

AD-A140 709

PROJMG FORTRAN: AN INTERACTIVE COMPUTER PROGRAM FOR
USE WITH THE DEFENSE MANAGEMENT SIMULATION EXERCISE(U)
NAVAL POSTGRADUATE SCHOOL MONTEREY CA G W SCHULTZ

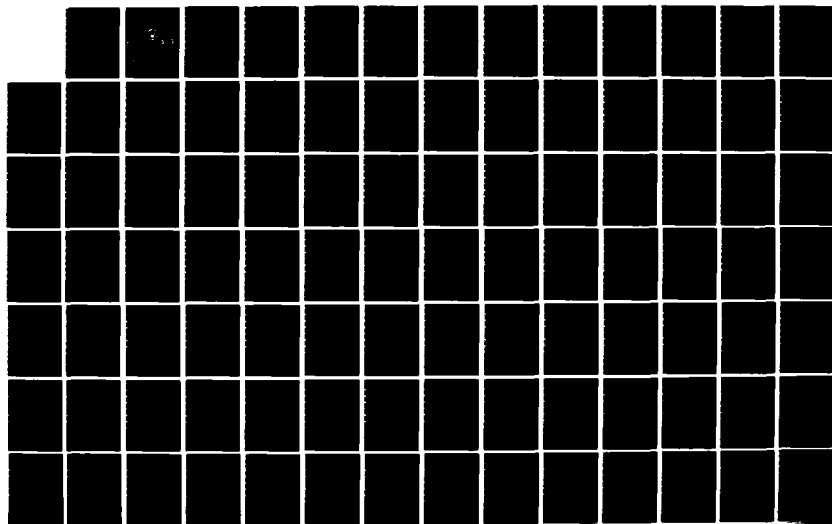
1/4

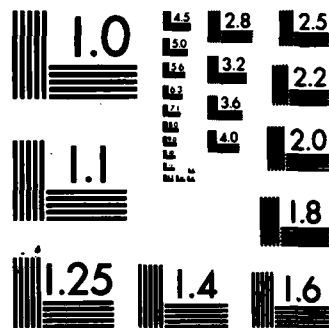
UNCLASSIFIED

MAR 84

F/G 9/2

NL





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

2

NAVAL POSTGRADUATE SCHOOL

Monterey, California

AD-A140 709



DTIC
ELECTE
MAY 4 1984
S B D

THESIS

DTIC FILE COPY

PROJMNG FORTRAN:
AN INTERACTIVE COMPUTER PROGRAM FOR USE WITH
THE DEFENSE MANAGEMENT SIMULATION EXERCISE

by

George W. Schultz

March, 1984

Thesis Advisor:

M. B. Kline

Approved for public release, distribution unlimited

84 05 03 C48

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. A140 709	3. REPORTING ORGANIZATION'S CATALOG NUMBER
4. TITLE (and Subtitle) PROJMNG FORTRAN: An Interactive Computer Program for Use with the Defense Management Simulation Exercise		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis March, 1984
7. AUTHOR(s) George W. Schultz		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		8. CONTRACT OR GRANT NUMBER(s)
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE March, 1984
		13. NUMBER OF PAGES 323
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release, distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Project Management Training Contract Negotiation Acquisition Management Training Computer Aided Training Defense Management Simulation PROJMNG Contract Negotiation Training		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) Contract Negotiation Package (CNP), the supporting computer program for the Defense Management Simulation (DMS), is revised and embedded into a program which makes it user-friendly, and which provides sensitivity analysis capa- bility to it. The program includes a plotting function for the sensitivity analysis. Exercise records are established for review of contracting team performance. Database files are generated which permit teams to submit reports, which provide a baseline for subsequent game sessions, and which		

permit monitor evaluation of team progress.

The text provides a description for the operation of both CNP and PROJMG. It documents the new program, PROJMG.

Appendices include the Fortran code, and Conversational Monitor System (CMS) executive machine language programs for the new programs operation. It contains instruction manuals which depict operation for both programs.



Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
PER CALL JC	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

Approved for public release; distribution unlimited.

**PROJING FORTRAN:
An Interactive Computer Program for Use with
the Defense Management Simulation Exercise**

by

George W. Schultz
Lieutenant Commander, United States Navy
E.S.E., University of New Mexico, 1971

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY
(COMMAND, CONTROL AND COMMUNICATIONS)**

from the

NAVAL POSTGRADUATE SCHOOL
March 1984

Author:

George W. Schultz

Approved by:

Melvin B. Klue

Thesis Advisor

Robert A. Pally

Second Reader

Michael A. Ferguson

Chairman, Command, Control and Communications Academic Group

W. H. Murray

Academic Dean

ABSTRACT

Contract Negotiation Package (CNP), the supporting computer program for the Defense Management Simulation ~~(DMS)~~, is revised and embedded into a program which makes it user-friendly, and which provides sensitivity analysis capability to it. The program includes a plotting function for the sensitivity analysis. Exercise records are established for review of contracting team performance. Database files are generated which permit teams to submit reports, which provide a baseline for subsequent game sessions, and which permit monitor evaluation of team progress.

The text provides a description for the operation of both CNP and PROJHNG. It documents the new program, PROJHNG.

Appendices include the Fortran code, and Conversational Monitor System (CMS) executive machine language programs for the new programs operation. It contains instruction manuals which depict operation for both programs.

TABLE OF CONTENTS

I.	INTRODUCTION	12
	A. HISTORICAL BACKGROUND	12
	B. SCENARIO	13
	C. SIMULATION	15
	D. SUMMARY	16
II.	CNP OPERATION	18
	A. PROCEDURES	18
	B. BENEFITS	22
	C. DEFICIENCIES	23
	D. MONITOR FUNCTIONS	25
	E. SUMMARY	26
III.	PROJNG OPERATION	27
	A. CNP MODERNIZATION	27
	B. PROJNG PACKAGE	28
	1. Programs	28
	2. Data files	29
	3. LINK	30
	4. PROJNG EXEC	31
	5. PROFILE EXEC	32
	C. DATABASE SECURITY	33
	D. EXIT	34
	1. Fatal Errors	35
	2. Security Violation	36
	3. Menu	36
	4. YES/NO Query	36
	5. Return to PROJNG EXEC	36
	E. PROJNG PROCEDURES	37

1.	Instructions	37
2.	Link	37
3.	PROJMNG EXEC Execution	38
4.	Running PROJMNG	40
F.	DATABASE CHANGE	41
G.	MENU	42
H.	REPORT SUBMISSION	44
1.	Progress Reports	44
2.	Decision Submission	46
I.	CALCULATION CORRECTION	51
J.	SUMMARY	53
IV.	PROJMNG MONITOR OPERATION	54
A.	CNP MONITOR UNIQUE FUNCTIONS	54
B.	PROJMNG MONITOR UNIQUE FUNCTIONS	56
C.	MONITOR UNIQUE SECURITY SYSTEM	56
D.	QUERY SUPPRESSION FOR THE MONITOR	57
E.	MONITOR'S MAIN MENU	59
1.	Change Team Number	59
2.	The DP Selection Query	63
3.	The Input Selection Menu	63
4.	Rerun the Data Calculation	64
5.	Do a Sensitivity Analysis	64
6.	Printout Cost + Factors	64
7.	EXIT	64
8.	Flexible Menu	65
F.	MONITOR SESSION PROCEDURES	65
G.	INITIALIZATION	66
1.	PROJMNG EXEC Operation	66
2.	Available Files	66
3.	Purging DATAFILES	67
H.	BASIC PROGRAM ROUTINES	69
1.	Start of PROJMNG's Run	70
2.	Summary of Basic Capabilities	74

I.	SENSITIVITY ANALYSIS ROUTINES	74
1.	Sensitivity Analysis Basic Menu	
	Options	74
2.	Sensitivity Analysis Table	78
3.	Menu Expansion	78
4.	Sensitivity Analysis Plotting	79
5.	Effectiveness Index	86
6.	Flexible Menu Reset	86
J.	SUMMARY	87
V.	SUMMARY AND CONCLUSIONS	89
A.	SUMMARY	89
B.	LMS	89
1.	Benefit	89
2.	Computer Support	90
3.	CNP	90
C.	PROJMG	90
1.	Benefits	90
2.	Shortcomings	91
3.	First Usage	91
4.	Transportability	92
D.	DOCUMENTATION	93
1.	CNP	93
2.	PROJMG	93
3.	Glossary	93
4.	Debugging	94
VI.	RECOMMENDATIONS FOR FUTURE CONSIDERATION	95
A.	INCREASED EXERCISE SUPPORT	95
1.	Discussion	95
2.	Recommendation	96
B.	INCLUDE ALL DECISION POINTS	96
1.	Discussion	96
2.	Recommendation	97

C.	DEVELOP A DMS TUTORIAL	97
1.	Discussion	97
2.	Recommendation	97
D.	VALIDATE PROJMG	98
1.	Discussion	98
2.	Recommendation	98
E.	REPLACE THE DATA TABLES WITH EQUATIONS	98
1.	Discussion	98
2.	Recommendation	99
F.	INCREASE SENSITIVITY ANALYSIS OPTIONS	100
1.	Discussion	100
2.	Recommendation	100
G.	IMPROVED GRAPHICS	100
1.	Discussion	100
2.	Recommendation	102
H.	PLATE DISPLAYS	102
1.	Discussion	102
2.	Recommendation	103
I.	A SCENARIO GENERATING PROGRAM	104
1.	Discussion	104
2.	Recommendation	104
J.	USE PROJMG TO ENCOMPASS OTHER CNP	
	VARIATIONS	105
1.	Discussion	105
2.	Recommendation	105
	APPENDIX A: CONTRACT NEGOTIATION PACKAGE (CNP)	106
	APPENDIX B: DMS DATASHEETS	123
	APPENDIX C: LINKPROJ EXEC	132
	APPENDIX D: PROJMG EXEC	134
	APPENDIX E: PROFILE EXEC FOR PROJMG	141

APPENDIX F: PROJMG STUDENT INSTRUCTIONS	143
A. I N T R O D U C T I O N	144
B. I N S T R U C T I O N S	147
APPENDIX G: PROJMG FORTRAN	193
APPENDIX H: PROJMG PROGRAM VARIABLES GLOSSARY	312
INITIAL DISTRIBUTION LIST	322

LIST OF TABLES

I.	Development Contract Values Summary	20
II.	Page Data Strings	21
III.	CCST + FACTORS	25
IV.	DATABASE PRINTCUT as SELECTION MENU	43
V.	Development Contract Summary	45
VI.	Proposed Values Table	47
VII.	Final Contract Values Table	48
VIII.	Accepted Values Table	49
IX.	DATAFILE TEAM03	50
X.	CCST + FACTORS	55
XI.	SELECTION MENU	58
XII.	CCST + FACTORS	65
XIII.	Selection Menu	72
XIV.	Development Contract Summary	76
XV.	Sensitivity Analysis Menu	77
XVI.	Sensitivity Table	80
XVII.	Development Contract Summary	109
XVIII.	Page Data Strings	111
XIX.	Development Contract Summary	112
XX.	Page Data Strings	115
XXI.	Page Data Strings	117
XXII.	Development Contract Achieved Value Summary . .	118
XXIII.	Development Contract Achieved Value Summary . .	121
XXIV.	PBCJMNNG FILEDEFS	144
XXV.	P B C J M N G FILES	146

LIST OF FIGURES

1.1	Defense Management Simulation Decision Points	14
3.1	MAIN MENU	42
4.1	Monitor's Main Menu	59
4.2	Monitor Mode Menu	60
4.3	MAIN MENU in Mode 1	61
4.4	MAIN MENU in Mode 2	61
4.5	Initial Monitor Display	68
4.6	Monitor Session File Purge Option	69
4.7	PROJMG Initial Displays	70
4.8	Team Number Responses	71
4.9	Monitor's Main Menu	74
4.10	Flight Tests Demonstration	75
4.11	Monitor's Main Menu in Mode 1	79
4.12	Plot Character Key Definitions	81
4.13	Sensitivity Table	82
4.14	Sensitivity Analysis Percentages Graph	83
4.15	Monitor's Main Menu in Mode 1	84
4.16	Selected Vertical Axis	85
4.17	Selection Menu with Optimized Flight Tests	88

I. INTRODUCTION

A. HISTORICAL BACKGROUND

During the Presidential Administration of John F. Kennedy, Secretary of Defense Robert S. McNamara instituted a major revision for management of defense resources. These procedures were instituted as a revolution to the military procurement system. Among the revisions he instituted were the Planning, Programming and Budgeting System and a management system for acquisition of new major systems.

The impact of these new procedures on the military was to create an immediate need for extensive education of the armed services' acquisition organizations. The personnel in charge of ongoing system procurements, the project managers, needed a means of being educated about the system. New personnel to the field of acquisition also had to have a means of learning about the system.

In response to the services' need, in 1963 the Defense Weapons Systems Management Center developed the Project Management Simulation Exercise (PMSE) to aid in DOD-wide training in project management. The exercise simulated procurement of a missile system to aid the DOD-wide training in PFES.

In the later 1960's, the Industrial College of the Armed Forces (ICAF) obtained the simulation and revised it into the Defense Management Simulation (DMS). DMS was installed at the Naval Postgraduate School in 1971.

The exercise currently has international application. It is used by Israel, Sweden and Singapore to train their defense personnel in the use of acquisition management systems.

Agencies within the U. S. government currently using DMS or related products for training exercises include:

- Industrial College of the Armed Forces
- Naval Postgraduate School
- U. S. Air Force Academy

It is this wide acceptance which has maintained DMS as an active management training exercise.

B. SCENARIO

The exercise developed by DWSMC in 1963 consisted of a time-sequenced set of decision points in the life cycle of a missile system acquisition (see Figure 1.1). The decision points (DPs) simulate the equivalent functions of these major milestones in the system life cycle. Within a prescribed time period each team of students participating in the exercise reviews the information materials for the specific decision point and submits its decision for recommended action at that point in the game, as would be done in actuality. The objective is to optimize the cost effectiveness for procurement, production and development of the proposed new missile system, the Zebra Missile System.

ZEPHA PROGRAM SCHEDULE	zero	one	two	three	four	five	six	seven	eight
DECISION POINTS	1	2 3		4	5	6			
CONCEPTUAL PHASE									
Validation Phase									
Contract Definition		1	1	1	1	1	1	1	1
Development Phase									
Sub-system Design & Test		1	1	1	1	1	1	1	1
Proto-type Assembly & Check		1	1	1	1	1	1	1	1
Fabricate for Qual Tests		1	1	1	1	1	1	1	1
Qualification Tests		1	1	1	1	1	1	1	1
Fabricate for Flight Tests		1	1	1	1	1	1	1	1
Flight Tests		1	1	1	1	1	1	1	1
PRODUCTION PHASE									
Block I long-lead Procure & Tool				1	1	1	1	1	1
Block I Mfr. Eng. & Procure				1	1	1	1	1	1
Block I Delivery by lots				1	1	1	1	1	1
Block II Mfr. Eng. & Procure				1	1	1	1	1	1
Block II Delivery by lots				1	1	1	1	1	1
Deployment Phase									
System Deployment by group						1	23 45	6 78 9 10	

Figure 1.1 Defense Management Simulation Decision Points.

C. SIMULATION

The team recommendations require monitor evaluation. Monitors review the student recommended performance reliabilities, costs, incentives, contractor selection processes, milestone dates and fees. They prepare materials for teams to interact with as they would with the Secretary of Defense and with the contractor. An extensive study of the materials from each team is required by the monitor.

In order for the monitors to not spend as much time on the simulation as the students, the monitors need aid in processing the team data. ICAF developed an Fortran program called Contract Negotiation Package (CNP) to assist the monitor in the evaluation of student team contract negotiations.

The DMS program is more than twenty years old. The program has experienced several modifications and offshoots. Its 1960's programming state-of-the-art remains mostly intact. By 1984 standards of ease of operation and user-friendliness, it is difficult to utilize.

Efforts to modify the program occurred in 1972 and in 1977. The first attempt to upgrade the DMS program in 1972 modified several internal algorithms and produced an offshoot program which provided fewer but more rapid results. In 1977 this version of the program, the Contract Negotiation Package, was again revised. This later revision provided predictions of achieved values to the student teams.

Team access to the program's simulated results grew out of the modernization of management processes. Managers are using the benefits of modern computer systems to improve their own performance. Student team use of the CNP program for the DMS exercise was a logical evolution. CNP use by the teams enables the DMS exercise to reach the level of current

project management technological support in computer-aided acquisition management techniques. The program also remains abreast through student and monitor interest in maintaining the program at the computer support state-of-the-art.

The CNP program available on the IBM-3033 in the W. R. Church Computer Center at the Naval Postgraduate School campus is discussed further in Chapter 2. Its displays reveal CNP's batch program origins. The data input and output formats are batch processing type fields. The lack of interactive capability typifies a program which is batch processed. In its batch processing use, the computer did not have terminals or users with which to interact. Originally CNP had no hardware capable of directly interacting with the users and its design developed accordingly.

Modern menu-driven designs have made user-friendly programming standard practice. As a viable training exercise for acquisition management, the DMS game continues to be valuable. In order to maintain its wide acceptance and expand its efficient utilization, a major revision to the program is needed.

The following chapters discuss a package of programs designed to give CNP an improved interactive program capability.

D. SUMMARY

The DMS training program incorporates an exercise which simulates a project management scenario. DMS is a successful training device which has been in use for 20 years. In support of DMS, the Fortran computer program CNP was developed to provide computer-aided assistance for monitor evaluation of team performance. Later, CNP was released to the student teams in a format which provides them with simulated performance data. It enables team

interaction with the computer. In so interacting, the team gains experience using the computer as an analysis tool and to perform sensitivity analysis ("what if questions") on major project performance and incentive parameters. As a result of CNP's age, its design did not permit adequate interactive capability for easy operation. In order to resolve the need for greater user-friendliness in CNP, this thesis develops, studies and reports on a CNP program revision entitled PRCJMNNG FORTRAN.

II. CNP OPERATION

A. PROCEDURES

Proficient use of the Contract Negotiation Program (CNP) depends upon the user's familiarity with both the computer facilities and with the instructions in Appendix A. The "Defense Management Simulation Instructions for Using CNP" presents computer terminal display lines as seen during a terminal session. In following sequentially through the instructions, difficulties are apparent with the program both in user-friendliness and in design. Appendix A demonstrates a session in CNP, and pertains to the following discussion of that program.

The procedures for accessing CNP require the operator to link to the monitor's class disk, then to access the disk, and finally to execute CNP. Accessing the program in this manner normally requires at least a conceptual knowledge of computer command and response terminology. In the event of an accessing malfunction, or if the monitor is using the CNP program's class disk and prevents read-only execution of the disk files, the user may be confronted with unfamiliar terminology. He would need more than a casual knowledge of computer operation to proceed into CNP.

The first function performed by CNP is to input administrative data. Included in the administrative data are the team number, the decision point (DP), and the DP pages for which data are to be entered. Complications with the program quickly arise with these procedures. The data must be input in the exact format prescribed on the instruction sheet. Too many digits, or too many spaces in the data result in erroneous output, and/or error statements. Each

page's data string requires confirmation. This becomes laborious in repeated runs of the program, or repeated executions of the calculations for a Summary Table, Table I.

Frequently explanation is required for the procedure of binary coded response to the query for pages of data to be entered. Coding the page selection for data input can be confusing. Each of the four pages requiring input data must be indicated by '1'(on) or '0'(off) in the four digit code string. Indicating the page number in Arabic characters would be a more commonly expected procedure. Again, an error in entry is irrevocable. The change pages cannot be rescinded once the binary code has been entered.

Having completed the administrative inputs, the team proceeds with providing its proposed DP data. Prior to running CNP, team participants are to have used the DMS Decision Point handouts (see Appendix B) to determine and analyze their tentative DP positions. These results are listed as data lines in the DMS Decision Point Sheets, (Appendix B). Each character in the page's data lines requires precise replication when placed in the page data string input to CNP. The program types the page's data back onto the screen and asks for confirmation. Through relooping to the query for pages of input, the program does allow revision of data. However, each data item to be changed requires an entire page change. The team is confronted with reentering the data string each time. This procedure has proven to be tedious. It frequently results in teams redoing data strings several times to get them correct before a calculation of achieved values can be made. Successive loops through the program have the added check feature of receiving a four line printout of the page data for confirmation. The redundant queries for confirmation of data values have benefit in reducing erroneous data calculations, (see Table II).

