

AD-A238 087**DOCUMENTATION PAGE**Form Approved
OMB No. 0704-01882b. DECLASSIFICATION/DOWNGRADING SCHEDULE

4. PERFORMING ORGANIZATION REPORT NUMBER(S)

ARI Research Note 91-49

6a. NAME OF PERFORMING ORGANIZATION
U.S. Army Research Institute6b. OFFICE SYMBOL
(If applicable)
PERI-II1b. RESTRICTIVE MARKINGS
---3. DISTRIBUTION/AVAILABILITY OF REPORT
Approved for public release;
distribution is unlimited.5. MONITORING ORGANIZATION REPORT NUMBER(S)
---6c. ADDRESS (City, State, and ZIP Code)
5001 Eisenhower Avenue
Alexandria, VA 22333-56007b. ADDRESS (City, State, and ZIP Code)
---8a. NAME OF FUNDING/SPONSORING
ORGANIZATION U.S. Army Research
Institute for the Behavioral
and Social Sciences8b. OFFICE SYMBOL
(If applicable)
---9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
DAAL03-86-D-0001

8c. ADDRESS (City, State, and ZIP Code)

5001 Eisenhower Avenue
Alexandria, VA 22333-5600

10. SOURCE OF FUNDING NUMBERS

PROGRAM
ELEMENT NO.
63007APROJECT
NO.
795TASK
NO.
3310WORK UNIT
ACCESSION NO.
H01

11. TITLE (Include Security Classification)

Automating 76C-AIT Course Test Development and Analysis Procedures

12. PERSONAL AUTHOR(S)

Fritz, Donald T. (Oxon Hill Science/Technology High School)

13a. TYPE OF REPORT

Final

13b. TIME COVERED

FROM 90/06 TO 90/09

14. DATE OF REPORT (Year, Month, Day)

1991, April

15. PAGE COUNT

83

16. SUPPLEMENTARY NOTATION

Contracting Officer's Representative, Angelo Mirabella

17. COSATI CODES

FIELD GROUP SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

Test construction

Performance evaluation

Performance support technology

Job aid

Logistics

Criterion referenced testing

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

This report discusses the design of a software tool to aid researchers and instructors in development of a test plan matrix, construction of a test item bank, collection of student test results, analysis of student test data, and creation of class tests from the test item bank. It explains the initial program specifications, the refined specifications, the program pseudo-code, the data base design and operators instructions, and includes full program documentation.

**Reproduced From
Best Available Copy**

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT

☒ UNCLASSIFIED/UNLIMITED
 ☐ SAME AS RPT.
 ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION

Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL

Angelo Mirabella

22b. TELEPHONE (Include Area Code)

(703) 274-8827

22c. OFFICE SYMBOL

PERI-II

Automating 76C-AIT Course Test Development and Analysis Procedures

Donald T. Fritz

Independent Contractor

for

Contracting Officer's Representative
Angelo Mirabella

| | |
|--------------------|-------------------------|
| Accession For | |
| ADP | GN&A |
| DTIC | Doc |
| Unannounced | |
| Justification | |
| By | |
| Distribution/ | |
| Availability Codes | |
| Dist | Avail and/or Special |
| A-1 | |

Automated Instructional Systems Technical Area
Robert J. Seidel, Chief

Training Research Laboratory
Jack H. Hiller, Director



April 1991



91-04794



United States Army
Research Institute for the Behavioral and Social Sciences

Approved for public release; distribution is unlimited.

91 7 11 090

Reproduced From
Best Available Copy

U.S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES

**A Field Operating Agency Under the Jurisdiction
of the Deputy Chief of Staff for Personnel**

EDGAR M. JOHNSON
Technical Director

JON W. BLADES
COL, IN
Commanding

Technical review by

Angelo Mirabella

NOTICES

DISTRIBUTION: This report has been cleared for release to the Defense Technical Information Center (DTIC) to comply with regulatory requirements. It has been given no primary distribution other than to DTIC and will be available only through DTIC or the National Technical Information Service (NTIS).

FINAL DISPOSITION: This report may be destroyed when it is no longer needed. Please do not return it to the U.S. Army Research Institute for the Behavioral and Social Sciences.

NOTE: The views, opinions, and findings in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other authorized documents.

**Reproduced From
Best Available Copy**

AUTOMATING 76C-AIT COURSE TEST DEVELOPMENT AND ANALYSIS PROCEDURES

CONTENTS

| | Page |
|--|------|
| INTRODUCTION | 1 |
| Purpose | 1 |
| Background | 2 |
| INITIAL SPECIFICATIONS FOR SYSTEM DESIGN | 2 |
| Problem Statement | 2 |
| Scope | 3 |
| System Hardware Specifications | 3 |
| Software Specifications | 3 |
| Suggested Program Capabilities | 3 |
| Suggested Program Modules | 4 |
| Files Required | 4 |
| FINAL SYSTEM DESIGN SPECIFICATIONS | 5 |
| System Hardware Specifications | 5 |
| Software Specifications | 5 |
| Program Capabilities | 5 |
| Program Modules | 6 |
| Files Required | 7 |
| PSEUDO-CODE OF MODULE FUNCTIONS | 9 |
| PMAIN.PRG | 9 |
| PLANMAT.PRG | 10 |
| PSTUD.PRG | 11 |
| PTQUEST.PRG | 12 |
| PMTESMAT.PRG | 13 |
| PRCTMAT.PRG | 13 |
| PMSTUD.PRG | 15 |
| PMTQUES.PRG | 16 |
| PRTP.PRG | 16 |
| PROB.PRG | 17 |
| PRCLSFIL.PRG | 17 |
| PRTES.PRG | 18 |
| PNRSTP.DBF | 19 |
| PRNTB.PRG | 21 |
| PRNTES.PRG | 21 |
| PSECUR.PRG | 22 |
| PTESMAK.PRG | 23 |
| PTAKTEST.PRG | 24 |

CONTENTS (Continued)

| | Page |
|---|------|
| PROGRAM SECURITY | 25 |
| How It Works | 26 |
| Audit Trail | 27 |
| PROGRAM FILE CREATION TECHNIQUES | 28 |
| REFERENCES | 29 |
| USING THE 76C TEST DEVELOPMENT AND ANALYSIS PROGRAM . . . | 30 |
| Getting Started | 30 |
| Starting the Program | 31 |
| Logging In for the First Time | 32 |
| Opening Screen Forms | 32 |
| UNDERSTANDING THE MAIN MENU | 34 |
| Option A - Enter Data : New Test Plan | 35 |
| Option B - Enter Data : Student Test Results | 38 |
| Option C - Enter Data : Test Questions | 41 |
| Option D - Edit Data : Modify Test Plan | 44 |
| Option E - Modify Data: Student Test Data | 47 |
| Option F - Modify Data: Modify Test Questions | 50 |
| Option G - Review Data: Student/Class Test Analysis | 52 |
| Option H - Review Data: Test Plan Matrix | 55 |
| Option I - Review Data: Test Item Bank | 57 |
| Option J - Review Data: Course Test File Names | 59 |
| Option K - Review Data: Class Test Form | 60 |
| Option L - Print Data : Student Test Performance | 62 |
| Option M - Print Data : Student/Class Test Analysis | 65 |
| Option N - Print Data : Course Test File Names | 68 |
| Option O - Print Data : Test Item Bank | 69 |
| Option P - Print Data : Class Test Form | 70 |
| Option S - Security | 71 |
| Option T - Security : Create Test | 75 |
| Option U - Security : Take Test | 77 |

AUTOMATING 76C-AIT COURSE TEST DEVELOPMENT AND ANALYSIS PROCEDURES

INTRODUCTION

As new training methods and technologies are being applied to the 76C advanced individual training (AIT) course and to on-the-job training in the 76C military occupational skill (MOS), the accurate assessment of the impact of these modifications on trainee performance is essential. Performance measures must be developed that can provide a valid index of trainee skills in order to measure the effects of such course changes.

The development of performance measures should be conducted with the following questions in mind:

- . do assessment tests contain an accurate and representative sample of tasks performed on the job.
- . do assessment tests reliably measure the trainee's ability to perform critical tasks.
- . are there additional elements of the job that are critical to successful performance.
- . how can the performance measures reliably assess trainee knowledge in these additional areas.
- . do the tests have some predictive utility in assessing the level of performance that the trainee will exhibit on the job.
- . are the performance measures scorable in a consistent, objective fashion.

The last of these points is of particular importance if these performance measures are to be placed within a computer managed instructional (CMI) system, which would provide automatic scoring of the tests. [CORM90]

Purpose:

This report will describe the design of an automated system that will assist an experimenter in the design of a performance measurement instrument and the operational procedure to be followed when using the system to achieve that design.

Background:

In their report entitled, "The Development of Content-Valid Performance Measures for 76C Course and Field Assessment" [CORM90], the authors describe a methodology for 76C test battery construction that includes the following five phases:

- . Job Analysis
- . Test Plan
- . Item construction
- . Validation
- . Reliability Analysis

The automated system described in this report will assist the experimenter in the test plan, item construction, and reliability phases of this test battery design process.

During the test plan phase, the experimenter is required to develop a template for the construction of alternate versions of a performance measurement test. By using this template, the experimenter will achieve a high degree of consistency between test versions. This is accomplished by developing a test plan matrix which maps the critical functions and task operations of the job. By arranging the job functions as columns and the operations as rows, a matrix is created which specifies the kind of functional and operational skills that need to be measured. This matrix is then used to drive the development of test items which can evaluate trainees on the skills represented by each of the matrix cells. Next, during the item construction phase, the experimenter creates items which systematically represent each of the test plan categories and uses those items as an item bank from which different versions of the performance test can be created. Finally, during the reliability phase, the experimenter makes an assessment of the degree to which the performance test provides a stable index of an individual's knowledge, skills or opinions. This is normally accomplished by hand scoring and posting the student's performance or a group of student's performances into the cells of the test plan matrix.

INITIAL SPECIFICATIONS FOR SYSTEM DESIGN

Problem Statement:

Design a system that can be used by researchers to aide in the development of performance measurement instruments for use in the accurate assessment of the impact of training method modifications on trainee performance.

Scope:

The system will assist the experimenter in the construction of a test matrix, the building of a test item bank based on the test matrix, the production of an actual test from information supplied through the test matrix, facilities to allow the student to take the test on the computer, and on-line analysis of a student's or group of student's performance on that test.

System Hardware Specifications:

- . IBM or Compatible PC
- . 640 K main memory
- . 360 K or 1.2 M floppy drive
- . 20 Megabyte or larger hard drive system
- . Dot matrix or laser printer
- . Monochrome or Color monitor

Software Specifications:

Requires dBase III plus preferably on a hard drive

Suggested Program Capabilities:

- . Capability to provide secure access.
- . Capability to produce an audit trail.
- . Capability to browse audit trail
- . Capability to print audit trail
- . Capability to backup audit train separate from other files
- . Capability to encode passwords
- . Capability to add new users to security file.
- . Capability to edit user information in security file.
- . Capability to create test bank, student test and class performance file names from course data.
- . Capability to create new class test bank file from test bank structure.
- . Capability to backup security file separate from other files.
- . Capability to modify existing test bank.
- . Capability to print the test question bank file.
- . Capability to design a test plan matrix from user input.
- . Capability to modify a test plan matrix.
- . Capability for entry of student test result data.
- . Capability to allow review of student test data.
- . Capability to perform student test result data analysis in test matrix.
- . Capability to perform class test result data analysis in test matrix.
- . Capability to generate different test forms based on instructor input from test plan matrix.

