	28.2
Org unsch-MAF	10.2	7.3	7.4
Org unsch-SAF	3.7	11.1	11.4
Flight hrs	80.0	122,558.0	143,305.0
Flight hrs/ac	6.7	33.8	38.4

Table F.1 (cont.)

	1976	1977	1978
	----	----	----
EA-6B			
Total DMMH/FH	48.9	51.6	60.5
Org unsch-MAF	15.2	16.6	19.3
Org unsch-SAF	13.5	12.5	15.5
Flight hrs	15,348.0	17,134.0	19,003.0
Flight hrs/ac	32.4	33.2	36.7
E-2C			
Total DMMH/FH	27.2	29.8	28.0
Org unsch-MAF	8.4	7.9	8.0
Org unsch-SAF	7.5	8.4	8.2
Flight hrs	11,653.0	12,943.0	15,246.0
Flight hrs/ac	44.7	41.1	45.1
F-14A			
Total DMMH/FH	56.2	52.4	56.9
Org unsch-MAF	18.4	16.9	18.6
Org unsch-SAF	16.0	16.1	17.1
Flight hrs	32,100.0	45,538.0	49,297.0
Flight hrs/ac	19.7	24.7	24.8
F-4J			
Total DMMH/FH	43.3	47.9	46.2
Org unsch-MAF	15.2	16.7	15.4
Org unsch-SAF	12.2	13.3	12.9
Flight hrs	80,311.0	79,482.0	82,245.0
Flight hrs/ac	25.4	23.7	25.8
P-3C			
Total DMMH/FH	17.0	19.7	19.2
Org unsch-MAF	5.9	6.4	6.2
Org unsch-SAF	3.6	4.2	3.9
Flight hrs	84,221.0	93,315.0	102,814.0
Flight hrs/ac	57.4	58.1	62.7
S-3A			
Total DMMH/FH	30.1	28.0	29.6
Org unsch-MAF	9.3	8.7	9.2
Org unsch-SAF	9.6	8.7	9.7
Flight hrs	35,550.0	54,097.0	62,555.0
Flight hrs/ac	36.7	35.5	34.5
F-4J			
Total DMMH/FH	43.3	47.9	46.2
Org unsch-MAF	15.2	16.7	15.4
Org unsch-SAF	12.2	13.3	12.9
Flight hrs	80,311.0	79,482.0	82,245.0
Flight hrs/ac	25.4	23.7	25.8

Table F.1 (cont.)

	1976	1977	1978
F-4M			
Total DMMH/FH	NA	29.4	NA
Org unsch-MAF	NA	10.7	NA
Org unsch-SAF	NA	6.6	NA
Flight hrs	NA	16,361.0	NA
Flight hrs/ac	NA	26.9	NA
F-5E			
Total DMMH/FH	9.4	10.9	13.6
Org unsch-MAF	1.6	1.6	2.2
Org unsch-SAF	4.4	4.4	5.1
Flight hrs	2,051.0	2,776.0	2,496.0
Flight hrs/ac	34.9	32.3	25.3

Table F.2

PM/CM DATA BANK
(October 1976)

A/C	W/C	RTG	PMMH Sortie	PMMH WK/AC	PMMH FH	PMMH DLY IN	CM MAF	CM SAF
A-4C	110	AD	.000	.000	.256	.000	20.28	10.71
	120	AMH	.000	.000	.044	.033	10.96	3.26
	120	AMS	.000	1.342	.123	.050	13.00	3.87
	121	AMS	.000	.000	.000	.000	6.96	7.39
	130	PR	.000	.167	.000	.000	.33	.50
	131	AME	.000	.319	.006	.020	3.17	5.84
	210	AT	.000	.688	.009	.000	18.88	1.46
	220	AE	.000	.097	.054	.167	24.77	3.53
	230	AO	.000	.314	.041	.000	1.65	17.88
	310	PC	1.000	1.500	.001	1.100	.00	45.56
A-4E	110	AD	.000	.000	.098	.000	19.17	5.44
	120	AMH	.000	.000	.046	.000	11.44	2.50
	120	AMS	.000	3.292	.095	.050	15.92	3.47
	121	AMS	.000	.000	.000	.000	11.92	4.61
	130	PR	.000	.150	.000	.000	.38	.59
	131	AME	.000	.369	.001	.000	5.37	5.33
	210	AT	.000	1.110	.008	.000	10.43	3.23
	220	AE	.083	.025	.044	.167	22.86	3.92
	230	AO	1.817	.025	.049	.000	2.50	14.77
	310	PC	.850	.000	.000	.917	.00	56.12
A-6A	110	AD	.050	.500	.138	.000	9.97	6.37
	120	AMH	.000	.875	.154	.000	10.11	5.15
	120	AMS	.000	6.129	.327	.000	10.22	5.21
	121	AMS	.000	.000	.000	.000	11.57	3.22
	130	PR	.000	.167	.000	.000	.13	.18
	131	AME	.000	2.135	.052	.667	3.08	1.53
	210	AT	.000	3.104	.011	.000	10.14	4.22
	211	AO	.000	.500	.012	.000	28.94	3.22
	220	AE	.000	.615	.018	.000	14.50	9.93
	230	AO	.000	1.375	.008	.000	1.34	12.15
	310	PC	.430	1.000	.004	3.567	.00	48.82
A-6E	110	AD	.050	1.000	.138	.000	16.28	9.65
	120	AMH	.000	.875	.154	.000	8.13	2.99
	120	AMS	.000	6.129	.319	.000	7.38	2.72
	121	AMS	.000	.000	.000	.000	13.47	5.68
	130	PR	.000	.167	.000	.000	.27	.24
	131	AME	.000	2.135	.052	.667	3.37	2.64
	210	AT	.000	3.104	.010	.000	11.41	6.70
	211	AO	.000	.500	.004	.000	18.54	4.76
	220	AE	.000	.615	.018	.000	18.25	6.84
	230	AO	.000	1.375	.008	.000	2.91	15.18
	310	PR	.433	1.000	.004	3.567	.00	42.60

Table F.2 (cont.)

A/C	W/C	RTG	PMMH Sortie	PMMH WK/AC	PMMH FH	PMMH DLY IN	CM MAF	CM SAF
A-7E	110	AD	.000	.233	.011	.333	16.99	4.29
	120	AMH	.000	.036	.053	.000	13.02	6.70
	120	AMS	.000	2.590	.106	.000	7.72	3.97
	121	AMS	.000	.000	.000	.000	13.60	5.99
	130	PR	.000	.163	.000	.000	.16	.11
	131	AME	.000	1.247	.006	.000	4.56	2.21
	210	AT	.000	.044	.004	.000	13.14	6.67
	211	AO	.000	.272	.032	.000	3.24	1.62
	220	AE	.000	.389	.017	.000	17.39	5.07
	230	AC	.000	2.275	.016	.000	10.18	23.50
	310	PR	1.000	1.050	.000	1.167	.00	39.87
EA-6B	110	AD	.000	.996	.275	.000	10.04	10.46
	120	AMH	.000	.300	.095	.000	9.58	7.44
	120	AMS	.000	4.447	.194	.000	6.21	4.82
	121	AMS	.000	.000	.000	.000	9.90	5.14
	130	PR	.000	.500	.000	.000	.43	.54
	131	AME	.000	3.123	.024	.750	4.67	5.67
	210	AT	.000	1.083	.032	.920	31.83	1.65
	211	AO	.000	.000	.000	.000	9.85	.22
	220	AE	.000	.478	.029	.000	16.83	9.51
	230	AO	.000	.724	.000	.500	.66	2.80
	310	PC	.500	.500	.005	2.917	.00	51.75
E-2C	110	AD	.000	.000	.004	.000	19.74	5.21
	120	AMH	.000	.000	.037	.000	8.64	4.41
	120	AMS	.000	6.675	.122	.000	9.42	4.81
	121	AMS	.000	.000	.000	.000	8.92	19.36
	130	PR	.000	.250	.000	.000	.19	.74
	131	AME	.000	.136	.019	.250	2.95	11.98
	210	AT	.000	.000	.009	.000	34.01	5.81
	220	AE	.000	.658	.014	.000	16.14	7.07
	230	AO	.000	.000	.000	.000	.00	.13
	310	PC	.870	1.000	.000	2.370	.00	40.46
F-14A	110	AD	.000	.000	.397	.500	19.98	2.17
	120	AMH	.000	.339	.142	.500	7.49	2.90
	120	AMS	.000	6.585	.564	.500	10.45	4.05
	121	AMS	.000	.000	.000	.000	3.62	12.85
	130	PR	.000	.167	.000	.000	.16	.17
	131	AME	.000	1.166	.110	.417	5.69	5.28
	210	AT	.000	.250	.020	.000	8.70	2.66
	211	AQ	.000	1.063	.130	.000	2.19	.70
	220	AE	.000	1.000	.109	.000	23.40	7.39
	230	AC	.000	1.573	.091	.000	18.31	18.95
	310	PC	.500	1.000	.008	1.700	.00	42.87

Table F.2 (cont.)

A/C	W/C	RTG	PMMH Sortie	PMMH WK/AC	PMMH FH	PMMH DLY IN	CM MAF	CM SAF
F-4J	110	AD	.000	1.000	.277	.317	14.48	2.55
	120	AMH	.000	1.557	.184	.080	10.14	4.59
	120	AMS	.000	2.291	.682	.667	12.83	5.80
	121	AMS	.000	.000	.000	.000	11.35	15.79
	130	PR	.000	.272	.000	.000	.21	.53
	131	AME	.000	1.468	.042	.200	6.08	2.26
	210	AT	.000	3.981	.009	.000	7.63	1.61
	211	AQ	.000	1.817	.048	.000	17.28	1.62
	220	AE	.000	1.425	.054	.167	17.87	2.11
	320	AO	.000	4.717	.085	.000	2.12	17.29
	310	PC	.500	2.600	.013	1.333	.00	45.83
P-3C	110	AD	.000	.542	.089	.333	18.93	8.06
	120	AMH	.000	.042	.007	.000	4.12	3.10
	120	AMS	.000	2.290	.041	.417	13.15	9.91
	121	AMS	.000	.000	.000	.000	4.06	17.10
	130	PR	.000	.250	.000	.000	.48	.85
	131	AME	.000	.175	.010	.000	2.87	5.03
	210	AT	.000	1.195	.009	.000	23.53	4.27
	210	AX	.000	.000	.000	.000	12.60	2.47
	220	AE	.000	.500	.025	.000	19.06	5.46
	230	AO	.000	.125	.004	.000	1.11	17.19
	240	PH	.000	.000	.000	.000	.09	.40
	310	PC	3.000	.000	.000	.000	.00	26.18
	350	FC	.000	.000	.003	.000	.00	.00
S-3A	110	AD	.000	3.000	.093	.000	12.82	5.24
	120	AMH	.000	.031	.003	.000	6.67	5.32
	120	AMS	.000	2.963	.124	.000	13.17	10.51
	121	AMS	.000	.000	.000	.000	3.30	6.73
	130	PR	.000	.167	.000	.000	.55	.31
	131	AME	.000	.029	.008	.000	5.63	3.21
	210	AT	.000	.000	.001	.217	14.27	1.34
	210	AX	.000	.350	.000	.000	12.48	.84
	220	AE	.000	.031	.004	.000	30.77	3.79
	230	AO	.000	.008	.002	.000	.34	11.32
	310	PC	.500	2.167	.000	.567	.00	51.39

Table F.3

PM/CM DATA BANK
(July 1977)

A/C	W/C	RTG	PMMH Sortie	PMMH WK/AC	PMMH FH	PMMH DLY IN	CM MAF	CM SAF
A-4C	110	AD	.000	.000	.256	.000	20.28	10.70
	120	AMH	.000	.000	.044	.033	10.96	3.26
	120	AMS	.000	1.342	.123	.050	13.00	3.87
	121	AMS	.000	.000	.000	.000	6.96	7.39
	130	PR	.000	.167	.000	.000	.33	.50
	131	AME	.000	.336	.006	.200	3.17	5.84
	210	AT	.000	.688	.009	.000	18.88	1.46
	220	AE	.000	.097	.054	.167	24.77	3.53
	230	AD	.000	.314	.041	.000	1.65	17.88
	310	PC	1.000	1.500	.001	1.100	.00	45.56
A-4E	110	AD	.000	.000	.099	.000	19.17	5.44
	120	AMH	.000	.000	.046	.000	11.44	2.50
	120	AMS	.000	3.292	.095	.050	15.92	3.47
	121	AMS	.000	.000	.000	.000	11.92	4.61
	130	PR	.000	.150	.000	.000	.38	.59
	131	AME	.000	.369	.001	.000	5.37	5.33
	210	AT	.000	1.077	.008	.000	10.43	3.23
	220	AE	.083	.025	.044	.167	22.86	3.92
	230	AD	1.817	.025	.049	.000	2.50	14.77
	310	PC	.850	.000	.000	.917	.00	56.12
A-6A	110	AD	.000	1.583	.349	.083	9.97	6.37
	120	AMH	.000	1.250	.150	.000	10.11	5.15
	120	AMS	.000	8.012	.499	.000	10.22	5.21
	121	AMS	.000	.000	.000	.000	11.57	3.22
	130	PR	.000	.000	.000	.000	.13	.18
	131	AME	.000	1.641	.067	.333	3.08	1.53
	210	AT	.000	5.336	.012	.000	10.14	4.22
	211	AQ	.000	.687	.011	.000	28.94	3.22
	220	AE	.000	2.167	.025	.000	14.50	9.93
	230	AO	.000	1.625	.020	.000	1.34	12.15
	310	PC	.500	1.000	.004	4.517	.00	48.82
A-6E	110	AD	.000	1.583	.352	.083	16.28	9.65
	120	AMH	.000	1.250	.156	.000	8.13	2.99
	120	AMS	.000	8.012	.524	.000	7.38	2.72
	121	AMS	.000	.000	.000	.000	13.47	5.68
	130	PR	.000	.000	.000	.000	.27	.24
	131	AME	.000	1.557	.077	.333	3.37	2.64
	210	AT	.000	5.336	.013	.000	11.41	6.70
	211	AQ	.000	.687	.011	.000	18.54	4.76
	220	AE	.000	2.167	.025	.000	18.25	6.84
	230	AO	.000	1.625	.020	.000	2.91	15.18
	310	PC	.500	1.000	.004	4.350	.00	42.60

Table F.3 (cont.)

A/C	W/C	RTG	PMMH Sortie	PMMH WK/AC	PMMH FH	PMMH DLY IN	CM MAF	CM SAF
A-7E	110	AD	.000	.259	.011	.333	16.99	4.29
	120	AMH	.000	.036	.053	.000	13.02	6.70
	120	AMS	.000	2.590	.107	.000	7.72	3.97
	121	AMS	.000	.000	.000	.000	13.60	5.99
	130	PR	.000	.163	.000	.000	.16	.11
	131	AME	.000	1.247	.006	.583	4.56	2.21
	210	AT	.000	.044	.004	.000	13.14	6.67
	211	AQ	.000	.272	.032	.000	3.24	1.62
	220	AE	.000	.395	.017	.000	17.39	5.07
	230	AO	.000	2.275	.016	.000	10.18	23.50
	310	PC	1.000	1.050	.000	1.167	.00	39.87
EA-6B	110	AD	.000	1.521	.309	.000	10.04	10.46
	120	AMH	.000	.750	.061	.000	9.58	7.44
	120	AMS	.000	6.222	.339	.000	6.21	4.82
	121	AMS	.000	.000	.000	.000	9.90	5.14
	130	PR	.000	.500	.000	.000	.43	.54
	131	AME	.000	4.051	.031	.750	4.67	5.67
	210	AT	.000	1.908	.041	.917	31.83	1.65
	211	AQ	.000	.000	.000	.000	9.85	.22
	220	AE	.000	2.428	.036	.000	16.83	9.51
	230	AO	.000	.724	.005	.500	.66	2.80
	310	PC	.500	.500	.003	2.917	.00	51.75
E-2C	110	AD	.000	.813	.041	.000	19.74	5.21
	120	AMH	.000	.000	.030	.000	8.64	4.41
	120	AMS	.000	8.057	.187	.000	9.42	4.81
	121	AMS	.000	.000	.000	.000	8.92	19.36
	130	PR	.000	.250	.000	.000	.19	.74
	131	AME	.000	.139	.023	.000	2.95	11.98
	210	AT	.000	.000	.023	.000	34.01	5.81
	220	AE	.000	1.172	.005	.000	16.14	7.07
	230	AO	.000	.000	.000	.000	.00	.13
	310	PC	.500	2.250	.000	1.917	.00	40.46
F-14A	110	AD	.000	.077	.533	.500	19.98	2.17
	120	AMH	.000	.339	.100	.833	7.49	2.90
	120	AMS	.000	6.585	.392	.500	10.45	4.05
	121	AMS	.000	.000	.000	.000	3.62	12.85
	130	PR	.000	.167	.000	.000	.16	.17
	131	AME	.000	1.438	.072	.417	5.69	5.28
	210	AT	.000	.250	.014	.000	8.70	2.66
	211	AQ	.000	1.455	.086	.000	13.43	.70
	220	AE	.000	1.000	.111	.000	23.40	7.39
	230	AO	.000	1.073	.061	.000	7.08	18.95
	310	PC	.500	1.000	.006	2.167	.00	42.87

Table F.3 (cont.)

A/C	W/C	RTG	PMMH Sortie	PMMH WK/AC	PMMH FH	PMMH DLY IN	CM MAF	CM SAF
F-4J	110	AD	.000	1.733	.313	.233	14.48	2.55
	120	AMH	.000	1.250	.143	.000	10.14	4.59
	120	AMS	.000	1.534	.293	.000	12.83	5.80
	121	AMS	.000	.000	.000	.000	11.35	15.79
	130	PR	.000	.272	.000	.000	.21	.53
	131	AME	.000	1.408	.031	.200	6.08	2.26
	210	AT	.000	4.001	.008	.000	7.63	1.01
	211	AQ	.000	1.817	.023	.000	17.28	1.62
	220	AE	.000	1.244	.051	.000	2.12	2.11
	230	AO	.000	4.783	.112	.000	.00	17.29
	310	PC	.500	2.583	.062	1.433	11.76	45.83
P-3C	110	AD	.000	.542	.083	.417	18.93	8.06
	120	AMH	.000	.042	.007	.083	4.12	3.10
	120	AMS	.000	2.352	.034	.667	13.15	9.91
	121	AMS	.000	.000	.000	.000	4.06	17.10
	130	PR	.000	.250	.000	.000	.48	.85
	131	AME	.000	.191	.010	.000	2.87	5.03
	210	AT	.000	2.127	.008	.000	23.53	4.27
	210	AX	.000	.000	.000	.000	12.60	2.47
	220	AE	.000	.500	.020	.000	19.06	5.46
	230	AO	.000	.104	.004	.000	1.11	17.19
	240	PH	.000	.000	.000	.000	.09	.40
	310	PC	.000	.000	.000	.000	.00	26.18
	350	FC	3.000	.000	.003	.000	.00	.00
S-3A	110	AD	.000	3.033	.093	.000	12.82	5.24
	120	AMH	.000	.048	.004	.000	6.67	5.32
	120	AMS	.000	3.568	.124	.000	13.17	10.51
	121	AMS	.000	.000	.000	.000	3.30	6.73
	130	PR	.000	.167	.000	.000	.55	.31
	131	AME	.000	.029	.008	.167	5.63	3.21
	120	AT	.000	.031	.001	.217	14.27	1.34
	210	AX	.000	.350	.000	.000	12.48	.84
	220	AE	.000	.031	.004	.000	30.77	3.79
	230	AO	.000	.610	.002	.000	.34	11.32
	310	PC	.500	2.417	.000	.567	.00	51.39

Table F.4

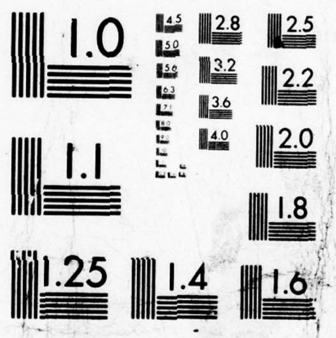
INTERMEDIATE LEVEL MAINTENANCE MANHOURS
PER AIRCRAFT PER MONTH
(b value)

Type Aircraft	b value
A-3B	37.3846
EA-3B	312.5494
KA-3B	125.1246
RA-3B	28.4396
ERA-3B	330.2738
TA-3B	131.7015
NA-3B	27.4833
NRA-3B	31.4797
A-4C	105.3328
A-4E	75.5632
A-4F	54.7682
EA-4F	86.0743
TA-4F	79.4524
TA-4J	79.8912
A-4L	45.4140
A-4M	69.1994
NA-4E	14.8308
NA-4F	32.5990
NTA-4F	36.5508
NA-4M	58.5542
RA-5C	578.0963
A-6A	316.5105
EA-6A	331.5851
A-6B	422.3289
EA-6B	414.3836
A-6C	263.0222
KA-6D	156.0498
A-6E	202.2598
NA-6A	20.0369
NEA-6B	6.8981
A-7A	145.0487
A-7B	152.1806
A-7C	93.0960
A-7E	154.2860
NA-7A	42.5617
NA-7C	23.9800
NA-7E	24.7737
C-1A	100.6243
C-2A	173.0653
TC-4C	118.4421
C-9B	1.6856

AD-A073 037 RAND CORP SANTA MONICA CA F/G 5/1
NAVMAN: A MODEL FOR ESTIMATING MAINTENANCF PERSONNEL REQUIREMEN--ETC(U)
JUN 79 B ARMSTRONG, J SCHANK, G BLAIS MDA903-77-C-0107
UNCLASSIFIED RAND/R-2402/2-PA/E NL

2 OF 2
AD
A073037

END
DATE
FILMED
9-79
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

Table F.4 (cont.)