Based on the author's experience, about one out of four occasions through the four line recap of data pages, Table II, the team finds an error.

TABLE I
Development Contract Values Summary

DP-3 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 1

	INCENTIVE PROVISIONS			INCENTIVE ACHIEVEMENTS	
INCENTIVE AREA	WORST VALUE	BEST VALUE	MAX FEE ALLOWED	ACHIEVED VALUE	
DEV. COST	\$ 53.0M	\$ 47.0M	4.0%	\$ 48.5M	\$
FLT TEST COMPL	236 WK	202 WK	3.5%	199 WK	\$
RELIABILITY	75.0%	81.0%	3.5%	80.0%	\$
ACCURACY	160YDS	140 YDS	4.0%	143 YDS	\$
TOTALS			15.0%		\$
TOTAL CONTRACT PRICE = \$ 54.9M					

The Development Contract Summary Table, Table I, neatly displays the decision point incentive input data as the 'Incentive Provisions'. Only five calculated values are displayed to the team. They are the achieved values for development cost, flight test completion date, reliability, accuracy, and total contract price. Also, displayed on the Summary Table is the computed total of fee percentages obligated. The total fees allocated must be 15%. Allocation of fees to other than 15% results in a warning on the Summary Table. The significance of the Development Contract Summary table and CNP rests in these five achieved values.

TABLE II
Page Data Strings

10301	1	1	2	1.75	0.75	0.0	
10302	92.00 3.00	97.00 3.00	98.50 6.00	80.00 17.00	70.00	25.00	6.00
10303	53.00 75.00	47.00 3.50	4.00	238.00	202.00	3.50	81.00
10304	160.00	140.00	4.00	337.00			

Compared with the original CNP design, there are missing values from the Summary Table. It does not print out ten additional calculated values: the fee percentages achieved, their fee cost, and the totals of fees earned and of fees cost. These values were suppressed from the original program table due to their computation by incorrect algorithms. Their suppression is evidenced by the dollar signs at the right of the Summary Table.

To reloop through the input of data and receive another estimate of the achieved values, the team answers an unessential query. This question has no application to the teams. The "COSTS+FACTORS(NO)" query provides monitor access to the tables of design performance factors achieved and test costs in the form of unit change cost factors. The team's nc('N') response to, or an incorrect response to, this query reloops the program to the top of the main routine. Again, each question: decision point, team, pages, data, yes or no, and confirm must be answered.

It is extremely difficult to exit from the program if the team attempts to stop at any point before the reloop query. At the query "DO YOU WISH TO CONTINUE, Y FOR YES, N

FOR NC", the program provides its only graceful exit. Any other attempts to exit the program must be accomplished by dumping (stopping) the program. Inputs that do not match the data types required for data read-statement formats result in error statements, and with sufficient error occurrence count the program will dump.

B. BENEFITS

The benefits for team utilization of CNP are in its ability to provide some semblance of the achieved values which the main DMS batch program, provides. In so doing, the team uses the computer as a tool with which to view the sensitivity of its incentivization parameters. It can evaluate effects of varying available DMS critical incentive parameters: development cost, flight test completion date, reliability, accuracy and development contract price. This aspect of analysis simulates the real world potential for computers to expedite the computations and as a tool in these assessments. It further provides an environment in which team participation and success depends on understanding the system acquisition concepts that DMS illustrates. In order to expeditiously vary the parameters within the computer the team must have competent knowledge of these DMS concepts.

CNP's best feature appears in its structure. It closely follows the procedures in the DMS Decision Point Handouts, ¹ From its inception as an offshoot of the main CMS program, it has inherently followed the decision point sequence and structure of DMS.

¹Industrial College of the Armed Forces, Defense Management Simulation Participant's Manual, Industrial College of the Armed Forces

The wide acceptance of CNP at many institutions can be found in the attributes it brought to DMS. In meeting the objectives of providing a program with the ability to quickly reloop through computations, and the ability to be operated from a terminal, the program fulfills a major need in performing certain sensitivity analyses over the DMS program. Once accessed, the program can be rerun innumerable times without completely reinputting all the data. The data inputs can be accomplished from a terminal. Computer support for DMS on computer terminals has made DMS accessible to more users without adding expense. It has enabled many students to simultaneously utilize DMS as a tool in acquisition management training.

C. DEFICIENCIES

Areas which need improvement in CNP focus on user-friendliness. The first of these is the inability to gracefully exit from the program. Some means must be available to permit the team to stop the program. Currently the program can only be stopped by expending time to finish the session or by generating sufficient syntax errors to force the computer to dump the program. In some cases, sessions have been in sufficient difficulty that the program would not reach the query "DO YOU WISH TO CONTINUE...". For this situation the team is forced to generate error counts which will dump the program. These situations have been exceedingly frustrating for the program users.

A means to change data entries without redoing entire pages is needed. The team must completely redo a DP page in order to change one data item. Changing one item should only require reentering one value. Additionally, data strings from one decision point to another are not readily duplicated for input. Similar strings of data for pages on

DP-3 and DP-4 do not have the same field format. Matching field formats for identical pages in different DP's would enhance the ability of teams to input data strings.

CNP receives all of the input data but cannot retain it between sessions. The team starts from a baseline of no data each time CNP is called. Most significant of the changes needed to modernize CNP is the creation of a database file to retain team data from session to session. A foundation data base of the initial paged input could provide a means of replicating session results, of documenting data inputs, and of proceeding to successive DP levels without laboriously reentering previously studied data.

Excessive recurrences of calculating the Summary Table could be eliminated to save team time in the program. Preventing runs which produce Summary Tables without correct incentive fee percentage totals would reduce wasted program run time. Utilizing data that does not meet the DMS parameters calculates erroneous values, and may provide Summary Tables which mislead the team. Repeatedly running calculations to evaluate wide ranges of parameter values without conducting proper preliminary analyses not only detracts from the timeliness of the DMS studies, it defeats the purpose of DMS training teams to understand the concepts of project management. Teams have occasionally found greater ease in working through the program and produce a Summary Table in order to reach the "DO YOU WISH TO CONTINUE...?" query as an expedient to reloop through or exit from the program. This process detracts from the educational benefit and timeliness of DMS.

As part of the training simulation, monitors spend many hours in analyzing the progress of the teams and the positions for contract negotiation. Analysis done by each team must also be accomplished by the monitor to assess student

D. MONITOR FUNCTIONS

CNP provides one additional table for monitors. COST+FACTORS, Table III, provides the monitor with the cost for a unit increase in a system test category as well as with the design factors achieved.

TABLE III
COST + FACTORS

COSTS OF TESTS PER UNIT (IN THOUSANDS OF DOLLARS)

QUALIFICATION TESTS

MOTOR :	32.034
AIR FRAME :	126.355
GUIDANCE :	309.906
FIRE CONTROL :	938.186
LAUNCHER :	424.555
FLIGHT TESTS :	316.855

DESIGN FACTORS ACHIEVED

COMPONENT	TABLE	ROW	FACTOR	ACHIEVED
MOTOR :	1	25	0.825	0.877
AIR FRAME :	2	15	0.958	0.980
GUIDANCE A :	5	22	0.666	0.630
GUIDANCE B :	6	25	0.580	0.0
FIRE CONTROL :	4	16	0.845	0.802
LAUNCHER :	3	7	0.936	0.988

IHO001A PAUSE ; PRESS <<ENTER>> TO CONTINUE.

progress. The monitor must also create positions from which to interact with the teams as the contractor. Within the COST+FACTORS Table, some aid exists for the monitor. However, his understanding of the impact of unit-cost on the contract requires additional analysis time over what the teams spend on incentives analysis. In general, because of limited time the monitor has to spend analyzing each team's

proposals compared with the amount of time put in by the team, coupled with the inconvenience in making repeated CNP runs, most monitors produce an actual DMS table printout corresponding to each team's proposed decision in order to provide more detailed and accurate information for use during contract negotiations. It would be better for CNP to provide the information in a more usable format than to produce the lengthy and time consuming main exercise reports of the DMS.

E. SUMMARY

The difficulties encountered in CNP detract from its usefulness. Time spent fighting the program problems are excessive compared to the hours available for productive DMS analysis. In an effort to correct these time-sinks in CNP, PROJMNG FORTRAN and its associated EXEC routines have been created.

III. PROJMNG OPERATION

A. CNP MODERNIZATION

The task in creating PROJMNG FORTRAN as a revision to CNP was first to retain the same functional structure relationship with DMS. This goal has been accomplished by incorporating the CNP subroutines into a larger program. The expansion has occurred by the added Fortran code for the human-factors engineered subroutines, and new sensitivity analysis features. The human-factors designs provide additional user friendliness to PROJMNG. The program calculations continue to center around the original CNP program routines. Many of the achieved value parameters and performance calculations are accomplished using the same Fortran code as used by CNP. CNP is encompassed in PROJMNG.

Other tasks for PROJMNG FORTRAN were to correct as many of the discrepancies in CNP as possible, to provide an enhanced user-friendly training device and to improve the analysis capability available to the monitor.

In modernizing CNP, the criticisms specified in Chapter 2 were highlighted. The following areas were examined.

1. Link
2. Database
3. Database Security
4. Exit
5. Data Change
6. Menu
7. Report Submission
8. Monitor Access
9. Sensitivity Analysis
10. Calculation Correction

B. PROJMG PACKAGE

This chapter discusses the features of PROJMG used to support team analysis of the exercise. The monitor capabilities are discussed in Chapter 4.

1. Programs

PROJMG consists of more files (executable programs) than does CNP. There are four routines utilized in executing PROJMG FORTRAN. In addition to the PROJMG FORTRAN file, the package includes LINKPROJ EXEC (Appendix C), PROJMG EXEC (Appendix D), and the PROJMG disk's PROFILE EXEC (Appendix E) which are discussed in detail throughout this thesis.

PROJMG FORTRAN is designed to function alone provided team files exist and have been defined for PROJMG FORTRAN as files 9 and 10: "FILEDEF 09 DISK DATAFILE TEAMxx" and "FILEDEF 10 DISK DATACODE TEAMxx". It can be used without the exec routines but at a lower level of performance. PROJMG can be executed through the routine PROJMG EXEC.

PROJMG FORTRAN and PROJMG EXEC are designed to reside on a publicly accessible disk. In this manner multiple teams can use the program simultaneously. Their database files will be on this one disk where they are conveniently accessible for the program and the monitors. PROJMG is read from this disk by the team's user disk and executed from the team's disk.

PROJMG both reads and writes onto its data files. The IBM-3033 public, "all", link access is read-only (R/O) and prevents writing onto the disk. The PROJMG disk is only read capable when linked to for the exercise. However, the disk linking to PROJMG can be written onto by the program. A team disk linking to PROJMG acts as the write disk for the program.

The ability for teams to write directly onto the PROJMG disk would be unsatisfactory. This access to the program would permit two detrimental effects. The program might be either altered or destroyed by overly curious students entering the files, and team files might also be altered or destroyed. Inadvertent alteration of the files is prevented by student access being limited to read only.

2. Data files

There are four files of data tables that were created for CNP. These tables contain the resource parameters to generate achieved values for the program calculations. These data tables remain in files FT17F001, FT18F001, FT19F001, and FT20F001 as in CNP. No contract simulation calculations can be accomplished without them. The program will dump when it requires a table be read and does not find it. PFCJMG will also stop if the tables are not in the required format.

Preformatted database files must be available when the STORE and read routines are called by PROJMG. Whenever it attempts to write or read these data files and does not find them, the program will dump (stop). Two of the remaining files are these database files. One is the FILE DATAFILE which is preformatted for the DMS data. It serves as a dummy file which can be copied to initiate the team's database, DATAFILE TEAMxx. The other file, FILE DATACODE, serves as a similar dummy file in the creation of the team's security file, DATACCDE TEAMxx.

DATACODE stores the team security code, the security code count for attempts to breach the code, and the USERID of the last team which used the file. Its data enables the program to use routines to check the team and validate the team identity through the security code.

If formatted files for the team are not available on the PROJMNG disk or the team disk, the dummy files are copied onto the team disk. PROJMNG EXEC reproduces the two dummy files on the team disk in the names' DATAFILE TEAMxx and DATACODE TEAMxx. They are used interactively by PROJMNG during the game sessions. The PROJMNG Instructions (Appendix F) discuss the DATABASE and DATACODE file generation on the team's disk.

When the team finishes its session and exits from PROJMNG FORTRAN, the last copy of the data which was stored by the program remains on the team's disk. As the computer execution finishes the routine PROJMNG EXEC, these two files are transmitted to the PROJMNG disk.

3. LINK

LINKPROJ establishes the link between the team's disk and the PROJMNG disk. It requires PROJMNG EXEC in order to run the program.

The routine LINKPROJ is a CMS executive level program which utilizes the link feature to establish the connection between the team's disk and the disk on which PROJMNG resides. It begins the recording of the team's session, "RECORD ON". This first step in LINKPROJ provides a record of the session for review in the event PROJMNG does not perform as desired. Once the link has been established, the routine sets the function of the programmable function key 9 (PF9) to enable release of the link. This feature enables the team to release the link after a session has been aborted and the program stopped without completing the LINKPROJ EXEC. -The team would have to abort a session should PROJMNG be stuck in a continuous loop.- PF9 will not succeed in breaking the link when PROJMNG is operating normally.

Team members should be aware of the IBM-3033 capability to be dumped (stopped) out of CMS by executing <<ALT>> and <<PA1>> simultaneously. Executing <<PA1>> aborts the program without completing LINKPROJ. The pressing of these two keys on the IBM 3278 terminals will place the virtual system into the CP level of operator interaction. The team can reaccess its "A" disk by executing the command "I CMS" and pressing <<ENTER>>; wait for the display to read "VM READ" and again press <<ENTER>>. The disk will again be in CMS and existing links will remain intact. In order to release the link to PROJ MNG press the <<ALT>> and <<PF9>> simultaneously.

LINKPROJ breaks the link. It releases the PROJ MNG disk (0276P). Once the disk is released if no users are accessed to it, the PROJ MNG disk can be logged onto in the read/write mode to enable files to be written onto it.

When LINKPROJ sends the special message to PROJ MNG's disk, "MSG QACNT KLINE" it tells the computer to automatically log (autolog) onto the disk. The disk is activated in the read/write mode only if no other user is on the PROJ MNG disk. The files which the team sent are copied by the disk's PROFILE EXEC.

LINKPROJ also terminates the recording of the session, "RECORD OFF".

4. PROJ MNG EXEC

a. Initiate Files

PROJ MNG EXEC performs the checks of available files. It determines if a formatted data file and a security file are available for the team. If it detects the monitor unique file, DATAINST DISKNUM, it initiates the monitor flag to tell PROJ MNG FORTRAN the user is a monitor.

PROJMNG EXEC checks first for the monitor's file, determines if the user is a team or a monitor. If the user is a team, it checks for the team files on the PROJMNG disk and uses them if available. If these files are not found, it will check the team disk for the files and use them. If the files are not found at all, the program copies the dummy files and generates the team label and files.

b. Terminating the Session

If the user is a monitor, PROJMNG EXEC queries him for whether he desires a concatenated listing of all of the team files. If he does, it spools a file with all team data files on it. It then completes its execution and returns operation to LINKPROJ.

For team operation, when the computer leaves PROJMNG FORTRAN control, execution returns to PROJMNG EXEC. In order to complete the session, PROJMNG EXEC has been designed to transmit the data files to the PROJMNG disk. This feature is accomplished by spooling the disk, "SPCOL PUNCH CONT CL X 0276P". The punch file created is sent to userid 0276P as a class x file. PROJMNG EXEC punches the data files onto the spool, "PUNCH DATAFILE TEAMxx", and closes the spool, "SPOOL PUNCH CLOSE NOCONT". The file is queued in the reader file of the PROJMNG disk (0276P).

5. PROFILE EXEC

When LINKPROJ sends the automatic log-on message to the PROJMNG disk, if no other user is on the PROJMNG disk, the disk is activated in the read/write mode. The main computer executes the disk's profile routine. PROFILE EXEC is performed whenever the disk is accessed in the write capable mode. This exec has been uniquely tailored for this PROJMNG specific disk (see Appendix E). It reads up to six class x files queued to the disk. First the PROFILE EXEC

sets the disk reader file to look for class x, "SPOOL RDR CL X", and then reads the files, "READ *". The program has been designed to reset the reader spool to spool any class. Upon completion of writing files onto the disk, the command "SPOOL RDR CL *" returns normal spooling capability

The PROFILE read function would nominally expect to find a set of files, DATAFILE TEAMxx and DATACODE TEAMxx. The read limit was expanded from two files to six, for three pairs, in expectation that there will be occasions when the disk will remain in use by other teams. When other teams are still using the disk the autolog will be unable to perform its one shot access to the disk. SMSG will not remain queued to access the disk. In order to read those class x files that may have been left waiting, subsequent running of the PROFILE EXEC will read in several more files than those transmitted in the current PROJMNNG session. The exec will read up to three additional pairs.

The benefit in using the class x spooling capability is the cue which this class gives. It tells the disk that the file is a PROJMNNG file. Normal spooling usage within the IBM 3033 does not use x. 'X' has a uniqueness within the IBM system and will reduce the possibility of PROJMNNG EXEC reading non-PROJMNNG files.

The files are not read onto the PROJMNNG disk while other users remain on the disk. Attempts to write onto the disk in this situation result in garbling the data files.

C. DATABASE SECURITY

The first query the team answers in PROJMNNG is to provide the team number.

The second question asks for the team to create a team security code. The team may enter any code of up to eight characters. If the eight characters are all zeros or

contain nothing (nulls) the program will again ask the team to change the code. PROJMG begins succeeding sessions by asking the team to match the security code stored in database, the team's security code.

The security code system gives the team access to its database while assuring that data has not been altered by anyone except the team or the monitor. This has benefits to the team and to the simulation. The team doesn't have to worry about data being changed accidentally and the monitor has less concern that the team's work might be compromised. The security code is also used for proposal submission. Due to its length and improbability of being entered correctly by mistake, the security code performs the function of assuring the monitor of the team's intent to submit the proposal as indicated.

The monitor can change the database to correct errors in the team files. Corrections may become necessary if the team submits a report and then wants to change it. The proposed and final data reports for a DP are not accessible by the team. Lack of access to submitted reports increases team commitment to the report. It also, provides the monitor with a fixed team position reference point. This is critical if the monitor is to create a contractor position by which to interact with the team.

The submission of reports and their purpose are discussed in greater detail later in this chapter.

D. EXIT

In PROJMG several methods of stopping the program have been provided. These are:

1. Fatal errors
2. Security violation
3. Menu option

4. Yes/No query

1. Fatal Errors

The computer capability to stop the program when a programmed statement cannot be performed (executed) is referred to as a fatal error. PROJMNG has selectively modified the fatal error response.

The program termination in response to a null entry (pressing the <<ENTER>> key without providing a character including the character blank ' ') was unsatisfactory. Inadvertent null responses repeatedly dumped users out of the program.

To resolve the fatal error dump (stopping the program), a GLOBAL routine (program which can be called from the IBM 3033's resident routine files) is called. The 'CALL ERRSET' enables the program to ignore fatal errors and proceed with the program. When PROJMNG finds an error during the process of reading information from the terminal, it performs a jump to the statement specified as the error default by the READ statement. Very serious errors in syntax will continue to be dumped by the computer.

Failure to answer PROJMNG's questions satisfactorily will cause it to relcop to either the same question or to call out the EXITS routine. If an error is serious enough, the EXITS routine will be called. EXITS allows termination of the program, or if the team wants to continue, EXITS will return to the top of the last query routine and allow the program to reloop. Correct entry of the security code will give the team access to the main program and existing data files with that team's number.

2. Security Violation

During the start of a session at the terminal, the user will be queried for his previously entered security code. The program will permit the user to answer the security code query incorrectly only five times during the course of a DMS exercise. At occurrences of incorrect security code entries, a count is kept. On the fifth occasion when the code is entered incorrectly, the counter will tell the program to stop. Successive attempts to enter a team file whose security file counter has reached five will terminate at the security code query.