- . Capability to print test form .
- . Capability to print student test result analysis report.
- . Capability to print a class test result analysis report.
- . Capability to print a test plan report.
- . Capability to create new class files from class file structure.
- . Capability to read student test answer files (interactive mode)

Suggested Program Modules :

Program modules should be developed to accomplish the following tasks:

- . Security Identification entry/modification/verification
- . Test Bank Question Entry/modification
- . Student Personal and test result data entry/modification
- . Test Plan Matrix development/modification
- . Test Data Analysis for student/class
- . Report Generator for test plan/student analysis/class analysis
- . Test Generator/printer
- . Interactive test routine

Files Required:

Files to hold the following information will need to be designed:

- . Class test results information
- . Test question data bank for each course test
- . Test questions for student test
- . Test plan matrix labels for course
- . Program security - passwords
- . Audit trail
- . Student responses and score (interactive mode)
- . Class test results file structure
- . Test Bank File structure
- . Class test file structure
- . Class test/test plan list
- . Temporary file for building test
- . Randomly generated numbers

FINAL SYSTEM DESIGN SPECIFICATIONS

System Hardware Specifications:

- . IBM or Compatible PC
- . 640 K main memory
- . 360 K or 1.2 M floppy drive
- . 20 Megabyte or larger hard drive system
- . Dot matrix or laser printer
- . Monochrome or Color monitor

Software Specifications:

dBase III Plus - full version or runtime version preferably installed on a hard drive.

Program Capabilities:

- . Password access security system in which user can change his own password once admitted to the system by system manager.
- . Produces an audit trail for each attempted log-in, each password entry, and each menu selection.
- . System manager can browse or print audit trail.
- . Passwords are kept in a security file in encoded form.
- . System manager can add new users to the system.
- . System manager can edit user information in security file.
- . Individual unique files are automatically created for each test bank, student test and class results file.
- . test
- . Allows modification to an existing test bank.
- . Allows printing the test question bank items.
- . Aids in the design a test plan matrix by the user
- . Allows modification of a test plan matrix.
- . Allows user or automatic entry of student test result data.
- . Allows review of student test data.
- . Performs student test result data analysis using test plan matrix.
- . Performs class test result data analysis using test plan matrix.
- . Randomly generates different test forms based on user input from test plan matrix.
- . Prints student test form .
- . Prints student test performance report.
- . Prints student test result analysis.
- . Prints class test result analysis.
- . Prints a test plan report.
- . Creates new class files from class file structure definition.
- . Allows student to take test on the computer.

- . Allows user to back up files to floppy disk when desired.

Program Modules :

This program consists of nineteen (19) modules that make up the automated system. They are:

- . PMAIN.PRG Controls access and displays the main program menu.
- . PLANMAT.PRG Creates test plan matrix.
- . PSTUD.PRG Entry of student test data.
- . PTQUEST.PRG Creates the test item data bank and allows test item entry.
- . PMTESMAT.PRG Modifies the pre-existing test plan matrix.
- . PMSTUD.PRG Modified student test performance data.
- . PMTQUES.PRG Modifies test items in test bank.
- . PRCTMAT.PRG Used to review a test plan matrix, review a student's performance, a class' performance, print a copy of both.
- . P RTP.PRG Review any test plan matrix that has been designed in the past.
- . PRQB.PRG Scans the test item data file.
- . PRCLSFIL.PRG Scans/prints the class file that contains the file names of all test banks, student tests and class data files for any given course.
- . PRTES.PRG Scans individual course annex test forms.
- . PNRSTP.PRG Makes a hard copy of a student's test performance.
- . PRNTB.PRG Produces a hard copy of the contents of the test item data bank.
- . PRNTES.PRG Produces a hard copy of a course annex testform.
- . PSECUR.PRG Allows addition of users to the program security file.
- . PTESMAK.PRG Produces a new test form from information from test plan matrix.
- . PTAKTEST.PRG Allows user to take a course test on the computer.
- . PBACK.PRG Makes backup copies of files to floppy disk.

Files Required:

Eleven (11) data files and seven (7) index files are used to store all of the data required for proper operation of this system. Their names and use are as follows:

Data Files:

- | | |
|---------------------|---|
| . PAAUDIT.DBF | Contains the audit trail kept by the program. |
| . PASEC.DBF | Contains the login/password/capabilities and personal data of legal users. |
| . PBNKSTRU.DBF | Contains the file structure description used in creating new test item bank files. |
| . PSTUDSTR.DBF | Contains the file structure description used in creation new class data files. |
| . PTESSTRUC.DBF | Contains the file structure description for new class test files. |
| . PLASLIST.DBF | Contains data showing the name of every course annex and testform created and the file names used for the test bank, class test, and class performance data for each new test form. |
| . PMAT.DBF | Contains the test plan matrix information for each test plan created along with the number of questions in the test bank for each cell in the matrix. |
| . &MFILNAM.dbf | Contains a pointer to the proper class file to be used. |
| . &MTFIINAM.DBF | Contains a pointer to the proper test bank or test file to be used. |
| . TEMP.DBF | Used as a temporary file from which different test forms are created to prevent contamination of actual test item bank. File sorted in row-column number order. |

- . PSHUFFLE.DBF Used to hold randomly picked question numbers during test creation.

Index Files:

- . MROW.NDX Indexed on RNUM. Used with TEMP.DBF.
- . PASEC.NDX Indexed on LOGIN. Used with PASEC.DBF.
- . PMAT.NDX Indexed on CNUMBER. Used with PMAT.DBF.
- . PLASLIST.NDX Indexed on CNUM. Used with PLASLIST.DBF
- . &FILNAM.NDX Points to class file index. Indexed on SID.
- . &MTFILNAM.NDX Points to test file index. Indexed on QNUM
- . PSHUFFLE.NDX Indexed on SHUFIND. Used with PSHUFFLE.DBF

PSEUDO-CODE OF MODULE FUNCTIONS

PMAIN.PRG

Pmain is designed to perform two major functions - security and module access control. The security function is accomplished using the PSEC.DBF which contains the log-in code, encrypted password and the capability level, that the system manager has assigned to the user. The user is prompted for a log-in code which if not entered correctly will cause a message of warning to appear. If the log-in is not completed correctly the second time the program will terminate. The access function is accomplished by checking the capability code that has been assigned to the user against a capability list assigned to each menu option. If the proper capability level is not presented, the user will be warned of an illegal access attempt and will then be returned to the main menu. All attempts at log-in, password entry, and menu selection are recorded in the audit trail data base. The module performs these functions by carrying out these processes.

1. Set dBASE system configuration by setting the following functions off: talk,confirm,status,history,bell,help and safety.
2. Open the security and audit trail files.
3. Set module constants to starting values.
4. Display opening screen and prompt user for login.
5. Write login into audit trail file. Check login against security file, if legal move to step 5, otherwise warn user of incorrect login and loop to step 4.
6. If acceptable login is not entered in two tries close files and exit program.
7. Extract user capability code and password from security file.
8. Prompt user for password.
9. Accept password one character at a time.
10. Encode password and check against security code,
11. Write to audit trail file, if legal password move to step 12, otherwise warn user of incorrect password and loop to step 8.
12. If acceptable password is not entered in three tries close files and exit program.
13. Close security file.
14. Display main menu program and prompt for menu choice.
14. Process user choice in case structure checking for proper capability level and writing choice to audit trail before continuing with selected process.
15. If proper capability is not found then warn user of illegal entry attempt and loop to main menu.
16. If user choice is Q or R close all files and return to either dBASE or DOS.
17. End module.

PLANMAT.PRG:

The main function of this module is to assist the user in developing the test plan matrix. The matrix design is limited to a maximum matrix size of six (6) rows by eight (8) columns. It is designed by entering the operation labels for each row and the job function labels for each column. Once that is done, a graphical form of the matrix is displayed on the screen and the user can then make any changes required before the form is stored in the PMAT data file. Once the - all is correct response - is issued the module sets all cell values to zero and stores the design information in the PMAT.DBF.

1. Close all databases.
2. Open PMAT using index PMAT.
3. Set program constants to opening values.
4. Prompt user for course number and annex letter.
5. Check PMAT to see if course test plan already exists.
6. If a test plan already exists notify the users of the fact that a test matrix already exists and ask user if he would like to enter another course number if the answer is no then return to the main menu.
7. If an existing plan was not found set all row and column labels to spaces.
8. Do the following steps until the test matrix is entered correctly
9. Do the following steps until the row and column labels have been entered correctly.
10. Prompt user for row and column labels.
11. If all is correct end loop. Otherwise loop to step 9.
12. Break the labels entered for the rows and column headings into their proper length to be displayed on the screen in matrix form. (Two strings having a length of 14 characters for the rows and three strings having a length of 7 for the columns)
13. Display the matrix as designed.
14. Prompt user for correctness.
15. If not correct loop to step 8.
16. Set all cell values to zero.
17. Store matrix plan in PMAT
18. Record the date+time+login+course number+annex letter+label for the first row of matrix in the audit trail file.
19. Return to main menu.

PSTUD.PRG:

The purpose of this module is to allow the entry of student test performance data into the class database (CL*****.DBF). This data includes a student ID, class ID, course number and annex code, and incorrect responses by question number that the student made on the test. The name of the class file is created by the module by encrypting the character string formed by concatenating the course number, annex letter, class number and test form. The encrypted value is then concatenated to the characters CL to form the complete class file name [example: CL5675].

1. Close all data files.
2. Open file PLASLIST using index PLASLIST.
3. Display title of module and ask if user wishes to continue.
4. If user doesn't wish to continue, close all files and return to main menu.
5. Do the following until more = NO.
6. Set screen values to spaces.
7. Prompt user for date of test, test form, course instructor, class number, annex letter.
8. Create file name string.
9. Encrypt file name.
10. Concatenate "CL" to front of encrypted file name.
11. Check to see if file already exists.
12. If file does not exist, search the PLASLIST file to see if the course data is already there. If it is then store the new class file name. If it isn't then create a new record for the class containing all class information including class file name. Finally create the class file from the pstudstr.dbf.
13. If the class file already exists, open it for data entry.
14. Do the next set of processes while there is another student to enter.
15. Do while data is not correct.
16. Prompt for student performance data.
17. Prompt for correctness.
18. End is it correct loop
19. Store data in class data file.
20. record date+time+login+student ID+course number+annex+class in the audit trail data file.
21. Prompt for another student
22. End another student loop.
23. End more loop.
23. Close all data files
24. Return to main menu.

FTQUEST.PRG:

The purpose of this module is to allow the user to build a test item data bank from which questions can be randomly selected to build a student test form.

1. Close all data files.
2. Display module title and prompt user for course number. Entry of the letter Q should allow the user to exit the module.
3. Prompt user for annex letter.
4. Open PMAT.bdf.
5. Check PMAT to determine if a test plan exists for the course-annex.
6. If a test plan doesn't exist notify the user and direct him to make a test plan first then exit to the main menu.
7. Copy test plan matrix into memory.
8. Create a test item data file name by concatenating the course name and annex letter.
9. Encrypt the file name and concatenate the characters "TB" to the beginning of the encrypted name.
10. If the test bank file already exists, open it. Otherwise open PLASLIST and create a record containing the course information and the test bank file name. Then create the test bank file and open it.
11. Do while there is another question to enter.
12. Prompt user for the row number and column number of the cell to which the question belongs.
13. Add 1 to the cell accumulator to record the total number of questions entered for that cell.
14. Extract the row and column labels for that cell from the test plan and concatenate them to form a concept label.
15. Set question variables to spaces.
16. Do while question item is not correct.
17. Prompt for question and answers.
18. If everything is correct end loop.
19. Store test item in test bank.
20. Record activity in Audit trail.
21. Prompt for another question.
22. If no more questions end loop.
23. Store total number of questions added for each cell of the test plan to the test plan file.
24. Close all data files
25. Exit module.