Type Aircraft	b value
C-117D	118.3298
TC-117D	114.7170
C-118B	236.8508
VC-118B	349.4821
EC-121K	204.8831
NC-121K	241.1442
DC-130A	154.3167
C-130F	328.4957
KC-130F	293.2852
LC-130F	225.1896
EC-130G	545.2754
LC-130H	60.5400
EC-130Q	509.1160
KC-130R	123.3165
LC-130R	152.7133
C-131F	74.2234
C-131G	128.4875
E-1B	117.5527
TE-2A	103.3385
E-2B	289.8170
E-2C	219.3658
TE-2C	43.7143
F-4B	213.0874
EF-4B	98.8930
QF-4B	28.3656
RF-4B	366.5163
F-4J	204.8710
NF-4J	59.5417
YF-4J	38.0893
F-4N	183.5623
F-5E	36.0150
RF-8G	216.4609
F-8H	76.5467
F-8J	213.4764
DF-8L	36.4398
F-14A	242.9594
QF-86H	11.0618
UH-1E	58.4336
AH-1G	29.6287
UH-1H	43.4216
AH-1J	64.3724
HH-1K	22.5604
TH-1L	76.3485
UH-1L	40.4633
UH-1N	28.7497
VH-1N	2.0037

Table F.4 (cont.)

Type Aircraft	b value
NUH-1E	1.3617
UH-2A	28.8000
HH-2D	58.6072
SH-2D	97.6190
SH-2F	95.2420
NHH-2D	16.4356
HH-3A	89.7911
SH-3A	110.6316
SH-3D	130.0747
SH-3G	127.4133
SH-3H	105.5128
UH-3A	95.9279
VH-3A	19.5989
VH-3D	4.6698
YSH-3J	11.2286
CH-46A	104.5250
HH-46A	62.1392
CH-46D	79.3345
UH-46D	66.0274
CH-46E	48.5683
CH-46F	78.8277
NCH-46A	29.1108
CH-53A	89.4456
CH-53D	96.2272
RH-53D	68.4092
CH-53E	22.4389
DP-2E	206.6904
EP-2H	305.5764
SP-2H	128.8860
NP-2H	144.1308
P-3A	236.5391
EP-3A	120.5829
RP-3A	228.1090
P-3B	383.2156
EP-3B	477.2564
P-3C	256.2503
RP-3D	167.3317
EP-3E	416.3184
TS-2A	95.0656
US-2A	50.5629
US-2B	64.7171
US-2C	82.0309
S-2D	180.1728
ES-2D	76.9111
US-2D	41.1777
S-2E	52.2[illegible]

Table F.4 (cont.)

Type Aircraft	b value
S-2G	98.1207
S-3A	223.1657
US-3A	11.1111
T-2B	82.9258
DT-2B	61.0154
T-2C	76.7140
T-28B	39.7869
T-28C	30.8450
T-29B	22.5087
T-29C	49.3815
QT-33A	10.5208
T-34B	7.0393
T-38A	36.2355
QT-38A	2.2995
T-39D	172.1076
CT-39E	57.9767
CT-39G	36.7162
NU-1B	7.8000
U-3A	7.3472
U-11A	13.1447
HU-16D	111.0156
AV-8A	102.8507
TAV-8A	66.1179
OV-10A	109.3294
X-26A	.3792

SOURCE: ACM-02.