3. Menu

A standard option in the main menu of PRCJMNG is EXIT. Selecting the EXIT option directs the program to stop. Again, this operation returns computer execution to the PRCJMNG EXEC program. Completion of the PROJ MNG EXEC run will return control to the LINKPROJ routine which presents the query for another run.

4. YES/NO Query

Each query in PROJ MNG which can be answered with a Y for YES or N for NO, can also be answered with an E for EXIT. This E response either calls the program subroutine EXITS or stops the program. EXITS gracefully stops the program by permitting either return of the computer's operation to PRCJMNG EXEC or by relooping into the last query routine.

5. Return to PRCJMNG EXEC

The termination of PROJ MNG FORTRAN returns the computer execution to PROJ MNG EXEC. This routine executes a transmittal of all team files to the PROJ MNG disk.

E. PRCJMNG PROCEDURES

1. Instructions

The procedures used with PROJ MNG FORTRAN are significantly different from CNP. Appendix F contains the PRCJMNG Student Instructions. These instructions replicate the program's screen displays. Appendix F demonstrates a session using PRCJMNG. Checks and corrective procedures in PROJ MNG render the making of errors more difficult. However, some examples are demonstrated in the PRCJMNG Student Instruction's tutorial.

2. Link

The instructions ask the team to access PRCJMNG solely by executing the command LINKPROJ. This command runs the program LINKPROJ EXEC. LINKPROJ EXEC is a routine which creates the disk link and accesses PROJ MNG as the user's 'C' disk, as discussed above. This exec must be placed on the team disk. From there it is able to link and access PROJ MNG's disk for the team.

Among the capabilities of LINKPROJ are the definition of programmable key PF9. This function can be executed by pressing <<ALT>> and <<PF9>> simultaneously.

LINKPROJ also executes a pair of calls to the macro computer program routine "RECORD". The first of this pair enables the recording of all session terminal activity. The latter executes after the session has been completed; it terminates the record of the session. Teams should be aware that if the program dumps or they exit the program without performing the "RECCED OFF", the record will continue in effect. It can be closed by inputting the command "RECCRD OFF". Termination of the record permits the team to either name and save the session's record on disk, to edit it, to print it, to quit it ("Q") which places it on the disk

spool, or to purge the record. Addition of this capability permits review of team session. Undetected difficulties or errors that result in program malfunction can be reviewed.

In order to facilitate rerunning of LINKPRCJ, the routine does query whether the team wishes to run "ANCTHER RUN, Y/N". This feature is one which LINKPROJ holds in common with the other programs in the PROJMNG package of routines. PROJMNG is designed to allow the team to terminate or reloop through program execution.

The running of PROJMNG FORTRAN does not occur from LINKPRCJ. LINKPROJ runs limited functions already specified, and calls the PROJMNG disk's program PROJMNG EXEC.

3. PROJMNG EXEC Execution

PROJMNG EXEC serves several functions in preparing the team disk for the session.

a. Team Disk Status

The PROJMNG EXEC routine provides several queries to the team disk. Through the response error return codes it receives indication of the availability of team data base files and security code files. The purposes accomplished by this procedure are:

(1). File Availability. The error return codes tell PROJMNG EXEC whether there are files available for use. Neither PROJMNG EXEC nor PROJMNG FORTRAN can proceed without files from which data can be read or to which data can be written.

(2). File Integrity. Having found a usable DATAFILE on the team disk, PROJMNG EXEC attempts to read the contents. It looks specifically for the team number. The team number cues PROJMNG EXEC not only as to what the team should be called, it also reveals the prior use of the DATAFILE to store data.

b. Monitor File

In a similar manner, PROJMNNG EXEC determines the availability and validity of the monitor file "DATAINST DISKNUM". The presence of data in the file and the security field format within DATAINST DISKNUM tells PROJMNNG if the disk gaining access to it belongs to a monitor. Disks belonging to monitors are not intended to be the disk which students access to run PROJMNNG.

c. DATAFILE Creation

PROJMNNG EXEC determines the team disk status. If DATAFILE and DATAACCDE files are not found, PROJMNNG EXEC proceeds to determine the team number. It will ask "WHAT IS YOUR TEAM NUMBER?" before it copies the files. After the team number is evaluated for being in the range of 1 to 20, the program generates the "TEAMxx" label. It will accept 1 to 20 with any number of preceding 0's. The TEAMxx label will include up to four numbers in the xx field. This label is used in the process of copying the dummy files. PROJMNNG copies the dummy files using the labels DATAFILE TEAMxx and DATAACCDE TEAMxx as the new file name and file type.

In the event that more than 20 team files are required, PROJMNNG will accept team file labels with additional leading zeroes. Teams with files labeled TEAM01 and TEAM001 can both be created.

The availability of the DATAFILES also gives PROJMNNG EXEC the team number. Lack of DATAFILES is synonymous with no team number. The team number is determined only once during the course of the gameplay. It is PROJMNNG EXEC that makes that query. It forms the team number into the "xx" portion of the files' type labels. PROJMNNG stacks the team number into its virtual reader to enable PROJMNNG FORTRAN to copy the team number. PROJMNNG FORTRAN uses the

team number to label displays and, when storing the database, places the number at the head of the record. It is this storing process and filling in of the first DATAFILE field which later enables PROJ MNG EXEC to find the team number without additional queries.

4. Running PROJ MNG

PROJ MNG EXEC completes determining if the disk belongs to a team or monitor, gets the team number or sets the monitor's flag of team number 21 (FL21), insures the availability of write files or creates them, and then loads and starts PROJ MNG FORTRAN's text file. The team database capability has been set for 20 teams. The data file for monitors was chosen in excess of the team database design size of 20. Having chosen TEAM21 to be the monitor's file label, it was convenient to utilize the label TEAM21 as a flag between PROJ MNG EXEC and PROJ MNG FORTRAN. It indicates to PROJ MNG FORTRAN that the computer operator for the session has been determined by PROJ MNG EXEC to be a monitor. Upon termination of PROJ MNG FORTRAN, PROJ MNG EXEC again takes control. If the operator is a monitor, the EXEC allows the monitor to request the creation of a concatenated file of all current team files on the PROJ MNG disk. For team disks, the exec sends a copy of the files DATAFILE and DATAACDE to the master disk. This file transmission enables the monitor to determine team progress and to accept the DP decision reports. PROJ MNG erases unnecessary residual files such as the LOAD MAP and then returns execution control to LINKPROJ to perform the "ANOTHER RUN, Y/N?" query. The team's response to this question will either terminate the session or reloop the program to the beginning of LINKPROJ.

F. DATABASE CHANGE

After data files have been created, PROJMG uses the DATAFILES as a baseline of team information. Once the file is created the program assumes the data is accurate and satisfactory for use in all subsequent sessions. All further Selection Menu displays will display the DMS data items as they are currently found in the team's file DATAFILE TEAMxx.

At the beginning of the team's first session all of their data items must be submitted. A database is generated from which successive sessions will obtain their baseline of information. The process of asking for all data items occurs through the program's subroutine PAGEIN.

The pages of the DMS decision sheets, Appendix E, list the data entries necessary for CNP. These same entries are queried once in PROJMG by PAGEIN as demonstrated in Appendix F. Each page of data queried coincides with the page and the data item of the decision sheets. Available parameter ranges are listed with the item name in PAGEIN. Correct completion of a page of data permits the program to check the data input for errors. If the program does not understand the input, or requires revision, a *****WARNING***** is displayed and questions designed to correct the discrepancy are asked. If the data entered cannot be read by PROJMG the EXIT query "DO YOU WISH TO CONTINUE: Y/N?" will occur. Teams experiencing confusion, or wishing to exit the program during PAGEIN, can do so by pressing <<ENTER>> without any input. During PAGEIN this action will enable the team to use the exit query to stop the program.

The automatic call to the PAGEIN routine occurs only when the DATAFILE contains 0's in all of the data base data fields. Successive sessions in PROJMG by a team utilize

the database DATAFILE to fulfil the data field input function. PAGEIN can be called at any time from the Selection Menu. This option permits the team to make mass changes to data items.

Once the program has a set of data it will produce a table containing those values, Table IV. The SELECTION MENU gives a review of the current data values and acts as a selection menu for items the team wishes to change. SELECTION MENU presentation is achieved by either completion of PAGEIN, by beginning a subsequent session, by completion of a DP proposal submission, or by team choice from the MAIN MENU.

DO YOU WISH TO:

1. RECEIVE THE TABLE OF ACHIEVED VALUES.
*****WARNING*****AFTER THE TENTH DP- 3 RUN
AN ADDITIONAL DEVELOPMENT COST OF \$100,000
WILL BE INCURRED FOR EACH ADDITIONAL RUN.
THIS IS RUN 1 OF YOUR DP- 3.
2. INPUT SELECTION MENU.
3. SUBMIT A CONTRACT PROPOSAL
4. EXIT

Figure 3.1 MAIN MENU.

G. MENU

Continuing sequentially through an initial terminal session with PROJMG, the next feature which differentiates this program from CNF is the MAIN MENU, Figure 3.1. From this routine in the main program, any of the options performed by PROJMG for the team can be executed. This menu places the PROJMG FORTRAN program in the category of being a menu-driven program.

TABLE IV
DATABASE PRINTOUT as SELECTION MENU

NO.	ITEM	CURRENT VALUE	DP-3	ITEM	CURRENT VALUE
1.	CONTRACTOR APPROACH	:	1	16.	GUIDANCE QUAL TESTS
2.	GUIDANCE CONFIGURATION	:	1	17.	PLIGHT COST
3.	GUIDANCE RELIABILITY	\$:	2	18.	MAXIMUM COST
4.	MAINTAINING DEVELOP	\$:	1.75	19.	MAXIMUM COST INCENTIVE
5.	VALUE ENG DEVELOP	\$:	0.0	20.	MAX COST WEEK
6.	PARALLEL RELIABILITY	\$:	0.0	21.	LATEST WEEK INCENTIVE
7.	MOTOR RELIABILITY	\$:	92.00	22.	EARLIEST WEEK INCENTIVE
8.	AIRFRAME RELIABILITY	\$:	97.00	23.	MAX DELIVER RELIABILITY
9.	AIRFRAME RELIABILITY	\$:	98.50	24.	MAXIMUM RELIABILITY
10.	AIRFRAME RELIABILITY	\$:	80.00	25.	MAXIMUM RELIABILITY
11.	FIRE CONTROL ERROR YDS	\$:	70.00	26.	MAX RELIABILITY INCENTIVE
12.	GUIDANCE ERROR S	\$:	25.00	27.	MAXIMUM ERROR
13.	MOTOR QUAL TESTS	\$:	6.00	28.	MINIMUM ERROR INCENTIVE
14.	AIRFRAME QUAL TESTS	\$:	3.00	29.	MAX ERROR LOT 10
15.	AIRFRAME QUAL TESTS	\$:	3.00	30.	WEEK PCB BY PAGE IN
				31.	INPUT BE CHANGED FOR DP-3

SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-3

The options available in the MAIN MENU are to print the table of achieved values, to return to the database table, to submit a proposal, or to exit.

The achieved values option directs the program to execute the routines associated with creating the Development Contract Summary table, Table V. First the data entries for fee percentages are checked. The fee percentages are added together and evaluated to determine if they total 15%. If they do not equal 15%, PROJ MNG FORTRAN requests corrections through a sequence which displays each current fee percentage value and asks for a value to be submitted. The program performs the computations for the table created in subroutine PROSUM. Then the table is displayed. Each time the computations are made the run is counted. After ten runs, the program adds a fee of \$100,000 onto the contract price for each subsequent run at that DP level.

The second option returns the display of the data table. It provides the team with the table, Selection Menu, of data items which are used for reference in changing data values.

The third capability from the MAIN MENU is proposal submission. Submission of proposals is discussed in the next section.

To enable the team to stop the program, the fourth option in MAIN MENU allows them to exit.

B. REPORT SUBMISSION

1. Progress Reports

Submission of progress reports has been readily accomplished in PROJ MNG EXEC. The process of completing PROJ MNG EXEC when leaving PROJ MNG FORTRAN serves to execute the commands for file transmission as discussed

TABLE V
Development Contract Summary

DP-3 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 3

INCENTIVE AREA	INCENTIVE PROVISIONS			INCENTIVE ACHIEVEMENTS		
	HORST VALUE	BEST VALUE	MAX FEE ALLOWED	ACHIEVED VALUE	FEE EARNED	FEE % EARNED
DEV. COST	\$ 53.0M	\$ 47.0M	4.0%	\$ 48.54M	\$ 1.486M	2.97%
FLT. TST COMPI	238 HK	202 HK	3.5%	199 HK	\$ 1.750M	3.50%
RELIABILITY	75.0%	81.0%	3.5%	80.08%	\$ 1.480M	2.96%
ACCURACY	160 YDS	140 YDS	4.0%	143 YDS	\$ 1.681M	3.36%
TOTALS			15.0%		\$ 6.397M	12.79%

TOTAL CONTRACT PRICE = \$ 54.9M

IH0001A PAUSE ;PRESS <<ENTER>> TO CONTINUE.

in subsection 3.B.4.2. The process of data file transmission to the PROJ MNG disk provides those files for access by the monitor. In the monitor's session, all team data files which have been sent to the PROJ MNG disk are copied onto the monitor's personal disk for read/write use. This feature was also discussed in 3.B.5. and is covered in greater detail in Chapter 4 with the discussion of monitor functions.

2. Decision Submission

Submission of the team's decision occurs with the transmittal of the team files to the PROJ MNG disk. Within the data fields transmitted, there is a label field which contains the status of the team's work on a given decision point. This field is either blank, "00000000", "PROPCSED" or "FINAL". These labels are sequentially exclusive. They specify the exact status for the DP level of the set of database fields in which they are contained. Each DP level has four lines of data in the DATAFILE TEAMxx. The DP status label is the last field of data in a set of lines.

The DP field labels are assigned in the FINISH subroutine. By accepting the values in the table of proposed values for the contract proposal, Proposed Values Table (Table VI), or of proposed final values as the final approved contract, Final Contract Values Table (Table VII), the team causes the label to be written. These labels occur only when the team enters its security code to validate the proposed values. Completion of the value assignments to the database for transmittal appears with the table of accepted values, Table VIII .

The label is written as the last field in each of DP level database records. These labels can be seen in the 33rd field of each set of four lines of a DP data, Table IX.

TABLE VI

ENTER YOUR
:OR "CONT".

TABLE VII
Final Contract Values Table

NO.	ITEM	CURRENT VALUE	@ DP-3	CURRENT	TESTS	VALUE
1.	CONTRACT APPROACH	1	1	16.	GUIDANCE QUAL	6.00
2.	GUIDANCE CONFIGURATION	1	1	17.	FLIGHT TESTS	17.00
3.	GUIDANCE RELIABILITY	1	1	18.	MAXIMUM COST	53.00
4.	MAINTAINING DEVELOP	1.75	1.75	19.	MINIMUM COST	47.00
5.	VALUE ENG	0.00	0.00	20.	MAX COST WEEK	4.00
6.	PARALLEL RELIABILITY	0.00	0.00	21.	LATEST WEEK INCENTIVE	238.00
7.	MOTOF RELIABILITY	92.00	92.00	22.	EARLIEST WEEK INCENTIVE	202.00
8.	AIRFRAMER RELIABILITY	97.00	97.00	23.	MAX DELIVER RELIABILITY	3.50
9.	LAUNCHER RELIABILITY	98.00	98.00	24.	MAXIMUM RELIABILITY	81.00
10.	FIRE CONTROL ERROR YDS	80.00	80.00	25.	MINIMUM RELIABILITY	75.00
11.	GUIDANCE ERROR YDS	70.00	70.00	26.	MAX RELIABILITY INCENTIVE	3.50
12.	MOTOF QUAL TESTS	25.00	25.00	27.	MAXIMUM ERROR	160.00
13.	AIRFRAMER QUAL TESTS	6.00	6.00	28.	MINIMUM ERROR INCENTIVE	140.00
14.	LAUNCHER QUAL TESTS	3.00	3.00	29.	MAX ERROR INCENTIVE	4.00
15.	FIRE CONTROL QUAL TESTS	3.00	3.00	30.	WEEK FOR LOT 10	337.00
MUST CHECK THE ABOVE ENTRIES WITH THE APPROPRIATE FINAL DECISION:						
YOU MUST CHECK THE ABOVE ENTRIES WITH THE APPROPRIATE FINAL DECISION:						
ENTER YOUR TEAM SECURITY CODE IF YOU WISH TO COMMIT TO A DECISION:						
OR "CONT"						

TABLE VIII
Accepted Values Table

```

*****THE FOLLOWING PARAMETERS WERE ACCEPTED*****
AS TEAM 3 DP-3 PROPOSED INPUT NO. ITEM*****CONTRACTOR.
*****CURRENT VALUE @ DP-3 16. GUIDANCE QUAL TESTS : 6.00
1. CONTRACTOR APPROACH : 1 : 17.00
2. GUIDANCE CONFIGURATION : 1 : 53.00
3. GUIDANCE CONFIGURATION : 1 : 47.00
4. MAINTAINING AILTY ENG : 1 : 4.00
5. VALUE ENG DEVELOP : 0.0 : 238.00
6. PARALLEL DEVELOP : 0.0 : 202.00
7. MOTOR RELIABILITY : 92.00 : 3.50
8. AIRFRAMES RELIABILITY : 97.00 : 81.00
9. LAUNCHER RELIABILITY : 98.50 : 75.00
10. FIRE CONTROL ERROR : 80.00 : 3.50
11. GUIDANCE ERROR : 70.00 : 160.00
12. MOTOR QUAL TESTS : 25.00 : 140.00
13. AIRFRAMES QUAL TESTS : 6.00 : 4.00
14. LAUNCHER QUAL TESTS : 3.00 : 337.00
15. FIRE CONTROL QUAL TESTS : 3.00 : 10
*****SEE YOUR MONITOR IF YOU HAVE A PROBLEM.*****

```

IHO001A PAUSE ;PRESS <ENTER>> TO CONTINUE.

TABLE IX

3	3	112	00	1	75	0	75	0	0	92	00	97	00	58	50	80	00	70	00
238	00	6	00	2	00	3	50	3	00	75	00	17	00	53	00	47	00	4	00
160	00	1	40	0	00	4	00	3	37	6	00	FINAL	3	50					
3	4	112	00	1	75	0	75	0	0	92	00	97	00	58	50	80	00	70	00
238	00	6	00	2	00	3	50	3	00	75	00	17	00	53	00	47	00	4	00
160	00	1	40	0	00	4	00	3	37	6	00	PROPOSED	3	50					
0	0	000	00	0	00	0	00	0	0	0	00	0	0	0	00	0	00	0	00
		0	00	0	00	0	00	0	0	0	00	0	0	0	00	0	00	0	00
		0	00	0	00	0	00	0	0	0	00	0	0	0	00	0	00	0	00
1981	P	0	00	0	00	0	00	0	0	0	00	00000000							

Use of the security code in this feature was designed for 'human factor' engineering. The conscious process of entering the team's security code indicates the user's deliberate entry of the data. Inadvertent acceptance of the database for the team's position cannot occur.

The team does retain the responsibility for reading and understanding the data displayed in the table of data, Table VII.

The process of the team submitting a final contract proposal provides the program with the position agreed to during contract negotiations. Submission of the final report indicates to the program that the team has completed a DP level. PROJMG assumes the team is ready to proceed to the next DP level, for example DP-4. It utilizes the previous DP level data entries from the submitted final report and prepares the new DP data table. The team is sequenced to the next higher level whenever a final proposal is detected for a given DP level.

I. CALCULATION CORRECTION

As discussed in Chapter 2 several of the computations which were designed for CNP to accomplish were not satisfactory. Consequently their display was suppressed.

In PROJMG these calculations have been reformulated. Their algorithm is based on the linear extrapolation of a fee over the range of its allotted parameter values. The contract cost for the fee incentive is calculated by:

1. Assume achieving optimum performance results in total fee payment.
2. Assume achieving the worst performance expected results in no fee payment.

3. Determine the ratio of satisfactory achievement; the difference between the achieved value and the worst value expected divided by the range of value between worst and optimum values expected.
4. Multiply the ratio determined by the maximum fee percentage offered for the incentive.
5. Multiply the resulting percentage by the target development cost, the average value of the maximum and minimum desired development cost.