PMTESMAT.PRG:

The purpose of this routine is to allow the user to modify a test plan matrix before the building of the test bank is begun. Operation or job function labels should not be changed once the test bank has been created or the test items will not match the new labels without time consuming modification to the test question items themselves.

1. Close all databases.
2. Prompt user for course number and annex letter.
3. Open PMAT using index PMAT.
4. Check PMAT to see if course test plan exists.
5. If a test plan does not exist notify the users of the fact then return to the main menu.
6. If an existing plan was found copy test plan data to memory variables.
7. Do the following steps until the test matrix is entered correctly
 8. Display the matrix as it already exists.
 9. Do the following steps until the row and column labels have been entered correctly.
 10. Prompt user for row and column labels.
 11. If all is correct end loop. Otherwise loop to step 9.
 12. Break the labels entered for the rows and column headings into their proper length to be displayed on the screen in matrix form. (Two strings having a length of 14 characters for the rows and three strings having a length of 7 for the columns)
 13. Display the matrix as designed.
 14. Prompt user for correctness.
 15. If not correct loop .
16. Store matrix plan in PMAT
17. Record the date+time+login+course number+annex letter+label for the first row of matrix in the audit trail file.
18. Return to main menu.

PRCTMAT.PRG:

The purpose of this module is to allow the user to analyze a student's or class' test performance in relation to the test plan matrix from which the test was designed.

1. Close all files.
2. Determine if the program was selected from a review menu choice or a print menu choice from the main menu.
3. If from a print menu then display printer choice menu.
4. Display module menu.
5. Prompt user for menu choice.

6. Load accumulator starting values.
7. Open Plaslist.dbf
8. Set initial row and column numbers for screen display.
9. Do the following while course number = ?
 10. Prompt user for course number.
 11. If course number = Q then close all files and exit module.
 12. If course number = ?
 13. Display all unique course numbers, annex letters and test forms on screen.
 14. If screen is full erase screen and continue till end of file.
 15. End course number loop.
16. Prompt user for annex letter.
17. Prompt user for class number.
18. Prompt user for test form.
19. Locate a record for that class in the class list file.
20. If class not found display message and exit module.
21. if found copy class file name and test file name to memory.
22. Open the class file and test file.
23. if menu choice was A.
 24. Do while student number = ?
 25. Display all student numbers from class on screen.
 26. End do
 27. Find requested student in class file.
 28. Load student performance data into memory.
 29. Do while not end of performance data.
 30. Extract question number.
 31. Determine row, column number for question from test file.
 32. Add 1 to the row accumulator, the column accumulator and the cell accumulator.
 33. End do performance data.
 34. Else
 35. Do while not end of student file.
 36. Find next student in class file.
 37. Load student performance data into memory.
 38. Do while not end of performance data.
 39. Extract question number.
 40. Determine row, column number for question from test file.
 41. Add 1 to the row accumulator, the column accumulator and the cell accumulator.
 42. End do performance data.
 43. End if.
 44. Open test plan file.
 45. Load test plan into memory.
 46. If user entered from print selection turn on printer.
 47. Display test plan with completed matrix on screen or printer.
 48. Turn off printer if on.

48. Close all files.
49. Exit module.

PMSTUD.PRG:

This module is designed to allow the user to modify student test performance data.

1. Close all data files.
3. Open file PLASLIST using index PLASLIST.
4. Do the following until more = NO.
5. Do the following while course number = ?
6. Prompt user for course number.
7. If course number = Q then close all files and exit module.
8. If course number = ?
9. Display all unique course numbers, annex letters and test forms on screen.
10. If screen is full erase screen and continue till end of file.
11. End course number loop.
12. Prompt for annex letter, class number, and test form.
13. Find record matching course number, annex, class number and test form. If found copy class file name to memory, otherwise loop for new course information.
14. Open class file.
15. do while student ID = ?
16. Prompt for student ID number.
17. If student ID is Q then close files and exit module.
18. If student id = ?
19. List student numbers on screen until screen is full or end of file is reached.
20. End student number loop.
21. Locate student data and store to memory variables.
22. Do while data is not correct.
23. Display student performance data.
24. Prompt for corrections.
25. Prompt for correctness.
26. End is it correct loop
27. Store data in class data file.
28. record date+time+login+student ID+course number+annex+class in the audit trail data file.
29. Prompt for another student
30. End another student loop.
31. close all data files
32. return to main menu.

PMTQUES.PRG:

The purpose of this module is to allow the user to modify a test item data bank.

1. Close all data files.
2. Display module title and prompt user for course number. Entry of the letter Q should allow the user to exit the module.
3. Prompt user for annex letter.
4. Prompt user for question number to modify.
5. Open Plaslist index Plaslist.
6. Find the record that matches the course number and test annex.
7. Store the test bank file name to memory.
8. Do while there is another question to modify.
9. Store row number and column number to memory variables.
10. Display present test item.
11. Do while question item is not correct.
12. Prompt for corrections.
13. Extract concept from test plan
14. If everything is correct end loop.
15. If row or column number has changed update proper row totals and cell totals in test plan.
16. Store test item in test bank.
17. Record activity in Audit trail.
18. Prompt for another question.
19. If no more questions end loop.
20. Store total number of questions added for each cell of the test plan to the test plan file.
21. Close all data files
22. Exit module.

PRTP.PRG:

This routine is designed to allow the user to review any test plan matrix that is contained in the test plan file. It displays the matrix and the number of questions in the test item bank for each cell.

1. Close all data files.
2. Open PMAT index PMAT.
3. Prompt user for course number.
4. If course number is Q then close all files and exit module.
5. Prompt user for annex letter.
6. Find plan for course number, annex letter in data file.
7. If found copy to memory, otherwise inform user that record of plan does not exist and exit module.
8. Display plan on screen.
9. Wait while plan is reviewed.

10. Close all files.
11. Exit module.

PROB.PRG:

This module is designed to allow the user to browse through the test item bank file to look for errors without danger of accidentally entering errors into the database.

1. Close all files.
2. Open Plaslist index Plaslist.
3. Do while more = yes.
4. Set program memory variables to spaces.
5. Do while course number = ?
6. Prompt user for course number.
7. If course number = Q then close all files and exit module.
8. If course number = ?
9. Display all unique course numbers and annex letters.
10. If screen is full erase screen and continue till end of file.
11. End course number loop.
12. Prompt for annex letter.
13. Find record matching course number, and annex. If found copy test bank file name to memory, otherwise loop for new course information.
14. Open test bank file.
15. Do browse while not end of file.
16. Store test item to memory variables.
17. Display test item.
18. Prompt for continuation of browse or abandon browse.
19. End browse loop.
20. End more loop.
21. Close all files
22. Exit module.

PRCLSFIL.PRG:

The purpose of this module is to allow the user to scan or make a hardcopy of the records of all the courses for which test plan matrix's, test banks, class files, and student tests have been created.

1. Close all files.
2. Open Plaslist index plaslist
3. Display module title and prompt for continue.
4. Check to seen if main menu option was from the review choice section or the print choice section.

5. If from print choice then set device to printer.
6. Display screen headings.
7. Do while not end of file.
8. Display file data on screen if review, send to printer if print.
9. If printer and row coordinate is > 56 eject page and print headings on new page, set row coordinate to 4.
10. If screen and row coordinate is equal to 21 prompt for continued browse.
11. If answer to prompt is no, close files and exit module.
12. If answer to prompt is yes then erase screen starting at row coordinate 4 ending at row coordinate 21, set row coordinate to 4.
13. End of end of file loop.
14. If printer then set device back to screen.
15. Notify user that end of file has been reached.
16. close files
17. Exit module.

PRTES.PRG:

The purpose of this module is to allow the user to scan the student test file to get an idea of what questions the program chose from the test item bank for the student to answer.

1. Close all files.
2. Open Plaslist index Plaslist.
3. Do while more = yes.
4. Set program memory variables to spaces.
5. Do while course number = ?
6. Prompt user for course number.
7. If course number = Q then close all files and exit module.
8. If course number = ?
9. Display all unique course numbers and annex letters.
10. If screen is full erase screen and continue till end of file.
11. End course number loop.
12. Prompt for annex letter.
13. Prompt for test form.
13. Find record matching course number, and annex and test form. If found copy student test file name to memory, otherwise loop for new course information.
14. Open student test file.
15. Do browse while not end of file.
16. Store test item to memory variables.
17. Display test item.

18. Prompt for continuation of browse or abandon
browse.
19. End browse loop.
20. End more loop.
21. Close all files
22. Exit module.

PNRSTP.DBF:

The purpose of this module is to allow the user to obtain a report of a student's test performance. More specifically it produces a report containing the student information, question number answered incorrectly and the job operations/job function to which that question referred on the screen, on the printer, or both.

1. Close all files
2. Display module menu on screen.
3. Prompt user for menu choice.
4. If choice is D close files and exit module.
5. Set memory variables to spaces.
6. Open Plaslist file.
7. Do while course number = ?
8. Prompt user for course number.
9. If course number = Q then close all files and exit module.
10. If course number = ?
11. Display all unique course numbers and annex letters.
12. If screen is full erase screen and continue till end of file.
13. End course number loop.
14. Prompt user for annex, class number, and test form.
15. Find a record having course number, annex letter, class number and test form.
16. If matching record is found extract the class file name and student test file name from the file, otherwise display file not found message, close files and exit.
17. Open the class file and student test file.
18. Do while student ID = ?
19. Prompt user for student ID.
20. If student ID = ?
21. Display all unique student ID numbers.
22. If screen is full erase screen and continue till end of file.
23. End course number loop.
24. Do case based on menu choice.
25. Case A
26. Use student file.
27. Store student performance to a memory variable.
28. Display analyzing message.
29. Convert student performance string into separate

question numbers.

30. Display question number and concept on screen
31. Check to see if screen is full.
32. If full erase screen, otherwise continue analysis.
33. When end of file reached, display message that analysis is complete
34. End case A.
35. Case B
36. Use student file.
37. Store student performance to a memory variable.
38. Display analyzing message.
39. Do while not end of performance string.
40. Convert student performance string into separate question numbers.
41. Display question number and concept on screen
42. Check to see if screen is full.
43. If full erase screen, otherwise continue analysis.
44. End analysis loop.
45. Set device to printer.
46. Store student performance to a memory variable.
47. Print analyzing heading message.
48. Do while not end of performance string.
49. Convert student performance string into separate question numbers.
50. Print question number and concept .
51. Check to see if page is full.
52. If full eject page, Print headings on new page, continue with analysis.
53. End of analysis loop.
54. Set device to screen.
55. When end of performance string is reached, display message that analysis is complete on screen.
56. End case B.
57. Case C
58. Set device to printer.
59. Store student performance to a memory variable.
60. Print analyzing heading message.
61. Do while not end of performance string.
62. Convert student performance string into separate question numbers.
63. Print question number and concept .
64. Check to see if page is full.
65. If full eject page, Print headings on new page, continue with analysis.
66. End of analysis loop.
67. Set device to screen.
68. When end of performance string is reached, display message that analysis is complete on screen.

69. End case C.
70. Exit module.

PRNTB.PRG:

This module is designed to allow the user to send a copy of the test item bank file contents to the printer.

1. Close all files.
2. Open Plaslist index Plaslist.
3. Set program memory variables to spaces.
4. Do while course number = ?
5. Prompt user for course number.
6. If course number = Q then close all files and exit module.
7. If course number = ?
8. Display all unique course numbers and annex letters.
9. If screen is full erase screen and continue till end of file.
10. End course number loop.
11. Prompt for annex letter.
12. Find record matching course number, and annex. If found copy test bank file name to memory, otherwise loop for new course information.
13. Open test bank file.
14. Set device to printer
15. Do while not end of file.
16. Store test item to memory variables.
17. Print test item and answer.
18. End of end of file browse loop.
19. Set device to screen.
20. Display end of file message.
21. Close all files
22. Exit module.