Appendix G

PROGRAM LISTING

This appendix contains the computer listing of the PL/1 NAVMAN program.

```
/******************************************************************
                              NAVMAN

          NAVMAN ESTIMATES MANPOWER REQUIREMENTS FOR THE ORGANIZATIONAL
          AND INTERMEDIATE MAINTENANCE OF NAVY AIRCRAFT.THE MODEL
          UTILIZES NAVY METHODOLOGY FROM THE SQUADRON MANNING DOCUMENTS
          PROCESS AND FROM ACM-02.USER INPUTS INCLUDE FLYING PROGRAM
          PARAMETERS,TYPE AND NUMBER OF AIRCRAFT,AND RELIABILITY AND
          MAINTAINABILITY DATA.WORK CENTER,SQUADRON,AND FLEET MANPOWER
          REQUIREMENTS FOR SEA AND SHORE ARE GENERATED.SENSITIVITY
          ANALYSIS AND THE CAPABILITY TO OVERRIDE CERTAIN STORED FACTORS
          ARE MODEL OPTIONS
  ******************************************************************/

DCL   WORK_CENTER_CODES (23)          CHAR(3)  INIT('010',
                                         '020','030','040',
                                         '050','060','100',
                                         '110','120','121',
                                         '130','131','140',
                                         '200','210','211',
                                         '220','230','240',
                                         '300','310','320','999');
DCL WORK_CENTER_NAMES (23)            CHAR(30)  INIT(
                                   'MAINTENANCE OFFICER
```

```
                                  'MAINTENANCE/MATERIAL CONTROL ',
                                  'MAINTENANCE ADMINISTRATION   ',
                                  'QUALITY ASSURANCE            ',
                                  'MATERIAL CONTROL             ',
                                  'DATA ANALYSIS                ',
                                  'AIRCRAFT DIVISION            ',
                                  'POWER PLANTS BRANCH          ',
                                  'AIRFRAMES BRANCH             ',
                                  'CORROSION CONTROL            ',
                                  'AVIATOR EQUIPMENT            ',
                                  'SAFETY EQUIPMENT             ',
                                  'PLANNED MAINTENANCE          ',
                                  'AVIONICS/ARMAMENT DIVISION   ',
                                  'ELECTRICAL BRANCH            ',
                                  'ELECTRONIC FIRE CONTROL      ',
                                  'ELECTRICAL/INSTRUMENTS       ',
                                  'WEAPONS BRANCH               ',
                                  'PHOTO SHOP                   ',
                                  'LINE DIVISION                ',
                                  'PLANE CAPTAINS               ',
                                  'TROUBLE SHOOTERS             ',
                                  'AGGREGATE                    ');
        DCL  PROD_DELAY_FACTOR_SEA  (23)      FLOAT(6)  INIT((7) 0,
                                                   .30,.20,.10,.05,.10,
                                                   0,0,.35,.35,.35,.30,
                                                   .35,0,.20,0);
        DCL PROD_DELAY_FACTOR_SHORE    (23)   FLOAT(6)  INIT((7) 0,
                                                   .10,.15,.10,.05,.10,
                                                   0,0,.30,.30,.20,.10,
                                                   .30,0,.10,0);
        DCL FACILITIES_MAINTENANCE_FACTORS (23) FLOAT(6)  INIT(0,.063,0,0,
```

```
                                                       .0653,0,0,.0956,.0998,
                                                       .0621,.0590,.0696,.0923,
                                                       0,.0769,.106,.0578,.0891,
                                                       .0408,0,.3182,0);
        DCL UTILITY_TASK_HOURS1  (23)        FLOAT(6)  INIT(0,10.4,0,0,
                                                       10.4,0,0,41.5,41.5,0,10.4
                                                       10.4,0,0,62.2,62.2,41.5,
                                                       41.5,0,0,103.7,0);
        DCL UTILITY_TASK_HOURS2  (23)        FLOAT(6)  INIT(0,10.4,0,0,
                                                       10.4,0,0,20.7,20.7,0,10.4
                                                       10.4,0,0,20.7,0,20.7,
                                                       20.7,0,0,62.2,0);
        DCL UTILITY_TASK_HOURS3  (23)        FLOAT(6)  INIT(0,10.4,0,0,
                                                       10.4,0,0,0,0,0,10.4,10.4,
                                                       (10) 0);
        DCL AVAILABILITY_SHORE               FLOAT(6) INIT(31.9);
        DCL AVAILABILITY_VP                  FLOAT(6) INIT(51.0);
        DCL AVAILABILITY_SEA                 FLOAT(6) INIT(63.0);
        DCL ADMIN_SUPPORT_SPREADS (22,10)    FLOAT(6)  INIT((70) 0.0,
                                                     .088,.10,.125,.095,.13,
                                                      .127,.125,.11,.095,.095,
                                                      .09,.105,.145,.058,.193,
                                                      .163,.145,.091,.058,.108,
                                                      .066,.054,.088,.088,.088,
                                                      .053,.088,.042,.126,.126,
                                                      .065,.051,.066,.085,.04,
                                                      .044,.066,.046,.128,.128,
                                                      .079,.084,.084,.091,.087,
                                                      .091,.084,.063,0.0,0.0,
                                                      .067,.045,.097,.024,0.0,
                                                      .059,.097,0.0,.024,.024,
```

```
                                        (10) 0.0,
                                        .086,.082,.113,.174,.138,
                                        .141,.187,.076,.174,.144,
                                        .108,.118,0.0,0.0,0.0,
                                        0.0,0.0,0.0,0.0,0.0,
                                        .106,.108,.095,.106,.142,
                                        .113,.115,.087,.106,.136,
                                        .123,.078,.094,.092,0.0,
                                        .07,0.0,.074,.092,.132,
                                        (7) 0.0,.25,0.0,0.0,
                                        (10) 0.0,
                                        .122,.174,.093,.187,.182,
                                        .139,.093,.161,.197,.107,
                                       (10) 0.0);
      DCL PAYGRADE_MATRIX020 (9,20)     FLOAT(6)  INIT((40) 0,
                                        0,1,1,2,2,2,3,3,4,4,4,
                                        5,5,6,6,7,7,8,9,10,
                                        0,0,1,1,1,2,2,2,2,3,3,3,
                                        4,4,4,4,5,5,5,5,
                                        (10) 1,2,2,2,2,
                                        (6) 3,
                                        (4) 0,(16) 1,
                                        (7) 0,(13) 1,
                                        (40) 0);
      DCL PAYGRADE_MATRIX050 (9,20)     FLOAT(6)  INIT((40) 0,
                                        0,1,1,1,2,2,2,2,3,3,
                                        4,4,4,5,6,(5) 0,
                                        0,0,0,1,1,2,2,2,2,3,3,3,
                                        4,4,4,(5) 0,
                                        (6) 1,(9) 2,(5) 0,
                                        0,0,(9) 1,(4) 2,(5) 0,
```

```
                                                   (7) 0,(8) 1,(5) 0,
                                                   (40) 0);
     DCL PRODUCTION_MATRIX (9,80)          FLOAT(6)  INIT((160) 0,
                                                   0,1,1,2,2,2,3,3,3,4,4,4,4,
                                                   5,5,6,7,7,7,8,9,10,10,10,
                                                   11,11,12,12,(4) 13,14,14,
                                                   (3) 15,(3) 16,17,(3) 18,
                                                   (4) 19,(3) 20,21,21,
                                                   (4) 22,23,24,24,24,24,
                                                   25,26,26,26,27,28,28,28,
                                                   29,30,30,30,31,(5) 0,
                                                   0,0,1,1,1,2,2,3,3,3,3,
                                                   (7) 4,5,5,5,5,(5) 6,
                                                   (4) 7,8,8,(5) 9,10,10,
                                                   10,10,11,11,11,(4) 12,
                                                   (3) 13,(3) 14,(6) 15,
                                                   (4) 16,(4) 17,(4) 18,
                                                   19,19,(5) 0,
                                                   (8) 1,2,2,2,2,(5) 3,4,4,4,
                                                   4,4,4,(7) 5,(5) 6,(4) 7,8,
                                                   8,8,8,9,9,9,(4) 10,(4) 11,
                                                   (6) 12,(4) 13,(4) 14,
                                                   (4) 15,(3) 16,(5) 0,
                                                   (4) 0,(10) 1,(11) 2,(11) 3,
                                                   (11) 4,(9) 5,(19) 6,(5) 0,
                                                   (10) 0,(19) 1,(10) 2,1,
                                                   (19) 1,2,(15) 2,(5) 0,
                                                   (39) 0,(36) 1, (5) 0,
                                                   (80) 0);
     DCL LINE_DIVISION_MATRIX (9,120)      FLOAT(6)  INIT(
                                                   (240) 0,0,1,2,3,4,5,5,
```

```
                                                        6,7,8,8,9,9,10,11,12,13,
                                                        14,(3) 15,16,17,18,18,
                                                        19,20,20,21,21,22,23,24,
                                                        25,26,26,27,28,29,29,30,
                                                        31,32,32,33,34,35,36,37,
                                                        38,39,39,40,41,42,43,44,
                                                        45,46,46,47,48,49,50,51,
                                                        52,53,53,54,55,56,57,58,
                                                        59,60,60,61,62,63,64,65,
                                                        66,67,67,68,69,70,71,72,
                                                        73,74,74,75,76,77,78,79,
                                                        80,81,81,82,83,84,85,86,
                                                        87,88,88,89,90,91,92,93,
                                                        94,95,95,96,97,98,99,
                                                        (6) 0,(6) 1,(8) 2,(7) 3,
                                                        (8) 4,(8) 5,(8) 6,(8) 7,
                                                        (8) 8,(8) 9,(8) 10,
                                                        (8) 11,(8) 12,(8) 13,
                                                        (8) 14,(5) 15,
                                                        (6) 1,(13) 0,1,(4) 1,
                                                        (96) 2,
                                                        (6) 0,(23) 1,(91) 2,
                                                        (10) 0,(29) 1,(81) 2,
                                                        (240) 0);
      DCL PAYGRADE_MATRIX230 (9,40)          FLOAT(6)   INIT(
                                                        (80) 0,0,1,1,2,2,2,
                                                        3,3,3,4,4,4,4,5,5,
                                                        6,6,6,7,7,7,7,8,
                                                        9,9,9,10,10,10,
                                                        11,11,11,12,12,13,
                                                        14,14,14,15,15,
```

```
                                                  0,0,1,1,1,2,2,2,2,
                                                  2,2,2,3,3,4,4,4,5,5,6,
                                                  (5) 6,7,7,8,9,8,9,9,9,
                                                  (4) 10,11,11,12,
                                                  (7) 1,2,2,2,2,(5) 3,
                                                  (4) 4,4,5,5,5,(4) 6,
                                                  7,7,7,(5) 8,(4) 9,
                                                  (4) 0,(4) 1,(12) 2,
                                                  (20) 3,
                                                  (10) 0, (10) 1,(20) 1,
                                                  (80) 0);
        DCL REQUISITION_FACTORS  (10)    FLOAT(6)  INIT(
                                                  1.2723,1.9962,2.3956,
                                                  1.5376,3.2059,1.9962,
                                                  2.3956,3.1333,1.5376,
                                                  1.5376);
        DCL ROUNDOFF_TABLE_SEA  (10)     FLOAT(6)  INIT(1.050,
                                                  2.100,3.150,4.20,5.25,
                                                  6.3,7.35,8.4,9.45,10.5);
        DCL ROUNDOFF_TABLE_SHORE  (10)   FLOAT(6)  INIT(1.078,
                                                  2.156,3.234,4.312,5.391,
                                                  6.469,7.5,8.5,9.5,10.5);
        DCL 1 TAC_INPUT,
              2 TYPE_OF_AIRCRAFT                CHAR (15),
              2 TAC_FILLER                      CHAR(65);
        DCL AIRCRAFT_CODES (10,2)        CHAR(15)  INIT(
                                                'VA','ATTACK',
                                                'VF','FIGHTER',
                                                'VP','PATROL',
                                                'VS','ANTI SUBMARINE',
                                                'VAW','EARLY WARNING',
```

```
                                                      'VAQ','ECM',
                                                      'VQ','INTELLIGENCE',
                                                      'RVAH','PHOTO',
                                                      'HM','MINE SWEEPING',
                                                      'HS','ANTI SUB HELIO');
/* V = FIXED WING,      H = HELICOPTER      */
DCL MINIMUM_MEN   (10)                          FLOAT(6) INIT(2.2,
                                          1.583,.556,1.0,0,1.0,0,0,.50,.75);
      DCL CM_PRCT_VAF  (23)                     FLOAT(6) INIT(
                                                (7) 0,.66,.61,1.0,.84,.60,0,0,
                                                .82,.86,.83,.64,0,0,.55,0,0);
      DCL CM_PRCT_OTHER (23)                    FLOAT(6) INIT(
                                                (7) 0,.81,.72,1.0,.95,.77,0,0,
                                                .93,0,.90,.92,.90,0,.67,0,0);
DCL HOURS_SEA                                         FLOAT(6);
DCL HOURS_SHORE                                       FLOAT(6);
DCL VAR_L                                             FLOAT(6);
DCL VAR_X                                             FLOAT(6);
DCL MAKE_READY_PUTAWAY_FACTOR                         FLOAT(6)  INIT(.30);
DCL PRODUCTIVITY_ALLOWANCE_FACTOR                     FLOAT(6)  INIT(.20);
DCL AIRCRAFT_PER_SQUADRON                            FLOAT(6);
DCL NUMBER_OF_SQUADRONS                              FLOAT(6);
DCL SORTIE_RATE_SEA                                   FLOAT(6);
DCL SORTIE_RATE_SHORE                                 FLOAT(6);
DCL SORTIE_LENGTH_SEA                                 FLOAT(6);
DCL SORTIE_LENGTH_SHORE                               FLOAT(6);
DCL SORTIES_WEEK_SEA                                  FLOAT(6);
DCL SORTIES_WEEK_SHORE                                FLOAT(6);
DCL FLYING_DAYS_WEEK_SEA                              FLOAT(6);
DCL FLYING_DAYS_WEEK_SHORE                            FLOAT(6);
DCL NUMBER_DEFAULT_INPUTS                             FLOAT(6);
```

```
DCL 1 DC_INPUT,
          2 DEFAULT_CODE                      CHAR(1),
          2 DC_FILLER                         CHAR(79);
DCL CM_PERCENT                                FLOAT(6);
DCL PM_PERCENT                                FLOAT(6);
DCL AS_COEFF1                                 FLOAT(6) INIT(306.9048);
DCL AS_COEFF2                                 FLOAT(6) INIT(.38519);
DCL AS2_COEFF1                                FLOAT(6) INIT(124.6715);
DCL AS2_COEFF2                                FLOAT(6) INIT(.3652);
DCL AS5_COEFF1                                FLOAT(6) INIT(57.7481);
DCL AS5_COEFF2                                FLOAT(6) INIT(.3625);
DCL M_SEA    (23)                             FLOAT(6) INIT((23) 0);
DCL M_SHORE      (23)                         FLOAT(6) INIT((23) 0);
DCL MINUS_HOURS_SEA  (23)                     FLOAT(6) INIT((23) 0);
DCL MINUS_HOURS_SHORE (23)                    FLOAT(6) INIT((23) 0);
DCL PLUS_HOURS_SEA  (23)                      FLOAT(6) INIT((23) 0);
DCL PLUS_HOURS_SHORE (23)                     FLOAT(6) INIT((23) 0);
DCL GRADE_LEVEL_SEA (23,10)                   FLOAT(6) INIT((230) 0);
DCL GRADE_LEVEL_SHORE (23,10)                 FLOAT(6)  INIT((230) 0);
DCL TOTAL_AIRCRAFT                            FLOAT(6);
DCL FLYING_HOURS_WEEK_SEA                     FLOAT(6);
DCL FLYING_HOURS_WEEK_SHORE                   FLOAT(6);
DCL TOTAL_PERSONNEL_SEA                       FLOAT(6) INIT(0);
DCL TOTAL_PERSONNEL_SHORE                     FLOAT(6) INIT(0);
DCL WC_CODE                                   CHAR(3);
DCL 1  RM_DATA,
          2 AA_TYPE                                CHAR(2),
          2 FILLER1                                CHAR(1),
          2 I_TYPE                                 CHAR(1),
          2 FILLER2                                CHAR(1),
          2 J_TYPE                                 CHAR(1),
```

```
                2 FILLER3                     CHAR(1),
                2 K_TYPE                      CHAR(1),
                2 FILLER4                     CHAR(1),
                2 XXX_CODE                    CHAR(3),
                2 FILLER5                     CHAR(3),
                2 V1X                         CHAR(4),