The following equation is used in PROJMG to determine fee ratio.

$$\text{FEERATIO} = \text{FEEMAX} * (\text{abs}(\text{WORST} - \text{ACHIEVED})) / (\text{RANGE}) \quad (\text{eqn 3.1})$$

This formulation for fee costs has proven to be more satisfactory than that of CNP.

The difficulty in CNP arises from numerical manipulation. Parameters in CNP frequently have their power-of-ten changed to permit calculations at different resource multiples. This is particularly true in the case of achieved development cost. Frequent multiplication by 1,000,000 with division by 1,000,000 to return the dollar values to millions of dollars incurs computer roundoff. The roundoff errors when repeated several times become sufficiently large to be apparent in the ratio calculations.

PROJMG does not use the numbers which the computer used in computing achieved value resources to determine the fee values. Its numbers are taken directly from the database array, and are only mathematically manipulated by this one calculation. This process results in only one iteration of roundoff error.

PROJMNG is not able to eliminate computer roundoff error, but the roundoff effect has been significantly reduced.

J. SUMMARY

PROJMNG's design has its foundation in CNP's routines which were designed around DMS. It uses this foundation from CNP, and develops an extended program. The program extension uses the current IBM-3033 facilities to drive user-directed options in a manner which provides the team control over program execution for completion of the DMS exercise. These features for team use of PROJMNG make PROJMNG easier to use in analyzing team positions. Chapter 4 discusses the analysis capabilities of PROJMNG which are available to the monitor.

IV. PROJING MONITOR OPERATION

A. CNP MONITOR UNIQUE FUNCTIONS

CNP has one routine capable of providing support to the monitor for his evaluations of the scenario progress. This singular routine is not available to the teams. The monitor can access the COST + FACTORS Table, Table X as discussed in the previous chapter. Access to the monitor routine occurs by the computer operator's response to the query for "COST + FACTORS ?". This response is coded to provide security of the COST + FACTORS table from team access. In order to preserve its integrity, the security code will not be given or demonstrated here. Correct security code responses to both the first "COST + FACTORS ?" and a second query of "COST + FACTORS ?" are necessary. The first response is a three character code word. The second query response is based on a data field used in the computations which has already been entered for the session. It constitutes a variable security code.

In CNP, an additional monitor capability is provided through the confirmation query, "CONFIRM ?", at the DP number verification sequence, "IS THIS DP NUMBER ** xx;"Y"=YES OR "N"=NC ?". Response to this query with a fixed single letter, places the operator in the "INSTRUCTOR" mode. Analysis of the program's Fortran code indicates that the INSTRUCTOR mode simply suppresses the queries for confirmation and the repeat back display of input data. Answering the statements "IS THIS DP NUMBER ** xx;"Y"=YES OR "N"=NC ?" with a "Y" reduces the displayed queries. It eliminates the repeated displays of "RECEIVED: xxxxx; CONFIRM ?". It also eliminates the repeat back printout of input page data.

TABLE X
COST + FACTORS

CCSIS OF TESTS PER UNIT, THOUSANDS OF DOLLARS

QUALIFICATION TESTS
MOTO F:
32.034
AIR FRAME:
126.355
GUIDANCE:
309.906
FIRE CONTROL:
938.186
LAUNCHER:
424.555
FLIGHT TESTS:
316.555

DESIGN FACTORS ACHIEVED
COMPONENT TABLE ROW FACTOR ACHIEVED

1	25	MOTO F: 0.825
0.877		
2	15	AIR FRAME: 0.958
0.980		
5	22	GUIDANCE A: 0.666
0.630		
6	25	GUIDANCE E: 0.580
4	16	FIRE CONTROL: 0.845
0.802		
3	7	LAUNCHER: 0.963
0.988		

The loss of page data repeat back and confirmation, and the format of Table X have not been appealing to users of the program. The "T" letter code system was not human factor engineered. Loss of the data item confirmation sequence increases the occasions when inaccurate results are received in the Development Contract Summary Table. Users

have not found benefit in these options. Additional experience with CNP is necessary to use these features successfully. Consequently, use of the "T" feature has not been used. The COST + FACTORS table has been found to be less than adequate.

B. PROJENG MONITOR UNIQUE FUNCTIONS

In PROJENG a variety of options are available for the monitor. They include:

1. Monitor Unique Security System
2. Query Suppression for the Monitor
3. Monitor's MAIN MENU
4. File Access Options
5. COST + FACTORS
6. Sensitivity Analysis
7. Graphing
8. Copies of Student Files

This chapter will review the intended capability and operation of these monitor unique functions.

C. MONITOR UNIQUE SECURITY SYSTEM

The PROJENG EXEC routine looks through the computer operator's disk for a file titled "DATAINST DISKNUM". Having located this file, the computer assumes that the operator is a monitor, and proceeds to validate the monitor identity. The program contains three file checks to determine if the operator is a monitor. Unsuccessful completion of the first two checks will cause the computer to ignore the DATAINST DISKNUM file and assume the computer user is a student.

The first of these checks is the file format. The file first line must contain the userid. The second line must contain a monitor specified security code. This monitor security code is queried in the same manner in which the team security code is queried for a team disk.

The third verification is a check of the stored monitor security code against the individual's ability to reproduce that code. A code match allows the program to proceed. Failure to accurately match the code causes the program to stop. PROJMNNG EXEC takes control to finish its execution. The user can reaccess PROJMNNG FORTRAN in his response to "ANOTEER RUN, 'Y'/'N' ?".

D. QUERY SUPPRESSION FOR THE MONITOR

Successfully accessing the program as a monitor suppresses the student's Main Menu, discussed in chapter 3. Consequently, the monitor does not receive the student's Main Menu after each alteration of the Selection Menu. Based on the assumption that the computer operator is a monitor the program immediately follows the Selection Menu with the Development Contract Summary, Table XI.

Table XI demonstrates the display sequence for the Selection Menu with the execution of option "0". The monitor's response to the Selection Menu with a zero causes the immediate execution of the achieved value calculations. Other selections from this table are performed similarly to those done for the team. The changes entered by the monitor in this first mode will not be stored in the database by the program. In monitor mode 1, the call to the routine STORE does not occur. Only the array of data in the disk's buffer will be altered. The monitor sees the computer response to his entry of a "0" as an immediate display of the Development Contract Summary.

TABLE XI
SELECTION MENU

NO.	ITEM	CURRENT VALUE @ DP-3	NO.	ITEM	CURRENT VALUE
1.	CONTRACT APPROACH	1	16.	GUIDANCE QUAL TESTS	6.00
2.	GUIDANCE CONFIGURATION	1	17.	FLIGHT TESTS	17.00
3.	GUIDANCE RELIABILITY	2	18.	MAXIMUM COST	53.00
4.	MAINTAINING ENG	1.75	19.	MINIMUM COST	47.00
5.	VALUE ENG DEVELOP	0.00	20.	MAX CCST INCENTIVE	4.00
6.	PARALLEL RELIABILITY	0.00	21.	LATEST WEEK	238.00
7.	MOTOR RELIABILITY	92.00	22.	EARLIEST WEEK	202.00
8.	AIRFRAME RELIABILITY	97.00	23.	MAX DELIVER INCENTIVE	3.50
9.	LAUNCHER RELIABILITY	98.50	24.	MAXIMUM RELIABILITY	81.00
10.	FIRE CONTROL ERROR	80.00	25.	MINIMUM RELIABILITY	75.00
11.	GUIDANCE ERROR	70.00	26.	MAXIMUM RELIABILITY	3.50
12.	MOTOR QUAL TESTS	25.00	27.	MAXIMUM ERROR	160.00
13.	AIRFRAME QUAL TESTS	6.00	28.	MINIMUM ERROR	140.00
14.	LAUNCHER QUAL TESTS	3.00	29.	MAX ERROR INCENTIVE	4.00
15.	FIRE CONTROL QUAL TESTS	3.00	30.	WEEK FOR PAGE	337.00
		*****31.		CHANGE BY PAGE	

SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-3.

<==== "NOTE: The query '?' response "0".

DP-3 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 3

INCENTIVE PROVISIONS		INCENTIVE ACHIEVEMENTS	
INCENTIVE AREA	WORST VALUE	BEST VALUE	MAX FEE ALLOWED
DEV. COST	\$ 53.0M	\$ 47.0M	4.0%
FLT TST COMPI	238 WK	202 WK	3.5%
RELIABILITY	75.0%	81.0%	3.5%
ACCURACY	160 YDS	140 YDS	4.0%
TOTALS			15.0%
TOTAL CONTRACT PRICE			= \$ 54.9M

E. MONITOR'S MAIN MENU

The PROJMNNG monitor's MAIN MENU contains options to allow the monitor access to any of the routines in the program. It has a unique feature in its flexible length. This feature provides menu listings to the monitor which are determined by his status in the program. If the monitor has just entered PROJMNNG or has changed the team number, and has not completed the computations for other display values, the program will suppress those options from the menu display. Options which require calculations or selections which have not been processed for the team are assumed to present results which are either not applicable or misleading.

```
MAIN MENU:
1. CHANGE TEAM NUMBER
2. THE DP SELECTION QUERY
3. THE INPUT SELECTION MENU
4. RERUN THE DATA CALCULATION
5. DO A SENSITIVITY ANALYSIS
6. PRINTOUT COST+FACTORS
7. EXIT
```

Figure 4.1 Monitor's Main Menu.

Figure 4.1, demonstrates the basic monitor's Main Menu.

1. Change Team Number

Option 1 when selected will return the monitor to the program's team input routine (TEAMIN). The execution of this routine begins by displaying a menu, Figure 4.2. The selections on this menu are:

```

SELECT WHICH INSTRUCTOR MODE YOU DESIRE.
1. RUN TEAM SCENARIOS WHICH ARE ON THE MONITOR'S DISK.
2. CHANGE TEAM FILES.
   THIS MODE WILL ALTER THE STUDENT'S FILES.
3. RUN THE MONITOR'S TEAM FILE 'TEAM21'.
4. EXIT

```

Figure 4.2 Monitor Mode Menu.

a. Run Team Scenarios from the PROJMG Disk

Selecting this choice enables the monitor to run evaluations and displays of the teams whose DATAFILES have been written on the PROJMG disk. PROJMG has at this point already written these files to the individual monitor's disk for use. As the monitor selects a team to review, the program determines if the team file exists. If the file exists, the monitor may proceed with the evaluation. If no files exist for the team requested, PROJMG responds with 'FILES DO NOT EXIST FOR TEAMxx', and proceeds to ask for a different team number, 'WHAT TEAM NUMBER?', until a team number is submitted for which files are present.

This option selects a mode of operation in which team data files are not changed by the monitor. The menus displayed are only those associated with the monitor's options. The Monitor's Main Menus are flexible menus. The discussion of this flexibility and of the basic menu will be provided in the following sections of this chapter. These differences are demonstrated by the two monitor menu displays for mode 1, Figure 4.3, and for mode 2, Figure 4.4

```

MAIN MENU:
1. CHANGE TEAM NUMBER
2. THE DP SELECTION QUERY
3. THE INPUT SELECTION MENU
4. RERUN THE DATA CALCULATION
5. DO A SENSITIVITY ANALYSIS
6. PRINTCUT COST+FACTORS
7. EXIT

```

Figure 4.3 MAIN MENU in Mode 1.

```

MAIN MENU:
1. CHANGE TEAM NUMBER
2. THE IP SELECTION QUERY
3. THE INPUT SELECTION MENU
4. RERUN THE DATA CALCULATION
5. DO A SENSITIVITY ANALYSIS
6. PRINTCUT COST+FACTORS
7. FILE TEAM PROPCAL
8. EXIT

```

Figure 4.4 MAIN MENU in Mode 2.

e. Change Team Files

Option 2 has an entirely different affect on the database. Again, the files accessed are those previously in existence. The monitor receives a different version of the MAIN MENU, The added option in this menu, "7. FILE TEAM PROPOSAL" permits the monitor to change the DP report level for any of the team's DPs irregardless of the team's current level.

The monitor can perform all of the functions he could in mode 1 plus changing the report level and placing these selections in the team database. This mode should only

be used to correct team errors. Team database manipulation for any other reason will lead to team confusion.

The query which finalizes the replacement of the team database, "SHOULD THE REWRITTEN TEAM xx FILE BE SENT TO THE PROJ MNG disk: Y/N ?". Unless this query is answered in the affirmative, the mode 2 capability to overwrite the team file is not performed.

Answering yes to the query to rewrite the team file directs PROJ MNG through a routine which performs three calls to the subroutine FRTCMS which permits CMS executive level commands to be executed from within a Fortran program. The three commands in this routine are the same as the spool commands in PROJ MNG EXEC which send the file DATAFILE TEAMxx to the PROJ MNG disk. The disk punch file is spooled as a class x file. The DATAFILE TEAMxx is punched into the spool. Finally the spool is closed and sent to the PROJ MNG disk.

The program does allow the DATAFILES on the monitor's personal disk to be rewritten whenever mode 2 operation is selected. Transmission of these files to the master PROJ MNG disk can be accomplished after responding in the negative to the query "SHOULD THE REWRITTEN TEAM xx FILE BE SENT TO THE PROJ MNG disk: Y/N ?". Sending these files to the master disk must be a deliberate act of the monitor. The monitor may either use one of the macro routines in general usage on the IBM-3033 for sending files between users, "SEND". Or, he may spool the files for immediate read by the PROJ MNG file using the following sequence of steps. (The lower case characters in the following command demonstration are optional inputs.)

1. "SPool PUNCH CCNT CLASS X 0276P"
2. "PUNCH DATAFILE TEAM (XX)", where xx is the two digit team number.
3. "SPool PUNCH CLOSE NCCNT"

In mode 1 or mode 2 operation, only the Development Contract Summary table resembles the same display for the teams.

c. Run the monitor's team file "TEAM21"

The third option provides a team file for the monitor. TEAM21 is the monitor's own team file to simulate the team interaction with the program. This provides a scratch pad file for the monitor's use.

d. EXIT

Option 4 from the monitor's Main Menu stops the program. Entering a '4' in response to this menu directs the program to stop. PROJNGN EXEC resumes control of the computer processing.

2. The DP Selection Query

Number 2 on the monitor's mode menu reloops the program to the DP query for which DP the monitor desires to evaluate. This query will not prevent the monitor from accessing DP levels for which the teams have not submitted files. The level which the team has completed is displayed on the Selection Menu, with the DP query, and on the proposed report table. This information should be known by the monitor based on the progress of the DMS scenario.

3. The Input Selection Menu

Again, the menu routine simply reloops the program execution to a previously executed question to permit the monitor to change his mode of analysis. In this case he returns to the team Selection Menu. The last team and DP inputs entered during the monitor's session will apply to the team database which is displayed.

Charges to a team's list of data in the program buffer which have been entered during a session remain applicable until the team number is changed. Any variations to the Selection Menu which have been selected for the team currently being evaluated will be in effect. In mode one, a team's data will revert to the PROJ MNG disk's team datafile values when changing team numbers. Mode 2 will keep the changed values until the PROJ MNG disk values are reccpied down onto the monitor's disk.

4. Rerun the Data Calculation

This option causes PROJ MNG to redo the achieved value calculations. The display for this selection is a repeat of the Development Cost Summary.

5. Do a Sensitivity Analysis

This option creates a major option for exploration by the monitor. It is discussed as part of the sequential demonstration of the monitor functions which follows in Section I below.

6. Printout Cost + Factors

Similar to the capability for COST + FACTORS demonstrated by CNP, PROJ MNG has an equivalent feature for the monitor. Table XII shows the improved layout of the data. The table has the benefit over CNP's version by not requiring a separate code to access it. The table is simply accessed as a menu selection from the monitor's Main Menu.

7. EXIT

This is another example of PROJ MNG's ability to stop the program.

TABLE XII
COST + FACTORS

COSTS OF TESTS PER UNIT (IN THOUSANDS OF DOLLARS)

QUALIFICATION TESTS

MOTOR	:	32.034
AIR FRAME	:	126.355
GUIDANCE	:	309.906
FIRE CONTROL	:	938.186
LAUNCHER	:	424.555
FLIGHT TESTS	:	316.855

DESIGN FACTORS ACHIEVED

CCOMPONENT	TABLE	ROW	FACTOR	ACHIEVED
MOTOR	1	25	0.825	0.877
AIR FRAME	2	15	0.958	0.980
GUIDANCE A	5	22	0.666	0.630
GUIDANCE B	6	25	0.580	0.0
FIRE CONTROL	4	16	0.845	0.802
LAUNCHER	3	7	0.936	0.988

IHC001A FAUSE ;PRESS <<ENTER>> TO CONTINUE.

8. Flexible Menu

The options presented in this section are the basic capabilities of PROJMN. However, additional capabilities become possible as the evaluation progresses. With the completion of the sensitivity analysis additional tables and graphs are within the capability of PROJMN. These flexible menu capabilities are explained and demonstrated in Section I of this chapter.

P. MONITOR SESSION PROCEDURES

The remainder of the discussion on menus will be presented sequentially in the following sections. Their relative position in the textual sequence corresponds to their occurrence in the process of stepping through the program's routines.

These sequential steps of PROJMNG are demonstrated as the program would be operated. The material in this demonstration is preliminary to the preparation of an instructor's manual. The sections are:

1. Initialization
2. Basic routines of the Main Menu
3. Sensitivity analysis
4. Logoff unique routines

G. INITIALIZATION

1. PROJMNG EXEC Operation

The monitor's access to the PROJMNG package of programs is also accomplished by LINKPROJ. The determination that the disk accessing PROJMNG is a monitor is made, as discussed in Chapter 3. Having completed the checks of the monitor's file DATAINST DISKNUM, the PROJMNG EXEC program conducts a check of files for a DATAFILE TEAM21. This file must be present on the monitor's disk if he is going to use it in PROJMNG. PROJMNG EXEC either finds the file or creates one in order to have the file ready for PROJMNG FORTRAN's read and write instructions.

2. Available Files

The lack of presence of the files which PROJMNG EXEC looks for is an automatic function of the IBM-3033. This capability was retained to display to the monitor a list of teams from which he has not received reports, as shown in Figure 4.5. Offsetting of the display from the left margin was performed in order to highlight between the computer printout and the typed queries.

A benefit of the "PRESS <<ENTER>>" set of statements is the ability to begin a display on a fresh screen. Displays which should appear as one screen occur more readily when the pause has finished the previous display to keep it displayed while the user is reading it. The pause can be followed by an immediate command in FRTCMS to 'CLRSCFN ', clear the screen. This sequence of commands leaves a fresh screen for the next screen display. The first line that PROJMG prints out appears as the top line of a display (see Figure 4.7).

Cueing the user to press <<ENTER>> identifies a pause which has been placed in order to organize screen displays. The pause in Figure 4.5 enables the display of the teams 'not found' to be isolated on one display. There can be up to 21 teams displayed, as for example at the start of an exercise scenario. Without the pause to segregate the displays, all 21 teams would not appear on one screen. The single screen full of one type of data aids in user ability to study and analyze the data. Splitting the display to a second view eliminates the first screen's data and taxes the user's recall to do any comparisons. Isolating the data onto one screen eliminates these disruptions to the user's study of the data. This technique of segregating related information onto a screen has been used throughout this program.

3. Purging DATAFILES

In order to provide a convenient method of purging files from the completed exercises, a routine was generated which permits the monitor to purge all of the existing files, to conduct normal operation of the program or to exit. These options are depicted in the display as shown in Figure 4.6. The purge option purges only the available 'A' disk DATAFILE TEAMxx and DATACODE TEAMxx. Should the

```

BEGIN RECORDING OF TERMINAL SESSION
* NOTE!: YOU ARE NOW LINKED TO PROJ MNG
              ON YOUR 192 DISK, MODE C
* NOTE!: PRESS PFC9 TO BREAK THE LINK
          PRESS <<ENTER>> WHEN YOU ARE READY TO CONTINUE.

```

```

*****...EXECUTION IS IN PROGRESS.
          WAIT. DO NOT PRESS ENTER.*****
FILE 'DATAFILE TEAM21 A' NCT FOUND.
FILE 'DATAFILE TEAM11 C' NCT FOUND.
FILE 'DATAFILE TEAM12 C' NCT FOUND.
FILE 'DATAFILE TEAM13 C' NCT FOUND.
FILE 'DATAFILE TEAM14 C' NCT FOUND.
FILE 'DATAFILE TEAM15 C' NCT FOUND.
FILE 'DATAFILE TEAM16 C' NCT FOUND.
FILE 'DATAFILE TEAM17 C' NCT FOUND.
FILE 'DATAFILE TEAM18 C' NCT FOUND.
FILE 'DATAFILE TEAM19 C' NCT FOUND.
FILE 'DATAFILE TEAM20 C' NCT FOUND.