PRNTES.PRG:

The purpose of this module is to print a hard copy of the student test without the correct answers listed.

1. Close all files.
2. Open Plaslist index Plaslist.
3. Set program memory variables to spaces.
4. Do while course number = ?
5. Prompt user for course number.
6. If course number = Q then close all files and exit module.
7. If course number = ?
8. Display all unique course numbers and annex letters.
9. If screen is full erase screen and continue till end

- of file.
- 10. End course number loop.
- 11. Prompt for annex letter.
- 12. Find record matching course number, and annex. If found copy student test file name to memory, otherwise loop for new course information.
- 13. Open student test file.
- 14. Set device to printer
- 15. Do while not end of file.
- 16. Store test item to memory variables.
- 17. Print test item without correct answer..
- 18. End of end of file loop.
- 19. Set device to screen.
- 20. Display end of file message.
- 21. Close all files
- 22. Exit module.

PSECUR.PRG:

The purpose of this module is to allow the system manager to set up user login, password, and capabilities and to allow a user to change his own password. The system manager's part of the module can only be accessed by someone with a capability value of F, while the user part of the module can be accessed by anyone.

- 1. Close all files.
- 2. Open PASEC index PASEC.
- 3. Check for capability level.
- 4. If capability level = F
- 5. Set choice variable to space.
- 6. Display module menu.
- 7. Get menu choice.
- 8. Do case choice A
- 9. Set variables to spaces.
- 10. do while not correct.
- 11. Display screen prompt for user information.
- 12. Read user information.
- 13. Encode user password.
- 14. Prompt for correctness of data.
- 15. end do loop correct.
- 16. Store user data in security data base.
- 17. Open Paaudit
- 18. Record new user in audit trail file
- 19. End case A.
- 20. Do case B.
- 21. Display menu
- 22. Prompt user for choice
- 23. do case choice
- 24. choice = A
- 25. Browse audit trial.
- 26. choice = B

```

27.          set device to printer
28.          report form paudit to printer
29.          set device to screen.
30.          Display report complete message.
31.      End case B.
32.      Do case C.
33.          Set variables to spaces.
34.          Retrieve old user information from data file.
35.          Do while not correct.
36.              Display screen prompt for user
                  information.
37.              Read user information.
38.              Encode user password.
39.              Prompt for correctness of data.
40.          End do loop correct.
41.          Store user data in security data base.
42.          Open Paaudit
43.          Record new user in audit trail file
44.      End case C.
45.  Else
46.      Clear screen.
47.      Prompt user for login code.
48.      Locate user in security file.
49.      Display user data.
50.      Prompt user for new password.
51.      Encode new password and record in security file.
52.  Endif
53.  Close all files
54.  Exit module.

```

PTESMAK.PRG:

The purpose of this module is to allow the user to create student test forms from the test plan matrix and test item question bank.

1. Close all files.
2. Store 0 to all accumulator variables.
3. Prompt user for course number.
4. Prompt user for annex letter.
5. Prompt user for method of selecting questions either by row or cell.
6. Open PMAT file for test plan.
7. Locate required test plan.
8. If test plan found
9. Display explanation on how to make a test.
10. Store test plan data to memory variables.
11. Create test bank file name from course name and annex letter.
12. Erase temporary file and index.

13. Open test bank file.
14. Sort test bank file on row number and column number to temporary file.
15. Index temp file on row number to mrow index.
16. Else
17. Notify user that the test plan file could not be found.
18. Close all files.
19. Exit module.
20. End if found.
21. Display test plan matrix.
22. Display number of questions available in test bank for each row and individual cell.
23. If user wishes to pick questions by row, prompt user for number of questions desired from each row.
24. If user wishes to pick questions by cell, prompt user for number of questions from each cell.
25. Convert the users input from character data to numeric for numbers of questions per cell or row.
26. Do while more questions to select.
27. Open shuffle file.
28. Determine where the first set of desired questions starts and ends in the temporary data file.
29. Generate a set of random numbers from the record number range for that row or cell.
30. Store the numbers in the shuffle file.
31. End do loop.
32. Scan the class list data file to determine the last test form that was created for the given course-annex.
33. Add one to the last annex and use the new value for the new test form letter.
34. Create a new test form file name from the course-annex-test form letter from the ptesstruc file.
35. Index the test file on question number.
36. Do while not end of file shuffle.
37. Read number in first record.
38. Go to shuffle record value in Temporary file.
39. Copy question and answers to test file.
40. Go to next shuffle record.
41. End loop.
42. Close all files
43. Exit module.

PTAKTEST.PRG:

1. Close all files.
2. Set all variables to opening values.
3. Prompt user for course number.
4. Prompt user for annex letter.
5. Prompt user for class number.
6. Prompt user for test form letter.

7. Prompt user for ID number.
8. Open class list file.
9. Locate the proper testform file name.
10. If test file is found go on, otherwise close all files and exit module.
11. Store zeros to accumulators.
12. Do while not end of test file.
13. Display test question and answer choices on screen.
14. Prompt user for correct answer.
15. Check user's answer against correct answer.
16. If answer correct add one to correct accumulator.
17. If answer is incorrect add one to incorrect accumulator and add the question number to questions wrong string variable
18. End loop.
19. Create the name of the student answer file from the course number+annex letter+class number+testform letter.
20. Check to see if the student answer file already exists.
21. If file does not exist create it and index it on student id.
22. If file exists open it.
23. Store student data and performance data in student file.
24. Close all files
25. Exit module.

PROGRAM SECURITY

The security aspect of this system has been addressed by incorporating a log-in/password system, a user capability and an audit trail system into the program code. This is accomplished by following these four steps.

First, the system manager must decide who is to have access to the program itself and then assign that user a log-in code. The log-in code consists of a set of up to ten characters which is used as a unique identifier for that user. Second, the user is also given an initial password which is stored in the security file in an encoded form. This password can be changed by the user at anytime after the initial log-in has been accomplished. Third, system manager must also assign a capability code to the user. This code will determine which of the main menu options the user can legally access. If a capability code is not assigned the user will not be allowed any access to the system. Finally, the program itself tracks the progress of the user through the program as the user makes menu selections and accomplishes his tasks.

How It Works:

The log-in code is checked by comparing the code entered by the user against the code that has been stored in the security file for that user. This code is used not only for log-in but also to track the user as he progresses through the system. If the user does not enter the correct log-in code the first time he is warned and allowed to try again. If the correct log-in is not entered the second time, the user is signed off the system and all program access is terminated.

After receiving an acceptable log-in code, the program transfers the users capability code to a memory variable and then prompts the user for his password. The password can be up to eight (8) characters long and eight (8) characters must be entered at the password prompt. If the password is less than eight characters long then spaces should be entered to complete the eight characters. The characters are accepted one character at a time and concatenated unto the users login code to form a complete string .

Once the complete password has been entered, any trailing blanks are truncated from the string. This isolates the password and it can now be encrypted. The encryption process is accomplished as follows:

- (1) Form a string from the letters of the password.
- (2) Trim off any trailing blanks from the login code and the password string and concatenate them.
- (1) Extract the first letter of the password.
- (2) Get the ASCII value of that letter.
- (3) Multiply the ASCII value by the letters position in the password string.
- (4) Sum the resulting product in step three.
- (5) Repeat the process until the whole password has been encrypted.

The actual code used for the process is :

```
MPAS = MPAS1 + MPAS2 + MPAS3 + MPAS4 + MPAS5 + MPAS6 +  
      MPAS7 + MPAS8  
MFILNAM = TRIM(MLOG) + TRIM(MPAS)  
MSUM = 0  
MSTAR = 1  
DO WHILE MSTAR <= LEN(MFILNAM)  
    MSUM = MSUM + (ASC(SUBSTR(MFILNAM,MSTAR,1)) * MSTAR  
    MSTAR = MSTAR + 1  
ENDDO
```

This encryption process results in a multidigit number which is the actual password for that user. The user get three tries to enter the correct password before his session is terminated. The password check completes the sign-on security check. The next check is made when the user makes selections from the main menu. Accessibility to menu options is controlled by the capability code (A, B, C or F) assigned the user. These codes limit user access as follows:

User capability code .. A .. can access menu options

- (B) Enter student test results
- (C) Enter test questions
- (F) Modify test questions
- (I) Review test questions
- (S) Security - change password only.
- (U) Take test

User capability code .. B .. can access all menu options except:

- (S) Security - set up user login codes and passwords

User capability code .. C .. can only access menu option

- (U) Take test

User capability code .. F .. can access all system options, this code should be assigned to the system manager only, since it gives access to the part of the security module that allows the set up of new user accounts.

Audit trail:

The final component of the security package is the audit trail. The audit trail is a record of which user attempted to sign on the system, the attempts at entering an incorrect password, the menu options selected, and selected information about what was done in certain modules to which the user was allowed access. A few samples of an audit trail record can be seen below.

Audit Trail Examples

| <u>Date</u> | <u>Time</u> | <u>User</u> | <u>Activity</u> | <u>Menu option</u> | <u>Comment</u> |
|-------------|-------------|--------------------|-------------------|----------------------|----------------|
| 07/31/90 | 11:44:41 | SYSTEM | Main menu choice: | G | |
| 07/25/90 | 13:40:53 | ***** | LOG-IN for: | SYSTEM | No tries = 1 |
| 07/26/90 | 12:47:03 | system | Main menu choice: | C | |
| 07/26/90 | 12:48:28 | system-D12345-A-51 | system | Procedures/Follow Up | |

The audit trail examples above should give you a general idea of what is stored in the audit trail file. For more specific information on the audit trail record for each module see appendix E.

PROGRAM FILE CREATION TECHNIQUES

This program is designed to create its own data files for holding test items for individual course annexes, student performance data for individual classes, and test items for individual class test forms. Each of these files require a unique file structure and a unique file name. The structures for these files are contained in three files used by and supplied with this program. These structure files are PBNKSTRU.DBF, PTESSTRU.DBF AND PSTUDSTR.DBF. The contents of these files are listed in appendix D under file names TB*****.dbf, CL*****.DBF and TS*****.DBF. The actual values of the asterix in the file names used by the program are generated using the encryption algorithm described in the security section of this report on a string of characters generated from a combination of course name, annex letter, class number, and test form letter.

The file name for the course test item data bank is created by concatenating the letters "TB" with the encrypted string made from the course number and course annex letter. For example, the test item bank name for the course number 76C-AIT-PLL annex B when encrypted would become the file " TB5363.dbf ".

The file name for the class file would be created by building a string from the course number, annex letter, class number and test form letter. Encrypting that string and concatenating it to the letters "CL". For example, the class data file name for the course 76C-AIT-PLL, annex B, class number 13, test form A would become " CL7689 ".

The file name for the class test form would be created by building a string from the course number, annex letter and test form letter. The test form letter is automatically determined by scanning the Plaslist data file, determining the last test form letter used and taking the next letter as the letter designation for the new test form. This string is then encrypted and the resulting code concatenated to the letters "TS". For example, the student test file name for the course 76C-AIT-PLL, annex B, test form A would become " TS6208 ".

Once these files are created, their names are stored in the file Plaslist and can be accessed by using main menu option H. The PLASLIST file is accessed by various program modules when the file names are needed for updating, browsing or printing.

REFERENCES

- CORM90 Comier, S., Dressel, J.D, Mirabella, A. (1990). The Development of Content-Valid Performance Measures for 76C Course and Field Assessment (Equipmant and Parts Specialist), US Army Institute for the Behavioral and Social Sciences. April 1990.