                2 FILLER6                     CHAR(2),
                2 V2X                         CHAR(4),
                2 FILLER7                     CHAR(2),
                2 V3X                         CHAR(4),
                2 FILLER8                     CHAR(2),
                2 V4X                         CHAR(4),
                2 RND_FILLER                  CHAR(43);
          DCL INDX                        FIXED BIN(31);
          DCL V1                          FLOAT(6);
          DCL V2                          FLOAT(6);
          DCL V3                          FLOAT(6);
          DCL V4                          FLOAT(6);
          DCL AIRCRAFT_INDX               FIXED BIN(31);
          DCL RAW_PM_WORKLOAD_SEA      (23)   FLOAT(6) INIT((23) 0);
          DCL RAW_PM_WORKLOAD_SHORE    (23)   FLOAT(6) INIT((23) 0);
          DCL RAW_TM_WORKLOAD_SEA      (23)   FLOAT(6) INIT((23) 0);
          DCL RAW_TM_WORKLOAD_SHORE    (23)   FLOAT(6) INIT((23) 0);
          DCL RAW_CM_WORKLOAD_SEA      (23)   FLOAT(6) INIT((23) 0);
          DCL RAW_CM_WORKLOAD_SHORE    (23)   FLOAT(6) INIT((23) 0);
          DCL TOTAL_RAW_PM_PLUS_CM_SEA        FLOAT(6)  INIT(0);
          DCL TOTAL_RAW_PM_PLUS_CM_SHORE      FLOAT(6) INIT(0);
          DCL RAWER_CM_WORKLOAD_SEA    (23)   FLOAT(6) INIT((23) 0);
          DCL RAWER_CM_WORKLOAD_SHORE  (23)   FLOAT(6) INIT((23) 0);
          DCL TOTAL_PM_WORKLOAD_SEA    (23)   FLOAT(6) INIT((23) 0);
          DCL TOTAL_PM_WORKLOAD_SHORE  (23)   FLOAT(6) INIT((23) 0);
```

```
DCL TOTAL_CM_WORKLOAD_SEA    (23)          FLOAT(6) INIT((23) 0);
DCL TOTAL_CM_WORKLOAD_SHORE  (23)          FLOAT(6) INIT((23) 0);
DCL TOTAL_TM_WORKLOAD_SEA    (23)          FLOAT(6) INIT((23) 0);
DCL TOTAL_TM_WORKLOAD_SHORE  (23)          FLOAT(6) INIT((23) 0);
    DCL TOTAL_FLEET_SEA                    FLOAT(6);
DCL TOTAL_FLEET_SHORE                      FLOAT(6);
DCL ISNR_SEA                               FLOAT(6);
DCL ISNR_SHORE                             FLOAT(6);
DCL AS_HOURS_SEA  (23)                     FLOAT(6) INIT((23) 0);
DCL AS_HOURS_SHORE  (23)                   FLOAT(6) INIT((23) 0);
DCL OTHER_HOURS_SEA (23)                   FLOAT(6) INIT((23) 0);
DCL OTHER_HOURS_SHORE (23)                 FLOAT(6) INIT((23) 0);
DCL WC_OTHER_HOURS                         FLOAT(6) INIT(0);
DCL TOTAL_AS_HOURS_SEA                     FLOAT(6) INIT(0);
DCL TOTAL_AS_HOURS_SHORE                   FLOAT(6) INIT(0);
DCL WORKCENTER_TM_SPREAD_VFA (23)          FLOAT(6) INIT(
                                       (7) 0,.096,.188,.042,.009,
                                       .058,(2) 0,.064,.068,.099,
                                       .128,(2) 0,.248,(2) 0);
DCL WORKCENTER_PM_SPREAD_VFA (23)          FLOAT(6) INIT(
                                       (7) 0,.096,.235,.0,.003,
                                       .065,(2) 0,.034,.030,.044,
                                       .162,(2) 0,.331,(2) 0);
DCL WORKCENTER_CM_SPREAD_VFA (23)          FLOAT(6) INIT(
                                       (7) 0,.095,.173,.063,.011,
                                       .053,(2) 0,.079,.090,.124,
                                       .110,(2) 0,.202,(2) 0);
DCL WORKCENTER_TM_SPREAD_OTHER (23)        FLOAT(6) INIT(
                                       (7) 0,.117,.166,.069,.022,
                                       .056,(2) 0,.241,.0,.126,
                                       .036,(2) 0,.167,(2) 0);
```

```
DCL WORKCENTER_PM_SPREAD_OTHER (23)     FLOAT(6) INIT(
                                   (7) 0,.139,.296,.0,.005,
                                   .084,(2) 0,.108,.0,.058,
                                   .008,(2) 0,.302,(2) 0);
    DCL WORKCENTER_CM_SPREAD_OTHER (23)     FLOAT(6) INIT(
                                   (7) 0,.112,.143,.083,.025,
                                   .051,(2) 0,.271,.0,.137,
                                   .041,(2) 0,.137,(2) 0);
    DCL STORE_TITLE                      CHAR(80);
    DCL STORE_PM_MMH_WEEK    (23)        FLOAT(6) INIT((23) 0);
    DCL STORE_PM_MMH_DAY     (23)        FLOAT(6) INIT((23) 0);
    DCL STORE_PM_MMH_PH      (23)        FLOAT(6) INIT((23) 0);
    DCL STORE_PM_MMH_S       (23)        FLOAT(6) INIT((23) 0);
    DCL STORE_CM_MMH_PH      (23)        FLOAT(6) INIT((23) 0);
    DCL STORE_CM_MMH_S       (23)        FLOAT(6) INIT((23) 0);
    DCL STORE_CM_MTBF        (23)        FLOAT(6) INIT((23) 0);
    DCL STORE_CM_MTTR        (23)        FLOAT(6) INIT((23) 0);
    DCL NUMBER_OF_SHIFTS                 FLOAT(6);
   /* AIMD CALCULATION VARIABLES */
   DCL I_LEVEL_MANHOURS_WEEK          FLOAT(6);
   DCL I_LEVEL_SPREAD       (5,10)      FLOAT(6) INIT(
                                      .20,.25,.28,.25,.20,.15,.35,
                                      .20,.33,.33,
                                      .12,.15,.16,.11,.10,.11,.18,
                                      .23,.15,.15,
                                      .60,.50,.52,.57,.65,.69,.50,
                                      .55,.50,.50,
                                      .05,.05,.01,.01,0,0,0,0,.01,
                                      .01,
                                      .03,.05,.03,.06,.05,.05,.02,
                                      .02,.01,.01);
```

```
DCL SUPPORT_EQUIPMENT_HOURS_SEA  (5) FLOAT(6) INIT(
                                     .52,.42,1.93,.17,.39);
DCL SUPPORT_EQUIPMENT_HOURS_SHORE (5) FLOAT(6) INIT(
                                     .15,.19,.95,.09,.29);
DCL AIMD_FLAG                          CHAR(1)  INIT('0');
DCL TOTAL_I_LEVEL_MANHOURS               FLOAT(6);
DCL TEMPMEN                              FLOAT(6);
DCL I_LEVEL_MANPOWER_SEA    (5)          FLOAT(6);
DCL I_LEVEL_MANPOWER_SHORE (5)           FLOAT(6);
DCL I_LEVEL_AVAILABILITY_SHORE           FLOAT(6) INIT(31.9);
DCL I_LEVEL_AVAILABILITY_SEA             FLOAT(6) INIT(60.0);
DCL I_LEVEL_AS_COEFF     (5)             FLOAT(6) INIT(
                                         2.350,4.5139,4.7271,5.2731,
                                         6.1751);
DCL I_LEVEL_ROUNDOFF     (7)             FLOAT(6) INIT(
                                         1.076,2.151,3.227,4.302,
                                         5.378,6.453,7.5);
DCL TOTAL_FLEET_I_LEVEL_SEA   (5)        FLOAT(6);
DCL TOTAL_FLEET_I_LEVEL_SHORE (5)        FLOAT(6);
DCL NUMBER_SQ_ON_SEA                     FLOAT(6);
DCL NUMBER_AC_ON_SEA                     FLOAT(6);
DCL NUMBER_OF_WAS_DEPLOYED             FLOAT(6);
DCL NUMBER_OF_AVIONICS_SKILLS_REQ      FLOAT(6);
DCL SHORE_AC_BEFORE  (5)                 FLOAT(6);
DCL SHORE_SQ_ADDED   (5)                 FLOAT(6);
DCL GSE_HOURS_PER_AC_SEA                 FLOAT(6) INIT(1.95);
DCL GSE_HOURS_PER_AC_SHORE               FLOAT(6) INIT(1.02);
DCL BEFORE_SEA_X                         FLOAT(6);
DCL AFTER_SEA_X                          FLOAT(6);
DCL BEFORE_SHORE_X  (5)                FLOAT(6);
DCL AFTER_SHORE_X  (5)                 FLOAT(6);
```

```
DCL AIMD_CADRE_ADDED_SEA                    FLOAT(6);
DCL AIMD_CADRE_ADDED_SHORE (5)           FLOAT(6);
DCL AIMD_TOTAL_CADRE_ADDED                  FLOAT(6);
DCL SHORE_HOURS_XB    (7)                   FLOAT(6);
DCL SEA_HOURS_XB                             FLOAT(6);
DCL SHORE_HOURS_XA    (7)                   FLOAT(6);
DCL SEA_HOURS_XA                             FLOAT(6);
DCL SHORE_MEN_XB    (7)                      FLOAT(6);
DCL SEA_MEN_XB      (7)                     FLOAT(6);
DCL SHORE_MEN_XA    (7)                     FLOAT(6);
DCL TOT_SHORE_XB   (7)                      FLOAT(6) INIT((7) 0);
DCL TOT_SHORE_XA   (7)                      FLOAT(6) INIT((7) 0);
DCL SEA_MEN_XA    (7)                       FLOAT(6);
DCL TOTAL_I_LEVEL_SEA                       FLOAT(6) INIT(0);
DCL TOTAL_I_LEVEL_SHORE                     FLOAT(6) INIT(0);
DCL WC230_FLAG_SEA                          CHAR(1) INIT('0');
DCL WC230_FLAG_SHORE                        CHAR(1) INIT('0');
DCL 1 SENSITIVITY_INPUT,
      2 SENSITIVITY_CODE                     CHAR(1),
      2 SV1                                    CHAR(4),
      2 SV2                                    CHAR(4),
      2 SI_FILLER                            CHAR(71);
  DCL SENSITIVITY_VALUE1                     FLOAT(6);
  DCL SENSITIVITY_VALUE2                     FLOAT(6);
DCL FACTOR1                                 FLOAT(6);
DCL FACTOR2                                 FLOAT(6);
DCL FACTOR3                                 FLOAT(6);
DCL SENSITIVITY_FLAG                        FLOAT(6) INIT(0);
DCL RM_MARKER                              CHAR(1) INIT('0');
DCL WUC_PTR                                FLOAT(6) INIT(0);
 DCL WUC_XXX                        (25)   CHAR(3);
```

```
     DCL WUC_J_TYPE              (25)    CHAR(1);
     DCL WUC_V1                  (25)    CHAR(4);
     DCL WUC_V2                  (25)    CHAR(4);
    DCL 1 INPUT_ARRAY,
          2 IA_DATA  (12)             CHAR(4),
          2 IA1_FILLER               CHAR(32);
    DCL 1 INPUT_ARRAY2,
          2 IA_DATA21                CHAR(3),
          2 IA_DATA22                CHAR(4),
          2 IA2_FILLER1              CHAR(2),
          2 IA_DATA23               CHAR(4),
          2 IA2_FILLER3             CHAR(2),
          2 IA_DATA24               CHAR(4),
          2 IA2_FILLER4             CHAR(61);
    DCL 1 TEMP_INPUT,
          2 TI_DATA                 CHAR(4),
          2 TI_FILLER                CHAR(76);
    /*   DECLARE THE INPUT-OUTPUT FILES   */
      DCL INFILE FILE;
      DCL OUTFILE FILE;
      DCL RHINPTS FILE;
   /*
        OPEN THE FILES
                        */
      OPEN FILE(INFILE) RECORD INPUT;
      OPEN FILE(OUTFILE) PRINT LINESIZE(132);
      OPEN FILE(RHINPTS) RECORD OUTPUT;
   /*
      READ AIRCRAFT DESCRIPTIVE DATA INCLUDING THE TYPE OF AIRCRAFT
        NUMBER OF AIRCRAFT IN THE SQUADRON, NUMBER OF SQUADRON IN THE
        FLEET AND FLYING PROGRAM VALUES FOR SEA AND SHORE
```

```
                                                          */
       INPUT:
               READ FILE(INFILE) INTO(STORE_TITLE);
               READ FILE(INFILE) INTO(TAC_INPUT);
               AIRCRAFT_INDX = 0;
               DO K = 1 TO 2;
               DO J = 1 TO 10;
                  IF TYPE_OF_AIRCRAFT = AIRCRAFT_CODES (J,K)
                    THEN AIRCRAFT_INDX = J;
               END;
               END;
               IF AIRCRAFT_INDX = 0 THEN GO TO INPUT_ERROR_EXIT3;
               IF AIRCRAFT_INDX > 4 & AIRCRAFT_INDX < 9 THEN DO;
                    SUPPORT_EQUIPMENT_HOURS_SHORE(4) = 0;
                    SUPPORT_EQUIPMENT_HOURS_SEA(4) = 0;
                    END;
               READ FILE(INFILE) INTO(TEMP_INPUT);
               AIRCRAFT_PER_SQUADRON = TI_DATA;
               READ FILE(INFILE) INTO(TEMP_INPUT);
               NUMBER_OF_SQUADRONS = TI_DATA;
               TOTAL_AIRCRAFT = AIRCRAFT_PER_SQUADRON*NUMBER_OF_SQUADRONS;
               READ FILE(INFILE) INTO(TEMP_INPUT);
               SORTIE_RATE_SEA = TI_DATA;
               READ FILE(INFILE) INTO(TEMP_INPUT);
               SORTIE_RATE_SHORE = TI_DATA;
               READ FILE(INFILE) INTO(TEMP_INPUT);
               SORTIE_LENGTH_SEA = TI_DATA;
               READ FILE(INFILE) INTO(TEMP_INPUT);
               SORTIE_LENGTH_SHORE = TI_DATA;
               READ FILE(INFILE) INTO(TEMP_INPUT);
               FLYING_DAYS_WEEK_SEA = TI_DATA;
```

```
        READ FILE(INFILE) INTO(TEMP_INPUT);
        FLYING_DAYS_WEEK_SHORE = TI_DATA;
        SORTIES_WEEK_SEA = SORTIE_RATE_SEA * FLYING_DAYS_WEEK_SEA
         *AIRCRAFT_PER_SQUADRON;
        SORTIES_WEEK_SHORE=SORTIE_RATE_SHORE*FLYING_DAYS_WEEK_SHORE
         *AIRCRAFT_PER_SQUADRON;
        FLYING_HOURS_WEEK_SEA = SORTIES_WEEK_SEA*SORTIE_LENGTH_SEA;
        FLYING_HOURS_WEEK_SHORE = SORTIES_WEEK_SHORE *
         SORTIE_LENGTH_SHORE;
        READ FILE(INFILE) INTO(TEMP_INPUT);
        NUMBER_OF_SHIFTS = TI_DATA;
        CALL PAGEONE_REPORT;;
/*
  READ ANY OVERRIDE VALUES SPECIFIED BY THE USER
                                                            */
        READ FILE(INFILE) INTO(TEMP_INPUT);
        NUMBER_DEFAULT_INPUTS = TI_DATA;
        IF NUMBER_DEFAULT_INPUTS = 0 THEN GO TO READ_NEXT_RM_INPUT;
        PUT FILE(OUTFILE) EDIT(STORE_TITLE) (PAGE,COL(10),A);
        PUT FILE(OUTFILE) EDIT('OVERRIDE INPUTS','CODE','VALUES')
            (SKIP,SKIP,COL(10),A,SKIP,SKIP,COL(15),A,COL(25),A);
        DO I = 1 TO NUMBER_DEFAULT_INPUTS;
        READ FILE(INFILE) INTO(DC_INPUT);
/* OVERRIDE THE TM TO CM/PM PERCENTS FOR THE WORK CENTERS  */
        IF DEFAULT_CODE = '1' THEN DO;
          IF AIRCRAFT_INDX > 2 THEN DO;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
        PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
            (SKIP,COL(17),A,COL(25),12 (A(4)));
            DO I = 1 TO 12;
            CM_PRCT_OTHER(I) = IA_DATA(I);
```

```
            END;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
        PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
            (SKIP,COL(17),A,COL(25),12 (A(4)));
            DO I = 13 TO 23;
            J = I - 12;
            CM_PRCT_OTHER(I) = IA_DATA(J);
            END;
          END;
          ELSE DO;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
        PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
            (SKIP,COL(17),A,COL(25),12 (A(4)));
            DO I = 1 TO 12;
            CM_PRCT_VAF(I) = IA_DATA(I);
            END;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
        PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
            (SKIP,COL(17),A,COL(25),12 (A(4)));
            DO I = 13 TO 23;
            J = I - 12;
            CM_PRCT_VAF(I) = IA_DATA(J);
            END;
            END;
        END;
  /* OVERRIDE THE SPREAD OF TOTAL TH TO WORK CENTERS                    */
        IF DEFAULT_CODE = '2' THEN DO;
          IF AIRCRAFT_INDX > 2 THEN DO;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
        PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
            (SKIP,COL(17),A,COL(25),12 (A(4)));
```

```
            DO I = 1 TO 12;
            WORKCENTER_TM_SPREAD_OTHER(I) = IA_DATA(I);
            END;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
        PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
            (SKIP,COL(17),A,COL(25),12 (A(4)));
            DO I = 13 TO 23;
            J = I - 12;
            WORKCENTER_TM_SPREAD_OTHER(I) = IA_DATA(J);
            END;
           END;
           ELSE DO;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
        PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
            (SKIP,COL(17),A,COL(25),12 (A(4)));
            DO I = 1 TO 12;
            WORKCENTER_TM_SPREAD_VPA(I) = IA_DATA(I);
            END;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
        PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
            (SKIP,COL(17),A,COL(25),12 (A(4)));
            DO I = 13 TO 23;
            J = I - 12;
            WORKCENTER_TM_SPREAD_VPA(I) = IA_DATA(J);
            END;
           END;
        END;
 /* OVERRIDE THE SPREAD OF TOTAL PM TO WORK CENTERS              */
        IF DEFAULT_CODE = '3' THEN DO;
           IF AIRCRAFT_INDX > 2 THEN DO;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
```

```
            PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
                (SKIP,COL(17),A,COL(25),12 (A(4)));
                DO I = 1 TO 12;
                WORKCENTER_PM_SPREAD_OTHER(I) = IA_DATA(I);
                END;
                READ FILE(INFILE) INTO(INPUT_ARRAY);
            PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
                (SKIP,COL(17),A,COL(25),12 (A(4)));
                DO I = 13 TO 23;
                J = I - 12;
                WORKCENTER_PM_SPREAD_OTHER(I) = IA_DATA(J);
                END;
               END;
               ELSE DO;
                READ FILE(INFILE) INTO(INPUT_ARRAY);
            PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
                (SKIP,COL(17),A,COL(25),12 (A(4)));
                DO I = 1 TO 12;
                WORKCENTER_PM_SPREAD_VFA(I) = IA_DATA(I);
                END;
                READ FILE(INFILE) INTO(INPUT_ARRAY);
            PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
                (SKIP,COL(17),A,COL(25),12 (A(4)));
                DO I = 13 TO 23;
                J = I - 12;
                WORKCENTER_PM_SPREAD_VFA(I) = IA_DATA(J);
                END;
               END;
            END;
       /* OVERRIDE THE SPREAD OF TOTAL CM TO WORK CENTERS               */
            IF DEFAULT_CODE = '4' THEN DO;
```

```
        IF AIRCRAFT_INDX > 2 THEN DO;
         READ FILE(INFILE) INTO(INPUT_ARRAY);
     PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
         (SKIP,COL(17),A,COL(25),12 (A(4)));
         DO I = 1 TO 12;
         WORKCENTER_CH_SPREAD_OTHER(I) = IA_DATA(I);
         END;
         READ FILE(INFILE) INTO(INPUT_ARRAY);
     PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
         (SKIP,COL(17),A,COL(25),12 (A(4)));
         DO I = 13 TO 23;
         J = I - 12;
         WORKCENTER_CH_SPREAD_OTHER(I) = IA_DATA(J);
         END;
        END;
        ELSE DO;
         READ FILE(INFILE) INTO(INPUT_ARRAY);
     PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
         (SKIP,COL(17),A,COL(25),12 (A(4)));
         DO I = 1 TO 12;
         WORKCENTER_CH_SPREAD_VPA(I) = IA_DATA(I);
         END;
         READ FILE(INFILE) INTO(INPUT_ARRAY);
     PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
         (SKIP,COL(17),A,COL(25),12 (A(4)));
         DO I = 13 TO 23;
         J = I - 12;
         WORKCENTER_CH_SPREAD_VPA(I) = IA_DATA(J);
         END;
        END;
     END;
```

```
          IF DEFAULT_CODE = '5' THEN DO;
     /* OVERRIDE THE SPREAD OF TOTAL I LEVEL HOURS TO DIVISIONS */
          READ FILE(INFILE) INTO(INPUT_ARRAY);
          DO I = 1 TO 5;
          I_LEVEL_SPREAD(I,AIRCRAFT_INDX) = IA_DATA(I);
          END;
          PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
             (SKIP,COL(17),A,COL(25),12 (A(4)));
          END;
     /* OVERRIDE THE I LEVEL SUPPORT EQUIPMENT HOURS              */
          IF DEFAULT_CODE = '6' THEN DO;
            READ FILE(INFILE) INTO(INPUT_ARRAY);
            DO I = 1 TO 5;
            J = 5 + 1;
            SUPPORT_EQUIPMENT_HOURS_SEA(I) = IA_DATA(I);
            SUPPORT_EQUIPMENT_HOURS_SHORE(I) = IA_DATA(J);
            END;
          PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
             (SKIP,COL(17),A,COL(25),12 (A(4)));
          END;
     /* OVERRIDE THE I LEVEL GSE HOURS PER AIRCRAFT               */
          IF DEFAULT_CODE = '7' THEN  DO;
        READ FILE(INFILE) INTO(INPUT_ARRAY);
        GSE_HOURS_PER_AC_SEA = IA_DATA(1);