```

Figure 4.5 Initial Monitor Display.

PROJ MNG EXEC purge function be performed on the PROJ MNG FORTRAN's disk the entire exercise scenario would be eliminated. Only database files pre-existing on team files or monitor's files could be reaccessed for PROJ MNG. With a new play of the exercise, it is expected that a new group of teams will be accessing PROJ MNG. No conflict with former team databases should be experienced once the monitor's and the PROJ MNG's disks have been purged.

Normal operation initiates PROJ MNG FORTRAN execution. The standard procedure during a game exercise is designed to respond to this query with an "N". The word length difference between FURGE and N has been designed to make inadvertent destruction of the team files more difficult.

"E" again provides a means of stopping the program. In this instance, "E" will stop PROJ MNG EXEC and return the computer to running the LINKPROJ program. The final display

of LINKPROJ, "ANOTHER RUN, 'Y' OR 'N'?", would be the query following the request to stop.

"WAIT FOR EXECUTION TO BEGIN" became a necessary printout as the result of long waits for execution to begin on the NPS IBM-3033. (One monitor reported experiencing a 20 minute delay during very heavy use of the computer facility, with the "WAIT FOR EXECUTION..." label reminding him that the program was in the queue to be run.)

```
*****WARNING*****
*****ANSWERING 'PURGE' TO THE FOLLOWING QUERY WILL*****
*****PURGE ALL EXISTING STUDENT DATAFILES*****
*****ON THE DISK YOU ARE USING.*****

DC YOU WISH TO INITIALIZE ALL TEAM FILES
  FOR A NEW IMS EXERCISE?
'PURGE' =PURGE ALL TEAM DATAFILES
'N' =PROCEED WITH NORMAL OPERATION
'E' =STOP

====>n
EXECUTION BEGINS...      WAIT FOR "EXECUTION TO BEGIN".
```

Figure 4.6 Monitor Session File Purge Option.

These initial queries, printouts and displays by PROJMG in the monitor mode of operation should appear to the reader as very similar to those presented to the teams. It should also be noticed in Figure 4.7 that the program does not ask for a team security code input to create the team's code. Instead, it has already received the monitor's security code from PROJMG EXEC. The monitor's code is found by PROJMG FORTRAN in DATAINST DISKNUM.

H. BASIC PROGRAM ROUTINES

The program routines discussed in this section perform similar processes in both CNP and PROJMG.

1. Start of PROJMNNG's Run

The stacked data items are read by PROJMNNG as its first step in the routine for team number input "TEAMIN".

```
*****
      TO TERMINATE THE PROGRAM AT ANY POINT,
      TYPE "E"=EXIT IN RESPONSE TO ANY "YES/NO" QUERY.
*****
***CAUTION***IN RESPONSE TO A QUERY, PRESSING <<ENTER>>
      WITHOUT PROVIDING DATA WILL DUMP THE PROGRAM.
*****
?
PLEASE ENTER YOUR SECURITY CODE.
```

Figure 4.7 PROJMNNG Initial Displays. Having received the FL21 flag of TEAM21, and validated the monitor's security code, PROJMNNG can proceed. IF the security code should be entered incorrectly, the PROJMNNG execution is terminated. The monitor receives a warning of the error, and is asked to reinitiate his startup of the program. He may either answer 'Y' to "ANOTHER RUN, 'Y' OR 'N' ?" or reenter "LINKPROJ" after PROJMNNG EXEC and LINKPROJ EXEC have finished execution.

PROJMNNG FORTRAN asks for the team number when a monitor is running the program. Team selection within the program is necessary in order to permit the evaluation of more than one team. Whereas a team will only enter the program with one team scenario to perform, the monitor is able to change teams without reinitiating the program.

Figure 4.8 demonstrates the response given by PROJMNNG when it cannot find files for the team number requested in either options 1 or 2 of the monitor's Main Menu. The read cue '?' indicates to the operator that an

input is expected. In the operations demonstrated throughout this thesis '====>' indicates the user input in response to the PROJMNNG queries. Figure 4.8 also presents a case in which the program does find the requested files and proceeds to query the monitor for which DP level he desires to evaluate.

```
WHAT TEAM NUMBER?  
?  
====>2  
RECORDS DO NOT EXIST FOR TEAM 2  
WHAT TEAM NUMBER?  
?  
====>3  
WHAT DECISION POINT DO YOU WISH TO ANALYZE?  
?  
====>3
```

Figure 4.8 Team Number Responses.

The queries illustrated in Figure 4.8 complete the functions of the program routine TEAMIN. The program returns to the main program text and enters the SELECT routine which displays the Selection Menu as it did for the teams, Table XIII. Table XIII functions the same as before except that the entries are not stored in the team's database file. The changes are made only to the monitor's disk data buffer arrays. In option 2 of the monitor's main menu, the changes are also stored on the monitor's copy of the team files in his file listing. These changes are not applied to the PROJMNNG disk files unless the monitor specifically sends the changed files to that disk.

In response to option '0' in Table XIII PROJMNNG immediately completes the calculation of fee percentages in the program routine PRCTCK. Successful completion of the percentage check is followed with the calculation of achieved values. This process occurs with the program calls to the routines:

TABLE XIII
Selection Menu

NO.	ITEM	CURRENT VALUE @ DP-3	CURRENT TESTS	VALUE
1.	CONTRACTOR APPROACH	1		6.00
2.	GUIDANCE CONFIGURATION	1		17.00
3.	GUIDANCE CONFIGURATION	2		53.00
4.	MAINTAINABILITY ENG	1.75		47.00
5.	VALUE ENG DEVELOP	0.75		4.00
6.	PARALLEL RELIABILITY	0.0		238.00
7.	PARTIAL RELIABILITY	92.00		202.00
8.	AIRFRAME RELIABILITY	97.00		3.50
9.	AIRFRAME RELIABILITY	98.50		81.00
10.	FIRE CONTROL ERROR	80.00		75.00
11.	GUIDANCE QUAL TESTS	25.00		3.50
12.	NOTOR QUAL TESTS	6.00		160.00
13.	AIRFRAME QUAL TESTS	3.00		140.00
14.	LAUNCHER QUAL TESTS	3.00		4.00
15.	FIRE CONTROL QUAL TESTS	3.00		337.00
16.	ITEM			
17.	FLIGHT TESTS			
18.	MAXIMUM COST			
19.	MAXIMUM COST			
20.	MAXIMUM COST INCENTIVE			
21.	EARLIEST WEEK			
22.	MAXIMUM WEEK INCENTIVE			
23.	MAXIMUM RELIABILITY			
24.	MAXIMUM RELIABILITY			
25.	MAXIMUM RELIABILITY			
26.	MAXIMUM RELIABILITY			
27.	MAXIMUM RELIABILITY			
28.	MAXIMUM RELIABILITY			
29.	MAXIMUM RELIABILITY			
30.	MAXIMUM RELIABILITY			
31.	MAXIMUM RELIABILITY			

1. ZEW - sets all non-database variables used in the computations to zero.
2. INPUT3 - reads the data tables in files FT17F001, FT18F001, FT19F001 and FT20F001
3. PRESET - uses the database values to make initial adjustments to the achieved value indices
4. TRADE - evaluates the team's input tradeoffs and uses them to generate affects which to apply in calculating the achieved value indices
5. DESSES - degrades the performance factors based on requests for system test.
6. DESRET - degrades the performance for the effects of selected engineering values and performs the calculations for production costs. (This feature, production cost computation, was not used in CNP and has not been activated in PROJNNG.)
7. SE - applies resource parameters to the achieved value calculations.
8. MCD9 - combines the calculation results from the previous three routines to derive a total cost factor for development.
9. RERUN - selectively sets recursive variables to zero to enable the program to finish the calculations of one DP and to reloop to compute the values for successive DP's
10. REPORT - completes the calculations after all DP level computations have been completed.
11. PECSUM - performs conversion of the achieved values to the appropriate power of ten for display to the user and generates the display, Table XIV.

2. Summary of Basic Capabilities

To this point, the basic capabilities of PROCJMNNG which it has in common with CNP have been discussed. The improved displays, access and queries discussed parallel those features given to the students, with exception of student access to COST + FACTORS. In the next section, the 'flexible menu' feature of the monitor's main menu, and the associated capabilities for sensitivity analysis, for data displays, and for analysis graphing are described.

I. SENSITIVITY ANALYSIS ROUTINES

1. Sensitivity Analysis Basic Menu Options

Figure 4.9 shows again the monitor's menu in mode 1 operation for comparison to the expanded menu which is now described. In order to accomplish the calculation of a sensitivity analysis, option 5 from the menu is selected. In performing this selection, a flag is set 'on' to indicate the availability of values for use in the other displays of the expanded menu options.

```
MAIN MENU:
1. CHANGE TEAM NUMBER
2. THE DP SELECTION QUERY
3. THE INPUT SELECTION MENU
4. REBUN THE DATA CALCULATION
5. DO A SENSITIVITY ANALYSIS
6. PRINTOUT COST+FACTORS
7. EXIT
```

Figure 4.9 Monitor's Main Menu.

Selection of number 5 from the menu immediately accesses the programs routine SNSITIV. This subroutine provides a selection menu (Table XV), queries parameters and parameter data for variation in the analysis, edits the parameter data to remain within DMS scenario limitations, divides the variable parameter into 10 steps for 11 levels of evaluation, runs eleven cycles through the achieved values calculations, displays the table of achieved values at each level, and stores these values in an array for later use in redisplaying the table and for plotting the computations.

Following the display of Table XV, the monitor makes a selection of which DMS data item to analyze. His selection determines a series of questions which PROJMNNG displays in order to resolve the analysis parameter details. Figure 4.10 shows one of these queries. In this case the Flight Tests will be varied. The maximum and minimum values permitted are displayed with the queries for range values. These are not only cues to the user but also are the limitations which PROJMNNG imposes.

If the selection exceeds the scenario design limits, PROJMNNG will take one of two actions. In the first case, PROJMNNG will reset the value to be within the range

```
====>14
      INPUT THE VALUES TO VARY FLIGHT TESTS BETWEEN.
      WHAT IS THE LOWER VALUE: >10. ?
      ?
====>10
      WHAT IS THE UPPER VALUE: <25. ?
      ?
====>25
```

Figure 4.10 Flight Tests Demonstration.

TABLE XIV
Development Contract Summary

CP-3 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 3

INCENTIVE AREA	INCENTIVE PROVISIONS			INCENTIVE ACHIEVEMENTS		
	WORST VALUE	BEST VALUE	MAX FEE ALLOWED	ACHIEVED VALUE	FEE EARNED	FEE % EARNED
DEV. COST	\$ 53.0M	\$ 47.0M	4.0%	\$ 48.54M	\$ 1.486M	2.97%
FLT TST COMPI	238 WK	202 WK	3.5%	199 WK	\$ 1.750M	3.50%
RELIABILITY	75.0%	81.0%	3.5%	80.08%	\$ 1.480M	2.96%
ACCURACY	160 YDS	140 YDS	4.0%	143 YDS	\$ 1.681M	3.36%
TOTALS			15.0%		\$ 6.397M	12.79%
TOTAL CONTRACT PRICE	= \$ 54.9M					

TABLE IV
Sensitivity Analysis Menu

 ****SELECT THE ITEM WHICH IS TO BE SENSITIVITY ANALYZED.****

 1. MAINTAINABILITY ENG COST
 2. VALUE ENGINEERING COST
 3. PARALLEL GUIDANCE COST
 4. MOTOR RELIABILITY RANGE
 5. AIRFRAME RELIABILITY RANGE
 6. LAUNCHER/GSE RELIABILITY
 7. FIRE CONTROL IMPACT ERROR
 8. GUIDANCE IMPACT
 9. MOTOR QUAL TESTS
 10. AIRFRAME QUAL TESTS
 11. LAUNCHER/GSE QUAL TESTS
 12. FIRE CONTROL QUAL TESTS
 13. GUIDANCE QUAL TESTS
 14. FLIGHT TESTS
 15. DEVELOPMENT INCCENTIVE %
 16. DEV COST OF FLIGHT TEST RANGE
 17. END WK OF FLIGHT INCCENTIVE
 18. FLIGHT RELIABILITY RANGE
 19. SYS RELIABILITY INCCENTIVE %
 20. SYS RELIABILITY INCCENTIVE
 21. IMPACT ERROR RANGE
 22. IMPACT ACCURACY RANGE
 23. DEPLOYMENT DATE RANGE
 *****0. NONE*****

permitted. If the second case occurs, the user will be requiered for input. This occurs when PROJ MNG doesn't understand the limits requested.

2. Sensitivity Analysis Table

The Sensitivity Table, Table XVI provides a simple column display of the achieved values computed at each step of the analysis as a function of the related parameters. The steps are labeled in the lefthand column. The second column lists the value of the selected parameter for the step. Some of the parameters will result in duplicate steps. PROJ MNG divides the range of the parameter values by 10 which can result in a fraction. In creating a step for the table, the parameter value can be successively rounded to the same value. PROJ MNG is telling the monitor, it has recognized the parameter as requiring an integer value. It has provided the closest integer to the value. The duplication is the occurrence of the closest integer as the same number on successive occasions. The first occasion PROJ MNG rounded up to the value and on the second occasion it rounds down.

The remaining columns, except EI, are self explanatory. They correlate with the achieved values data given in the table of Development Contract Summary, Table XIV. EI is explained later in this section.

Upcn completion of the table, PROJ MNG displays the cue of "IH0001A PAUSE ;PRESS <<ENTER>> TO CONTINUE". This holds the display for review by the monitor and reminds the monitor of the action to perform after reviewing the table.

3. Menu Expansion

Pressing <<ENTER>> causes the program to continue. The sequential routing of PROJ MNG returns the monitor's MAIN

MENU display in the main program. Figure 4.11 demonstrates the flexible menu with the new options. These options are discussed as we proceed further into PROJMG.

```

MAIN MENU:
1. CHANGE TEAM NUMBER
2. TEE DP SELECTION QUERY
3. TEE INPUT SELECTION MENU
4. RERUN THE DATA CALCULATION
5. DC A SENSITIVITY ANALYSIS
6. ERINTOUT CCST+FACTORS
• 7. PRINT LAST SENSITIVITY TABLE.
• 8. PLOT LAST SENSITIVITY ANALYSIS ACHIEVED VALUES.
• 9. PLOT LAST SENSITIVITY ANALYSIS FEE PERCENTAGES.
• 10. PLOT WITH SELECTED Y-AXIS SCALES.
• 11. USE OPTIMUM EI VALUE FROM LAST SENSITIVITY
    ANALYSIS FOR DP-3 DATA.
12. EXIT

```

Figure 4.11 Monitor's Main Menu in Mode 1.

Number 7 on the menu returns to the last display of the Sensitivity Table viewed. PROJMG's subroutine SNSITIV is reentered through a call (entry) at the end of the subroutine, ENTRY SNSPRT (enter SNSITIV and print the table). SNSITIV's array of stored values enables PROJMG to reprint the table without any calculations.

4. Sensitivity Analysis Plotting

The second new feature available through the main menu is the capability to graph the table results. The stored values in the array are passed to the PLTSCH subroutine in a data buffer, COMMON FLOTVA. The graph is 20 lines high and fills the entire screen. Consequently a pause statement at the bottom of the graph could not appear on the same screen with the display.

TABLE XVI

Sensitivity Table

#	SENSITIVITY FLIGHT TESTS	DEVELOPMENT FEE %	FLIGHT FEE %	TEST DATE COMPLT	RELIABILITY FEE %	ACCURACY FEE YDS	TCT %	CONTRACT COST \$M	TEST EI
0	10.00	4.00	3.50	197	0.0	ACHIEVED	8.86	45.65	80.91
1	11.00	4.00	3.50	197	0.0	ACHIEVED	9.12	47.22	82.62
2	13.00	4.00	3.50	197	0.83	ACHIEVED	10.66	50.45	86.03
3	14.00	4.00	3.50	197	1.29	ACHIEVED	11.43	51.66	87.28
4	16.00	3.58	3.50	199	2.37	ACHIEVED	12.79	53.94	89.57
5	17.00	2.97	3.50	199	2.96	ACHIEVED	12.59	54.68	90.57
6	19.00	1.75	3.50	200	3.50	ACHIEVED	12.05	57.30	92.88
7	20.00	1.15	3.40	203	3.50	ACHIEVED	10.30	57.89	93.44
8	22.00	0.17	2.63	211	3.50	ACHIEVED	9.54	58.73	94.24
9	23.00	0.0	2.04	217	3.50	ACHIEVED	8.38	59.66	95.12
10	25.00	0.0	0.88	229	3.50	ACHIEVED			

IH0001A PAUSE ; PRESS <<ENTER>> TO CONTINUE.

IH0001A PAUSE ; PRESS <<ENTER>> TO CONTINUE.

In order to alleviate this difficulty the pause has been eliminated. The display appears on one screen with the addition of a blank line which forces the IBM to hold the display under the condition of a screen overflow, as indicated by the "MCRE" in the screen lower right corner. The screen can be held only temporarily in this IBM display mode. If the operator desires to extend the hold for closer analysis of the graph, pressing <<ENTER>> will change the IBM display mode to "HOLDING" which can remain indefinitely. Once the operator has finished with the "MORE" display mode, or the "HOLDING" mode he can continue by clearing the screen. On the IBM 3278 this is accomplished by pressing <<ALT>> and <<CLEAR>> simultaneously.

The definition key for characters is displayed on the graph, as given in Figure 4.12. A series of print statements has been placed in FLTSCH to list the key in the righthand graph margin.

```

• = DEVELOPMENT COST
T = FLIGHT TEST COMPLETION DATE
R = RELIABILITY
@ = ACCURACY
% = TOTAL FEE PERCENTAGE
$ = TOTAL COST
* = EFFECTIVENESS INDEX
& = Several curves intersect

```

Figure 4.12 Plot Character Key Definitions.

The table displayed by "8. PLOT LAST SENSITIVITY ANALYSIS ACHIEVED VALUES." divides the screen height of twenty lines into the largest value which is to be displayed. This functions as the ordinate scale for the graph, Figure 4.13. FLTSCH displays the entire set of data. In the next selection, "9. PLOT LAST SENSITIVITY ANALYSIS

?====>8		<====NOTE This is the selection from the MAIN MENU to view this graph.									
230.00	+	.	.	.	.	.	.	.	.	.	.
210.00	+	.	.	.	.	.	.	.	.	.	.
190.00	+	.	.	.	.	.	.	.	.	.	.
170.00	+	.	.	.	.	.	.	.	.	.	.
150.00	+	.	.	.	.	.	.	.	.	.	.
130.00	+	.	.	.	.	.	.	.	.	.	.
110.00	+	.	.	.	.	.	.	.	.	.	.
90.00	+	.	.	.	.	.	.	.	.	.	.
70.00	+	.	.	.	.	.	.	.	.	.	.
50.00	+	.	.	.	.	.	.	.	.	.	.
30.00	+	.	.	.	.	.	.	.	.	.	.
10.00	+	.	.	.	.	.	.	.	.	.	.

• = DEV COST
T = TESTCOMP
R = RELIABIL
@ = ACCURACY
\$ = TOTICOST
* = EFFINDEX
E = MULTPTS

Y-AXIS = UNITS
X-AXIS = TESTS

Figure 4.13 Sensitivity Table.