USING THE 76C TEST DEVELOPMENT AND ANALYSIS PROGRAM

Getting Started:

The first step in using this program involves either copying all the program and data files from the floppy disk to the dBASE III directory of your hard drive, to a subdirectory of the dBASE III directory, or simply setting the default drive for the dBASE III program to the floppy drive in which the program is located. Each of these procedures will be explained in the following sections.

Copying Files to the dBASE Directory

Copy the program files to your dBASE III directory as follows:

- (1) Boot your system and change directories to your dBASE III directory. CD\[name of directory]
- (2) Place the program disk in drive A:
- (3) Type copy A:*. * after the C prompt. C:Copy A:*. *

For example:

```
C:CD\dbase
C:Copy a:*. *
```

Each file name will appear on the screen as the copy process is carried out. When the files copied message appears you can remove the program disk from drive A and store it in a secure location.

Copying Files to a dBASE Subdirectory

Copy the program files to a dBASE III subdirectory using the following procedure:

- (1) Boot your system and change directories to your dBASE III directory. CD\<name of directory>
- (2) Create a subdirectory by typing the command MKDIR <directory name>
- (3) Change directories to the new directory. CD <name of directory>
- (4) Place the program disk in drive A:
- (5) Type copy A:*. * after the C prompt. C:Copy A:*. *

For example:

```
C:CD\DBASE
C:MKDIR PMAIN
C:CD PMAIN
C:COPY A:*. *
```

Each file name will appear on the screen as the copy process is carried out. When the files copied message appears you can remove the program disk from drive A and store it in a secure location.

Starting the Program:

Once the program and data files are either copied to the desired directory or placed in the B drive (when using dBASE from floppy disk) the user can start the program by doing the following:

- (1) Type DBASE at the C or A prompt. This will bring the dBase interpreter on line. A copyright message will appear and after a few seconds a dot (.) prompt will appear in the command line at the bottom of the screen.
- (2) This step depends on where your copy of the Test Development and Analysis program is located.
 - (a) If your program is in the dBASE directory type the command DO PMAIN to start the program.
 - (b) If your program is in a subdirectory then you will need to use the SET PATH command to tell the dBASE interpreter where the program and data files are located. This is accomplished by typing:

```
SET PATH TO \<dBASE directory>\<subdirectory>
```

For example, let's say your dBASE directory is called DB and the subdirectory is called PMAIN. You would set the path to the program files by typing:

```
SET PATH TO \DB\PMAIN
```

Once the path is set, the program is started by typing the command: DO PMAIN

- (c) If your program is on floppy disk and your dBASE interpreter is on the hard drive you must place the floppy disk in either drive A or B and then change the default drive name to that of the drive in which the program disk is located. This is accomplished by typing the command:

SET DEFAULT TO A: or SET DEFAULT TO B:

Once the default drive is set, the program is started by typing the command: DO PMAIN

Logging In For The First Time:

The initial security set up for this program uses the log-in code - SYSTEM with the password MANAGER. The person assigned as system manager for the program should immediately select the security option (S) from the main menu and change the initial security setup from SYSTEM and MANAGER to a personalized LOG-IN and PASSWORD and add the other personal information required by the security screen form.

WARNING: Do not change the capability code or you will not be able to get back into the security setup module once you log out of the program.

Once the initial security set up is modified, the access to the program can be limited by the security manager by issuing LOG-IN's, PASSWORDS and user capabilities as desired. Complete procedures for changing the initial security and creating new user access can be found in the section of this report that describes main menu option S.

Opening Screen Forms:

Screen 1

| |
|---|
| <p>WELCOME TO ARI'S TEST DEVELOPMENT AND ANALYSIS PROGRAM</p> <p>Written by Donald T. Fritz 7/90</p> <p>PLEASE LOG-IN: _____</p> |
|---|

Type your log-in code at the log-in prompt and press enter when your log-in is completed. The program will check its security file for your log-in code and if it can't be found a message will be displayed and you will be given another chance to enter your log-in code. If the correct code is not entered on the second trial the program will terminate. Once the correct log-in code has been entered the program will prompt you for your password.

Screen 2

WELCOME TO ARI'S TEST DEVELOPMENT AND ANALYSIS PROGRAM

Written by Donald T. Fritz 7/90

PLEASE LOG-IN:SYSTEM

PLEASE ENTER PASSWORD: _

The program will accept your password one letter at a time and replace that letter with an X immediately upon entry. You do not need to press the enter key; however, you must enter eight characters to complete the password. If extra characters are required simply press the space bar to supply the required number of extra characters. When the eighth character has been entered, the program will encode your password and check it against the password issued to you by the system manager. You have three chances to enter the correct password. If you do not enter the correct password on the third try, the program will revert back to screen 1 to give you a second complete try at logging in and giving your password. If the correct password is entered the main menu will appear as in screen 3 below.

Screen 3 - Main Menu

Enter Data:

- (A) New Test Plan
- (B) Student Test Results
- (C) Test Questions

Edit Data:

- (D) Modify Test Plan
- (E) Modify Student Test Data
- (F) Modify Test Questions

Review Data:

- (G) Student/Class Test Analysis
- (H) Course File Names
- (I) Test Plan matrix
- (J) Course Test File Names
- (K) Class Test Form

Print Data:

- (L) Student Performance Report
- (M) Student/Class Test Analysis
- (N) Course Test File Names
- (O) Test Bank
- (P) Class Test Form

Exit:

- (Q) Quit to Dbase
- (R) Quit to DOS

Security:

- (S) Change Log-in/Password
- (T) Create Test
- (U) Take Test

Enter Choice: Q

UNDERSTANDING THE MAIN MENU

The main menu is divided into six sections determined by the processes that are performed by the modules accessed from each section.

- | | |
|-------------|--|
| Section I | contains those options that deal with the creation of (A) Test Plans for a given course annex, (B) Student Performance data entry, and (C) Test Bank item entry. |
| Section II | contains those options that deal with the editing of data already contained in (D) the Test Plan file, (E) the Student Performance file and (F) the items already in the Test Item Bank file. |
| Section III | contains those options that allow the user to review data contained in or about (G) a Student/Class Test matrix analysis, (H) a Test plan, (I) the Test Item Bank file, (J) the course annex file names, and (K) a Class test item file. |
| Section IV | contains those options that allow the user to print hard copies of (L) a Student performance report, (M) a student performance analysis matrix, (N) Course Test file names, (O) the Test Item Question Bank, and (P) a pre-created test form to give to a class. |
| Section V | contains the exit options available to the user. (Q) exits the program to the dBASE III environment and (R) exits the program to the DOS environment. |
| Section VI | contains options to allow the user to (S) change his password or in the case of a user having system management capabilities to create new user access and assign user capabilities, (T) create a new test form from the test plan matrix, and (U) take a test that was created from the test plan matrix. |

Each of these options will be looked at in depth to explain their operation and use.

Option A - Enter Data : New Test Plan

Option A allows the user to store the labels for the test plan matrix rows and columns. The labels should be developed before using the program, since the program does not assist in the test plan development in any way except in allowing the instructor to try different arrangements of the matrix rows and columns. The test plan matrix will hold a maximum of six rows of job operations and a maximum of eight columns for job functions to produce a maximum of 48 cells.

The module will prompt the user for the following information:

- . Course Name. Example: 76C-AIT-PLL
- . Annex Letter. Example: B
- . Matrix Row Labels Example: Form Functions
- . Matrix Column Labels Example: Req. For Issue NSN

This information should be at hand when this module is going to be used. The actual prompt sequence is as follows:

Screen A.1

```
Enter Course number for the PLAN
you wish to CREATE [Q to quit] : _____
```

Type the name of the course and press enter.

Screen A.2

```
Enter Course number for the PLAN
you wish to CREATE [Q to quit] : 76C-AIT-PLL
```

```
Enter Annex Letter Code: ____
```


Type the annex letter and press enter. As this is done, the computer will check the test plan data file to see if a test plan for that course and annex already exists. If a plan is already in the data file this message will appear on the screen.

A Plan for that Course and Annex already exists. Would you like to try another course? Y/N: Y

The default value of the message option is YES. Therefore, all you need to do to enter another course number and annex is to press the enter key and screen A.1 will reappear to let you try again. If a test plan for the course-annex entered does not exist, the screen will clear and screen form A.3 will be displayed.

Screen A.3

| Test Plan Matrix Design Stage | |
|-------------------------------|-------|
| Row 1 Label: | _____ |
| Row 2 Label: | _____ |
| Row 3 Label: | _____ |
| Row 4 Label: | _____ |
| Row 5 Label: | _____ |
| Row 6 Label: | _____ |
| Column 1 Label: | _____ |
| Column 2 Label: | _____ |
| Column 3 Label: | _____ |
| Column 4 Label: | _____ |
| Column 5 Label: | _____ |
| Column 6 Label: | _____ |
| Column 7 Label: | _____ |
| Column 8 Label: | _____ |

Each of the Row Label fields can hold 28 characters and each of the Column Label fields can hold 21 characters. If your labels contain more characters than the field can hold, condense those labels by using clear abbreviations for terms that can be shortened. All of these fields are active and you can move from field to field at anytime by pressing the cursor keys on the keyboard. You can erase characters using the backspace key, insert characters using the insert key, type over characters using the insert key and delete characters using the delete key. Use the enter key to leave a data field. This will be true for all screen fields used in this program.

Now that you've filled in the fields you wish to use, you can skip through the rest by pressing the enter key or the down cursor key. As you complete or move through the last column field, the program will display this prompt:

Is everything correct Y/N/A(bort): N

You now have a chance to do one of three things

- . Correct any errors you may spot on your design stage by answering N to the prompt.
- . Abort the whole process by answering A to the prompt.
- . Accept the form as designed for the present by answering Y to the prompt.

Before responding to the prompt, read the screen carefully looking for typographical errors or content errors. If any errors are found press the enter key and the program will reactivate the Row and Column Label fields and you can correct any error you've spotted. If you've decided that you don't like the plan type an A and press enter to abort the plan. If you abort, none of the information supplied will be recorded in the data files. If everything seems OK at the moment, type a Y and press enter to see the actual test plan matrix (Screen A.4) that you have designed.

Screen A.4

Test Plan as Designed

| | | | | | | | | |
|--|--|--|--|--|--|--|--|--|
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |

Is everything correct Y/N/A(bort): N

You will again have the option of returning to screen A.3 and changing your design, accepting the design as is, or aborting the process completely. Once you have decided everything is the way you want it, respond yes to this prompt and your test plan will be recorded in the Test Plan data file PMAT.

Option B - Enter Data : Student Test Results

Option B allows the user to enter the student test performance results into a data file from which those results can be used to analyze student performance deficiencies or test performance qualities. To correctly use the B option you will need the following information:

- . Date of Test
- . Test form
- . Course Instructor
- . Class Number
- . Annex Letter
- . Student ID number
- . Question numbers of questions missed on the test.

Module B begins by identifying itself and asking if you wish to continue (Screen B.1).

Screen B.1

This is the student test data entry routine.
Do you wish to continue Y/N : Y

If you got here accidentally, you can now return to the main menu. Otherwise, press the enter key to continue on to screen B.2. Screen B.2 has two of the data entry fields filled in as default values. They are (1) the date of test field - contains the present date and needs to be changed to the test date by typing over the default date, and (2) the course instructor - contains the name of the person who logged in and if this person is not the instructor the name should be changed again by typing over the default name.

Screen B.2

| | | |
|--|----------------------|----------|
| Test Course and Class Data Entry | | |
| Date of Test: <u>08/09/90</u> | Test Form: __ | |
| Course Instructor: <u>Person who signed on</u> | | |
| Class Number: __ | Course Number: _____ | Annex: _ |

You also need to supply the test form, the class number, the course name and annex letter. As the last field is completed the program will display this prompt:

Is everything correct Y/N/A(bort): N

Read the screen form one more time before responding to be sure everything is correct because you will not get another chance to change any of this information. Type an A to abort the process.