        GSE_HOURS_PER_AC_SHORE = IA_DATA(2);
          PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
             (SKIP,COL(17),A,COL(25),12 (A(4)));
          END;
     /* OVERRIDE THE MINIMUM MANNING FOR WORK CENTER 230          */
          IF DEFAULT_CODE = '8' THEN  DO;
         READ FILE(INFILE) INTO(INPUT_ARRAY);
```

```
          MINIMUM_MEN(AIRCRAFT_INDX) = IA_DATA(1);
           PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
               (SKIP,COL(17),A,COL(25),12 (A(4)));
           END;
     /* READ ANY DIRECTED MANNING HOURS AND PLACE IN OTHER_ HOURS
        FOR THE APPROPRIATE WORK CENTERS                              */
            IF DEFAULT_CODE = '9' THEN DO;
           READ FILE(INFILE) INTO(INPUT_ARRAY2);
           WC_CODE = IA_DATA21;
           WC_OTHER_HOURS = IA_DATA22;
               INDX = 0;
               DO I = 1 TO 22;
               IF WC_CODE = WORK_CENTER_CODES(I) THEN
                          INDX = I;
               END;
            IF INDX = 0 THEN GO TO INPUT_ERROR_EXIT2;
            OTHER_HOURS_SEA(INDX) = WC_OTHER_HOURS;
            OTHER_HOURS_SHORE(INDX) = WC_OTHER_HOURS;
           PUT FILE(OUTFILE) EDIT(DEFAULT_CODE,IA_DATA)
               (SKIP,COL(17),A,COL(25),12 (A(4)));
            END;
      END;
     /*  READ THE RELIABILITY AND MAINTAINABILITY DATA AND CALCULATE
         THE RAW PREVENTIVE,CORRECTIVE,OR TOTAL MAINTENANCE
         WORKLOAD                                                      */
      READ_NEXT_RM_INPUT:
           IF RM_MARKER = '1' THEN GO TO READ_FROM_DISK;
           READ FILE(INFILE) INTO(RM_DATA);
           IF V1X = ' ' THEN V1X = '0';
           IF V2X = ' ' THEN V2X = '0';
           IF V3X = ' ' THEN V3X = '0';
```

```
            IF V4X = ' ' THEN V4X = '0';
            V1 = V1X;
            V2 = V2X;
            V3 = V3X;
            V4 = V4X;
            WRITE FILE(RMINPTS) FROM(RM_DATA);
            GO TO PROC_RM_CARD;
        READ_FROM_DISK:
            READ FILE(RMINPTS) INTO(RM_DATA);
            V1 = V1X;
            V2 = V2X;
            V3 = V3X;
            V4 = V4X;
        PROC_RM_CARD:
            IF XXX_CODE = '212' THEN XXX_CODE = '211';
            IF XXX_CODE = '888' THEN GO TO BEGIN_PROCESSING;
            IF (AA_TYPE ¬= 'CM' & AA_TYPE ¬= 'PM' & AA_TYPE ¬= 'TM')
                THEN GO TO INPUT_ERROR_EXIT;
            IF (XXX_CODE = '999' | K_TYPE = '1') THEN DO;
                                IF K_TYPE = '1' THEN DO;
                                    WUC_PTR = WUC_PTR + 1;
                                    WUC_XXX(WUC_PTR) = XXX_CODE;
                                    WUC_J_TYPE(WUC_PTR) = J_TYPE;
                                    WUC_V1(WUC_PTR) = V1X;
                                    WUC_V2(WUC_PTR) = V2X;
                                    END;
                                INDX = 23;
                                GO TO CALC_WORKLOAD;
                                END;
            INDX = 0;
            DO K = 1 TO 22;
```

```
            IF XXX_CODE = WORK_CENTER_CODES(K) THEN
                                INDX = K;
        END;
        IF INDX = 0 THEN GO TO INPUT_ERROR_EXIT2;
    CALC_WORKLOAD:
  /*   CALCULATE RAW CM,PM,TM WORKLOADS    */
  /*
       CORRECTIVE MAINTENANCE DATA
                                                          */
        IF AA_TYPE = 'CM'
            THEN DO;
            IF J_TYPE = '1' THEN DO;
                        RAWER_CM_WORKLOAD_SEA(INDX) =
                            V1*FLYING_HOURS_WEEK_SEA;
                        RAWER_CM_WORKLOAD_SHORE(INDX) =
                            V1*FLYING_HOURS_WEEK_SHORE;
                        STORE_CM_MMH_FH(INDX) = V1;
                        END;
            IF J_TYPE = '2' THEN DO;
                        RAWER_CM_WORKLOAD_SEA(INDX) =
                            V1*SORTIES_WEEK_SEA;
                        RAWER_CM_WORKLOAD_SHORE(INDX) =
                            V1*SORTIES_WEEK_SHORE;
                        STORE_CM_MMH_S(INDX) = V1;
                        END;
            IF J_TYPE = '3' THEN DO;
                        RAWER_CM_WORKLOAD_SEA(INDX) =
                            (FLYING_HOURS_WEEK_SEA/V1)*V2;
                        RAWER_CM_WORKLOAD_SHORE(INDX) =
                            (FLYING_HOURS_WEEK_SHORE/V1)*V2;
                        STORE_CM_MTBF(INDX) = V1;
```

```
                        STORE_CM_MTTR(INDX) = V2;
                        END;
        /*  ADD MR/PA AND PA TO CM IF NOT INCLUDED IN THE DATA      */
                IF I_TYPE = '1'
                  THEN DO;
                  RAWER_CM_WORKLOAD_SEA(INDX) =
                     RAWER_CM_WORKLOAD_SEA(INDX) *
                     (1.0 +MAKE_READY_PUTAWAY_FACTOR +
                     PRODUCTIVITY_ALLOWANCE_FACTOR);
                  RAWER_CM_WORKLOAD_SHORE(INDX)=
                     RAWER_CM_WORKLOAD_SHORE(INDX) *
                     (1.0 + MAKE_READY_PUTAWAY_FACTOR +
                     PRODUCTIVITY_ALLOWANCE_FACTOR);
                  END;
                RAW_CM_WORKLOAD_SEA(INDX) = RAW_CM_WORKLOAD_SEA(INDX) +
                  RAWER_CM_WORKLOAD_SEA(INDX);
                RAW_CM_WORKLOAD_SHORE(INDX)=RAW_CM_WORKLOAD_SHORE(INDX)+
                  RAWER_CM_WORKLOAD_SHORE(INDX);
                GO TO READ_NEXT_RM_INPUT;
                END;
        /*
           PREVENTIVE MAINTENANCE DATA
                                                                     */
           IF AA_TYPE = 'PM'
                THEN DO;
                STORE_PM_MMH_WEEK(INDX) = V1;
                STORE_PM_MMH_DAY(INDX) = V2;
                STORE_PM_MMH_FM(INDX) = V3;
                STORE_PM_MMH_S(INDX) = V4;
                RAW_PM_WORKLOAD_SEA(INDX) =
                  RAW_PM_WORKLOAD_SEA(INDX) +
```

```
                    V1 + V2*FLYING_DAYS_WEEK_SEA +
                    V3*FLYING_HOURS_WEEK_SEA + V4*SORTIES_WEEK_SEA;
          RAW_PM_WORKLOAD_SHORE(INDX) =
                    RAW_PM_WORKLOAD_SHORE(INDX) +
                    V1 + V2*FLYING_DAYS_WEEK_SHORE +
                    V3*FLYING_HOURS_WEEK_SHORE + V4*SORTIES_WEEK_SHORE;
          GO TO READ_NEXT_RM_INPUT;
          END;
     /*
          TOTAL MAINTENANCE DATA
                                                            */
          IF AA_TYPE = 'TM'
             THEN DO;
               IF J_TYPE = '1' THEN DO;
                              RAW_TM_WORKLOAD_SEA(INDX) =
                                   RAW_TM_WORKLOAD_SEA(INDX) +
                                   V1*FLYING_HOURS_WEEK_SEA;
                              RAW_TM_WORKLOAD_SHORE(INDX) =
                                   RAW_TM_WORKLOAD_SHORE(INDX) +
                                   V1*FLYING_HOURS_WEEK_SHORE;
                              STORE_CM_MMH_PM(INDX) = V1;
                              END;
               IF J_TYPE = '2' THEN DO;
                              RAW_TM_WORKLOAD_SEA(INDX) =
                                   RAW_TM_WORKLOAD_SEA(INDX) +
                                   V1*SORTIES_WEEK_SEA;
                              RAW_TM_WORKLOAD_SHORE(INDX) =
                                   RAW_TM_WORKLOAD_SHORE(INDX) +
                                   V1*SORTIES_WEEK_SHORE;
                              STORE_CM_MMH_S(INDX) = V1;
                              END;
```

```
            IF J_TYPE = '3' THEN DO;
                                   RAW_TM_WORKLOAD_SEA(INDX) =
                                       RAW_TM_WORKLOAD_SEA(INDX) +
                                       (FLYING_HOURS_WEEK_SEA/V1)*V2;
                                   RAW_TM_WORKLOAD_SHORE(INDX) =
                                       RAW_TM_WORKLOAD_SHORE(INDX) +
                                       (FLYING_HOURS_WEEK_SHORE/V1)*V2;
                                   STORE_CM_MTBF(INDX) = V1;
                                   STORE_CM_MTTR(INDX) = V2;
                                   END;
                   END;
               GO TO READ_NEXT_RM_INPUT;
       /*  SPREAD ANY AGGREGATE OR WUC DATA TO APPRPRIATE WORK CENTERS */
        BEGIN_PROCESSING:
            IF RAW_TM_WORKLOAD_SEA(23) > 0 THEN
                  CALL AD_SPREAD(RAW_TM_WORKLOAD_SEA,
                     RAW_TM_WORKLOAD_SHORE,
                     WORKCENTER_TM_SPREAD_VFA,
                     WORKCENTER_TM_SPREAD_OTHER);
    /*
       BREAK TM INTO CM AND PM FOR EACH WORK CENTER
                                                                   */
                  DO I = 1 TO 22;
                  INDX = I;
                  IF RAW_TM_WORKLOAD_SEA(I) ¬= 0 THEN
                  CALL CWTM_CALC;
                  END;
            IF RAW_CM_WORKLOAD_SEA(23) > 0 THEN
                  CALL AD_SPREAD(RAW_CM_WORKLOAD_SEA,
                     RAW_CM_WORKLOAD_SHORE,
                     WORKCENTER_CM_SPREAD_VFA,
```

```
                    WORKCENTER_CM_SPREAD_OTHER);
          IF RAW_PM_WORKLOAD_SEA(23) > 0 THEN
                    CALL AD_SPREAD(RAW_PM_WORKLOAD_SEA,
                    RAW_PM_WORKLOAD_SHORE,
                    WORKCENTER_PM_SPREAD_VFA,
                    WORKCENTER_PM_SPREAD_OTHER);
    /*  ADD INDIRECT FACTORS TO RAW DATA TO GET TOTAL PM AND CM
        WORKLOADS IN EACH WORK CENTER                             */
      ENTRYLEVEL:
                DO INDX = 1 TO 22;
            TOTAL_PM_WORKLOAD_SEA(INDX) =
                (RAW_PM_WORKLOAD_SEA(INDX) * (1.0 +
                MAKE_READY_PUTAWAY_FACTOR)) * ( 1.0 +
                PRODUCTIVITY_ALLOWANCE_FACTOR +
                PROD_DELAY_FACTOR_SEA(INDX));
            TOTAL_PM_WORKLOAD_SHORE(INDX) =
                (RAW_PM_WORKLOAD_SHORE(INDX) * (1.0 +
                MAKE_READY_PUTAWAY_FACTOR)) * (1.0 +
                PRODUCTIVITY_ALLOWANCE_FACTOR +
                PROD_DELAY_FACTOR_SHORE(INDX));
            TOTAL_CM_WORKLOAD_SEA(INDX)=RAW_CM_WORKLOAD_SEA(INDX) *
                 (1.0 + PROD_DELAY_FACTOR_SEA(INDX));
            TOTAL_CM_WORKLOAD_SHORE(INDX) =
                RAW_CM_WORKLOAD_SHORE(INDX) *
                 (1.0 + PROD_DELAY_FACTOR_SHORE(INDX));
          END;
    /*  CALCULATE THE ADMINISTRATIVE SUPPORT WORKLOAD             */
      ADMIN_SUPPORT_SUM:
          TOTAL_RAW_PM_PLUS_CM_SEA = 0.0;
          TOTAL_RAW_PM_PLUS_CM_SHORE = 0.0;
          DO INDX = 1 TO 22;
```

```
            TOTAL_RAW_PM_PLUS_CM_SEA = TOTAL_RAW_PM_PLUS_CM_SEA +
                RAW_CM_WORKLOAD_SEA(INDX) +
                 RAW_PM_WORKLOAD_SEA(INDX);
            TOTAL_RAW_PM_PLUS_CM_SHORE = TOTAL_RAW_PM_PLUS_CM_SHORE+
                RAW_CM_WORKLOAD_SHORE(INDX)+
                RAW_PM_WORKLOAD_SHORE(INDX);
        END;
    AS_CALCULATIONS:
          TOTAL_AS_HOURS_SEA = AS_COEFF1 + AS_COEFF2 *
               TOTAL_RAW_PM_PLUS_CM_SEA;
          TOTAL_AS_HOURS_SHORE = AS_COEFF1 + AS_COEFF2 *
                TOTAL_RAW_PM_PLUS_CM_SHORE;
   /* SPREAD THE TOTAL AS WORKLOAD TO WORK CENTERS                  */
         K = AIRCRAFT_INDX;
         DO I = 1 TO 22;
            AS_HOURS_SEA(I) =
              TOTAL_AS_HOURS_SEA * ADMIN_SUPPORT_SPREADS(I,K);
            AS_HOURS_SHORE(I) =
              TOTAL_AS_HOURS_SHORE * ADMIN_SUPPORT_SPREADS(I,K);
         END;
   /* CALCULATE PM WORKLOAD AND STORE IN OTHER_HOURS                */
    PM_CALCULATIONS:
        DO INDX = 1 TO 22;
   /*
       STANDARD EQUATION FOR WORK CENTER 020
                                                                       */
         IF INDX = 2 THEN DO;
            AS_HOURS_SEA(2)=AS2_COEFF1+AS2_COEFF2*
               FLYING_HOURS_WEEK_SEA;
            AS_HOURS_SHORE(2)=AS2_COEFF1 + AS2_COEFF2*
               FLYING_HOURS_WEEK_SHORE;
```

```
                  OTHER_HOURS_SEA(2) = OTHER_HOURS_SEA(2) +
                     FACILITIES_MAINTENANCE_FACTORS(2)*AS_HOURS_SEA(2);
                  OTHER_HOURS_SHORE(2)=OTHER_HOURS_SHORE(2) +
                     FACILITIES_MAINTENANCE_FACTORS(2)*AS_HOURS_SHORE(2);
                  END;
         /*
            STANDARD EQUATION FOR WORK CENTER 050
                                                                          */
               ELSE IF INDX = 5 THEN DO;
                  AS_HOURS_SEA(5) = AS5_COEFF1 + AS5_COEFF2*
                     FLYING_HOURS_WEEK_SEA*REQUISITION_FACTORS(K);
                  AS_HOURS_SHORE(5) = AS5_COEFF1 + AS5_COEFF2*
                     FLYING_HOURS_WEEK_SHORE*REQUISITION_FACTORS(K);
                  OTHER_HOURS_SEA(5) = OTHER_HOURS_SEA(5) +
                     FACILITIES_MAINTENANCE_FACTORS(5)*AS_HOURS_SEA(5);
                  OTHER_HOURS_SHORE(5)=OTHER_HOURS_SHORE(5)+
                     FACILITIES_MAINTENANCE_FACTORS(5)*AS_HOURS_SHORE(5);
                  END;
               ELSE DO;
                  I = INDX;
                  OTHER_HOURS_SEA(I) = OTHER_HOURS_SEA(I) +
                        FACILITIES_MAINTENANCE_FACTORS(I)*AS_HOURS_SEA(I);
                  OTHER_HOURS_SHORE(I) = OTHER_HOURS_SHORE(I) +
                        FACILITIES_MAINTENANCE_FACTORS(I)*AS_HOURS_SHORE(I);
               END;
             END;
         /* ADD ANY UT HOURS TO SEA WORKLOAD AND STORE IN OTHER_HOURS */
           UT_CALCULATIONS:
               IF AIRCRAFT_INDX <=2 | AIRCRAFT_INDX = 10 THEN
                  DO I = 1 TO 22;
                  OTHER_HOURS_SEA(I) = OTHER_HOURS_SEA(I) +
```

```
                UTILITY_TASK_HOURS1(I);
          END;
          IF (AIRCRAFT_INDX >=4 & AIRCRAFT_INDX <=6) |
          AIRCRAFT_INDX = 8 THEN
          DO I = 1 TO 22;
          OTHER_HOURS_SEA(I) = OTHER_HOURS_SEA(I) +
                UTILITY_TASK_HOURS2(I);
          END;
       IF AIRCRAFT_INDX ¬= 3 & AIRCRAFT_INDX ¬= 7 &
          AIRCRAFT_INDX ¬= 9 THEN GO TO WC320_CALC;
       DO I = 1 TO 22;
          OTHER_HOURS_SEA(I) = OTHER_HOURS_SEA(I) +
                UTILITY_TASK_HOURS3(I);
       END;
    /* CALCULATE TROUBLESHOOTERS WORKLOAD FOR SHORE AND SEA(NOT VA,
       VP,VS) AND STORE IN OTHER HOURS FOR APPROPRIATE WORK CENTERS */
    WC320_CALC:
    /*   WC 320 TROUBLESHOOTERS CALCULATIONS     */
       VAR_L = 1.0;
       VAR_X = 1.0;
       IF AIRCRAFT_INDX <= 3 THEN DO;
                            VAR_L = .5;
                            VAR_X = 2.0;
                            END;
    /*  AT SHORE & (NAT VA,VP,VS) AT SEA CALCULATIONS FOR WC 320 */
       HOURS_SHORE = (SORTIES_WEEK_SHORE*VAR_L)/VAR_X;
       DO K = 8,15,17;
         OTHER_HOURS_SHORE(K) = OTHER_HOURS_SHORE(K) +
               HOURS_SHORE;
       END;
       OTHER_HOURS_SHORE(9) = OTHER_HOURS_SHORE(9) +
```

```
                    2*HOURS_SHORE;
         IF AIRCRAFT_INDX > 2 & AIRCRAFT_INDX ¬=4
            THEN DO;
                    HOURS_SEA = (SORTIES_WEEK_SEA*VAR_L)/VAR_X;
                    DO K = 8,15,17;
                    OTHER_HOURS_SEA(K) = OTHER_HOURS_SEA(K) +
                            HOURS_SEA;
                    END;
                    OTHER_HOURS_SEA(9) = OTHER_HOURS_SEA(9) +
                            2*HOURS_SEA;
                    END;
   /*
        TOTAL THE WORKLOADS FOR EACH WORK CENTER
                                                                  */
         DO I = 1 TO 22;
         TOTAL_TH_WORKLOAD_SEA(I) = TOTAL_CM_WORKLOAD_SEA(I) +
                                    TOTAL_PM_WORKLOAD_SEA(I) +
                                    AS_HOURS_SEA(I) +
                                    OTHER_HOURS_SEA(I);
         TOTAL_TM_WORKLOAD_SHORE(I) = TOTAL_CM_WORKLOAD_SHORE(I) +
                                      TOTAL_PM_WORKLOAD_SHORE(I) +
                                      AS_HOURS_SHORE(I) +
                                      OTHER_HOURS_SHORE(I);
         END;
   /* TRANSLATE HOURLY WORKLOAD INTO MANPOWER REQUIREMENTS          */
   AVAILABILITY_CALC:
         IF AIRCRAFT_INDX ¬= 3
            THEN DO K = 1 TO 22;
            M_SEA(K) = TOTAL_TM_WORKLOAD_SEA(K) / AVAILABILITY_SEA;
            M_SHORE(K)=TOTAL_TM_WORKLOAD_SHORE(K)/AVAILABILITY_SHORE;
            END;
```

```
            ELSE DO K = 1 TO 22;
                M_SEA(K) = TOTAL_TM_WORKLOAD_SEA(K)/AVAILABILITY_VP;
                M_SHORE(K)=TOTAL_TM_WORKLOAD_SHORE(K)/AVAILABILITY_VP;
                END;
            IF AIRCRAFT_INDX > 3 & AIRCRAFT_INDX ¬=6 & AIRCRAFT_INDX ¬= 10
              THEN GO TO INTEGERIZE;
      /*
           SET MINIMUM MANNING FOR WEAPONS LOADERS (WC 230)
                                                                        */
            IF M_SEA(18) < (MINIMUM_MEN(AIRCRAFT_INDX)*
              AIRCRAFT_PER_SQUADRON) THEN DO;
              M_SEA(18) = MINIMUM_MEN(AIRCRAFT_INDX)*
                AIRCRAFT_PER_SQUADRON;
              WC230_FLAG_SEA = '1';
              END;
            IF M_SHORE(18) < (MINIMUM_MEN(AIRCRAFT_INDX)*
              AIRCRAFT_PER_SQUADRON) THEN DO;
              M_SHORE(18) = MINIMUM_MEN(AIRCRAFT_INDX) *
                  AIRCRAFT_PER_SQUADRON;
              WC230_FLAG_SHORE = '1';
              END;
      /* ROUNDOFF MANPOWER TO GET INTEGER MEN                           */
      /*
           AND SET PLUS AND MINUS HOURS
                                                                        */
       INTEGERIZE:
            DO K = 1 TO 22;
              IF M_SEA(K) > 10.5 THEN DO;
                M_SEA(K) = TRUNC(M_SEA(K)+.4999);
                IF AIRCRAFT_INDX ¬= 3 THEN DO;
                    MINUS_HOURS_SEA(K)=TOTAL_TM_WORKLOAD_SEA(K) -
```

```
                    (N_SEA(K) - .5)*AVAILABILITY_SEA;
                    PLUS_HOURS_SEA(K)=((N_SEA(K) + .5)*
                     AVAILABILITY_SEA) - TOTAL_TM_WORKLOAD_SEA(K);
                    END;
          ELSE DO;
                    MINUS_HOURS_SEA(K)=TOTAL_TM_WORKLOAD_SEA(K) -
                    (N_SEA(K) - .5)*AVAILABILITY_VP;
                    PLUS_HOURS_SEA(K)=((N_SEA(K) + .5)*
                     AVAILABILITY_VP) = TOTAL_TM_WORKLOAD_SEA(K);
                    END;
          END;
     ELSE DO;
                   IJ = 0;
                   II = 10;
                   DO I = 1 TO 9;
                   II = II - 1;
                   IF IJ = 0 THEN
                      IF N_SEA(K) > ROUNDOFF_TABLE_SEA(II)
                         THEN DO;
                         IF AIRCRAFT_INDX ¬= 3 THEN DO;
                            MINUS_HOURS_SEA(K) = (N_SEA(K) -
                             ROUNDOFF_TABLE_SEA(II))*AVAILABILITY_SEA;
                            PLUS_HOURS_SEA(K)=(ROUNDOFF_TABLE_SEA(II+1)
                             - N_SEA(K)) * AVAILABILITY_SEA;
                            END;
                         ELSE DO;
                            MINUS_HOURS_SEA(K) = (N_SEA(K) -
                             ROUNDOFF_TABLE_SEA(II))*AVAILABILITY_VP;
                            PLUS_HOURS_SEA(K)=(ROUNDOFF_TABLE_SEA(II+1)
                             - N_SEA(K)) * AVAILABILITY_VP;
                            END;
```

```