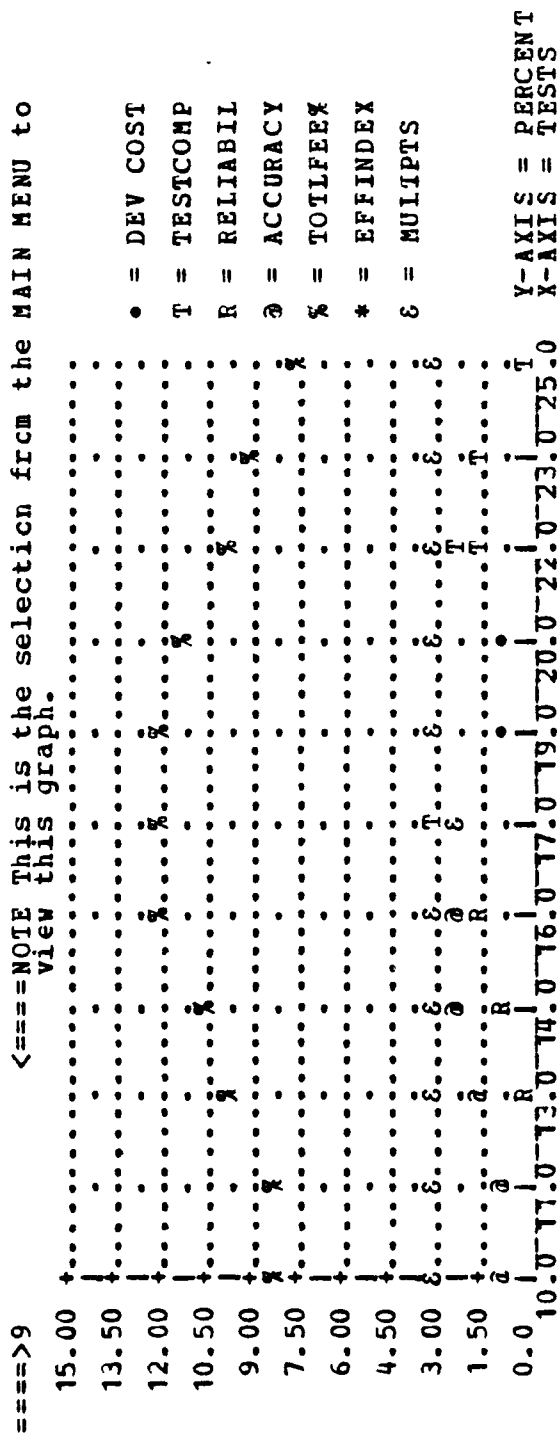


Figure 4.14 Sensitivity Analysis Percentages Graph.

FEE PERCENTAGES." the vertical axis is provided with a scale of 0.5 per line. The height of the ordinate becomes 10, which will display the fees up to 10%. The benefit of this change in scales is in the presentation of a magnified view in the y-axis between the range of 0 and 10 where the fee percentages are concentrated. Figure 4.14 demonstrates the percentage graph.

Graphing can also be scaled by the monitor. Choosing Sensitivity Menu option number 10 directs PROJMNNG to ask for the ranges which are to be used for the y-axis. Figure 4.15 demonstrates the interactive exchange between the program and user to create the desired y-scale. The program accepts any values for the minimum and maximum values for the axis. The program again divides this range into 20 steps. The step size becomes the y-axis scale. If the scale is less than 0.05 per line, PROJMNNG resets the scale to 0.05. Figure 4.16 demonstrates the graph resulting from the steps in figure 4.15.

```

MAIN MENU:
1. CHANGE TEAM NUMBER
2. THE CP SELECTION QUERY
3. THE INPUT SELECTION MENU
4. REFUN THE DATA CALCULATION
5. DO A SENSITIVITY ANALYSIS
6. PRINTCUT COST+FACTORS
7. PRINT LAST SENSITIVITY TABLE.
8. PLOT LAST SENSITIVITY ANALYSIS ACHIEVED VALUES.
9. PLOT LAST SENSITIVITY ANALYSIS FEE PERCENTAGES.
10. PLOT WITH SELECTED Y-AXIS SCALES.
11. USE OPTIMUM EI VALUE FROM LAST SENSITIVITY
    ANALYSIS FOR CP-3 DATA.
12. EXIT
    ?

```

Figure 4.15 Monitor's Main Menu in Mode 1.

[illegible]

5. Effectiveness Index

Selection 11 on the Sensitivity Menu provides the monitor with a means of directly copying into the data array (Selection Menu) the best sensitivity parameter value for the highest EI value. SNSITIV determines the best occurrence of the highest value of Effectiveness Index. The EI with the lowest parameter value is used for cost analysis, accuracy analysis, and completion date analysis. For parameters involving 'reliability', the EI of the highest parameter value is stored in EICPT. It will not make a change of the optimum selection stored in those cases where the minimum value is zero. It stores the sensitivity parameter's step number in variable EICPT. When option 11 is selected, the sensitivity parameter being evaluated has its value changed to that of the evaluation step which had been saved in EICPT. Figure 4.17 demonstrates the revised Selection Menu.

6. Flexible Menu Reset

Should the monitor change the team, the DP level or the data in the database the calculations made by SNSITIV are no longer applicable. PROJNG flags the use of the subroutines which generate these changes. Through the running of these subroutines, the flag which expands the Sensitivity Menu is reset to 'off'. The option to replot graphs or tables based on the previous set of data is invalidated.

Selection of the mode 2 option to "FILE TEAM PROPOSAL" generates the monitor mode flag MR. For mode 2, MR=2. This flag causes the menu statement to file the proposal, and flags the program when leaving mode 2 either at exit or on return to the Monitor's Mode Menu to ask if the rewritten file should be sent to the PROJNG disk.

J. SUMMARY

The PROJMG FORTRAN program retains its basic computations and organization in common with the original CNP program. These basic features have been enhanced by improved user/program interaction, and by the reformulation of the achieved value fee percentage computation algorithm.

PROJMG FORTRAN has an additional benefit over CNP in the extended sensitivity analysis capability it provides to the monitor. Both the sensitivity tables and graphs enhance the monitor's ability to evaluate the team's progress when using PROJMG.

====>11

NO.	ITEM	TEAM	3 IS IN	DP-3	DP-3	FINAL	*****CURRENT	VALUE
1.	CONTRACTOR APPREACH	VALUE	1		16.	GUIDANCE QUAL TESTS	17.00	6.00
2.	GUIDANCE CONFIGURATION	1		17.	FLIGHT TESTS	18.	53.00	17.00
3.	GUIDANCE CONFIGURATION	1		18.	MAXIMUM CCST	19.	47.00	53.00
4.	MAINTAINABILITY ENG	1.75		19.	MINIMUM CCST	20.	4.00	47.00
5.	VALUE ENG DEVELOP	0.0		20.	MAX COST WEEK	21.	202.00	4.00
6.	PARALLEL RELIABILITY	92.00		21.	LATEST WEEK	22.	3.50	202.00
7.	NOTCH RELIABILITY	97.00		22.	EARLIEST WEEK	23.	81.00	3.50
8.	AIRFRAMES RELIABILITY	98.50		23.	MAX DELIVER INCENTIVE	24.	75.00	81.00
9.	AIRFRAMES RELIABILITY	80.00		24.	MAXIMUM RELIABILITY	25.	3.50	75.00
10.	FIRE CONTROL ERROR YDS	70.00		25.	MINIMUM RELIABILITY	26.	160.00	3.50
11.	GUILLANCE ERROR YDS	25.00		26.	MAX RELIABILITY	27.	154.00	160.00
12.	NOTCH QUAL TESTS	6.00		27.	MAXIMUM ERROR	28.	4.00	154.00
13.	AIRFRAMES QUAL TESTS	3.00		28.	MINIMUM ERROR	29.	337.00	4.00
14.	AIRFRAMES QUAL TESTS	3.00		29.	MAX ERROR INCENTIVE	30.		337.00
15.	FIRE CONTROL QUAL TESTS	3.00		30.	WEEK FOR LOT 10			

*****31. CHANGE BY PAGE*****
 SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-3.

Figure 4.17 Selection Menu with Optimized Flight Tests.

V. SUMMARY AND CONCLUSIONS

A. SUMMARY

The DMS exercise provides a time proven means of educating project managers in the potentials, sensitivities and incentives of acquisition management. This same capability is the significance of CNP and of PROJMNG as training aids in the DMS exercise. The support of project management training by computer programs has existed for the past twenty years. This thesis has upgraded the available computer program support for DMS. PROJMNG provides DMS with a user-friendly interactive tool to assist in the acquisition management training process.

B. DMS

DMS has proven to be an effective training exercise. It provides personnel in the Department of Defense, and personnel in other system acquisition environments, with relevant management training.

1. Benefit

The benefit of DMS as a training exercise is in the requirement it places on students to understand acquisition management concepts. It provides a sequenced set of events to develop, produce and deploy a missile. The Decision Point's serve as planning milestones that the student teams must prepare for in order to successfully manage their acquisition project. Their understanding of the concepts of acquisition management is challenged by the exercise in a pseudo-real time scenario.

2. Computer Support

During the twenty years in which DMS has been available, there has been a revolution in the tools available to support project management. The ability of computers to rapidly organize, process and correlate data has changed the acquisition management environment.

3. CNP

DMS has adapted somewhat to management evolution by its incorporation and development of CNP.

CNP is an interactive Fortran program which has been used for the past decade. In its current configuration, it is difficult to use. Since it is not user friendly.

C. PROJMG

In order for DMS to remain a relevant management training exercise, it must be kept current in its state-of-the-art management concepts, training facility, and computer support. The Fortran program PROJMG has been created to provide DMS a program which is more user-friendly. It is not just an off-shoot of CNP. It incorporates portions of CNP by virtue of utilizing most of the subroutines and algorithms of CNP. PROJMG improves the Fortran support for DMS in its refinement of CNP and by expanding the programmed support for analysis.

1. Benefits

PROJMG's benefits include:

- program access
- program exit
- session record
- data file of input parameters
- menus for operation selection

- expanded sensitivity analysis
- tables and graphs to study parameter sensitivity
- user-friendly displays
- monitor-team communication interaction through datafile exchange

2. Shortcomings

PROJMNG has not been completely validated. Its achieved values have not been tested sufficiently to determine its ability to react in a pseudo-realistic manner to parameter variations.

Additionally, the programming facilities utilized in PROJMNG are not as modern as they could be. The PROJMNG capability has been hampered by the limitations of the IBM-3033. Areas in which these limitations have impacted PROJMNG are discussed in Chapter 6, Recommendations.

3. First Usage

PROJMNG has undergone a field test. It was used to support the Winter 1984 course in Project Management, AS-3501, taught at the Naval Postgraduate School by LCDR. J. Ferris. This field test resulted in the following program enhancements:

- Ability to mass purge data files
- Ability of the monitor to send files to the PROJMNG disk
- The SMSG autolog feature operates only when no users are accessed to the PROJMNG disk
- All displays are formatted for 80 character wide personal computer (PC) display.
- All exec query responses are tested for input errors.
- PROJMNG EXEC tests files for format before replacing the team disk copy. The file used

for the team database is sequentially the first one which exhibits the correct format from either the PROJMNNG disk or the team, and defaults to the file "FILE DATAFILE".

During this field test, the major problem in facilitating the program occurred in the autolog procedure. The SMSG automatic log-on of the PROJMNNG disk occurred regardless of users being on the disk. This situation resulted in files being read into the disk without proper organization. The files subjected to this garbling were unusable.

SMSG is now programmed to write the team files onto the PROJMNNG disk only when the disk is not accessed by any user.

4. Transportability

The wide usage of DMS in the U. S. and other countries provided interest in making PROJMNNG capable of use on various computer systems. CMS unique features and IBM extensions to fortran have been discussed in this thesis to enable adaptation. In its present form PROJMNNG relies heavily on CMS and its executive language facilities to manipulate the data files outside of the Fortran program environment. These facilities require assessment on a case by case basis in order for PROJMNNG to be usable on any system other than IBM's CMS with the FRTCMS extension.

a. FORTRAN

The program has been compiled both on the FORTRAN H and FORTRAN GI compilers of the NPS IEM-3033. PROJMNNG is unable to compile in WATFIV (WATERLOO FORTRAN). The complication which precludes the WATFIV compilation is the call to FRTCMS. It is desirable to not remove the FRTCMS calls. They provide a major enhancement to the program by

enabling the program to clear the screen, 'CLRSCRN '. Loss of this feature would disorganize the PROJMNNG displays.

D. DOCUMENTATION

1. CNP

A major problem in working with a program is its documentation. CNP has several styles of documentation. There are periodic lines of text to label or explain the programs operation. These comments were beneficial in reconstructing some of the program operation, but inadequate in deciphering the program operation.

The best documentation present in CNP were the names selected for variables. Each of the input variables, a majority of the computational variables and the routine names were carefully selected to provide a relation with their data definition or with the process they represent. The variables in most of CNP are acronyms.

2. PROJMNNG

PROJMNNG contains additional comments throughout the program code taken from CNP. Frequent comments have been added to explain the processes accomplished by groups of code and routines. All of the terms added to the program by PROJMNNG are meaningful acronyms for the routines and variables.

3. Glossary

Appendix H provides a glossary of the terms used in both CNP and PROJMNNG. Compilation of the glossary provides a documentation method which supplements the benefit obtained from selecting variables which are acronyms for the content definition. Some definitions for CNP terms in the glossary remain undetermined.

4. Debugging

Terms used in a program without adequate documentation may interact in unexpected areas of the program. With 3063 variables in the program, CNP is a maze of connections between variables. Tracking the details of terms used in CNP was difficult. The two worst aspects of terms in CNP are multiple definitions for a term and manipulation of the variables to change the value's power-of-ten multiple. For example, RATIO is used in CNP to determine the percentage of a fee to be awarded, it is also used as a value of millions of dollars paid for fees, and it is used in a resource ratio calculation. An example of the manipulation of variables is the array CTC(8). This array is used in millions of dollars and in dollars.

The disadvantage in using mixed definitions for a variable is the impact which occurs on calculations which depend on the variable. If a calculation is based on a specific multiple of the variable and it has varying values dependent on which subroutine used it last, the programmer cannot utilize the variable. He must consider the inability to predict the occurrence of changes to the variables definition.

VI. RECOMMENDATIONS FOR FUTURE CONSIDERATION

The planning, design and execution of this revision to CNP have generated the following recommendations.

- Increase the program support to the exercise
- Expand the program to support more Decision Points in the DMS exercise
- Develop a DMS computer assisted tutorial
- Validate the achieved values
- Change the data tables to formulas
- Increase the sensitivity analysis capabilities
- Improve the graphics resolution
- Change the menu displays to 'plate' displays
- Develop a program to create project management exercise scenarios for actual system acquisitions
- Use PRJCMNG to encompass other CNP program variations

A. INCREASED EXERCISE SUPPORT

1. Discussion

The wide availability of personal computers provides project managers and students with improved data processing capability. They now have computation capability available to them instantaneously throughout the acquisition process for storing and analyzing their own specific acquisition data. If DMS is to continue to provide project managers with training at the level now available in the management environment, it must have software facility equivalent to at least that of their personal systems. It is feasible to make the DMS program compatible with PCs. DMS must be supported

AD-A140 709

PROJMG FORTRAN: AN INTERACTIVE COMPUTER PROGRAM FOR
USE WITH THE DEFENSE MANAGEMENT SIMULATION EXERCISE(U)
NAVAL POSTGRADUATE SCHOOL MONTEREY CA G W SCHULTZ

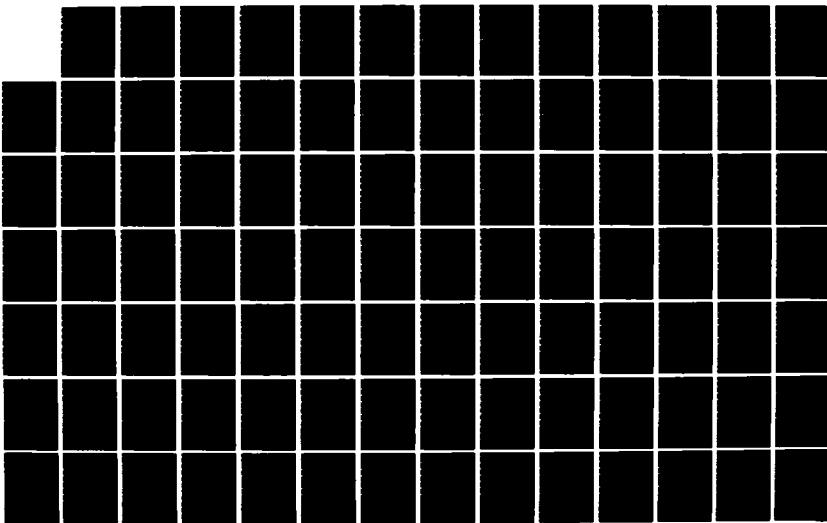
2/4

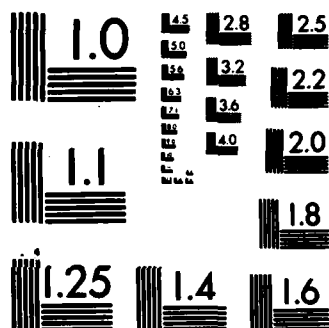
UNCLASSIFIED

MAR 84

F/G 9/2

NL





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

at the state-of-the-art. It must be contemporary in providing a program which has the degree of computer facility which the managers experience in day-to-day use and in actual acquisition planning. As this capability develops, the program supporting DMS should expand to:

- Support all DMS Decision Point levels
- Interface with student and manager PCs
- Support actual acquisition scenarios

2. Recommendation

Design new program capabilities to permit DMS adaptation for broader application to PC's and to change the acquisition management process.

B. INCLUDE ALL DECISION POINTS

1. Discussion

Both CNP and PROJMG support the DMS exercise only for DP-3 and DP-4. The present DMS scenario includes six Decision Points. All of the DP's require team analysis of the exercise parameters and submission of Decision Sheets (see Appendix B). These characteristics are conducive to incorporation of these DP's into other PROJMG like analyses.

Use of an interactive computer program to support DP's 1 and 2 would provide an opportunity for each team to do a better analysis in these early Decision Points, and to initiate the team database for later use. Inclusion of all DP's in the program would permit documentation of the entire team exercise performance in one computer record.

2. Recommendation

A future computer support program for DMS should be designed to expand exercise support from the PROJMG and CNP capability of DP 3 and 4 evaluation to all exercise DP's.

C. DEVELOP A DMS TUTORIAL

1. Discussion

Computer tutorials have become a popular method of training. The availability of computer assets for CNP or PROJMG indicates the presence of computer hardware to support a tutorial program. Student interaction with a computer supported DMS tutorial could be used to introduce the student to the DMS exercise, to perform sample problems, and to give examples.

The one-on-one capability for student response to a DMS tutorial should be limited to exercise orientation and familiarization. DP's 2 through 6 provide an opportunity for a team-work experience. This team interaction should remain an integral part of the DMS exercise.

2. Recommendation

Incorporate the DMS training scenario into an acquisition management tutorial. The tutorial is seen as providing:

- scenario presentation to the individual
- review of initial positions
- aid to students to remedy difficulty with concepts

D. VALIDATE PROJHNG

1. Discussion

Correlation of achieved values between the CNP/PROJHNG programs and the DMS data values has not been demonstrated. An analysis should be conducted in order to determine if the algorithms in PROJHNG provide realistic approximations to the full DMS batch program.

Evidence exists of differences between some CNP and PROJHNG results. Either program may be at fault. PROJHNG's errors result from its acceptance of the algorithms and calculations contained in the original CNP program. The values computed by CNP can not be assumed to be accurate.

Current classroom usage of PROJHNG has been predicated on the assumption that these values indeed perform as should be expected. This assumption was based on the incorporation of the Fortran code for the achieved values calculation from CNP into PROJHNG.

2. Recommendations

A series of evaluations should be conducted to compare the actual DMS batch outputs and with the performance of PROJHNG.

E. REPLACE THE DATA TABLES WITH EQUATIONS

1. Discussion

The tables of values included in CNP/PROJHNG have limited range in which performance data can be obtained. Beyond these limits data values are not available for calculations. In most cases, CNP does not warn the user that he has performed a calculation which is not a feasible solution. In most cases, PROJHNG forces the team to resubmit data items which are outside the range capabilities

discussed in the DMS instructions. There are parameters for which limits are not specified in the DMS handouts, and which can result in unusual calculations such as achieving a flight test completion date of 0.

There are two reasons for redesigning the program resource value algorithms which are used to calculate achieved values. First is to provide a continuous function instead of a step function. The table values read for calculations in both CNP and PROJNMG do not provide continuous value functions. Second the values have a limited range. Student selections which are beyond the capability of the table result in erroneous achieved values, such as the flight test example above.

The results provided by CNP and PROJNMG, as discussed in Chapter 1, are not actual results. They represent approximate results to reflect the real world issue of contract risk.