When you are sure everything is correct and respond Y to the prompt, the screen will clear and display the student data entry form (Screen B.3).

Screen B.3

| | | |
|--|------------------------|--------------|
| Student ID: _____ | Date of test: 08/09/90 | Test Form: # |
| Course Instructor: Person entered in screen B.2 | | |
| Class Number: ## | Course Number: ##### | Annex: # |
| Test questions missed [Use two digits separated by a space Ex 03 08] | | |
| _____ | | |
| _____ | | |

This is the student performance data screen and only requires the entry of the student ID number and the question numbers that were answered incorrectly on the test. All the other information on the screen is carried over from screen B.2 . When the test question numbers are entered they should all consist of two digits. Put a zero in front of any question number less than 10 and type one and only one space after the number before typing the next number. A test performance entry screen might look like the following example screen when finished.

Sample Completed Screen B.3

| |
|---|
| Student ID:2346 Date of test:08/09/90 Test Form:A |
| Course Instructor: Doe, John |
| Class Number: 13 Course Number:76C-ALT Annex:B |
| Test questions missed [Use two digits separated by a space Ex 03 08] |
| 03 05 10 13 16 23 25 |

Is everything correct Y/N/A(bort): N

At this point be sure to recheck the student ID number and the question numbers entered before responding to the prompt. Double check the question numbers to be sure they begin on the first space of the field, contain two digits, and are separated by only one space. If this procedure is not followed the program will not be able to correctly analyze this student's performance against the test plan matrix.

A Yes response to this prompt will cause the next prompt to appear on the screen:

Do another student for this class? Y/N: Y

The default to this prompt is Yes to change it type a N and press enter. Once the response is entered this final prompt will appear:

Do you wish to enter student data from another class? Y/N: Y

Again the default is set to Yes. Type N to change the response and exit the program module.

Option C - Enter Data - Test Questions

Option C allows you to create a test item data bank and enter test items, or enter test items to an already existing test data bank. This is accomplished automatically by the module. All you have to do is to enter the Course number and annex letter. If a data bank already exists the program will open the file, determine the number of questions already in the data bank and add any new items to the end of the data file. If a data bank does not exist the computer will build one for you.

The information required to use this module consists of:

- . Course number
- . Annex Letter
- . Row and Column number of test plan cell to which the item belongs.
- . The test item itself

The data entry sequence begins with a welcome to the module and a prompt for the course number (Screen C.1).

Screen C.1

Welcome to the Test Question Entry Routine.

Course Number [Q to quit]: _____

Type the number or the course you wish to work with and press enter when finished.

Screen C.2

Welcome to the Test Question Entry Routine.

Course Number [Q to quit]: 76C-AIT-PLL

Annex Letter for Course: _ _ _

Enter the annex letter for the course and press the enter key. The module will now do two things. First, it will determine if a test plan for the course and annex entered in screen C.1 exists. If it does not a message to that effect will appear on the screen and you will be reminded that a plan must be entered

before you can create a test bank. Second, if a test plan exists the module will then determine if the test bank already exists or if it needs to be created. Once the determination is made the next screen (Screen C.3) form will be displayed.

Screen C.3

The course number of this test is #####.
The annex letter for this test is #.
Test question #
Enter the Row and Column number of this question _

The only response from you that is required is the row number and the column number from the test matrix that identifies the test plan cell to which the question belongs. Type the row number first followed by the column number. There is no need to press the enter key between entries. With the entry of the column number the screen will clear and the test item entry screen (Screen C.4) will appear.

Screen C.4

| Question # | Course: | Annex: | Row: | Column: |
|---------------------------------------|---------|--------|------|---------|
| | | | | |
| | | | | |
| | | | | |
| A. | | | | |
| | | | | |
| B. | | | | |
| | | | | |
| C. | | | | |
| | | | | |
| D. | | | | |
| | | | | |
| | | | | |
| Correct answer: _ | | | | |
| Job Operations/functions: #####/##### | | | | |

The first set of four - seventy five character - fields are supplied to hold the test item. If you do not need all four fields simply skip the ones not needed by pressing the enter key. The next four sets of two - seventy five character - fields are

supplied to hold the four multiple choice answers for the test item and as in the case of the question fields it is not necessary to use all the fields.

You can move from field to field in one of two ways. First, you can move by pressing the enter key. This will move you from the last character in the active field to the first character in the next field. Secondly, you can move from field to field or within a given field by using the cursor keys on the keyboard.

All the fields in this screen form are active as long as the cursor is within the boundaries of the form and the fields can be treated as if they were a word processing document where characters can be added using the insert key, or deleted by using the delete key, or whole lines removed using the backspace key.

Upon entry of the correct answer, the cursor will leave the entry screen area and the prompt

Is everything correct Y/N/A(bort): N

will appear. The responses are again (Y) for yes, (N) for no, and (A) for abort the operation. Assuming everything was correct, the response to the prompt will be Y. As soon as that response is entered the test item and possible responses will be recorded in the test item file and you will be given the next prompt.

Enter another question? Y/N: N

At this point you can continue adding items to the bank or stop for now and come back later to finish up.

Option D - Edit Data - Modify Test Plan

Option D is designed to allow you to correct any errors that you may find in your test plan matrix. When you use this routine be careful that you do not change any of the job operations or functions once you have started adding questions to the test item bank, for if you do, you will also have to go into the item bank and change the row and column numbers of the cell questions to match the new cell and column locations of that cell.

This module operates much like the Option A module except that the modification screen will have the old row and column labels as default values in the label entry screen instead of the fields being empty. Another difference is in the design of the screen. The modification screen shows the old test plan along with the modification fields.

The module begins with an identification of the module and a prompt for the course number as shown in screen D.1 below.

Screen D.1

```
Welcome to the test plan modification routine.  
Course number [? for list, Q to quit]: ?
```

This prompt defaults to a ?. If you don't know the name of the course test plan you wish to modify or you simply wish to see a list of the test plans that are recorded in the test plan file press the enter key and a list of recorded plans will appear on the screen and the execution sequence will return you to the course number prompt for the course number. If you wish to quit at this point, just type a Q in response to the prompt.

Entering a course number causes screen D.2 to appear.

Screen D.2

```
Welcome to the test plan modification routine.  
Course number [? for list, Q to quit]: 76C-AIT-PLL  
Annex Letter: _
```

If you enter a nonexistent course number and annex letter, a message to that effect will be displayed and you will be returned to the main menu. If a test plan for the course number and annex entered is found it will be displayed on the screen as shown in screen D.3 .

Screen D.3

Test Plan Modification

| | Label | label | label | label | label | label | label | label |
|--------------|-----------------|-------|-------|-------|-------|-------|-------|-------|
| | | | | | | | | |
| | Row 1 Label: | | | | | | | |
| <u>Label</u> | Row 2 Label: | | | | | | | |
| | Row 3 Label: | | | | | | | |
| | Row 4 Label: | | | | | | | |
| <u>Label</u> | Row 5 Label: | | | | | | | |
| | Row 4 Label: | | | | | | | |
| | Row 5 Label: | | | | | | | |
| <u>Label</u> | Row 6 Label: | | | | | | | |
| | Column 1 Label: | | | | | | | |
| | Column 2 Label: | | | | | | | |
| <u>Label</u> | Column 3 Label: | | | | | | | |
| | Column 4 Label: | | | | | | | |
| | Column 5 Label: | | | | | | | |
| <u>Label</u> | Column 6 Label: | | | | | | | |
| | Column 7 Label: | | | | | | | |
| <u>Label</u> | Column 8 Label: | | | | | | | |

The labels for the rows and columns will be displayed in the center of the matrix and the cursor will be positioned at the first row label name. You can move through the labels using the cursor keys to make corrections. When you move out of the field for column 8 label this prompt will appear:

Is everything correct Y/N/A(bort): N

A no answer to this prompt will reposition the cursor at the first row label. A yes answer will cause the screen to clear and the new modified test plan matrix will be displayed as shown in screen D.4

Screen D.4

Test Plan Modification

| | Label | label | label | label | label | label | label | label |
|--------------|-------|-------|-------|-------|-------|-------|-------|-------|
| | | | | | | | | |
| <u>Label</u> | | | | | | | | |
| <u>Label</u> | | | | | | | | |
| <u>Label</u> | | | | | | | | |
| <u>Label</u> | | | | | | | | |
| <u>Label</u> | | | | | | | | |
| <u>Label</u> | | | | | | | | |

Is everything correct Y/N/A(bort): N

If everything is now the way you want it, respond by typing a Y, if it is not correct simply press the enter key and the row and column labels will again be displayed and modifications to them may be made.

Option E - Modify Data : Student Test Data

Option E allows the user to modify the student test performance results in the class data file from which those results can be used to analyze student performance deficiencies or test performance qualities.

Module E begins by identifying itself and asking for the course number as in the following screen display.

Screen E.1

```
Student Test Performance Data Modification
Course Number: [? for list, Q to quit:]: ?
```

If you got here accidentally, you can now return to the main menu by entering a Q in response to the prompt. If you do not know the course number, press the enter key and a list of those courses for which tests have been given will be displayed on the screen. If you already know the course number, enter the number and press the enter key. The screen will now display a prompt for the annex letter, class number and test form. (Screen E.2)

Screen E.2

```
Student Test Performance Data Modification
Course Number: [? for list, Q to quit:]: ?
```

```
Annex Letter:___ Class Number:___ Test Form:___
```

Enter the requested information to advance to screen E.3.

Screen E.3

Student Test Data Modification

Date modified: 08/09/90

Course Instructor: Person who signed on

Class Number: Student ID [? for list,Q to quit]:?

As in screen E.2, if you do not know the student number, press enter and a list of student numbers in the requested class will be displayed. As the last field is completed the program will display this prompt:

Is everything correct Y/N: N

Read the screen form one more time before responding to be sure everything is correct because you will not get another chance to change any of this information.

When you are sure everything is correct and respond Y to the prompt, the screen will clear and display the student data entry form (Screen E.4).

Screen E.4

Student ID:##### Date of test:08/09/90 Test Form:#

Course Instructor: Person entered in screen B.2

Class Number: ## Course Number:##### Annex:#

Test questions missed [Use two digits separated by a
space Ex 03 08]

03 05 08 10 16 24 _____

This is the student performance data screen and only allows the user to modify the question numbers that were answered incorrectly on the test. All the other information on the screen is carried over from screen E.3 . When the test question numbers are entered they should all consist of two digits. Put a zero in front of any question number less than 10 and type one and only one space after the number before typing the next number. When you exit the data field this prompt appears.

Is everything correct Y/N/A(bort): N

At this point be sure to recheck the question numbers entered before responding to the prompt. Double check the question numbers to be sure they begin on the first space of the field, contain two digits, and are separated by only one space. If this procedure is not followed the program will not be able to correctly analyze this student's performance against the test plan matrix.

A Yes response to this prompt will cause the next prompt to appear on the screen:

Do you wish to modify another student's data? Y/N: Y

The default to this prompt is Yes to change it type a N and press enter. You will now return to the main menu screen.

Option F - Modify Data - Modify Test Questions

Option F allows you to modify a test item that is stored in the test data bank. All you have to do is to enter the Course number and annex letter.

Screen F.1

Welcome to the Test Question Entry Routine.

Course Number [Q to quit]._____

Type the number of the course you wish to work with and press enter when finished.

Screen F.2

Welcome to the Test Question Entry Routine.

Course Number [Q to quit]: 76C-AIT-PLL

Annex Letter _____

Question Number: _____

To delete a question change its row number to zero!