                    N_SEA(K) = II + 1;
                    IJ = 1;
                    END;
               END;
               IF IJ = 0 & TOTAL_TM_WORKLOAD_SEA(K) ¬= 0 THEN DO;
                 N_SEA(K) = 1;
                 IF AIRCRAFT_INDX ¬= 3 THEN
                 PLUS_HOURS_SEA(K) = AVAILABILITY_SEA*
                  ROUNDOFF_TABLE_SEA(1) - TOTAL_TM_WORKLOAD_SEA(K);
                 ELSE
                 PLUS_HOURS_SEA(K) = AVAILABILITY_VP*
                  ROUNDOFF_TABLE_SEA(1) - TOTAL_TM_WORKLOAD_SEA(K);
                 END;
               END;
     /*
          ROUNDOFF SHORE MANPOWER
                                                                */
            IF N_SHORE(K) > 10.5 THEN DO;
               N_SHORE(K) = TRUNC(N_SHORE(K) + .4999);
               IF AIRCRAFT_INDX ¬= 3 THEN DO;
                MINUS_HOURS_SHORE(K)=TOTAL_TM_WORKLOAD_SHORE(K) -
                 (N_SHORE(K) - .5)*AVAILABILITY_SHORE;
                PLUS_HOURS_SHORE(K)=((N_SHORE(K) + .5)*
                 AVAILABILITY_SHORE) - TOTAL_TM_WORKLOAD_SHORE(K);
                END;
               ELSE DO;
                MINUS_HOURS_SHORE(K)=TOTAL_TM_WORKLOAD_SHORE(K) -
                 (N_SHORE(K) - .5)*AVAILABILITY_VP;
                PLUS_HOURS_SHORE(K)=((N_SHORE(K) + .5)*
                 AVAILABILITY_VP) = TOTAL_TM_WORKLOAD_SHORE(K);
                END;
```

```
          END;
       ELSE DO;
            IJ = 0;
            II = 10;
            DO I = 1 TO 9;
            II = II - 1;
            IF IJ = 0 THEN
                IF M_SHORE(K) > ROUNDOFF_TABLE_SHORE(II)
                    THEN DO;
                    IF AIRCRAFT_INDX ¬= 3 THEN DO;
                        MINUS_HOURS_SHORE(K) = (M_SHORE(K) -
                         ROUNDOFF_TABLE_SHORE(II))*
                         AVAILABILITY_SHORE;
                        PLUS_HOURS_SHORE(K)=
                         (ROUNDOFF_TABLE_SHORE(II+1)
                         - M_SHORE(K)) * AVAILABILITY_SHORE;
                        END;
                    ELSE DO;
                        MINUS_HOURS_SHORE(K) = (M_SHORE(K) -
                         ROUNDOFF_TABLE_SHORE(II))*
                         AVAILABILITY_VP;
                        PLUS_HOURS_SHORE(K)=
                         (ROUNDOFF_TABLE_SHORE(II+1)
                         - M_SHORE(K)) * AVAILABILITY_VP;
                        END;
                    M_SHORE(K) = II + 1;
                    IJ = 1;
                    END;
            END;
            IF IJ = 0 & TOTAL_TM_WORKLOAD_SHORE(K) ¬= 0 THEN DO;
          M_SHORE(K) = 1;
```

```
                              IF AIRCRAFT_INDX ¬= 3 THEN
                              PLUS_HOURS_SHORE(K) = AVAILABILITY_SHORE*
                               ROUNDOFF_TABLE_SHORE(1) -
                               TOTAL_TM_WORKLOAD_SHORE(K);
                              ELSE
                              PLUS_HOURS_SHORE(K) = AVAILABILITY_VP*
                               ROUNDOFF_TABLE_SHORE(1) -
                               TOTAL_TM_WORKLOAD_SHORE(K);
                              END;
                           END;
                  END;
                  IF WC230_FLAG_SEA = '1' THEN MINUS_HOURS_SEA(18) = 0.0;
                  IF WC230_FLAG_SHORE = '1' THEN MINUS_HOURS_SHORE(18) = 0.0;
            AVAIL_CALC_CONT:
          /* SET A MINIMUM OF 2 PLANE CAPTAINS PER AIRCRAFT FOR SEA SQUADRONS*/
                  IF M_SEA(21) < (2.0*AIRCRAFT_PER_SQUADRON) THEN DO;
                     M_SEA(21) = (2.0*AIRCRAFT_PER_SQUADRON);
                     MINUS_HOURS_SEA(21) = 0.0;
                     PLUS_HOURS_SEA(21) = (M_SEA(21)*AVAILABILITY_SEA) -
                      TOTAL_TM_WORKLOAD_SEA(21);
                     END;
           /*
                CALL AIMD IF THE BASE CASE OR A NEW NUMBER OF AIRCRAFT PER
                SQUADRON
                                                                              */
            GRADE_LEVEL_CALC:
                  IF (SENSITIVITY_FLAG = 0 |
                       SENSITIVITY_CODE = '1') THEN CALL AIMD_CALCULATIONS;
           /*  CALL ROUTINE TO CALCULATE I LEVEL REQUIREMENTS                 */
           /* DETERMINE PAYGRADES FOR FIXED POSITIONS IN WORK CENTERS
              010,030, AND 060                                                */
```

```
        GLC_FIXED:
            IF AIRCRAFT_PER_SQUADRON < 18 THEN I = 1;
            ELSE IF AIRCRAFT_PER_SQUADRON < 24 THEN I = 2;
                ELSE IF AIRCRAFT_PER_SQUADRON < 30 THEN I = 3;
                    ELSE I = 4;
            DO K = 1,3,6;
            GRADE_LEVEL_SEA(K,10) = I;
            GRADE_LEVEL_SHORE(K,10) = I;
            IF K = 3 THEN DO;
                    GRADE_LEVEL_SEA(K,5) = I;
                    GRADE_LEVEL_SHORE(K,5) = I;
                    END;
            IF K = 6 THEN DO;
                    GRADE_LEVEL_SEA(K,6) = I;
                    GRADE_LEVEL_SHORE(K,6) = I;
                    END;
        /*  WHEN CODING PRINT, REMEMBER THAT W C 010 ARE ALL LT. CMDRS.  */
            END;
            IF AIRCRAFT_PER_SQUADRON < 18 THEN I = 8;
        /* DETERMINE PERSONNEL REQUIREMENTS AND PAYGRADES FOR 040        */
            ELSE IF AIRCRAFT_PER_SQUADRON < 24 THEN I = 10;
                ELSE IF AIRCRAFT_PER_SQUADRON < 30 THEN I = 12;
                    ELSE I = 14;
            K = 4;
            GRADE_LEVEL_SEA(K,10) = I;
            GRADE_LEVEL_SHORE(K,10) = I;
            GRADE_LEVEL_SEA(K,8) = 1;
            GRADE_LEVEL_SHORE(K,8) = 1;
            GRADE_LEVEL_SEA(K,4) = 1;
            GRADE_LEVEL_SHORE(K,4) = 1;
            GRADE_LEVEL_SEA(K,6) = I - 2;
```

```
            GRADE_LEVEL_SHORE(K,6) = I - 2;
       /* DETERMINE PERSONNEL REQUIREMENTS FOR WORK CENTERS 100,200,300 */
            DO K = 7,14,20;
            GRADE_LEVEL_SEA(K,10) = 1;
            GRADE_LEVEL_SHORE(K,10) = 1;
            END;
       /*  DETERMINE PAYGRADES FOR WORK CENTER 140                      */
            GRADE_LEVEL_SEA(13,10) = GRADE_LEVEL_SEA(13,10) + M_SEA(13);
            GRADE_LEVEL_SHORE(13,10)=GRADE_LEVEL_SHORE(13,10)+M_SHORE(13);
            GRADE_LEVEL_SEA(13,6) = GRADE_LEVEL_SEA(13,6) + M_SEA(13);
            GRADE_LEVEL_SHORE(13,6)=GRADE_LEVEL_SHORE(13,6)+M_SHORE(13);
   GLC_VARIABLE:
       /*  DETERMINE PERSONNEL REQUIREMENTS FOR WORK CENTER 020 THAT
           ARE A FUNCTION OF THE NUMBER OF SHIFTS                       */
            IF NUMBER_OF_SHIFTS = 1 THEN DO;
               GRADE_LEVEL_SEA(2,10) = 2;
               GRADE_LEVEL_SEA(2,9)  = 1;
               GRADE_LEVEL_SEA(2,7)  = 1;
               GRADE_LEVEL_SHORE(2,10) = 2;
               GRADE_LEVEL_SHORE(2,9)  = 1;
               GRADE_LEVEL_SHORE(2,7)  = 1;
               END;
            IF NUMBER_OF_SHIFTS = 2 THEN DO;
               GRADE_LEVEL_SEA(2,10) = 3;
               GRADE_LEVEL_SEA(2,9) = 1;
               GRADE_LEVEL_SEA(2,8) = 1;
               GRADE_LEVEL_SEA(2,7) = 1;
               GRADE_LEVEL_SHORE(2,10) = 3;
               GRADE_LEVEL_SHORE(2,9) = 1;
               GRADE_LEVEL_SHORE(2,8) = 1;
               GRADE_LEVEL_SHORE(2,7) = 1;
```

```
            END;
         IF NUMBER_OF_SHIFTS = 3 THEN DO;
            GRADE_LEVEL_SEA(2,10) = 4;
            GRADE_LEVEL_SEA(2,9) = 1;
            GRADE_LEVEL_SEA(2,8) = 1;
            GRADE_LEVEL_SEA(2,7) = 2;
            GRADE_LEVEL_SHORE(2,10) = 4;
            GRADE_LEVEL_SHORE(2,9) = 1;
            GRADE_LEVEL_SHORE(2,8) = 1;
            GRADE_LEVEL_SHORE(2,7) = 2;
            END;
   /*  DETERMINE TOTAL PERSONNEL AND PAYGRADES FOR WORK CENTER 020  */
         GRADE_LEVEL_SEA(2,10) = GRADE_LEVEL_SEA(2,10) + M_SEA(2);
         GRADE_LEVEL_SHORE(2,10)=GRADE_LEVEL_SHORE(2,10)+M_SHORE(2);
         DO I = 1 TO 9;
            GRADE_LEVEL_SEA(2,I) = GRADE_LEVEL_SEA(2,I) +
                                 PAYGRADE_MATRIX020(I,M_SEA(2));
            GRADE_LEVEL_SHORE(2,I)=GRADE_LEVEL_SHORE(2,I) +
                                 PAYGRADE_MATRIX020(I,M_SHORE(2));
         END;
   /* DETERMINE PAYGRADES FOR WORK CENTER 050                      */
         GRADE_LEVEL_SEA(5,10) = M_SEA(5);
         GRADE_LEVEL_SHORE(5,10) = M_SHORE(5);
         DO J = 1 TO 9;
            GRADE_LEVEL_SEA(5,J) = PAYGRADE_MATRIX050(J,M_SEA(5));
            GRADE_LEVEL_SHORE(5,J)=PAYGRADE_MATRIX050(J,M_SHORE(5));
         END;
   /* SET TROUBLESHOOTER REQUIREMENTS = 5 FOR VA,VF,VS AT SEA      */
         IF AIRCRAFT_INDX < 5 & AIRCRAFT_INDX ¬=3
            THEN DO;
               GRADE_LEVEL_SEA(22,10) = 5;
```

```
                           GRADE_LEVEL_SEA(22,6) = 1;
                           GRADE_LEVEL_SEA(22,5) = 4;
                           END;
      /*  DETERMINE PAYGRADE REQUIREMENTS FOR PRODUCTION WORK CENTERS    */
                 DO K = 8,9,10,11,12,15,16,17;
                    IF TOTAL_TM_WORKLOAD_SEA(K) ¬= 0 THEN DO;
                    GRADE_LEVEL_SEA(K,10) = N_SEA(K);
                    GRADE_LEVEL_SHORE(K,10) = N_SHORE(K);
                    DO J = 1 TO 9;
                      GRADE_LEVEL_SEA(K,J) = PRODUCTION_MATRIX(J,N_SEA(K));
                      GRADE_LEVEL_SHORE(K,J)=PRODUCTION_MATRIX(J,N_SHORE(K));
      END;
      END;
                 END;
                 IF AIRCRAFT_INDX = 8
                    THEN DO;
                    GRADE_LEVEL_SEA(19,10) = N_SEA(19);
                    GRADE_LEVEL_SHORE(19,10) = N_SHORE(19);
                    DO J = 1 TO 9;
                      GRADE_LEVEL_SEA(19,J) = PRODUCTION_MATRIX(J,N_SEA(19));
                      GRADE_LEVEL_SHORE(19,J)=PRODUCTION_MATRIX(J,N_SHORE(19))
                    END;
                 END;
                 ELSE DO;
                   PLUS_HOURS_SEA(19) = 0.0;
                   PLUS_HOURS_SHORE(19) = 0.0;
                   END;
      /*  DETERMINE PAYGRADE REQUIREMENTS FOR PLANE CAPTAINS             */
                 GRADE_LEVEL_SEA(21,10) = N_SEA(21);
                 GRADE_LEVEL_SHORE(21,10) = N_SHORE(21);
                 DO J = 1 TO 9;
```

```
            GRADE_LEVEL_SEA(21,J)=LINE_DIVISION_MATRIX(J,M_SEA(21));
            GRADE_LEVEL_SHORE(21,J)=LINE_DIVISION_MATRIX(J,M_SHORE(21))
         END;
     /*  DETERMINE PAYGRADE REQUIREMENTS FOR WORL CENTER 230          */
      GLC_CONT:
         GRADE_LEVEL_SEA(18,10) = M_SEA(18);
         GRADE_LEVEL_SHORE(18,10) = M_SHORE(18);
         DO J = 1 TO 9;
            GRADE_LEVEL_SEA(18,J) = PAYGRADE_MATRIX230(J,M_SEA(18));
            GRADE_LEVEL_SHORE(18,J) =
                  PAYGRADE_MATRIX230(J,M_SHORE(18));
         END;
     /*  DETERMINE PAYGRADE REQUIREMENTS FOR DIVISION WORK CENTERS
         (WC 100,200,300) EQUAL TO ONE PAYGRADE GREATER THAN SUBORDINATE
         WORK CENTERS WITH AN E-8 AT MOST                              */
         ISNR_SEA = 0;
         ISNR_SHORE = 0;
         DO K = 8 TO 13;
            DO J = 1 TO 8;
            IF GRADE_LEVEL_SEA(K,J) > 0 & J > ISNR_SEA
               THEN ISNR_SEA = J;
            IF GRADE_LEVEL_SHORE(K,J) > 0 & J > ISNR_SHORE
               THEN ISNR_SHORE = J;
             END;
         END;
         ISNR_SEA = ISNR_SEA + 1;
         ISNR_SHORE = ISNR_SHORE + 1;
         IF ISNR_SEA > 8 THEN ISNR_SEA = 8;
         IF ISNR_SHORE > 8 THEN ISNR_SHORE = 8;
         IF ISNR_SEA ¬= 1 THEN GRADE_LEVEL_SEA(7,ISNR_SEA) =
            GRADE_LEVEL_SEA(7,ISNR_SEA) + 1;
```

```
IF ISNR_SHORE ¬= 1 THEN
   GRADE_LEVEL_SHORE(7,ISNR_SHORE) =
   GRADE_LEVEL_SHORE(7,ISNR_SHORE) + 1;
ISNR_SEA = 0;
ISNR_SHORE = 0;
DO K = 15 TO 19;
   DO J = 1 TO 8;
   IF GRADE_LEVEL_SEA(K,J) > 0 & J > ISNR_SEA
      THEN ISNR_SEA = J;
   IF GRADE_LEVEL_SHORE(K,J) > 0 & J > ISNR_SHORE
      THEN ISNR_SHORE = J;
    END;
END;
ISNR_SEA = ISNR_SEA + 1;
ISNR_SHORE = ISNR_SHORE + 1;
IF ISNR_SEA > 8 THEN ISNR_SEA = 8;
IF ISNR_SHORE > 8 THEN ISNR_SHORE = 8;
IF ISNR_SEA ¬= 1 THEN GRADE_LEVEL_SEA(14,ISNR_SEA) =
   GRADE_LEVEL_SEA(14,ISNR_SEA) + 1;
IF ISNR_SHORE ¬= 1 THEN
   GRADE_LEVEL_SHORE(14,ISNR_SHORE) =
   GRADE_LEVEL_SHORE(14,ISNR_SHORE) + 1;
ISNR_SEA = 0;
ISNR_SHORE = 0;
DO K = 21 TO 22;
   DO J = 1 TO 8;
   IF GRADE_LEVEL_SEA(K,J) > 0 & J > ISNR_SEA
      THEN ISNR_SEA = J;
   IF GRADE_LEVEL_SHORE(K,J) > 0 & J > ISNR_SHORE
      THEN ISNR_SHORE = J;
    END;
```

```
            END;
            ISNR_SEA = ISNR_SEA + 1;
            ISNR_SHORE = ISNR_SHORE + 1;
            IF ISNR_SEA > 8 THEN ISNR_SEA = 8;
            IF ISNR_SHORE > 8 THEN ISNR_SHORE = 8;
            IF ISNR_SEA ¬= 1 THEN GRADE_LEVEL_SEA(20,ISNR_SEA) =
               GRADE_LEVEL_SEA(20,ISNR_SEA) + 1;
            IF ISNR_SHORE ¬= 1 THEN
               GRADE_LEVEL_SHORE(20,ISNR_SHORE) =
               GRADE_LEVEL_SHORE(20,ISNR_SHORE) + 1;
     /* TOTAL THE PERSONNEL REQUIREMENTS IN THE ORGANIZATIONAL
        PAYGRADE MATRIX                                        */
            TOTAL_PERSONNEL_SEA = 0.0;
            TOTAL_PERSONNEL_SHORE = 0.0;
            DO I = 1 TO 22;
               TOTAL_PERSONNEL_SEA = TOTAL_PERSONNEL_SEA +
               GRADE_LEVEL_SEA(I,10);
               TOTAL_PERSONNEL_SHORE = TOTAL_PERSONNEL_SHORE +
               GRADE_LEVEL_SHORE(I,10);
               DO J = 1 TO 9;
                  GRADE_LEVEL_SEA(23,J) = GRADE_LEVEL_SEA(23,J) +
                  GRADE_LEVEL_SEA(I,J);
                  GRADE_LEVEL_SHORE(23,J) = GRADE_LEVEL_SHORE(23,J) +
                  GRADE_LEVEL_SHORE(I,J);
               END;
            END;
     /*   CALL THE OUTPUT ROUTINES                             */
            IF SENSITIVITY_FLAG = 1 THEN CALL PAGEONE_REPORT;
            CALL PAGETWO_REPORT;
            CALL PAGETWO_DETAIL_REPORT;
            CALL PAGETWO_SPREAD_REPORT;
```

```
            CALL PAGETHREE_REPORT;
            CALL PAGEFOUR_REPORT;
            CALL PAGEFIVE_REPORT;
      /*    PERFORM THE SENSITIVITY COMPUTATIONS IF REQUESTED  */
            READ FILE(INFILE) INTO(SENSITIVITY_INPUT);
             IF SV1 = ' ' THEN SV1 = '0';
             IF SV2 = ' ' THEN SV2 = '0';
             SENSITIVITY_VALUE1 = SV1;
             SENSITIVITY_VALUE2 = SV2;
            IF SENSITIVITY_CODE = 'Z' THEN GO TO ENDRUN;
            CLOSE FILE(RMINPTS);
            OPEN FILE(RMINPTS) RECORD INPUT;
            RM_MARKER = '1';
            SENSITIVITY_FLAG = 1;
      /*
         SENSITIVITY ON NUMBER OF AIRCRAFT PER SQUADRON
                                                                  */
            IF SENSITIVITY_CODE = '1' THEN DO;
                FACTOR1 = SENSITIVITY_VALUE1 / AIRCRAFT_PER_SQUADRON;
                FACTOR2 = FACTOR1;
                FACTOR3 = FACTOR1;
                CALL RESET;
                AIRCRAFT_PER_SQUADRON = SENSITIVITY_VALUE1;
                TOTAL_AIRCRAFT = AIRCRAFT_PER_SQUADRON*
                                 NUMBER_OF_SQUADRONS;
                GO TO ENTRYLEVEL;
                END;
      /*
         SENSITIVITY ON SORTIE RATES
                                                                  */
            IF SENSITIVITY_CODE = '2' THEN DO;
```

```
            FACTOR1 = 0.0;
            FACTOR2 = SENSITIVITY_VALUE1/SORTIE_RATE_SEA;
            FACTOR3 = SENSITIVITY_VALUE2/SORTIE_RATE_SHORE;
            CALL RESET;
            SORTIE_RATE_SEA = SENSITIVITY_VALUE1;
            SORTIE_RATE_SHORE = SENSITIVITY_VALUE2;
            GO TO READ_NEXT_RM_INPUT;
            END;
    /*
        SENSITIVITY ON NUMBER OF FLYING DAYS
                                                                  */
          IF SENSITIVITY_CODE = '3' THEN DO;
            FACTOR1 = 0.0;
            FACTOR2 = SENSITIVITY_VALUE1/FLYING_DAYS_WEEK_SEA;
            FACTOR3 = SENSITIVITY_VALUE2/FLYING_DAYS_WEEK_SHORE;
            CALL RESET;
            FLYING_DAYS_WEEK_SEA = SENSITIVITY_VALUE1;
            FLYING_DAYS_WEEK_SHORE = SENSITIVITY_VALUE2;
            GO TO READ_NEXT_RM_INPUT;
            END;
    /*
        SENSITIVITY ON R&M DATA
                                                                  */
          IF SENSITIVITY_CODE = '4' THEN DO;
            IF SENSITIVITY_VALUE1 = 1 | SENSITIVITY_VALUE1 = 3 THEN
            DO I = 1 TO 23;
              RAW_PM_WORKLOAD_SEA(I) = RAW_PM_WORKLOAD_SEA(I) *
                                        SENSITIVITY_VALUE2;
              RAW_PM_WORKLOAD_SHORE(I) = RAW_PM_WORKLOAD_SHORE(I) *
                                          SENSITIVITY_VALUE2;
            END;
```

```
              IF SENSITIVITY_VALUE1 = 2 | SENSITIVITY_VALUE1 = 3 THEN
              DO I = 1 TO 23;
                 RAW_CH_WORKLOAD_SEA(I) = RAW_CH_WORKLOAD_SEA(I) *
                                             SENSITIVITY_VALUE2;
                 RAW_CH_WORKLOAD_SHORE(I) = RAW_CH_WORKLOAD_SHORE(I) *
                                               SENSITIVITY_VALUE2;
              END;
              DO I = 1 TO 23;
              OTHER_HOURS_SEA(I) = 0.0;
              OTHER_HOURS_SHORE(I) = 0.0;
                DO J = 1 TO 10;
                GRADE_LEVEL_SEA(I,J) = 0.0;
                GRADE_LEVEL_SHORE(I,J) = 0.0;
                END;
              END;
            GO TO ENTRYLEVEL;
            END;
      INPUT_ERROR_EXIT:
            PUT FILE(OUTFILE) EDIT(
              ' ERROR --    AA INPUT IS NOT TM,PM OR CM ')
              (SKIP,COL(5),A);
            GO TO ENDRUN;
      INPUT_ERROR_EXIT2:
            PUT FILE(OUTFILE) EDIT(' ERROR --    XXX CODE IS NOT VALID ')
              (SKIP,COL(5),A);
            GO TO ENDRUN;
      INPUT_ERROR_EXIT3:
            PUT FILE(OUTFILE) EDIT(
              ' ERROR -- AIRCRAFT TYPE IS NOT VALID ')
              (SKIP,COL(5),A);
            GO TO ENDRUN;
```

```
         /*  ROUTINE TO ALLOCATE TM HOURS TO CM/PM HOURS            */
         CWTM_CALC:  PROC;
                     IF AIRCRAFT_INDX > 2 THEN GO TO CWTM_OTHER;
                          PM_PERCENT = 1.0 - CM_PRCT_VAF(INDX);
                          CM_PERCENT = CM_PRCT_VAF(INDX);
                          GO TO CWTM_CONT;
          CWTM_OTHER:
                     PM_PERCENT = 1.0 - CM_PRCT_OTHER(INDX);
                     CM_PERCENT = CM_PRCT_OTHER(INDX);
          CWTM_CONT:
                     RAW_PM_WORKLOAD_SEA(INDX) = PM_PERCENT *
                          RAW_TM_WORKLOAD_SEA(INDX) +
                          RAW_PM_WORKLOAD_SEA(INDX);
                     RAW_PM_WORKLOAD_SHORE(INDX) = PM_PERCENT *
                          RAW_TM_WORKLOAD_SHORE(INDX) +
                          RAW_PM_WORKLOAD_SHORE(INDX);
                     RAW_CM_WORKLOAD_SEA(INDX) = CM_PERCENT *
                          RAW_TM_WORKLOAD_SEA(INDX) *
                          (1.0 + MAKE_READY_PUTAWAY_FACTOR +
                          PRODUCTIVITY_ALLOWANCE_FACTOR) +
                          RAW_CM_WORKLOAD_SEA(INDX);
                     RAW_CM_WORKLOAD_SHORE(INDX) = CM_PERCENT *
                          RAW_TM_WORKLOAD_SHORE(INDX) *
                          (1.0 + MAKE_READY_PUTAWAY_FACTOR +