The programming of equations to obtain the resource values will result in the need to validate the achieved values predictions. Validating the values achieved as a result of changing the resource allocation algorithm would be of equal complexity to validating PROJNMG's current results. Both validation projects would verify the program's capability to produce realistic predicted contract performance values. In the interest of reducing the expenditure of effort between these two validations, the effort would be best applied in one project. To validate PROJNMG and later require revalidation for a new algorithm would obviate the former.

2. Recommendation

First, redesign and implement the equations for determining resource allocations. Second, validate the entire new program package of PROJNMG for DMS support.

F. INCREASE SENSITIVITY ANALYSIS OPTIONS

1. Discussion

The options available in the sensitivity analysis menu do not have as dramatic an effect in demonstrating the impact of incentives and other parameters as had been originally expected. Sensitivity curves showing definite peaks or valleys in the achieved effectiveness index are not demonstrated as readily as had been anticipated. The graphic presentation and sensitivity selection options should be designed to provide the user with a clear depiction of the effect of incentive variation. The user should be able to readily determine from the sensitivity analysis products the benefits or disadvantages of changing his decision above or below his current position.

2. Recommendation

Determine other analysis options which may have greater value to the monitors. Conduct a series of analyses to determine if they affect the sensitivity, and to eliminate those options which do not cause an achieved value fluctuation.

G. IMPROVED GRAPHICS

1. Discussion

One area of computer hardware and software development which is rapidly changing is graphics. The graph subroutine in PROJMG was specifically designed for the program and has a number of limitations. No other IBM-3033 graphing program was found which provides multiple curve graphing, selectable scales and the number of variations which PROJMG provides. A (continuous) curve plot is needed to enhance demonstration of parameter variations. The

graphic detail which is needed to improve the display does not appear to be within the capability of the IBM-3278 terminals which are currently a widely available computer peripheral for student use, and which are particularly available in large quantities at NPS. What is needed is the ability to control screen display for continuous line plots. This type of resolution for the graphics on the IBM-3033 is not within the IBM-3278 terminals capability. Terminals do exist which provide greater screen control and which can be programmed to plot continuous lines for the IBM.

Display detail is undergoing evolution. The advantages of color for highlighting presentations adds a perceptible benefit in human ability to comprehend. Color can be used to add a perception of relative weight of significance between curves and between points on a curve. The use of characters and figure presentations which exceed the standard typewriter characters in ASCII presentations increase the pictorial communication of data. Both of these factors -color graphics and detailed display pixle control- are receiving widespread attention in interactive computer systems and should be considered as enhancements for computer support programs which will be successors to CNP and PROJNG.

A benefit to plotting continuous curves is the ability to separate points of curve intersection. Another benefit is the ability of continuous curves to show the connection between points in the same curve. The PROJNG plots readily demonstrate the difficulty in presenting the different curves whose plots have a coinciding point.

In the short term, use of Fortran capable displays on the Tektronix 616 or the Rantek does not appear reasonable. Several factors degrade the significance of implementing operation on a more complex graphic system. One is the added processing time and cost for a display. The

second reason to not consider graphing the program on a graphics display system is the rapid rate of technical development currently occurring. Based on the presence of new IBM color graphics terminals at NPS, the facility at the W. R. Church Computer Center may be expected to provide high resolution color graphics for users in the not too distant future. In this case the effort to recode the PROJMNG program for high resolution graphics on the Tektronix or the Hamtek graphics terminals would be obviated both in comparative cost and in utility.

Another factor which should be considered before programming DMS support on a graphics unique system, such as DISSPA, is whether the system will allow DMS to remain transportable. CNP and PROJMNG were designed to be cost-effective by their potentially wide distribution and usage. The common availability of the systems on which these two programs are based permits them to be widely copied. Use of unique display systems would reduce the transportability and preclude spreading the cost over a wider user population.

2. Recommendation

Future computer programs to support DMS should demonstrate multi-color plotting capability and continuous line graphing.

B. PLATE DISPLAYS

1. Discussion

The Selection Menu display sets a pattern of presentation of data throughout the PROJMNG program. Menu heading and columns of data labels remain consistent in the monitor modes of operation, in the student mode of operation and in the display of data for the report submission tables. With the capability to produce plate displays, these factors

could be set into a file to be overlaid onto the screen. Items in the table are excellent candidates for a plate information display. The plate display capability would enable data item value change on the screen without reprinting the display and without the need to indicate which item is to be changed. The user will be able to place the terminal cursor on the data item value, change the values as desired, and submit the entire change as one input.

Not only would the plate display simplify data change and operation, it would also eliminate the need for four of the subroutines used in PROJMG. The two change routines, the PAGEIN subroutine and the FINISH subroutines could be combined into one menu with the Selection Menu format.

Plate display capability would also be useful for the Sensitivity Analysis Menu. The format of this display could be changed into the same format as the Selection Menu. Much of the sensitivity analysis routine could be eliminated with the saving in lines-of-code which the plate display represents.

The IBM CMS II version currently being implemented at the Naval Postgraduate School (NPS) contains this capability. Plate display capability for this program is expected to be available within the coming academic quarter.

2. Recommendations

Adapt PROJMG to display plates as soon as CMS II is implemented.

I. A SCENARIO GENERATING PROGRAM

1. Discussion

Within the computer simulation environment, conceptual level programs have been demonstrated which permit the user to create his own decision support and analysis scenario. Application of this technology to DMS is envisioned as providing a program which can simulate any actual acquisition scenario. These concept level programs utilize a hierarchy of menus to query the user for arrays to establish the program dimensions. The program then queries the user for array size, array name, variables names, parameter ranges for the variables and relative significance (weight) of the variables within the parameter. A hierarchy of menu-driven routines guides the user to create a program which supports the scenario to be studied. These programs enable the user to vary input variables in a matrix of parameters. The user constructs a scenario based on his concept of the parameters, on his feeling of their interaction and their relative significance. Based on these user perceptions such a program creates predictions for the effects of the parameter interaction. These predictions are heavily weighted by user bias to parameter interaction. Usually the program results predict what the user envisioned because of the impact of his biases. The main benefit of scenario creation has proven to be the opportunity for the user to conduct a detailed management study of the situation being modelled.

2. Recommendation

Develop a program whose design would permit project managers to create program simulations tailored to assess their specific real-world scenarios.

J. USE PROJMNNG TO ENCOMPASS OTHER CNP VARIATIONS

1. Discussion

The design of PROJMNNG around CNP routines with its own routines performing the features which make it different from CNP makes it possible for PROJMNNG to encompass other CNP variations. Any CNP style program which has the same common buffer of data could utilize PROJMNNG's enhancements. The buffer of common variables in PROJMNNG has been edited from that contained in CNP. The CNP buffer can be restored into PROJMNNG, and does permit PROJMNNG operation.

The design of PROJMNNG routines to support the calculations obtained in CNP suggest the potential of PROJMNNG as a set of routines to enhance the user-friendliness of other programs. Programs which could generate the same data items as CNP could be used as a nucleus program whose data would be used by PROJMNNG routines to conduct sensitivity analysis.

Programs which are candidates for consideration are:

- the main DMS batch game
- the latest ICAF DMS version (SAFE 84)
- the Swedish version called PNG/ZEBRA

2. Recommendation

Investigate the use of PROJMNNG as is or modified to be compatible with the latest ICAF DMS version (SAFE 84) and the Swedish version called PNG/ZEBRA. Test the concept of using PROJMNNG as an enhancement to the other management acquisition programs.

APPENDIX A
CONTRACT NEGOTIATION PACKAGE (CNP)

Instructions for Using CNP
DP-3 & DP-4

***** DO NOT TYPE DATA WHICH ARE IN BOXES *****
 * ON DECISION SHEETS *

Type in the data as shown by arrows(====>)	NOTES
DP-3	
Log onto the computer	

LINK TO CNP PROGRAM

CALLS CNP

FIRST TIME YOU MUST TYPE ALL 1'S

R; T=0.01/0.01 17:58:49
 0543P 191 192
 DASD 192 FORCED R/O; R/W BY 1681P
 R(00102); T=0.01/0.01 17:59:02
 C (192) R/O
 R; T=0.01/0.01 17:59:08
 EXECUTION BEGINS.
 TYPE DP NUMBER 3 OR 4
 THIS IS DP NUMBER ***** 3 *****
 Y OR N
 TO CONFIRM, TYPE 1 FOR OK, 0 IF
 CORRECTION MUST BE MADE.
 TEAM NUMBER XXX
 RECEIVED - TEAM NUMBER: 3
 CONFIRM:
 NEW OR CHANGES TO INPUT DATA
 RETYPE WHOLE PAGE CONTAINING ONE
 0 (ZERO) FOR NO - 1 (ONE) FOR YES
 TYPE IN COMBINATION, 4 0'S AND 1'S FOR
 PAGES 1 to 4
 TYPE PAGE 1
 7500750000
 RECEIVED FOR PAGE 1:
 1
 1
 1.75
 0.75
 0.0
 CONFIRM:

====>LINK

====>ACC

====>CNP

====>3

====>Y

====>003

====>1

====>1111

====>11201

====>1

====>92097098500702563617
TYPE PAGE 2:
RECEIVED FOR PAGE 2:

92.0
97.0
98.5
80
70

25
6
3
6
1

CONFIRM:

====>0

====>92097098508007025633617
TYPE PAGE 2:
RECEIVED FOR PAGE 2:

92.0
97.0
98.5
80
70
25

6
3
3
6
17

CONFIRM:

====>1

====>5304700502382020309185035
TYPE PAGE 3:
RECEIVED FOR PAGE 3:

53.0
47.0
5.0
238
202
3.0
91
85

CONFIRM:

====>1

====>160140035337
TYPE PAGE 4:
RECEIVED FOR PAGE 4:

160
140
3.5
337

CONFIRM:

====>1

<====ERROR MADE
If not corrected it will
result in an impossible
output 0, TO REINPUT PAGE 2
<====

TABLE XVII
Development Contract Summary

DP-3 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 3

INCENTIVE AREA	INCENTIVE PROVISIONS		INCENTIVE ACHIEVEMENTS	
	WORST VALUE	BEST VALUE	MAX FEE ALLOWED	ACHIEVED VALUE
DEV. COST	\$ 53.0M	\$ 47.0M	5.0%	\$ 50.6M
FLT TST COMPL	238 WK	202 WK	3.0%	199 WK
RELIABILITY	85.0%	91.0%	3.5%	83.7%
ACCURACY	160YDS	140 YDS		143 YDS
TOTALS			15.0%	\$
TOTAL CONTRACT PRICE = \$ 54.7M				

=====>NO
 COST + FACTORS (NO)
 DO YOU WISH TO CONTINUE, Y FOR YES,
 N FOR NO
 =====>Y
 TYPE DP NUMBER 3 OR 4
 =====>3
 THIS IS DP NUMBER ***** 3 *****
 Y OR N
 =====>Y
 TO CONFIRM, TYPE 1 FOR OK, 0 IF
 CORRECTION MUST BE MADE.
 TEAM NUMBER
 =====>003
 RECEIVED - TEAM NUMBER: 3
 CONFIRM:
 =====>1
 NEW OR CHANGES TO INPUT DATA
 RETYPE WHOLE PAGE CONTAINING ONE
 0 (ZERO) FOR NO - 1 (ONE) FOR YES
 TYPE IN COMBINATION, 4 0S AND 1S FOR
 PAGES 1 to 4
 =====>0110
 TYPE PAGE 2
 =====>88097098508007025633617
 RECEIVED FOR PAGE 2:
 88.0
 97.0
 98.5
 80
 70
 25
 6
 3
 6
 17
 CONFIRM:
 =====>1
 TYPE PAGE 3:
 =====>5424860502382020258781040
 RECEIVED FOR PAGE 3:
 54.2
 48.8
 5.0
 238
 202
 2.2
 87
 81
 4.0
 CONFIRM:
 =====>1

Y FOR ITERATION, N FOR STOP

CHANGES TO PAGES 2 AND 3 ONLY.

TABLE XVIII
Page Data Strings

	1	2	1.75	0.75	0.0	
02	88.00 17.00	97.00	98.50	80.00	70.00	25.00 6.00 3.00 6.00
10303	54.20	48.80	5.00	238.00	202.00	2.50 87.00 81.00 4.00
10304	160.00	140.00	3.50	337.00		

CHECK CAREFULLY (This is a repeat
of the input data).

TABLE XIX
Development Contract Summary

DP-3 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 3

INCENTIVE AREA	INCENTIVE PROVISIONS		INCENTIVE ACHIEVEMENTS	
	WORST VALUE	BEST VALUE	MAX FEE ALLOWED	ACHIEVED VALUE
DEV. COST	\$ 54.2M	\$ 48.8M	5.0%	\$ 50.2M
FLT TST COMPL	238 WK	202 WK	2.5%	199 WK
RELIABILITY	81.0%	87.0%	4.0%	83.7%
ACCURACY	160YDS	140 YDS	3.5%	145 YDS
TOTALS			15.0%	\$

TOTAL CONTRACT PRICE = \$ 56.2M

====>NO
COST + FACTORS (NO)
DO YOU WISH TO CONTINUE, Y FOR YES,
N FOR NO
====>N
R; T=0.73/1.28 18:02:42
====>LOGOFF

'Y' FOR ITERATION
'N' FOR STOP

	DP-4	NOTES
	Repeat login procedure.	
====>LINK	R: T=0.01/0.01 17:58:49 0543P 191 192 DASD 192 FORCED R/O: R/W BY 1681P R(00102); T=0.01/0.01 17:59:02 C (192) R/O	LINKTO CNP PROGRAM
====>ACC	R: T=0.01/0.01 17:59:08 C (192) R/O	
====>CNP	EXECUTION BEGINS. TYPE DP NUMBER 3 OR 4	
====>4	THIS IS DP NUMBER ***** 4 ***** Y OR N	
====>Y	TO CONFIRM, TYPE 1 FOR OK, 0 IF CORRECTION MUST BE MADE. TEAM NUMBER XXX	CALLS CNP
====>003	RECEIVED - TEAM NUMBER: 3 CONFIRM:	
====>1	HAS CORRECT DP-3 DATA BEEN ENTERED - Y, N	
====>N	TYPE PAGE 1 DP-3 INFO 7500750000 CONFIRM:	
====>1120	TYPE PAGE 2 DP-3 INFO 0985080070025633617 CONFIRM:	
====>1	TYPE PAGE 3 DP-3 INFO 5424880502382020258781040 CONFIRM:	
====>1	TYPE PAGE 4 DP-3 INFO 16014003537 CONFIRM:	
====>1		

<====MUST ENTER N EACH TIME YOU LOGON.
FIRST TIME THROUGH YOU MUST
ENTER FINAL DP-3 DATA.
CHECK EACH INPUT CAREFULLY
BEFORE CONFIRMING

TABLE XX

DP-3 PG-1 PG-2 PG-3 PG-4

DOUBLE CHECK OF CORRECT DP-3
FIRST ALL OUTPUT DEPENDS ON THIS.
ENTER ALL 1'S (IN THIS
EXAMPLE, ENTRY WAS 11110'
BECAUSE PAGE-4 IS THE
SAME AS FOR DP-3
THIS PRODUCES AN ERRONEOUS RESULT.
SEE BELOW.

{189} <====TEAM
{225} REVERSED LIMITS AND
PUT IN 189 BEFORE 225.
CMP MAKES THE CORRECTION
AND GIVES A WARNING

CONFIRM:
NEW OR CHANGES TO INPUT DATA
RETYPE WHOLE PAGE CONTAINING ONE
0 (ZERO) FOR NO - 1 (ONE) FOR YES
TYPE IN COMBINATION, 4 OS AND 1S FOR
PAGES 1 to 4

====>1110
TYPE PAGE 1
RECEIVED FOR PAGE 1:
1
2
2.00
0.75
0.0
CONFIRM:
====>1
TYPE PAGE 2:
RECEIVED FOR PAGE 2:
098508070021633613
88.0
97.3
98.3
80
70
27
6
3
6
13
CONFIRM:
====>1
TYPE PAGE 3:
RECEIVED FOR PAGE 3:
53.2
47.8
6.0
225
189
1.5
*** INCENTIVE LIMITS WERE REVERSED ***
87
81
4.0
CONFIRM:
*** WARNING ***
*** INCENTIVE LIMITS WERE REVERSED ***
*** DECISION SHEET MUST BE CORRECTED **

====>1

TABLE XXI
Page Data Strings

10301	1	1	2	2.00	0.75	0.0					
10302	66.00 13.00	97.00	98.50	80.00	70.00	25.00	6.00	3.00	3.00	6.00	
10303	53.20	47.80	6.00	225.00	189.00	1.50	87.00	81.00	4.00		
10304	0.0	0.0	0.0	0.00							

<====NOTE CMP FILLED IN
ALL 0'S FOR PAGE 4.

TABLE XXII
Development Contract Achieved Value Summary

DP-4 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 3

INCENTIVE AREA	INCENTIVE PROVISIONS		INCENTIVE ACHIEVEMENTS	
	WORST VALUE	BEST VALUE	MAX FEE ALLOWED	ACHIEVED VALUE
DEV. COST	\$ 53.2M	\$ 47.8M	6.0%	\$ 49.0M
FLY TST COMPL	225 WK	189 WK	1.5%	182 WK
RELIABILITY	81.0%	87.0%	4.0%	82.0%
ACCURACY	0 YDS	0 YDS	0.0%	149 YDS
TOTALS			11.5%	\$ 15.0%
TOT MAX FEE ALLOWED DOES NOT EQUAL				
TOTAL CONTRACT PRICE = \$ 53.1M				

119

====>5324 TYPE PAGE 3:
80602251890158781040
RECEIVED FOR PAGE 3:

53.2
47.8
6.0
225
189
1.5
1.87
81
4.0
CONFIRM:

====>1

====>160140035337 TYPE PAGE 4:
RECEIVED FOR PAGE 4:

160
140
3.5
337
CONFIRM:

====>1

<====TEAM CORRECTED LIMITS
ON DECISION SHEET.