Enter the annex letter for the course and press the enter key. Enter the question number of the item you wish to modify. This will cause screen F.3 to appear. If you want to delete this question from the data file simply change the row number to a zero. This process does not physically remove the question from the data bank, but simply removes it from possibility of selection when you make a test form from the item bank. You could reinstate the question at some later date by changing the row number back from zero to a real row number. (See screen F.3)

Screen F.3

Question # Course: Annex: Row: Column:

A.

B.

C.

D.

Correct answer:

Job Operations/functions: @@@@@@@@/@@@@@@@

You can move from field to field in one of two ways. First, you can move by pressing the enter key. This will move you from the last character in the active field to the first character in the next field. Secondly, you can move from field to field or within a given field by using the cursor keys on the keyboard.

All the fields in this screen form are active as long as the cursor is within the boundaries of the form and the fields can be treated as if they were a word processing document where characters can be added using the insert key, or deleted by using the delete key, or whole lines removed using the backspace key.

Upon entry of the correct answer, the cursor will leave the entry screen area and the prompt.

Is everything correct Y/N/A(bort): N

will appear. The responses are again (Y) for yes, (N) for no, and (A) for abort the operation. Assuming everything was correct, the response to the prompt will be Y. As soon as that response is entered the test item and possible responses will be recorded in the test item file and you will be given the next prompt.

Modify another question? Y/N: N

At this point you can continue modifying items in the bank or return to the main menu.

Option G - Review Data - Student/Class Test Analysis

Option G is designed to allow the user to review a student's test performance in reference to the test plan from which the test was created. This is accomplished by determining which cell of the test plan any incorrect question responses belong in and once all incorrect responses are mapped to their cells, displaying the test plan matrix and the number of student incorrect responses for each cell. The module begins by displaying screen G.1 .

Screen G.1

| |
|--|
| <div>Test Performance Matrix Review</div> <div>A. Review Individual Student Performance B. Review Class Performance C. Return to Main Menu</div> <div>Enter Choice: <u>G</u></div> |
|--|

Option C from this menu return you to the main menu. Option A will display the analysis of an individual student. Responding with either A or B to the above prompt will cause screen G.2 to appear.

Screen G.2

| |
|--|
| <div>Class Test Performance Review</div> <div>Course Number [? for list, Q to quit]: ?</div> |
|--|

Press enter if you don't know the course number and a list will be supplied. Type the course number in response to the prompt and screen G.3 will appear.

Screen G.3

| | |
|---|-------------------------------|
| <table border="1"><tr><td>Class Test Performance Review</td></tr></table> | Class Test Performance Review |
| Class Test Performance Review | |
| Course Number [? for list, Q to quit]: 76C-AIT-PLL | |
| | |
| Annex Letter:___ Class Number:___ Test form:___ | |

Supply the annex letter, class number and test form to continue. This will bring on screen G.4 .

Screen G.4

| | |
|---|-------------------------------|
| <table border="1"><tr><td>Class Test Performance Review</td></tr></table> | Class Test Performance Review |
| Class Test Performance Review | |
| Course Number [? for list, Q to quit]: 76C-AIT-PLL | |
| Annex Letter: B Class Number: 13 Test Form: A | |
| | |
| Student ID Number [? for list]: ? | |

Enter the student number to begin the analysis. The program will locate that student's performance data and extract that data while displaying the following:

Performance for student #####

01 05 07 10 13

Analyzing

When the analysis is complete the screen will clear and the test plan matrix with student response analysis will be displayed. (Screen G.5)

Screen G.5

Test analysis for Student #### Course 76C-AIT-PLL Annex B _

| | Label | label | label | label | label | label | label | label |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|
| | | | | | | | | |
| <u>Label1</u> | | 1 | | | | | | |
| <u>Label2</u> | | 1 | | 1 | | | | |
| <u>Label</u> | | | | | | | | |
| <u>Label2</u> | | | | 1 | | 1 | | |
| <u>Label</u> | | | | | | | | |
| <u>Label1</u> | | | | | | 1 | | |
| | | 2 | | 2 | | 2 | | |

Another: Y

Each number in a cell represents an incorrect question answer. The number at the beginning of the row indicates the total number incorrect in each row and the numbers at the bottom of the matrix represent the number of incorrect responses in each column. The matrix display will remain on the screen until a Y or N response is made to the prompt Another: Y displayed at the lower left corner of the screen.

The same basic procedure is used to display a whole class's response to the test questions.

Option H - Review Data - Test Plan Matrix

Option H is designed to allow the user to review any test plan matrix that has been developed and stored in the data files. If this option is used immediately after the plan is entered, all that will be displayed is a blank test plan matrix with the row and column labels in place; however, if the option is used after the test item bank has been filled, the test plan matrix will be displayed with numeric values in each of the cells. These values represent the total number of test items that are contained in the test item bank for each cell of the matrix.

This module starts by displaying screen H.1.

Screen H.1

| |
|---|
| <div>Welcome to the Test Matrix Review</div> <div>(P) Send Matrix to Printer (S) Send Matrix to Screen</div> <div>Where do you want it displayed? P/S: <u>S</u></div> |
|---|

This prompt is set to default to the screen. If you want to obtain a hard copy of the test plan matrix respond to the prompt with a P and press enter. For a screen display just press enter.

Screen H.2

| |
|--|
| Course Number [? for list, Q to quit]: ? |
|--|

Enter the course number for which you wish to see the test plan. Like the other modules, if you press enter with no response a list of course test plans will appear on the screen. The next prompt will ask for the annex letter of the test plan. (Screen H.3)

Screen H.3

Course Number [? for list, Q to quit]: ?

Annex Letter: __

Type in the annex letter for the course and the test plan matrix will appear on what ever device you selected from the opening screen prompt.

Screen H.4

Test Plan Matrix for Course 76C-AIT-PLL Annex B _

| | Label | label | label | label | label | label | label | label |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|
| | 2 | 1 | 4 | 2 | 3 | 6 | 5 | 3 |
| <u>Lab.26</u> | 3 | 1 | 4 | 1 | 3 | 3 | 4 | 2 |
| <u>Lab.21</u> | 5 | 2 | 4 | 4 | 4 | 3 | 3 | 3 |
| <u>Lab.28</u> | 3 | 4 | 5 | 1 | 2 | 1 | 5 | 4 |
| <u>Lab.25</u> | 4 | 4 | 4 | 3 | 1 | 3 | 6 | 5 |
| <u>Lab.30</u> | 5 | 4 | 5 | 2 | 3 | 1 | 3 | 2 |
| <u>Lab.25</u> | | | | | | | | |

Numbers indicate the number of questions in test bank for each row or cell C

When you have finished reviewing the test plan press the enter key to return to the main menu.

Option I - Review Data - Test Item Bank

Option I is designed to allow the user to browse through the test item bank file to look for errors in typing or answer selections. The module begins with the following screen.

Screen I.1

| |
|--|
| <p>Welcome to the Test Bank Review Routine</p> <p>Course Number [? for list, Q to quit]:</p> |
|--|

Enter the course number of the test bank you wish to review and press enter. Screen I.2 will appear.

Screen I.2

| |
|--|
| <p>Welcome to the Test Bank Review Routine</p> <p>Course Number [? for list, Q to quit]: 76C-AIT-PLL</p> <p>Annex Letter: __</p> |
|--|

Enter the annex letter and press enter. Screen I.3 will appear. With the question, answer selections and correct answer displayed on the screen.

Screen I.3

Question # Course: Annex: Row: Column:

A.

B.

C.

D.

Correct answer:

Job Operations/functions: eeeeeeeee/eeeeeeee

Press enter to continue, A to abandon: C

If you type an A and press enter, the browse will stop and this prompt will appear.

Browse another test: Y/N: Y

Option J - Review Data - Course Test File Names

Option J allows you to browse through the test file name data file. The module begins by displaying screen J.1.

Screen J.1

| |
|---|
| <div>Class File Names Browse Routine</div> <div>Do You Wish to Browse the Class Name File Y/N: <u>Y</u></div> |
|---|

To browse the class name file respond by pressing the enter key. Otherwise enter an N and then press enter. The file browse routine prints a series of column headings then displays the contents of the class name file (Plaslist.dbf).

Screen J.2

| Course Number | Annex | Class Number | Test Form | Test Bank File | Class Results File | Class Test File |
|------------------|-------|-----------------|--------------|-------------------|-----------------------|--------------------|
| 76C-AIT-PLL B | | 13 | A | TB5363 | TS4053 | TS4305 |

Press any key to continue. Y

The class names will list until the screen fills. Once the screen is full just press the enter key to continue the browse. A message will appear when the end of file is reached.

Option K - Review Data - Class Test Form

Option K is exactly like option I except it allows the user to review a class test form rather than the test item data file. It begins with screen K.1.

Screen K.1

Welcome to the Class Test Review Routine

Course Number [? for list, Q to quit]:

Enter the course number of the test bank you wish to review and press enter. Screen K.2 will appear.

Screen K.2

Welcome to the Test Bank Review Routine

Course Number [? for list, Q to quit]: 76C-AIT-PLL

Annex Letter: __

Enter the annex letter and press enter. Screen K.3 will appear. With the question, answer selections and correct answer displayed on the screen.

Screen K.3

| Question # | Course: | Annex: | Row: | Column: |
|--|---------|--------|------|---------|
| | | | | |
| | | | | |
| | | | | |
| A. | | | | |
| | | | | |
| B. | | | | |
| | | | | |
| C. | | | | |
| | | | | |
| D. | | | | |
| | | | | |
| | | | | |
| Correct answer: <u> </u> | | | | |
| Job Operations/functions: @@@@@@@@/@@@@@@@ | | | | |

Press enter to continue, A to abandon: C

If you type an A and press enter, the browse will stop and this prompt will appear.

Browse another test: Y/N: Y

Option L - Print Data - Student Test Performance

Option L is designed to allow you to obtain a hard copy of either an individual's test performance in a verbal format rather than in test plan form. The module begins by displaying screen L.1 .

Screen L.1

Student Test Performance Review

- (A) On Screen
- (B) On screen and Hard Copy
- (C) Hard Copy Only
- (D) Return to Main Menu

Enter Choice: L

Select where you'd like the performance report displayed and type that letter, then press enter. Screen L.2 will appear.

Screen L.2

Course Number [? for list, Q to quit]: ?

Enter the course number and screen L.3 will appear.

Screen L.3

| |
|---|
| <p>Student Test Performance Review</p> |
| <p>Course Number [? for list, Q to quit]: 76C-AIT-PLL</p> |
| <p>Annex Letter: __ Class Number: __ Test form: __</p> |

Enter the annex letter, class number and test form and the next screen will appear.

Screen L.4

| |
|---|
| <p>Class Test Performance Review</p> |
| <p>Course Number [? for list, Q to quit]: 76C-AIT-PLL Annex Letter: B Class Number: 13 Test Form: A</p> |
| <p>Student ID Number [? for list]: ?</p> |

Enter the student number to begin the analysis. The program will locate that student's performance data and extract that data while displaying the following:

Screen L.5

Performance of Student Number #####

Questions Missed: ## ## ## ## ## ##

Analyzing

- Job opportunity/function

- " " " " "

Press any key to continue.

Option M - Print Data - Student/Class Test Analysis

Option M is designed to allow the user to send a review of a student's test performance in reference to the test plan from which the test was created to the printer. This is accomplished by determining which cell of the test plan any incorrect question responses belong in and once all incorrect responses are mapped to their cells, displaying the test plan matrix and the number of student incorrect responses for each cell. The module begins by displaying screen M.1 .

Screen M.1

| |
|-------------------------|
| What Type Printer |
| (L) Laser |
| (D) Dot matrix/graphics |
| Choice: <u>L</u> |

Select the type printer you wish to use and press enter.
Screen M.2 will next appear.