                          PRODUCTIVITY_ALLOWANCE_FACTOR) +
                          RAW_CM_WORKLOAD_SHORE(INDX);
           END;
         /*  INTERMEDIATE MAINTENANCE REQUIREMENTS ROUTINE           */
          AIMD_CALCULATIONS:  PROC;
         /*   INPUT 1    */
                 AIMD_FLAG = '1';
```

```
IF SENSITIVITY_FLAG = 0  THEN DO;
    READ FILE(INFILE) INTO(INPUT_ARRAY2);
    I_LEVEL_MANHOURS_WEEK = IA_DATA22;
    IF IA_DATA23 = '    ' THEN IA_DATA23 = '0000';
    NUMBER_OF_AVIONICS_SKILLS_REQ = IA_DATA23;
    END;
  TOTAL_I_LEVEL_MANHOURS = I_LEVEL_MANHOURS_WEEK *
                             AIRCRAFT_PER_SQUADRON;
/*  CALCULATE TEMPORARY ASSIGNED DUTY I LEVEL PERSONNEL  */
  DO I = 1 TO 5;
  I_LEVEL_MANPOWER_SEA(I) = (TOTAL_I_LEVEL_MANHOURS *
        I_LEVEL_SPREAD(I,AIRCRAFT_INDX) +
        SUPPORT_EQUIPMENT_HOURS_SEA(I) *
        AIRCRAFT_PER_SQUADRON) / I_LEVEL_AVAILABILITY_SEA;
  I_LEVEL_MANPOWER_SEA(I) = I_LEVEL_MANPOWER_SEA(I) +
        (I_LEVEL_AS_COEFF(I) * I_LEVEL_MANPOWER_SEA(I)) /
         I_LEVEL_AVAILABILITY_SEA;
  I_LEVEL_MANPOWER_SHORE(I) = (TOTAL_I_LEVEL_MANHOURS *
        I_LEVEL_SPREAD(I,AIRCRAFT_INDX) +
        SUPPORT_EQUIPMENT_HOURS_SHORE(I) *
     AIRCRAFT_PER_SQUADRON) / I_LEVEL_AVAILABILITY_SHORE;
  I_LEVEL_MANPOWER_SHORE(I) = I_LEVEL_MANPOWER_SHORE(I) +
        (I_LEVEL_AS_COEFF(I) * I_LEVEL_MANPOWER_SHORE(I))/
         I_LEVEL_AVAILABILITY_SHORE;
  END;
AC_ROUND:
  DO K = 1 TO 5;
  CALL INTEGER(I_LEVEL_MANPOWER_SEA(K),TEMPMEN);
  I_LEVEL_MANPOWER_SEA(K) =   TEMPMEN;
  CALL INTEGER(I_LEVEL_MANPOWER_SHORE(K),TEMPMEN);
  I_LEVEL_MANPOWER_SHORE(K) =  TEMPMEN;
```

```
                END;
                IF I_LEVEL_MANPOWER_SEA(3) < NUMBER_OF_AVIONICS_SKILLS_REQ
                   THEN I_LEVEL_MANPOWER_SEA(3) =
                        NUMBER_OF_AVIONICS_SKILLS_REQ;
                IF I_LEVEL_MANPOWER_SHORE(3) <
                     NUMBER_OF_AVIONICS_SKILLS_REQ THEN
                   I_LEVEL_MANPOWER_SHORE(3) =
                     NUMBER_OF_AVIONICS_SKILLS_REQ;
                DO I = 1 TO 5;
                TOTAL_FLEET_I_LEVEL_SEA(I) = I_LEVEL_MANPOWER_SEA(I) *
                              NUMBER_OF_SQUADRONS;
                TOTAL_FLEET_I_LEVEL_SHORE(I) =
                      I_LEVEL_MANPOWER_SHORE(I) * NUMBER_OF_SQUADRONS;
                END;
             /*    INPUT 2  */
           /*  CALCULATE ANY CHANGES IN THE PERMANENT AIMD CADRE
               DUE TO ADDING THIS AIRCRAFT                               */
           AC_INPUT2:
                IF SENSITIVITY_FLAG = 0 THEN DO;
                 READ FILE(INFILE) INTO(INPUT_ARRAY2);
                 NUMBER_SQ_ON_SEA = IA_DATA22;
                 NUMBER_AC_ON_SEA = IA_DATA23;
                 NUMBER_OF_NAS_DEPLOYED = IA_DATA24;
                DO I = 1 TO NUMBER_OF_NAS_DEPLOYED;
                READ FILE(INFILE) INTO(INPUT_ARRAY2);
                SHORE_AC_BEFORE(I) = IA_DATA22;
                SHORE_SQ_ADDED(I) = IA_DATA23;
                END;
                END;
           /*
                AIMD CADRE ON CARRIER WITHOUT THIS AIRCRAFT
```

```
                                                                         */
                 BEFORE_SEA_X = NUMBER_AC_ON_SEA - (NUMBER_SQ_ON_SEA *
                            AIRCRAFT_PER_SQUADRON);
                 CALL AIMD_FIXED(BEFORE_SEA_X,SEA_MEN_XB,
                      I_LEVEL_AVAILABILITY_SEA);
                 SEA_HOURS_XB = (4.05029 * BEFORE_SEA_X) /
                      I_LEVEL_AVAILABILITY_SEA;
                 CALL INTEGER(SEA_HOURS_XB,SEA_MEN_XB(6));
                 SEA_HOURS_XB = (GSE_HOURS_PER_AC_SEA*BEFORE_SEA_X) /
                      I_LEVEL_AVAILABILITY_SEA;
                 CALL INTEGER(SEA_HOURS_XB,SEA_MEN_XB(7));
                 SEA_HOURS_XB = (46.25 + 5.0861*SEA_MEN_XB(7))
                           /I_LEVEL_AVAILABILITY_SEA + SEA_HOURS_XB;
                 CALL INTEGER(SEA_HOURS_XB,SEA_MEN_XB(7));
     /*
          AIMD CADRE ON CARRIER WITH ALL AIRCRAFT
                                                                         */
                APTER_SEA_X = NUMBER_AC_ON_SEA;
                 CALL AIMD_FIXED(APTER_SEA_X,SEA_MEN_XA,
                      I_LEVEL_AVAILABILITY_SEA);
                 SEA_HOURS_XA = (4.05029 * APTER_SEA_X) /
                      I_LEVEL_AVAILABILITY_SEA;
                 CALL INTEGER(SEA_HOURS_XA,SEA_MEN_XA(6));
                 SEA_HOURS_XA = (GSE_HOURS_PER_AC_SEA*APTER_SEA_X) /
                      I_LEVEL_AVAILABILITY_SEA;
                 CALL INTEGER(SEA_HOURS_XA,SEA_MEN_XA(7));
                 SEA_HOURS_XA = (46.25 + 5.0861*SEA_MEN_XA(7))
                           /I_LEVEL_AVAILABILITY_SEA + SEA_HOURS_XA;
                 CALL INTEGER(SEA_HOURS_XA,SEA_MEN_XA(7));
     /*
          CHANGE IN AIMD CADRE ON CARRIER
```

```
                                                          */
               AIMD_CADRE_ADDED_SEA = 0.0;
               DO I = 1 TO 7;
               AIMD_CADRE_ADDED_SEA = AIMD_CADRE_ADDED_SEA +
                  (SEA_MEN_XA(I) - SEA_MEN_XB(I));
               END;
     /*
            AIMD CADRE AT NAS BEFORE AIRCRAFT IS ADDED
                                                          */
               DO I = 1 TO NUMBER_OF_NAS_DEPLOYED;
                  BEFORE_SHORE_X (I) = SHORE_AC_BEFORE(I);
                  TOT_SHORE_XA(I) = 0.0;
                  TOT_SHORE_XB(I) = 0.0;
               END;
               DO IA = 1 TO NUMBER_OF_NAS_DEPLOYED;
               CALL AIMD_FIXED(BEFORE_SHORE_X(IA),SHORE_MEN_XB,
                  I_LEVEL_AVAILABILITY_SHORE);
               SHORE_HOURS_XB(6) =(87.666 + .37487 * BEFORE_SHORE_X(IA) +
                     .0022157*(BEFORE_SHORE_X(IA)*BEFORE_SHORE_X(IA))) /
                     I_LEVEL_AVAILABILITY_SHORE;
               CALL INTEGER(SHORE_HOURS_XB(6),SHORE_MEN_XB(6));
            SHORE_HOURS_XB(7) = (GSE_HOURS_PER_AC_SHORE*BEFORE_SHORE_X(IA))
                        / I_LEVEL_AVAILABILITY_SHORE;
               CALL INTEGER(SHORE_HOURS_XB(7),SHORE_MEN_XB(7));
               SHORE_HOURS_XB(7) =(46.25 + 5.0861*SHORE_MEN_XB(7))
                     / I_LEVEL_AVAILABILITY_SHORE + SHORE_HOURS_XB(7);
               CALL INTEGER(SHORE_HOURS_XB(7),SHORE_MEN_XB(7));
               DO J = 1 TO 7;
               TOT_SHORE_XB(IA) = TOT_SHORE_XB(IA) + SHORE_MEN_XB(J);
               END;
               END;
```

```
/*
         AIMD CADRE AT NAS AFTER AIRCRAFT IS ADDED
                                                                  */
              DO IA = 1 TO NUMBER_OF_NAS_DEPLOYED;
              AFTER_SHORE_X(IA) = SHORE_AC_BEFORE(IA) +
                SHORE_SQ_ADDED(IA)* AIRCRAFT_PER_SQUADRON;
              END;
              DO IA = 1 TO NUMBER_OF_NAS_DEPLOYED;
              CALL AIMD_FIXED(AFTER_SHORE_X(IA),SHORE_MEN_XA,
                   I_LEVEL_AVAILABILITY_SHORE);
              SHORE_HOURS_XA(6) =(87.666 + .37487 * AFTER_SHORE_X(IA) +
                     .0022157*(AFTER_SHORE_X(IA)*AFTER_SHORE_X(IA))) /
                     I_LEVEL_AVAILABILITY_SHORE;
              CALL INTEGER(SHORE_HOURS_XA(6),SHORE_MEN_XA(6));
          SHORE_HOURS_XA(7) = (GSE_HOURS_PER_AC_SHORE*AFTER_SHORE_X(IA))
                    / I_LEVEL_AVAILABILITY_SHORE;
              CALL INTEGER(SHORE_HOURS_XA(7),SHORE_MEN_XA(7));
              SHORE_HOURS_XA(7) =(46.25 + 5.0861*SHORE_MEN_XA(7))
                     / I_LEVEL_AVAILABILITY_SHORE + SHORE_HOURS_XA(7);
              CALL INTEGER(SHORE_HOURS_XA(7),SHORE_MEN_XA(7));
              DO J = 1 TO 7;
              TOT_SHORE_XA(IA) = TOT_SHORE_XA(IA) + SHORE_MEN_XA(J);
              END;
              END;
/*
         CHANGE IN AIMD CADRE AT NAS BECAUSE OF THIS AIRCRAFT
                                                                  */
              DO I = 1 TO NUMBER_OF_NAS_DEPLOYED;
              AIMD_CADRE_ADDED_SHORE(I) = 0.0;
              END;
              DO I = 1 TO NUMBER_OF_NAS_DEPLOYED;
```

```
                  AIMD_CADRE_ADDED_SHORE(I) = AIMD_CADRE_ADDED_SHORE(I) +
                      (TOT_SHORE_XA(I) - TOT_SHORE_XB(I));
                  END;
         END  AIMD_CALCULATIONS;
         /*  ROUTINE TO CALCULATE AIMD CADRE REQUIREMENTS              */
         AIMD_FIXED:   PROC(AIRCRAFT,AIMD_MEN,AVAIL);
            DCL  AIRCRAFT                 FLOAT(6);
            DCL  AIMD_MEN   (7)           FLOAT(6);
            DCL  AVAIL                    FLOAT(6);
                 SEA_HOURS_XB = (18.575 + .93871*AIRCRAFT -
                      .0006217*AIRCRAFT**2)/ AVAIL;
                 CALL INTEGER(SEA_HOURS_XB,AIMD_MEN(1));
                 IF AIRCRAFT < 76 THEN AIMD_MEN(2) = 1;
                    ELSE IF AIRCRAFT < 201 THEN AIMD_MEN(2) = 2;
                    ELSE IF AIRCRAFT < 301 THEN AIMD_MEN(2) = 3;
                    ELSE AIMD_MEN(2) = 4;
                 SEA_HOURS_XB = (11.855 + .08987*AIRCRAFT +
                      .0003166*AIRCRAFT**2)/AVAIL;
                 CALL INTEGER(SEA_HOURS_XB,AIMD_MEN(3));
                 SEA_HOURS_XB =  4.72708*AIMD_MEN(3) /
                                 AVAIL + SEA_HOURS_XB;
                 CALL INTEGER(SEA_HOURS_XB,AIMD_MEN(3));
                 SEA_HOURS_XB = (10.2240 + .2386*AIRCRAFT) /AVAIL;
                 CALL INTEGER(SEA_HOURS_XB,AIMD_MEN(4));
                 SEA_HOURS_XB = (4.86 + .2257*AIRCRAFT) /AVAIL;
                 CALL INTEGER(SEA_HOURS_XB,AIMD_MEN(5));
         END AIMD_FIXED;
         /*  ROUTINE TO ROUNDOFF FRACTIONAL PEOPLE                     */
         INTEGER:  PROC(MEN,RMEN);
            DCL  MEN                          FLOAT(6);
            DCL  RMEN                         FLOAT(6);
```

```
          RMEN = 0;
          IF MEN > 7.5 THEN
                RMEN   =
                  TRUNC(MEN + .4999);
          ELSE DO;
                 IJ = 0;
                 II = 7;
                 DO I = 1 TO 6;
                 II = II - 1;
                 IF IJ = 0 THEN
                       IF MEN > I_LEVEL_ROUNDOFF(II)
                             THEN DO;
                               RMEN   = II + 1;
                               IJ = 1;
                             END;
                 END;
                 END;
                 IF IJ = 0 & MEN ¬= 0 THEN RMEN = 1;
     END INTEGER;
     /*  ROUTINE TO SPREAD AN AGGREGATE WORKLOAD TO WORK CENTERS  */
      AD_SPREAD:   PROC(WORKLOAD_SEA,WORKLOAD_SHORE,FACTOR_VFA,
                 FACTOR_OTHER);
        DCL  WORKLOAD_SEA   (23)          FLOAT(6);
        DCL  WORKLOAD_SHORE  (23)          FLOAT(6);
        DCL  FACTOR_VFA     (23)          FLOAT(6);
        DCL  FACTOR_OTHER         (23)    FLOAT(6);
       DCL  LOAD_FACTOR_OTHER           FLOAT(6);
       DCL  LOAD_FACTOR_VFA        FLOAT(6);
       /*  CALCULATE THE LOAD FACTORS  */
            LOAD_FACTOR_VFA = 1.0;
            LOAD_FACTOR_OTHER = 1.0;
```

```
              DO I = 1 TO 22;
                 IF WORKLOAD_SEA(I) ¬= 0
                    THEN DO;
                    LOAD_FACTOR_VFA = LOAD_FACTOR_VFA -
                       FACTOR_VFA(I);
                    LOAD_FACTOR_OTHER = LOAD_FACTOR_OTHER -
                       FACTOR_OTHER(I);
                    END;
              END;
              LOAD_FACTOR_VFA = 1.0 / LOAD_FACTOR_VFA;
              LOAD_FACTOR_OTHER = 1.0 / LOAD_FACTOR_OTHER;
              IF AIRCRAFT_INDX > 2 THEN GO TO AD_OTHER;
              DO I = 1 TO 22;
              IF WORKLOAD_SEA(I) = 0 THEN
              WORKLOAD_SEA(I) =FACTOR_VFA(I) *
                    WORKLOAD_SEA(23)*LOAD_FACTOR_VFA;
              IF WORKLOAD_SHORE(I) = 0 THEN
              WORKLOAD_SHORE(I) =FACTOR_VFA(I) *
                    WORKLOAD_SHORE(23)*LOAD_FACTOR_VFA;
              END;
              GO TO END_AD_SPREAD;
         AD_OTHER:
              DO I = 1 TO 22;
              IF WORKLOAD_SEA(I) = 0 THEN
              WORKLOAD_SEA(I) =FACTOR_OTHER(I) *
                    WORKLOAD_SEA(23)*LOAD_FACTOR_OTHER;
              IF WORKLOAD_SHORE(I) = 0 THEN
              WORKLOAD_SHORE(I) =FACTOR_OTHER(I) *
                    WORKLOAD_SHORE(23)*LOAD_FACTOR_OTHER;
              END;
              GO TO END_AD_SPREAD;
```

```
     END_AD_SPREAD:
     END  AD_SPREAD;
    /*
          ROUTINE TO RESET VALUES IN THE MODEL BASED ON SENSITIVITY
          VARIABLES
                                                                        */
     RESET:  PROC;
              DO I = 1 TO 23;
                  RAW_PM_WORKLOAD_SEA(I) = RAW_PM_WORKLOAD_SEA(I) *
                                           FACTOR1;
                  RAW_PM_WORKLOAD_SHORE(I) = RAW_PM_WORKLOAD_SHORE(I) *
                                               FACTOR1;
                  RAW_CM_WORKLOAD_SEA(I) = RAW_CM_WORKLOAD_SEA(I) *
                                           FACTOR1;
                  RAW_CM_WORKLOAD_SHORE(I) = RAW_CM_WORKLOAD_SHORE(I) *
                                               FACTOR1;
                  RAW_TM_WORKLOAD_SEA(I) = RAW_TM_WORKLOAD_SEA(I) *
                                           FACTOR1;
                  RAW_TM_WORKLOAD_SHORE(I) = RAW_TM_WORKLOAD_SHORE(I) *
                                               FACTOR1;
                  OTHER_HOURS_SEA(I) = 0.0;
                  OTHER_HOURS_SHORE(I) = 0.0;
                    DO J = 1 TO 10;
                    GRADE_LEVEL_SEA(I,J) = 0.0;
                    GRADE_LEVEL_SHORE(I,J) = 0.0;
                    END;
                  END;
                  SORTIES_WEEK_SEA = SORTIES_WEEK_SEA*FACTOR2;
                  SORTIES_WEEK_SHORE = SORTIES_WEEK_SHORE*FACTOR3;
                  FLYING_HOURS_WEEK_SEA = SORTIES_WEEK_SEA*
                                          SORTIE_LENGTH_SEA;
```

```
                    FLYING_HOURS_WEEK_SHORE = SORTIES_WEEK_SHORE*
                                              SORTIE_LENGTH_SHORE;
          END RESET;
     PAGEONE_REPORT:   PROC;
     DCL FLYING_HOURS_AWEEK_SEA                   FLOAT(6);
     DCL FLYING_HOURS_AWEEK_SHORE                 FLOAT(6);
          PUT FILE(OUTFILE) EDIT(STORE_TITLE)  (PAGE,COL(10),A);
          PUT FILE(OUTFILE) EDIT(
            'I.   FLEET DESCRIPTION AND OPERATIONAL ASSUMPTIONS')
               (SKIP,SKIP,COL(10),A);
          PUT FILE(OUTFILE) EDIT(
            '    A.   AIRCRAFT TYPE',TYPE_OF_AIRCRAFT)
               (SKIP,SKIP,COL(10),A,COL(55),A);
          PUT FILE(OUTFILE) EDIT('    B.   AIRCRAFT PER SQUADRON',
               AIRCRAFT_PER_SQUADRON)  (SKIP,COL(10),A,COL(55),F(6));
          PUT FILE(OUTFILE) EDIT('    C.   NUMBER OF SQUADRONS',
               NUMBER_OF_SQUADRONS)  (SKIP,COL(10),A,COL(55),F(6));
          PUT FILE(OUTFILE) EDIT(
            '    D.   TOTAL FLEET SIZE',TOTAL_AIRCRAFT)
               (SKIP,COL(10),A,COL(55),F(6));
          PUT FILE(OUTFILE) EDIT('  SEA              SHORE')
            (SKIP,SKIP,COL(55),A);
          PUT FILE(OUTFILE) EDIT(
            '    E.   SORTIE RATE(SORTIES/AC/FLYING DAY)',
               SORTIE_RATE_SEA,SORTIE_RATE_SHORE)
               (SKIP,COL(10),A,COL(55),F(6,2),COL(71),F(6,2));
          PUT FILE(OUTFILE) EDIT('    F.   MEAN SORTIE LENGTH (HOURS)',
               SORTIE_LENGTH_SEA,SORTIE_LENGTH_SHORE)
               (SKIP,COL(10),A,COL(55),F(6,2),COL(71),F(6,2));
          PUT FILE(OUTFILE) EDIT('    G.   FLYING DAYS PER WEEK',
               FLYING_DAYS_WEEK_SEA,FLYING_DAYS_WEEK_SHORE)
```

```
                   (SKIP,COL(10),A,COL(55),F(6,1),COL(71),F(6,1));
         PUT FILE(OUTFILE) EDIT(
             '    H.  TOTAL FLYING HOURS/SQUADRON/WEEK',
                   FLYING_HOURS_WEEK_SEA,FLYING_HOURS_WEEK_SHORE)
                   (SKIP,SKIP,COL(10),A,COL(55),F(6,2),COL(71),F(6,2));
         FLYING_HOURS_AWEEK_SEA = FLYING_HOURS_WEEK_SEA /
                                    AIRCRAFT_PER_SQUADRON;
         FLYING_HOURS_AWEEK_SHORE = FLYING_HOURS_WEEK_SHORE /
                                      AIRCRAFT_PER_SQUADRON;
         PUT FILE(OUTFILE) EDIT(
             '    I.  TOTAL FLYING HOURS/AIRCRAFT/WEEK',
                   FLYING_HOURS_AWEEK_SEA,FLYING_HOURS_AWEEK_SHORE)
                   (SKIP,SKIP,COL(10),A,COL(55),F(6,2),COL(71),F(6,2));
     END PAGEONE_REPORT;
     PAGETWO_REPORT:   PROC;
          PUT FILE(OUTFILE) EDIT(STORE_TITLE) (PAGE,COL(10),A);
         PUT FILE(OUTFILE) EDIT(
             'II.  RELIABILITY AND MAINTAINABILITY VALUES',
             'A. INPUTS') (SKIP,SKIP,COL(1),A,SKIP,SKIP,COL(5),A);
         PUT FILE(OUTFILE) EDIT('  WORK CENTER','PM','CM/TM')
               (SKIP,SKIP,COL(6),A,
                  COL(43),A,COL(67),A);
         PUT FILE(OUTFILE) EDIT('MMH/W MMH/D MMH/FH MMH/S MMH/FH',
             ' MMH/S MTBF  MTTR') (SKIP,COL(33),A,A);
     END PAGETWO_REPORT;
     PAGETWO_DETAIL_REPORT: PROC;
         DO K = 8,9,10,11,12,15,16,17,18,19,21,23;
            PUT FILE(OUTFILE) EDIT(WORK_CENTER_CODES(K),
                WORK_CENTER_NAMES(K),
                      STORE_PM_MMH_WEEK(K),STORE_PM_MMH_DAY(K),
                      STORE_PM_MMH_FH(K),STORE_PM_MMH_S(K),
```

```
                STORE_CM_MMH_PM(K),STORE_CM_MMH_S(K),
                STORE_CM_MTBF(K),STORE_CM_MTTR(K))
          (SKIP,COL(5),A(3),COL(9),A(23),(8) (X(1),F(5,2)));
        END;
        IF WUC_PTR ¬= 0 THEN DO;
          PUT FILE(OUTFILE) EDIT('WUC DATA') (SKIP,SKIP,COL(1),A);
          DO I = 1 TO WUC_PTR;
          PUT FILE(OUTFILE) EDIT(WUC_XXX(I),WUC_J_TYPE(I),
             WUC_V1(I),WUC_V2(I)) (SKIP,COL(15),A,COL(20),A,
             COL(25),A,COL(30),A);
          END;
          END;
    P2DR_END:
    END PAGETWO_DETAIL_REPORT;
    PAGETWO_SPREAD_REPORT:     PROC;
        PUT FILE(OUTFILE) EDIT(
          'B.  PM AND CM SPREAD BY WORK CENTER (%)')
             (SKIP,SKIP,SKIP,COL(1),A);
        PUT FILE(OUTFILE) EDIT(
          '  110  120  121  130  131  140  210  211  220',
          '  230  310  320') (SKIP,SKIP,COL(25),A,A);
        PUT FILE(OUTFILE) EDIT('  ') (SKIP,A);
        IF AIRCRAFT_INDX > 2 THEN GO TO OTHER_SPREAD;
    VFA_SPREAD:
        PUT FILE(OUTFILE) EDIT(
          ' TM - VF,VA ',WORKCENTER_TM_SPREAD_VFA(8),
             WORKCENTER_TM_SPREAD_VFA(9),
             WORKCENTER_TM_SPREAD_VFA(10),
             WORKCENTER_TM_SPREAD_VFA(11),
             WORKCENTER_TM_SPREAD_VFA(12),
             WORKCENTER_TM_SPREAD_VFA(13),
```

```
                WORKCENTER_TM_SPREAD_VFA(15),
                WORKCENTER_TM_SPREAD_VFA(16),
                WORKCENTER_TM_SPREAD_VFA(17),
                WORKCENTER_TM_SPREAD_VFA(18),
                WORKCENTER_TM_SPREAD_VFA(21),
                WORKCENTER_TM_SPREAD_VFA(22))
                (SKIP,COL(10),A,COL(26),12 (F(4,3),X(1)));
        PUT FILE(OUTFILE) EDIT(
            ' PM - VF,VA ',WORKCENTER_PM_SPREAD_VFA(8),
                WORKCENTER_PM_SPREAD_VFA(9),
                WORKCENTER_PM_SPREAD_VFA(10),
                WORKCENTER_PM_SPREAD_VFA(11),