Screen M.2

| |
|--|
| Test Performance Matrix Review |
| A. Review Individual Student Performance |
| B. Review Class Performance |
| C. Return to Main Menu |
| Enter Choice: <u>G</u> |

Option C from this menu return you to the main menu. Option A will display the analysis of an individual student. Responding with either A or B to the above prompt will cause screen M.3 to appear.

Screen M.3

| |
|--|
| <div>Class Test Performance Review</div> <div>Course Number [? for list, Q to quit]: ?</div> |
|--|

Press enter if you don't know the course number and a list will be supplied. Type the course number in response to the prompt and screen M.4 will appear.

Screen M.4

| |
|---|
| <div>Class Test Performance Review</div> <div>Course Number [? for list, Q to quit]: 76C-AIT-PLL</div> <div>Annex Letter:___ Class NUmber:___ Test form:___</div> |
|---|

Supply the annex letter, class number and test form to continue. This will bring on screen M.5 .

Screen M.5

Class Test Performance Review

Course Number [? for list, Q to quit]: 76C-AIT-PLL
Annex Letter: B Class Number: 13 Test Form: A

Student ID Number [? for list]: ?

Enter the student number to begin the analysis. The program will locate that student's performance data and extract that data while displaying the following:

Performance for student #####

01 05 07 10 13

Analyzing

When the analysis is complete the screen will clear and the test plan matrix with student response analysis will be displayed on the selected printer.

The same basic procedure is used to display a whole class's response to the test questions.

Option N - Print Data - Course Test File Names

Option N allows you to browse through the test file name data file. The module begins by displaying screen N.1.

Screen N.1

| |
|---|
| <div><div>Class File Names Print Routine</div><div>Do You Wish to Print the Class Name File Y/N: <u>Y</u></div></div> |
|---|

To print the class name file respond by pressing the enter key. Otherwise enter an N and then press enter. The file print routine prints a series of column headings then displays the contents of the class name file (Plaslist.dbf).

Screen N.2

| Course Number | Annex | Class Number | Test Form | Test Bank File | Class Results File | Class Test File |
|------------------|-------|-----------------|--------------|-------------------|-----------------------|--------------------|
| 76C-AIT-PLL B | | 13 | A | TB5363 | TS4053 | TS4305 |

Press any key to continue. Y

Option 0 - Print Data - Test Item Bank

Option 0 is designed to allow the user to print a hard copy of the test item bank file to look for errors in typing or answer selections. The module begins with the following screen.

Screen 0.1

Welcome to the Test Bank Print Routine

Course Number [? for list, Q to quit]:

Enter the course number of the test bank you wish to review and press enter. Screen 0.2 will appear.

Screen 0.2

Welcome to the Test Bank Print Routine

Course Number [? for list, Q to quit]: 76C-AIT-PLL

Annex Letter: ____

Enter the annex letter and press enter. The contents of the test item bank will be sent to the printer.

Option P - Print Data - Class Test Form

Option P is designed to allow the user to print a hard copy of a class test form file to look for errors in typing or answer selections. The module begins with the following screen.

Screen P.1

Welcome to the Class Test Form Print Routine

Course Number [? for list, Q to quit]:

Enter the course number of the test bank you wish to review and press enter. Screen P.2 will appear.

Screen P.2

Welcome to the Class Test Form Print Routine

Course Number [? for list, Q to quit]: 76C-AIT-PLL

Annex Letter: Test Form:

Enter the annex letter and press enter. The contents of the test item bank will be sent to the printer.

Option S - Security

Option S is designed to work in two modes depending of the assigned capability of the user. The first mode is available to all users except the system manager. In this mode the user can change his own password any time he wants to. The second mode is available only to the system manager and allows him to enter new user names, log-in's, passwords and capabilities. It also allows the manager to browse the audit trail and print hard copies of the audit trail. Let's look at the first mode operation.

Mode 1:

In order to change your password, select option S from the main menu. The first screen to appear will be Screen S.1.

Screen S.1 Mode 1

| |
|--------------------|
| Enter Log-in Code: |
|--------------------|

Type your log-in code and press enter. Screen S.2 will next appear.

Screen S.2 Mode 1

| | |
|----------------------------|-------|
| User Password Setup Module | |
| User Name: | |
| Log in: | Rank: |
| Password: | |

The user name, log-in and rank will be displayed as stored in the data file. The only active field will be the password field. Enter the new password in the password field and press enter. The new password will be stored in the security file and will become active the next time you sign on the system.

Mode 2:

Entry into security mode 2 requires system manager capabilities. This mode begins with the following screen:

Screen S.1 Mode 2

| |
|--|
| <p>Program Security Module</p> <p>(A) Add New User (B) Browse/Print Audit Trail (C) Edit User Information (D) Return to Main Menu Enter Choice: <u>D</u></p> |
|--|

Option A from this menu allows the addition of new members to the security file. This option begins and ends with screen S.A.1.

Screen S.A.1

| |
|--|
| <p>User Password/Capability Set Up Module</p> <p>User Name:</p> <p>Log in: Rank:</p> <p>Password: Capability:</p> <p> [A - Limited Data Entry] [B - Data and Reports] [C - Take Test only]</p> |
|--|

Entry of a new user to the system requires the following input:

- . Users full name.
- . Users assigned log-in code.
- . Users Rank.
- . Users Password.
- . Users Capability [A|B|C]

The user's capability assignment will determine which of the main menu options the user will be able to access. A user with capability code A will only be allowed to access options B,C,F,I, and U. A user with capability code B can access all options with the exception of Security Mode 2. A user with capability code C can only access option U. The system Manager can access all the options plus security mode 2; however, the system manager cannot access security mode 1. Access to this mode by the system manager is unnecessary, since the system manager can change passwords in mode 2.

Option B from the security menu allows the system manager to browse through or print the contents of the audit trail file. The module opens with the following screen.

Screen M.5

| | |
|--|----------------------------------|
| <table border="1"><tr><td>Audit Trail Browse/Print Routine</td></tr></table> | Audit Trail Browse/Print Routine |
| Audit Trail Browse/Print Routine | |
| (A) Browse Audit Trail (B) Print Audit Trail (C) Return to Main Menu | |
| Enter Choice: <u>S</u> | |

Selection of option A from this menu will switch the program into dBASE III Browse mode. The directions for moving through the browse mode are located at the top of the screen. To exit from the browse mode you must hold down the Ctrl key and press the End key on the number pad.

Selection of option B from this menu will send a copy of the contents of the audit trail file to the printer.

Option C of the security menu will allow you to modify the information of any user found in the security file. It starts and ends with the following screen.

Screen S.C.1

| |
|----------------------|
| Enter Log-in Code: ? |
|----------------------|

Enter the log-in code of the user you wish to modify. If you don't know the user's log-in code press enter for a list of log-ins and users. Once the user log-in is entered screen S.C.2 will appear with the present user information as the default values.

Screen S.C.1

| | |
|--|----------------------------|
| User Password/Capability Set Up Module | |
| User Name: | |
| Log in: | Rank: |
| Password: | Capability: |
| | [A - Limited Data Entry] |
| | [B - Data and Reports] |
| | [C - Take Test only] |

Make the required modifications by typing over, inserting or deleting data as desired. When modification is complete and you move through the capability field, the following prompt will appear:

Is it Correct Y/N: N

A yes response will record the modifications in the security data file.

Option T - Security - Create Test

Option T will allow the user to produce a selectively randomized test from a selected test item bank. The module begins by displaying the following screen:

Screen T.1

```
Welcome to the Class Test Maker Routine  
Course Number [? for list, Q to quit]:
```

Enter the course number for which you wish to build the test. Press enter if you need a list of course test banks available. Screen T.2 will appear once the course number is entered.

Screen T.2

```
Welcome to the Class Test Maker Routine  
Course Number [? for list, Q to quit]:
```

Annex Letter:
Select questions by (R)ow or (C)ell : R

Enter the annex letter and the method you wish to use to select your questions - by row or by cell. If by row (R) is selected, the program will display the test plan matrix for the course selected with the number of questions available in each row and a highlighted field to enter the number of questions you want on the test for each row. If by cell (C) is selected, the program will display the test plan matrix with the number of test items available for each cell and a highlighted field to enter the number of questions you wish from each cell.

A set of instructions will be displayed for you to read while the program prepares the files from which the test will be created. When the prompt - press any key to continue - appears at the bottom of the instruction screen, the program is ready to proceed. Screen T.3 is the instruction screen you will see.

Screen T.3

HOW TO MAKE A TEST

Creating a test from your test plan and test question bank is a fairly simple process. While you are reading these instructions the computer is accessing your test plan and loading the value representing the number of questions available in the test bank either for each row or for each individual matrix cell depending on the choice you made to the prompt - Select questions by (R)ow or (C)ell. If you selected by row, you will see a numeric value printed at the end of the top label for each row., under which you will see a highlighted 0. If you selected by cell, you will see the numeric value at the top center of each cell and a highlighted 0 below each value at the bottom center of each cell. The upper value tells you the total number of questions available for each row or cell. You are to enter the number of questions you want on the test from that row or cell. Use the cursor arrows or enter key to move from highlighted field to highlighted field as you fill in the values. When you complete the last field, the computer will build your test from this input.

Press any key to continue..

Press any key to clear the instruction screen and the matrix form will then appear as in screen T.4.

Screen T.4

Test Plan for Course 76C-AIT-PLL Annex B _

| | Label | label | label | label | label | label | label | label |
|----------------|-------|-------|-------|-------|-------|-------|-------|-------|
| | 5 | 6 | 3 | 8 | 7 | 8 | 4 | 9 |
| <u>Label11</u> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 4 | 4 | 4 | 5 | 6 | 6 | 7 | 6 |
| <u>Label12</u> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 5 | 5 | 7 | 4 | 8 | 8 | 8 | 8 |
| <u>Label1</u> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 6 | 6 | 6 | 6 | 6 | 6 | 7 | 5 |
| <u>Label12</u> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 4 | 4 | 5 | 6 | 7 | 8 | 6 | 5 |
| <u>Label1</u> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 8 | 8 | 7 | 5 | 6 | 7 | 6 | 8 |
| <u>Label11</u> | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

After you've entered all the question number values the program will automatically produce the test from those values and will display its progress on the screen while it is accomplishing its task. The test form is automatically determined from previous data. You can tell what test form and file name the program is using by recording the number that appears near the progress screen. The created test form can be printed from option P.

Option U - Security - Take Test

Option U will allow you to take any course test for which a test form has been created. This module begins with screen U.1.

Screen U.1

Welcome to the 76C-AIT Automated Testing Program

Course Number [Q to quit]:

Enter the course number for which you'd like to take a test. Enter Q to return to the main menu. As you supply the course number screen U.2 will appear.

Screen U.2

Welcome to the 76C-AIT Automated Testing Program

Course Number [Q to quit]:

Annex Letter: _____

Class Number: _____

Test Form: _____

Student ID: _____

The class number requested on this screen is the number used by the instructor to designate the class in which you are registered. The student ID is the number used to identify yourself.

Enter all requested information. As the student ID field is completed, the screen will clear and the first test question will appear. (Screen U.3)

Screen U.3

Question # Course: Annex: Test Form:

| | |
|--------------------|--|
| | |
| | |
| | |
| A. | |
| | |
| B. | |
| | |
| C. | |
| | |
| D. | |
| | |
| | |
| Correct answer: __ | |

As you answer each question, the computer will display the next question until the test is completed. When you have answered all the questions the screen will clear and screen U.4 will be displayed showing your performance on the test.

Screen U.4

| | |
|-------------------------------|-------|
| Test Results | |
| Number of questions incorrect | = ### |
| Number of questions correct | = ### |

Press any key to continue...

To end the test taking routine press any key and you will return to the main menu. Select option R from the main menu to completely leave the program.