                WORKCENTER_PM_SPREAD_VFA(12),
                WORKCENTER_PM_SPREAD_VFA(13),
                WORKCENTER_PM_SPREAD_VFA(15),
                WORKCENTER_PM_SPREAD_VFA(16),
                WORKCENTER_PM_SPREAD_VFA(17),
                WORKCENTER_PM_SPREAD_VFA(18),
                WORKCENTER_PM_SPREAD_VFA(21),
                WORKCENTER_PM_SPREAD_VFA(22))
                (SKIP,COL(10),A,COL(26),12 (F(4,3),X(1)));
        PUT FILE(OUTFILE) EDIT(
            ' CM - VF,VA ',WORKCENTER_CM_SPREAD_VFA(8),
                WORKCENTER_CM_SPREAD_VFA(9),
                WORKCENTER_CM_SPREAD_VFA(10),
                WORKCENTER_CM_SPREAD_VFA(11),
                WORKCENTER_CM_SPREAD_VFA(12),
                WORKCENTER_CM_SPREAD_VFA(13),
                WORKCENTER_CM_SPREAD_VFA(15),
                WORKCENTER_CM_SPREAD_VFA(16),
                WORKCENTER_CM_SPREAD_VFA(17),
```

```
                    WORKCENTER_CM_SPREAD_VPA(18),
                    WORKCENTER_CM_SPREAD_VPA(21),
                    WORKCENTER_CM_SPREAD_VPA(22))
                    (SKIP,COL(10),A,COL(26),12 (F(4,3),X(1)));
            GO TO CONT_PAGETWO;
        OTHER_SPREAD:
            PUT FILE(OUTFILE) EDIT(
               ' TM - ALL OTHER ',WORKCENTER_TM_SPREAD_OTHER(8),
                    WORKCENTER_TM_SPREAD_OTHER(9),
                    WORKCENTER_TM_SPREAD_OTHER(10),
                    WORKCENTER_TM_SPREAD_OTHER(11),
                    WORKCENTER_TM_SPREAD_OTHER(12),
                    WORKCENTER_TM_SPREAD_OTHER(13),
                    WORKCENTER_TM_SPREAD_OTHER(15),
                    WORKCENTER_TM_SPREAD_OTHER(16),
                    WORKCENTER_TM_SPREAD_OTHER(17),
                    WORKCENTER_TM_SPREAD_OTHER(18),
                    WORKCENTER_TM_SPREAD_OTHER(21),
                    WORKCENTER_TM_SPREAD_OTHER(22))
                    (SKIP,COL(10),A,COL(26),12 (F(4,3),X(1)));
            PUT FILE(OUTFILE) EDIT(
               ' PM - ALL OTHER ',WORKCENTER_PM_SPREAD_OTHER(8),
                    WORKCENTER_PM_SPREAD_OTHER(9),
                    WORKCENTER_PM_SPREAD_OTHER(10),
                    WORKCENTER_PM_SPREAD_OTHER(11),
                    WORKCENTER_PM_SPREAD_OTHER(12),
                    WORKCENTER_PM_SPREAD_OTHER(13),
                    WORKCENTER_PM_SPREAD_OTHER(15),
                    WORKCENTER_PM_SPREAD_OTHER(16),
                    WORKCENTER_PM_SPREAD_OTHER(17),
                    WORKCENTER_PM_SPREAD_OTHER(18),
```

```
                    WORKCENTER_PM_SPREAD_OTHER(21),
                    WORKCENTER_PM_SPREAD_OTHER(22))
                    (SKIP,COL(10),A,COL(26),12 (F(4,3),X(1)));
          PUT FILE(OUTFILE) EDIT(
             ' CM - ALL OTHER ',WORKCENTER_CM_SPREAD_OTHER(8),
                    WORKCENTER_CM_SPREAD_OTHER(9),
                    WORKCENTER_CM_SPREAD_OTHER(10),
                    WORKCENTER_CM_SPREAD_OTHER(11),
                    WORKCENTER_CM_SPREAD_OTHER(12),
                    WORKCENTER_CM_SPREAD_OTHER(13),
                    WORKCENTER_CM_SPREAD_OTHER(15),
                    WORKCENTER_CM_SPREAD_OTHER(16),
                    WORKCENTER_CM_SPREAD_OTHER(17),
                    WORKCENTER_CM_SPREAD_OTHER(18),
                    WORKCENTER_CM_SPREAD_OTHER(21),
                    WORKCENTER_CM_SPREAD_OTHER(22))
                    (SKIP,COL(10),A,COL(26),12 (F(4,3),X(1)));
     CONT_PAGETWO:
          PUT FILE(OUTFILE) EDIT('C.   AIMD INPUTS')
             (SKIP,SKIP,SKIP,COL(1),A);
          PUT FILE(OUTFILE) EDIT('MMH PER AC PER WEEK',
             I_LEVEL_MANHOURS_WEEK) (SKIP,COL(10),A,COL(55),F(6));
          PUT FILE(OUTFILE) EDIT('NUMBER OF SQUADRONS ON A CARRIER',
             NUMBER_SQ_ON_SEA) (SKIP,COL(10),A,COL(55),F(6));
          PUT FILE(OUTFILE) EDIT('TOTAL NUMBER ALL AIRCRAFT ON A ',
             'CARRIER',NUMBER_AC_ON_SEA) (SKIP,COL(10),A,A,COL(55),
             F(6));
          PUT FILE(OUTFILE) EDIT('NUMBER OF NAS DEPLOYED',
             NUMBER_OF_NAS_DEPLOYED) (SKIP,COL(10),A,COL(55),F(6));
          PUT FILE(OUTFILE) EDIT('AIR STATION','NO. OF AC',
             'NO. SQ. ADDED') (SKIP,SKIP,COL(20),A,COL(40),A,COL(60),
```

```
              A);
          DO I = 1 TO NUMBER_OF_NAS_DEPLOYED;
          PUT FILE(OUTFILE) EDIT(I,SHORE_AC_BEFORE(I),SHORE_SQ_ADDED(I))
          (SKIP,COL(25),F(2),COL(41),F(6),COL(62),F(6));
          END;
          PUT FILE(OUTFILE) EDIT('NUMBER OF AVIONICS SKILLS',
              NUMBER_OF_AVIONICS_SKILLS_REQ) (SKIP,SKIP,SKIP,COL(10),A,
              COL(55),F(6));
     END_PROC_PAGETWO:
     END PAGETWO_SPREAD_REPORT;
     PAGETHREE_REPORT:  PROC;
          IF AIMD_FLAG = '0' THEN GO TO PAGETHREE_CONT;
          TOTAL_I_LEVEL_SEA = 0.0;
          TOTAL_I_LEVEL_SHORE = 0.0;
          DO I = 1 TO 5;
          TOTAL_I_LEVEL_SEA = TOTAL_I_LEVEL_SEA +
                  I_LEVEL_MANPOWER_SEA(I);
          TOTAL_I_LEVEL_SHORE = TOTAL_I_LEVEL_SHORE +
                  I_LEVEL_MANPOWER_SHORE(I);
          END;
          TOTAL_PERSONNEL_SEA = TOTAL_PERSONNEL_SEA +
                  TOTAL_I_LEVEL_SEA;
          TOTAL_PERSONNEL_SHORE = TOTAL_PERSONNEL_SHORE +
                  TOTAL_I_LEVEL_SHORE;
    PAGETHREE_CONT:
          TOTAL_PERSONNEL_SEA = TOTAL_PERSONNEL_SEA *
                                   NUMBER_OF_SQUADRONS;
          TOTAL_PERSONNEL_SHORE = TOTAL_PERSONNEL_SHORE *
                                   NUMBER_OF_SQUADRONS;
          PUT FILE(OUTFILE) EDIT(STORE_TITLE) (PAGE,COL(10),A);
          PUT FILE(OUTFILE) EDIT(
```

```
                'III.  TOTAL FLEET MAINTENANCE MANPOWER REQUIREMENTS')
                    (SKIP,SKIP,COL(10),A);
            PUT FILE(OUTFILE) EDIT(
                '     TOTAL PERSONNEL WHEN CARRIER DEPLOYED IS     ',
                TOTAL_PERSONNEL_SEA)   (SKIP,SKIP,COL(10),A,F(8,1));
            PUT FILE(OUTFILE) EDIT(
                '     TOTAL PERSONNEL WHEN AT NAVAL AIRSTATION IS ',
                TOTAL_PERSONNEL_SHORE) (SKIP,SKIP,COL(10),A,F(8,1));
            PUT FILE(OUTFILE) EDIT('  BY PAYGRADE:         ','SEA','SHORE')
                (SKIP,SKIP,COL(15),A,SKIP,SKIP,COL(32),A,COL(59),A);
            PUT FILE(OUTFILE) EDIT(
                'PER SQUADRON','TOTAL FLEET','PER SQUADRON'
                'TOTAL FLEET') (SKIP,
                 COL(19),A,COL(33),A,COL(48),A,COL(62),A,SKIP);
            DO K = 9 TO 2 BY -1;
               GRADE_LEVEL_SEA(23,10) = GRADE_LEVEL_SEA(23,10) +
                     GRADE_LEVEL_SEA(23,K);
               GRADE_LEVEL_SHORE(23,10) = GRADE_LEVEL_SHORE(23,10) +
                     GRADE_LEVEL_SHORE(23,K);
               TOTAL_FLEET_SEA=GRADE_LEVEL_SEA(23,K)*NUMBER_OF_SQUADRONS;
               TOTAL_FLEET_SHORE = GRADE_LEVEL_SHORE(23,K) *
                   NUMBER_OF_SQUADRONS;
               PUT FILE(OUTFILE) EDIT(
                'E-',K,GRADE_LEVEL_SEA(23,K),TOTAL_FLEET_SEA,
                   GRADE_LEVEL_SHORE(23,K),TOTAL_FLEET_SHORE)
                   (SKIP,COL(5),A(2),F(1),COL(14),(4)  (X(4),F(10,2)));
            END;
            TOTAL_FLEET_SEA=GRADE_LEVEL_SEA(23,10) *
               NUMBER_OF_SQUADRONS;
            TOTAL_FLEET_SHORE = GRADE_LEVEL_SHORE(23,10) *
               NUMBER_OF_SQUADRONS;
```

```
PUT FILE(OUTFILE) EDIT(
    '* TOTAL',GRADE_LEVEL_SEA(23,10),TOTAL_FLEET_SEA,
     GRADE_LEVEL_SHORE(23,10),TOTAL_FLEET_SHORE)
     (SKIP,COL(5),A(7),COL(14),(4) (X(4),F(10,2)));
IF AIMD_FLAG = '0' THEN GO TO END_PAGETHREE_REPORT;
TOTAL_FLEET_SEA = 0.0;
TOTAL_FLEET_SHORE = 0.0;
DO I = 1 TO 5;
TOTAL_FLEET_SEA = TOTAL_FLEET_SEA +
      TOTAL_FLEET_I_LEVEL_SEA(I);
TOTAL_FLEET_SHORE = TOTAL_FLEET_SHORE +
      TOTAL_FLEET_I_LEVEL_SHORE(I);
END;
PUT FILE(OUTFILE) EDIT(
    'AIMD TAD',TOTAL_I_LEVEL_SEA,TOTAL_FLEET_SEA,
     TOTAL_I_LEVEL_SHORE,TOTAL_FLEET_SHORE)
   (SKIP,SKIP,COL(5),A(8),COL(14),(4) (X(4),F(10,2)));
AIMD_TOTAL_CADRE_ADDED = AIMD_CADRE_ADDED_SEA;
DO I = 1 TO NUMBER_OF_NAS_DEPLOYED;
AIMD_TOTAL_CADRE_ADDED = AIMD_TOTAL_CADRE_ADDED +
          AIMD_CADRE_ADDED_SHORE(I);
END;
PUT FILE(OUTFILE) EDIT('ADDED AIMD CADRE PERSONNEL')
   (SKIP,SKIP,COL(5),A(35));
PUT FILE(OUTFILE) EDIT('PER CARRIER',AIMD_CADRE_ADDED_SEA)
      (SKIP,COL(40),A(12),COL(55),F(10,2));
DO I = 1 TO NUMBER_OF_NAS_DEPLOYED;
PUT FILE(OUTFILE) EDIT('NAS-',I,AIMD_CADRE_ADDED_SHORE(I))
     (SKIP,COL(44),A(4),F(2),COL(55),F(10,2));
END;
PUT FILE(OUTFILE) EDIT('* DOES NOT INCLUDE PERSONNEL IN ',
```

```
                      'MAINTENANCE OFFICE (WC010) WHICH ARE LT. CMDRS')
                      (SKIP,SKIP,COL(5),A,A);
           END_PAGETHREE_REPORT:
           END   PAGETHREE_REPORT;
           PAGEFOUR_REPORT:  PROC;
           DCL SUBTOTAL_ONE_MSEA                  FLOAT(6);
           DCL SUBTOTAL_TWO_MSEA                  FLOAT(6);
           DCL SUBTOTAL_THREE_MSEA                FLOAT(6);
           DCL SUBTOTAL_FOUR_MSEA                 FLOAT(6);
           DCL MAINTENANCE_TOTAL_MSEA             FLOAT(6);
           DCL SUBTOTAL_ONE_MSHORE                    FLOAT(6);
           DCL SUBTOTAL_TWO_MSHORE                    FLOAT(6);
           DCL SUBTOTAL_THREE_MSHORE                  FLOAT(6);
           DCL SUBTOTAL_FOUR_MSHORE                   FLOAT(6);
           DCL MAINTENANCE_TOTAL_MSHORE               FLOAT(6);
           DCL SUBTOTAL_ONE_HSEA                  FLOAT(6);
           DCL SUBTOTAL_TWO_HSEA                  FLOAT(6);
           DCL SUBTOTAL_THREE_HSEA                FLOAT(6);
           DCL SUBTOTAL_FOUR_HSEA                 FLOAT(6);
           DCL MAINTENANCE_TOTAL_HSEA             FLOAT(6);
           DCL SUBTOTAL_ONE_HSHORE                    FLOAT(6);
           DCL SUBTOTAL_TWO_HSHORE                    FLOAT(6);
           DCL SUBTOTAL_THREE_HSHORE                  FLOAT(6);
           DCL SUBTOTAL_FOUR_HSHORE                   FLOAT(6);
           DCL MAINTENANCE_TOTAL_HSHORE               FLOAT(6);
                  PUT FILE(OUTFILE) EDIT(STORE_TITLE) (PAGE,COL(10),A);
                  PUT FILE(OUTFILE) EDIT(
                      'IV.  DETAILED SQUADRON MAINTENANCE MANPOWER ',
                      'REQUIREMENTS') (SKIP,SKIP,COL(10),A,A);
                  PUT FILE(OUTFILE) EDIT('WORK CENTER','SEA','SHORE')
                      (SKIP,SKIP,COL(6),A,
```

```
            COL(50),A,COL(70),A);
        PUT FILE(OUTFILE) EDIT('MANHOURS  MANPOWER',
            'MANHOURS  MANPOWER')
            (SKIP,COL(43),A,COL(63),A);
        SUBTOTAL_ONE_MSEA = 0;
        SUBTOTAL_TWO_MSEA = 0;
        SUBTOTAL_THREE_MSEA = 0;
        SUBTOTAL_FOUR_MSEA = 0;
        MAINTENANCE_TOTAL_MSEA = 0;
        SUBTOTAL_ONE_HSEA = 0;
        SUBTOTAL_TWO_HSEA = 0;
        SUBTOTAL_THREE_HSEA = 0;
        SUBTOTAL_FOUR_HSEA = 0;
        MAINTENANCE_TOTAL_HSEA = 0;
        SUBTOTAL_ONE_MSHORE = 0;
        SUBTOTAL_TWO_MSHORE = 0;
        SUBTOTAL_THREE_MSHORE = 0;
        SUBTOTAL_FOUR_MSHORE = 0;
        MAINTENANCE_TOTAL_MSHORE = 0;
        SUBTOTAL_ONE_HSHORE = 0;
        SUBTOTAL_TWO_HSHORE = 0;
        SUBTOTAL_THREE_HSHORE = 0;
        SUBTOTAL_FOUR_HSHORE = 0;
        MAINTENANCE_TOTAL_HSHORE = 0;
        DO K =  1 TO 6;
            PUT FILE(OUTFILE) EDIT(WORK_CENTER_CODES(K),
                WORK_CENTER_NAMES(K),
                TOTAL_TM_WORKLOAD_SEA(K),GRADE_LEVEL_SEA(K,10),
                TOTAL_TM_WORKLOAD_SHORE(K),GRADE_LEVEL_SHORE(K,10))
               (SKIP,COL(5),A(3),COL(10),A(30),(4) F(10,2));
            SUBTOTAL_FOUR_MSEA = SUBTOTAL_FOUR_MSEA +
```

```
                GRADE_LEVEL_SEA(K,10);
            SUBTOTAL_FOUR_HSEA = SUBTOTAL_FOUR_HSEA +
               TOTAL_TM_WORKLOAD_SEA(K);
            SUBTOTAL_FOUR_MSHORE = SUBTOTAL_FOUR_MSHORE +
                GRADE_LEVEL_SHORE(K,10);
            SUBTOTAL_FOUR_HSHORE = SUBTOTAL_FOUR_HSHORE +
               TOTAL_TM_WORKLOAD_SHORE(K);
         END;
         PUT FILE(OUTFILE) EDIT('SUB TOTAL',' OVERHEAD',
             SUBTOTAL_FOUR_HSEA,SUBTOTAL_FOUR_MSEA,
             SUBTOTAL_FOUR_HSHORE,SUBTOTAL_FOUR_MSHORE)
            (SKIP,SKIP,A(9),COL(15),A(25),(4) F(10,2),SKIP,SKIP);
         PUT FILE(OUTFILE) EDIT('  ') (SKIP,SKIP,A);
         DO K = 7 TO 13;
            PUT FILE(OUTFILE) EDIT(WORK_CENTER_CODES(K),
                WORK_CENTER_NAMES(K),
                TOTAL_TM_WORKLOAD_SEA(K),GRADE_LEVEL_SEA(K,10),
                TOTAL_TM_WORKLOAD_SHORE(K),GRADE_LEVEL_SHORE(K,10))
               (SKIP,COL(5),A(3),COL(10),A(30),(4) F(10,2));
            SUBTOTAL_ONE_MSEA = SUBTOTAL_ONE_MSEA +
                GRADE_LEVEL_SEA(K,10);
            SUBTOTAL_ONE_HSEA = SUBTOTAL_ONE_HSEA +
               TOTAL_TM_WORKLOAD_SEA(K);
            SUBTOTAL_ONE_MSHORE = SUBTOTAL_ONE_MSHORE +
                GRADE_LEVEL_SHORE(K,10);
            SUBTOTAL_ONE_HSHORE = SUBTOTAL_ONE_HSHORE +
               TOTAL_TM_WORKLOAD_SHORE(K);
         END;
         PUT FILE(OUTFILE) EDIT('SUB TOTAL',WORK_CENTER_NAMES(7),
             SUBTOTAL_ONE_HSEA,SUBTOTAL_ONE_MSEA,
             SUBTOTAL_ONE_HSHORE,SUBTOTAL_ONE_MSHORE)
```

```
            (SKIP,SKIP,A(9),COL(15),A(25),(4) F(10,2),SKIP,SKIP);
      PUT FILE(OUTFILE) EDIT('  ') (SKIP,SKIP,A);
      DO K = 14 TO 19;
         PUT FILE(OUTFILE) EDIT(WORK_CENTER_CODES(K),
              WORK_CENTER_NAMES(K),
              TOTAL_TM_WORKLOAD_SEA(K),GRADE_LEVEL_SEA(K,10),
              TOTAL_TM_WORKLOAD_SHORE(K),GRADE_LEVEL_SHORE(K,10))
              (SKIP,COL(5),A(3),COL(10),A(30),(4) F(10,2));
         SUBTOTAL_TWO_MSEA = SUBTOTAL_TWO_MSEA +
              GRADE_LEVEL_SEA(K,10);
         SUBTOTAL_TWO_HSEA = SUBTOTAL_TWO_HSEA +
             TOTAL_TM_WORKLOAD_SEA(K);
         SUBTOTAL_TWO_MSHORE = SUBTOTAL_TWO_MSHORE +
              GRADE_LEVEL_SHORE(K,10);
         SUBTOTAL_TWO_HSHORE = SUBTOTAL_TWO_HSHORE +
             TOTAL_TM_WORKLOAD_SHORE(K);
      END;
      PUT FILE(OUTFILE) EDIT('SUB TOTAL',WORK_CENTER_NAMES(14),
          SUBTOTAL_TWO_HSEA,SUBTOTAL_TWO_MSEA,
          SUBTOTAL_TWO_HSHORE,SUBTOTAL_TWO_MSHORE)
         (SKIP,SKIP,A(9),COL(15),A(25),(4) F(10,2),SKIP,SKIP);
      PUT FILE(OUTFILE) EDIT('  ') (SKIP,SKIP,A);
      DO K = 20 TO 22;
         PUT FILE(OUTFILE) EDIT(WORK_CENTER_CODES(K),
             WORK_CENTER_NAMES(K),
              TOTAL_TM_WORKLOAD_SEA(K),GRADE_LEVEL_SEA(K,10),
              TOTAL_TM_WORKLOAD_SHORE(K),GRADE_LEVEL_SHORE(K,10))
              (SKIP,COL(5),A(3),COL(10),A(30),(4) F(10,2));
         SUBTOTAL_THREE_MSEA = SUBTOTAL_THREE_MSEA +
             GRADE_LEVEL_SEA(K,10);
         SUBTOTAL_THREE_HSEA = SUBTOTAL_THREE_HSEA +
```

```
          TOTAL_TM_WORKLOAD_SEA(K);
       SUBTOTAL_THREE_MSHORE = SUBTOTAL_THREE_MSHORE +
           GRADE_LEVEL_SHORE(K,10);
       SUBTOTAL_THREE_HSHORE = SUBTOTAL_THREE_HSHORE +
          TOTAL_TM_WORKLOAD_SHORE(K);
    END;
    PUT FILE(OUTFILE) EDIT('SUB TOTAL',WORK_CENTER_NAMES(20),
        SUBTOTAL_THREE_HSEA,SUBTOTAL_THREE_MSEA,
        SUBTOTAL_THREE_HSHORE,SUBTOTAL_THREE_MSHORE)
       (SKIP,SKIP,A(9),COL(15),A(25),(4) F(10,2),SKIP,SKIP);
    PUT FILE(OUTFILE) EDIT('  ') (SKIP,SKIP,A);
    MAINTENANCE_TOTAL_MSEA = SUBTOTAL_ONE_MSEA +
         SUBTOTAL_TWO_MSEA + SUBTOTAL_THREE_MSEA +
         SUBTOTAL_FOUR_MSEA;
    MAINTENANCE_TOTAL_HSEA = SUBTOTAL_ONE_HSEA +
         SUBTOTAL_TWO_HSEA + SUBTOTAL_THREE_HSEA +
         SUBTOTAL_FOUR_HSEA;
    MAINTENANCE_TOTAL_MSHORE = SUBTOTAL_ONE_MSHORE +
         SUBTOTAL_TWO_MSHORE + SUBTOTAL_THREE_MSHORE +
         SUBTOTAL_FOUR_MSHORE;
    MAINTENANCE_TOTAL_HSHORE = SUBTOTAL_ONE_HSHORE +
         SUBTOTAL_TWO_HSHORE + SUBTOTAL_THREE_HSHORE +
         SUBTOTAL_FOUR_HSHORE;
    PUT FILE(OUTFILE) EDIT('  ORGANIZATIONAL','MAINTENANCE TOTAL',
        MAINTENANCE_TOTAL_HSEA,MAINTENANCE_TOTAL_MSEA,
        MAINTENANCE_TOTAL_HSHORE,MAINTENANCE_TOTAL_MSHORE)
       (SKIP,A(16),SKIP,A(17),COL(40),(4) F(10,2));
    IF AIMD_FLAG = '0' THEN GO TO END_PAGEFOUR_REPORT;
    PUT FILE(OUTFILE) EDIT('AIMD TAD REQUIREMENTS') (SKIP,SKIP,
          A(35));
    PUT FILE(OUTFILE) EDIT('POWER PLANTS',I_LEVEL_MANPOWER_SEA(1),
```

```
                    I_LEVEL_MANPOWER_SHORE(1))
               (SKIP,A(16),COL(50),F(10,2),COL(70),F(10,2));
          PUT FILE(OUTFILE) EDIT('AIRFRAMES    ',I_LEVEL_MANPOWER_SEA(2),
                    I_LEVEL_MANPOWER_SHORE(2))
               (SKIP,A(16),COL(50),F(10,2),COL(70),F(10,2));
          PUT FILE(OUTFILE) EDIT('AVIONICS     ',I_LEVEL_MANPOWER_SEA(3),
                    I_LEVEL_MANPOWER_SHORE(3))
               (SKIP,A(16),COL(50),F(10,2),COL(70),F(10,2));
          PUT FILE(OUTFILE) EDIT('ARMAMENT     ',I_LEVEL_MANPOWER_SEA(4),
                    I_LEVEL_MANPOWER_SHORE(4))
               (SKIP,A(16),COL(50),F(10,2),COL(70),F(10,2));
          PUT FILE(OUTFILE) EDIT('AVIATOR EQUIPMENT',
               I_LEVEL_MANPOWER_SEA(5),
                    I_LEVEL_MANPOWER_SHORE(5))
               (SKIP,A(16),COL(50),F(10,2),COL(70),F(10,2));
     END_PAGEFOUR_REPORT:
     END  PAGEFOUR_REPORT;
      PAGEFIVE_REPORT:  PROC;
          PUT FILE(OUTFILE) EDIT(STORE_TITLE) (PAGE,COL(10),A);
          PUT FILE(OUTFILE) EDIT(
               'V.   WORK CENTER HOUR BREAKDOWNS ')
              (SKIP,SKIP,COL(10),A);
          PUT FILE(OUTFILE) EDIT(
               'SEA','SHORE') (SKIP,SKIP,COL(48),A,COL(83),A);
          PUT FILE(OUTFILE) EDIT(
               ' WORK CENTER',' CM',' PM',' AS','OTH','TOT',
                    ' CM',' PM',' AS','OTH','TOT')
              (SKIP,SKIP,COL(6),A,COL(31),(10) (X(4),A(3)));
          DO I = 1 TO 22;
          PUT FILE(OUTFILE) EDIT(WORK_CENTER_CODES(I),
               WORK_CENTER_NAMES(I),
```

```
                TOTAL_CM_WORKLOAD_SEA(I),TOTAL_PM_WORKLOAD_SEA(I),
                AS_HOURS_SEA(I),OTHER_HOURS_SEA(I),
                TOTAL_TM_WORKLOAD_SEA(I),
                TOTAL_CM_WORKLOAD_SHORE(I),
                TOTAL_PM_WORKLOAD_SHORE(I),
                AS_HOURS_SHORE(I),OTHER_HOURS_SHORE(I),
                TOTAL_TM_WORKLOAD_SHORE(I))
               (SKIP,COL(5),A(3),COL(9),A(23),(10) (X(1),F(6,1)));
         END;
         PUT FILE(OUTFILE) EDIT(
             '   MANPOWER SENSITIVITY TO WORKLOAD          ')
            (SKIP,SKIP,SKIP,SKIP,SKIP,COL(10),A);
         PUT FILE(OUTFILE) EDIT('SEA','SHORE')
             (SKIP,SKIP,COL(43),A,COL(76),A);
         PUT FILE(OUTFILE) EDIT(
             '  WORK CENTER','MINUS HOURS',' REQ HOURS ',
               'PLUS HOURS','MINUS HOURS',' REQ HOURS ','PLUS HOURS')
            (SKIP,SKIP,COL(6),A,COL(31),(6) (A(11)));
         DO I = 8,9,10,11,12,15,16,17,18,19,21;
         PUT FILE(OUTFILE) EDIT(WORK_CENTER_CODES(I),
            WORK_CENTER_NAMES(I),
                MINUS_HOURS_SEA(I),
                TOTAL_TM_WORKLOAD_SEA(I),
                PLUS_HOURS_SEA(I),MINUS_HOURS_SHORE(I),
                TOTAL_TM_WORKLOAD_SHORE(I),PLUS_HOURS_SHORE(I))
               (SKIP,COL(5),A(3),COL(9),A(18),(6) (X(1),F(10,1)));
         END;
      END PAGEFIVE_REPORT;
      ENDRUN:
    END NAVMAN;
  /*
```