TABLE XXIII
Development Contract Achieved Value Summary

DP-4 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 3				
INCENTIVE AREA	INCENTIVE PROVISIONS		INCENTIVE ACHIEVEMENTS	
	WORST VALUE	BEST VALUE	MAX FEE ALLOWED	ACHIEVED VALUE
DEV. COST	\$ 53.2M	\$ 47.8M	6.0%	\$ 49.0M
FLT TST COMPL	225 WK	189 WK	1.5%	186 WK
RELIABILITY	81.0%	87.0%	4.0%	82.0%
ACCURACY	160YDS	140 YDS	3.5%	149 YDS
TOTALS			15.0%	\$
TOTAL CONTRACT PRICE = \$ 54.3M				

COST + FACTORS (NO)

DO YOU WISH TO CONTINUE, Y FOR YES,
N FOR NO

R; T=0.73/1.28 18:02:42

====>LOGOFF

'Y' FOR ITERATION
'N' FOR STOP

APPENDIX B
DMS DATASHEETS

DEFENSE MANAGEMENT SIMULATION

DECISION SHEET 3

1. Of the total funds approved by the Secretary of Defense, the following distribution is made for procurement of the ZEBRA Weapon System:

a. Development Phase

\$ _____.____ Million [1-4]

b. Production Phase

\$ _____.____ Million [5-8]

2. The Development Contractor selected for ZEBRA System development is: (enter appropriate contractor number)

a. Midas Missile Corporation (contractor 1)

b. Apex Aerospace, Incorporated (contractor 2) _____ [9]

3. The basic guidance approach selected for the ZEBRA System is: (enter appropriate approach number)

a. Radar Inertial Guidance (G-1) (approach 1)

b. All-Inertial Guidance (G-2) (approach 2) _____ [10]

4. The guidance configuration selected for development is: (enter appropriate configuration number)

a. "A" as primary, "B" as parallel (configuration 1)

b. "A" alone (configuration 2)

c. "B" as primary, "A" as parallel (configuration 3)

d. "B" alone (configuration 4) _____ [11]

5. The development contract to be negotiated contains the following features:

a. Funds for the following activities. These funds are included in the total target development cost:

(1) Maintainability Engineering \$ _____.____ Million [12-15]

(2) Value Engineering \$ _____.____ Million [16-19]

(3) Parallel Guidance Development (enter Ø where not desired) \$ _____.____ Million [20-23]

Team Number DP-5

DECISION SHEET 3

b. The following minimum technical performance characteristics:
(demonstrated at completion of flight tests)

(1) Motor Sub-system Reliability	_____.____ % [24-26]
(2) Airframe Sub-system Reliability	_____.____ % [27-29]
(3) Launcher/GSE Sub-system Reliability	_____.____ % [30-32]
(4) Fire Control Sub-system Impact Error	____ _ yards [33-35]
(5) Guidance Sub-system Impact Error	____ _ yards [36-38]

c. The following numbers of sub-system Qualification Tests:

(1) Motor Sub-system	(20-40)	____ [39-40]
(2) Airframe Sub-system	(3-9)	____ [41]
(3) Launcher/GSE Sub-system	(2-6)	____ [42]
(4) Fire Control Sub-system	(2-4)	____ [43]
(5) Guidance Sub-system	(3-9)	____ [44]

d. The following number of Flight Tests: (10-25) ____ [45-46]

6. Decision Point

Team Number

Card Number

____ [76]
____ [77-78]
____ [80]

Team Number 01-3

DECISION SHEET 3

7. The incentive provisions for this development contract (subject to negotiations) are as follows:

NOTE: If any area is not incentivized, enter the target value for both the maximum and minimum fee values, enter 0 for the maximum fee percentage.

a. Development Program Cost:

Maximum Cost (minimum fee value) \$ _____. Million [1-3]
Minimum Cost (maximum fee value) \$ _____. Million [4-6]
Maximum Fee Percentage _____. % [7-9]

b. Schedule; Completion of Flight Tests:

Latest Week (minimum fee value) Week No. _____. [10-12]
Earliest Week (maximum fee value) Week No. _____. [13-15]
Maximum Fee Percentage _____. % [16-18]

c. System Reliability:

Maximum Reliability (maximum fee value) _____. % [19-20]
Minimum Reliability (minimum fee value) _____. % [21-22]
Maximum Fee Percentage _____. % [23-25]

Team Number DP-3

DECISION SHEET 3

d. System Impact Error:

Maximum Error (poorest accuracy)
(minimum fee value) yards [26-28]

Minimum Error (best accuracy)
(maximum fee value) yards [29-31]

Maximum Fee Percentage % [32-34]

8. Production Decisions:

a. The desired date for completion of
deployment of lot TEN is: Week No. [44-46]

b. Fabrication of Block I missiles is to start after:

- (1) Qualification Tests
 [concurrent (fabrication 1)]
- (2) Flight Tests
 [fly-before-buy (fabrication 2)]

(1 or 2) [47]

9. Class

Decision Point

Team Number

Card Number

 [73-74]

 [76]

 [77-78]

 [80]

Team Number DP-4

DEFENSE MANAGEMENT SIMULATION

DECISION SHEET 4

1. The Development Contractor selected for ZEBRA System development is: (enter appropriate contractor number)

a. Midas Missile Corporation (contractor 1)
b. Apex Aerospace, Incorporated (contractor 2) [1]

2. The basic guidance approach selected for the ZEBRA System is: (enter appropriate approach number)

a. Radar Inertial Guidance (G-1) (approach 1)
b. All-Inertial Guidance (G-2) (approach 2) [2]

3. The final guidance configuration selected for development and production is: (enter appropriate configuration number)

a. "A" configuration (configuration 2)
b. "B" configuration (configuration 4) [3]

4. The revised development contract contains the following features:

- a. Funds for the following activities:

(If you do not desire to expend additional funds in these areas, enter the appropriate amounts expended at DP-3.
If you do desire to expend additional funds in these areas, enter the total amounts you now desire.)

(1) Maintainability Engineering \$. Million [4-6]

(2) Value Engineering \$. Million [7-9]

- b. [For Monitor's Use Only]

 0 0 0 [10-12]

Team Number DP-4

DECISION SHEET 4

c. The following minimum technical performance characteristics:
(demonstrated at completion of flight tests)

(1) Motor Sub-system Reliability	___ . ___ % [33-35]
(2) Airframe Sub-system Reliability	___ . ___ % [36-38]
(3) Launcher/GSE Sub-system Reliability	___ . ___ % [39-41]
(4) Fire Control Sub-system Impact Error	___ ___ yards [42-44]
(5) Guidance Sub-system Impact Error	___ ___ yards [45-47]

d. The following numbers of sub-system Qualification Tests:

(1) Motor Sub-system	(20-40)	___ [48-49]
(2) Airframe Sub-system	(3-9)	___ [50]
(3) Launcher/GSE Sub-system	(2-6)	___ [51]
(4) Fire Control Sub-system	(2-4)	___ [52]
(5) Guidance Sub-system	(3-9)	___ [53]

e. The following number of Flight Tests: (10-25) ___ [54-55]

5. Decision Point

Team Number

Card Number

___ [76]
___ []
___ []

DECISION SHEET 4

6. The incentive provisions for this development contract (subject to negotiations) are as follows:

NOTE: If any area is not incentivized, enter the target value for both the maximum and minimum fee values, enter 0 for the maximum fee percentage.

a. Development Program Cost:

Maximum Cost	(minimum fee value)	\$	___	___	Million	[1-3]
Minimum Cost	(maximum fee value)	\$	___	___	Million	[4-6]
Maximum Fee Percentage			___	___	%	[7-9]

b. Schedule; Completion of Flight Tests:

Latest Week	(minimum fee value)	Week No.	___	___	___	[10-12]
Earliest Week	(maximum fee value)	Week No.	___	___	___	[13-15]
Maximum Fee Percentage			___	___	%	[16-18]

c. System Reliability:

Maximum Reliability	(maximum fee value)	___	___	%	[19-20]
Minimum Reliability	(minimum fee value)	___	___	%	[21-22]
Maximum Fee Percentage		___	___	%	[23-25]

Team Number DP-4

DECISION SHEET 4

d. System Impact Error:

Maximum Error (poorest accuracy)
(minimum fee value) _____ yards [26-28]

Minimum Error (best accuracy)
(maximum fee value) _____ yards [29-31]

Maximum Fee Percentage _____ % [32-34]

7. Production Decisions:

- a. The Development Contractor is to procure the following number of lots of long-lead time components in preparation for Block I production:

_____ [44-45]

- b. The earliest Lot TEN deployment date for which fixed price proposals are desired is:

Week No. _____ [46-48]

- c. Fabrication of Block I missiles is to start after:

- (1) Qualification Tests
[concurrent (fabrication 1)]
- (2) Flight Tests
[fly-before-buy (fabrication 2)]

(1 or 2) _____ [49]

8. Class

Decision Point

Team Number

Card Number

_____ [73-74]

_____ [76]

_____ [77-78]

_____ [80]

APPENDIX G
LINKPROJ EXEC

```

*THIS EXEC FACILITATES STUDENT LINK TO THE 0276P DISK FOR EXECUTION OF
*THE PROJECT MANAGER SIMULATION "PROJMMG FORTMAN".
&CONTROL OFF
-AGAIN
CP LINK 0276P 191 192 RR
ACCESS 192 C
CP SET PF09 IMM REL 192 (DET) #CP SET PF09
CLRSCRN
EXEC RECORD ON
&TYPE # NOTE!! YOU ARE NOW LINKED TO PROJMMG ON YOUR 192 DISK, MODE C
&TYPE # NOTE!! PRESS PF09 TO BREAK THE LINK
EXEC PROJMMG
REL 192 (DET)
CP MSG QACNT KLINE
&BEGTYPE
ANOTHER RUN, 'Y'/'N'?

&END
&READ ARGS
&IF .&1 EQ . &SKIP
&IF &1 EQ Y &GOTO -AGAIN
EXEC RECORD OFF
-END

```

APPENDIX D
PROJMG EXEC

```

**"PROJMG EXEC" PROVIDES ENTRY CHECKS FOR THE MAIN PROGRAM "PROJMG".
**AMONG THE CHECKS ARE THOSE HIGHLIGHTED BY COMMENTS BELOW. THEY FACIL-
**ITATE "TEAM" STORED FILE PROCESSING.
****
&CONTROL OFF
SET BLIP
SET CMSTYPE HT
GLOBAL TXLIB FORTMOD2 MOD2EEH IMLS P NONIMSL CMSLIB
&FLAG = 0
&ONE = 21
FILEDEF 5 TERMINAL (PERM
FILEDEF 6 TERMINAL (PERM
FILEDEF 13 DISK DATAINST DISKNUM A (PERM
FILEDEF 17 DISK FILE FT17F001 C (PERM
FILEDEF 18 DISK FILE FT18F001 C (PERM
FILEDEF 19 DISK FILE FT19F001 C (PERM
FILEDEF 20 DISK FILE FT20F001 C (PERM
FILEDEF 23 DISK FILE FT23F001 C (PERM
**FILE DATAINST--IDENTIFIES THE DISK AS BELONGING TO A MONITOR/INSTRUCTOR
**FILES 17, 18, 19 & 20 -- ARE DATA FOR ACHIEVED VALUE DETERMINATION IN
** EXECUTING THE PROGRAM'S SIMULATED INTERACTION WITH A
** CONTRACTOR.
****
SET CMSTYPE RT
** (THE FOLLOWING COPYFILES ARE PERFORMED TO PERMIT FORTRAN PROGRAM
** EXECUTION TO CHANGE THE TEAM FILES)
**
** USERID
** (USERID VARIABLES ARE USED AS SECURITY CHECKS)
&READ VARS &DUMMY &USERID
&USER = 0
** (CHECK FOR THE PRESENCE OF AN INSTRUCTOR PECULIAR FILE)
STATE DATAINST DISKNUM A
&RET = &RETCODE
&IF &RET NE 0 &GOTO -STUDENT
** (CHECK FOR A "P" IN THE FIRST LINE FIFTH POSITION)
FINDSTAK DATAINST DISKNUM A 5 P
&IF &RETCODE NE 0 &GOTO -STUDENT
&READ VARS &USER
&IF &USER NE &USERID &GOTO -STUDENT
&BEGTYPE

```



```

PRESS <<ENTER>> WHEN YOU ARE READY TO CONTINUE.

&END
&READ
&CLRSRN
&BEGTYPE
*****.....EXECUTION IS IN PROGRESS. WAIT. DO NOT PRESS ENTER.*****
&END
FILEDEF 11 DISK DATAINST DISKNUM A
      (ESTABLISH THE INSTRUCTOR UNIQUE FLAG OF 21)
&FLAG = TEAM21
STATE DATAFILE TEAM21 A
&IF &RETCODE NE 0 COPYFILE FILE DATAFILE C DATAFILE TEAM21 A (REPLACE
&NM = 0
      (CHECK FOR DATAFILES 1-20)
-NEXTIN
&NM = &NM + 1
&CHART = TEAM&NM
&IF &NM LT 10 &CHART = TEAM0&NM
STATE DATAFILE &CHART C
&IF &RETCODE EQ 0 COPYFILE DATAFILE &CHART C = A (REPLACE
&IF &NM LT 20 &GOTO -NEXTIN
FILEDEF 9 DISK DATAFILE TEAM21
&BEGTYPE

PRESS <<ENTER>> WHEN YOU ARE READY TO CONTINUE.

&END
&READ
&PROCS
&CLRSRN
*****
* THE FOLLOWING LINES PROVIDE A ROUTINE TO PURGE TEAM DATAFILES
* FROM THE DISK ON WHICH THE USER HAS A ACCESS. THE DESIGN OF
* THIS FEATURE IS TO PURGE ALL TEAM FILES FROM THE DISK ONTO WHICH
* TEAMFILES ARE SENT BY THIS EXEC ROUTINE. THIS OPERATION WILL BE
* PARTICULARLY BENEFICIAL IN HELPING MONITORS TO INITIALIZE THE
* DISK FOR A NEW GAME.
* THIS IS ACCOMPLISHED BY ANSWERING 'Y' AFFIRMATIVE TO PURGE TEAM
* DATAFILES WHILE LOGGED ONTO THE PROJMG MOUNTED FILE.
*****

```

```

&BEGTYPE
*****WARNING*****
*****ANSWERING 'Y' (=YES) TO THE FOLLOWING QUERY WILL*****
*****PURGE ALL EXISTING STUDENT DATAFILES*****
*****ON THE DISK YOU ARE USING.*****

DO YOU WISH TO INITIALIZE ALL TEAM FILES FOR A NEW DMS EXERCISE?
'Y'=PURGE ALL TEAM DATAFILES
'N'=PROCEED WITH NORMAL OPERATION
'E'=STOP

&END
&CMS MAKEBUF
&READ VARS &N
&CMS DROPEUF
&IF &N EQ . &GOTO -PROEX
&IF &N EQ PURGE &GOTO -REDUM
&IF &N EQ E &GOTO -END
&GOTO -PROGRUN
-PROEX
&BEGTYPE
DO YOU WISH TO CONTINUE; Y/N ?

&END
&READ VARS &NOGO
&IF &NOGO EQ . &GOTO -INSTEND
&IF &NOGO EQ Y &GOTO -PROCES
&GOTO -INSTEND
REDUM
STATE DATAFILE * A
&IF &RETCODE NE 0 &GOTO -END
ERASE DATAFILE * A
ERASE DATAFILE * A
&IF &RETCODE NE 0 &GOTO -END
&GOTO -REDUM
*STUDENT
*STATE DATAFILE * A
&RET = &RETCODE
&IF &RET NE 0 &GOTO -NEMNUM
*
( GET THE TEAM NUMBER )
( CHECK FOR A "3" IN THE FIRST LINE FOURTH POSITION )

```

```

FINDSTAK DATAFILE * A
&READ VARS &O.C &TWO &THREE
&IF &ONE EQ 0 &GOTO -NEWNUM
&GOTO -TEAMNUM
-NEWNUM
&BEGETYPE

WHAT IS YOUR TEAM NUMBER?

&END
CMS MAKEBUF
&READ VARS &ONE
CMS DROPBUF
&IF &ONE EQ 1 &GOTO -EXITTM
&IF &ONE GE 21 &GOTO -EXITTM
&IF &ONE LE 0 &GOTO -EXITTM
&IF &ONE GT 0 &GOTO -TEAMNUM
-EXITTM
&BEGETYPE

DO YOU WISH TO CONTINUE; Y/N ?

&END
&READ VARS &NOGO
&IF &NOGO EQ 1 &GOTO -INSTEND
&IF &NOGO EQ Y &GOTO -NEWNUM
&GOTO -INSTEND
-TEAMNUM
&1 = &ONE
&CHAR = TEAM&1
&IF &1 LT 10 &CHAR = TEAM0&1
(ESTABLISH THE STUDENT'S WORKING FILES)
FILEDEF 9 DISK DATAFILE &CHAR *(PERM
&IF &FLAG NE TEAM21 FILEDEF 11 DISK DATACODE &CHAR *(PERM
* (CHECK THE INSTRUCTOR FILES FOR A PREVIOUSLY STORED TEAM DATABASE)
-TEAMFILE
STATE DATAFILE &CHAR C
&IF &RETCODE NE 0 &GOTO -ADISKM
FINDSTAK DATAFILE &CHAR C
&READ VARS &THREE
&IF &THREE EQ 1 &GOTO -ADISKM
&IF &THREE EQ 0 &GOTO -ADISKM
COPYFILE DATAFILE &CHAR C = A (REPLACE)
-ADISKM

```

```

STATE DATAFILE &CHAR A
&IF &RETCODE EQ 0 &GOTO -CODE
COPYFILE FILE DATAFILE C DATAFILE &CHAR A (REPLACE)
-CODE
STATE DATAFILE &CHAR C
&IF &RETCODE NE 0 &GOTO -ADISK
FINDSTAK DATAFILE &CHAR C
&READ VARS &THREE
&IF &THREE EQ 0 &GOTO -ADISK
&IF &THREE EQ 0 &GOTO -ADISK
COPYFILE DATAFILE &CHAR C = A (REPLACE)
-ADISK
STATE DATAFILE &CHAR A
&IF &RETCODE EQ 0 &GOTO -PROGRAM
COPYFILE FILE DATAFILE C DATAFILE &CHAR A (REPLACE)
SET CMSTYPE RT
-PROGRAM
&BEGTYPE
WAIT FOR "EXECUTION TO BEGIN".

&END
SET BLIP OFF
&STACK &ONE
&STACK &FLAG
*****
LOAD PROJING (START
*****
&IF &READFLAG EQ STACK DESBUF
SET BLIP *
* (THE FORTRAN PROGRAM MANIPULATION OF FILES PREVENTS GENERATION OF NEW
* DATA FILES DURING THE PROGRAM EXECUTION. THE FOLLOWING ROUTINE PERMITS
* THE OPERATOR TO GENERATE ATLEAST ONE NEW DATABASE FILE BY PROGRAM RE-
* DEFINITION OF FILEDEFS AS APPLIED TO THE THREE DUMMY FILES GENERATED
* AT THE TOP OF THIS EXEC.)
* (IF THE OPERATOR IS NOT AN INSTRUCTOR, BYPASS DATAFILE PRINTOUT
* CAPABILITIES)
&IF &USERID NE &USER &GOTO -NOTINS
*
&BEGTYPE
DO YOU WISH TO RECEIVE A LISTING OF TEAMS ON THE CLASS DISK,
Y=YES?
      (GET THE TEAM NUMBER)

```

```

&END
&READ VARS &ANS
&IF &ANS NE ' ' &GOTO -INSTEND
&IF &ANS NE 'Y' &GOTO -INSTEND
*
&NM = 0
CP SPOOL PUNCH CONT TO &USER
-NEXTNM
&NM = &NM + 1
&CHART = TEAM&NM
&IF &NM LT 10 &CHART = TEAM0&NM
PUNCH DATAFILE &CHART C
&IF &NM LT 20 &GOTO -NEXTNM
CP SPOOL PUNCH CLOSE NOCONT
CP SPOOL PUNCH OFF
READCARD INST FILE A
&GOTO -INSTEND
-NOTINS
*
CP SPOOL PUNCH CL X CONT TO 0276P
PUNCH DATAFILE &CHAR A0
&PUNCH &USERID
CP SPOOL PUNCH CLOSE NOCONT
CP SPOOL PUNCH OFF
CP SPOOL PUNCH CL X CONT TO 0276P
PUNCH DATAFILE &CHAR A0
&PUNCH &USERID
CP SPOOL PUNCH CLOSE NOCONT
CP SPOOL PUNCH CL A
CP SPOOL PUNCH OFF
-INSTEND
SET CMSTYPE HT
ERASE LOAD MAP A
SET CMSTYPE RT
SET BLIP OFF
-END

```

(INITIALIZE THE DATAFILE INCREMENT TO 0)
(CHECK FOR DATAFILES 1-20)

(DATA DUMP THE TEAM'S DATABASE FILES TO THE INSTRUCTOR)

APPENDIX E
PROFILE EXEC FOR PROJING

```

*TRACE
* The following warning alerts operators logging onto the disk
* that they should access by ACC (NOPROF) and not permit the
* system to run the Profile Exec since it has a logoff command
* on its bottom line. Normal Profile type functions are performed
* by executing a separate command of MYPROF which performs the
* routines in MYPROF EXEC.
*BEGETYPE
*****WARNING***** <<ACC (NOPROF)>>, <<MYPROF>> VICE <<PROFILE>>
*END
CP SPOOL RDR CL X
READ *
READ *
READ *
READ *
READ *
READ *
READ *
CP SPOOL RDR CL *
CP LOGOFF

```

APPENDIX E
PROJMG STUDENT INSTRUCTIONS

A. I N T R O D U C T I O N

This pamphlet contains general instructions for student operation of the program "PROJMG FORTRAN" which is designed as a companion to the Industrial College of the Armed Forces' "DEFENSE MANAGEMENT SIMULATION" exercise.

The program functions as a computerized analysis tool assisting a project manager in evaluating his assessment of the goals and incentives for his contract to procure a new missile system. It does not provide answers to the simulation's main batch computer program.

Significant effort has been placed in making the program as easy to operate as possible. The user-friendly design permits the simulation's participants to correct, to rerun, to submit reports to the monitor, and to exit the game with the minimum of difficulty.

Additionally, consideration has been given to protecting the individual team's results. A generic security system guards access to the team's files to prevent unintentional destruction of the database, or database plagiarism. Proper entry of the team security code enables the team to access its stored database and permits submission of a team proposal to the class monitor. Attempts to circumvent these measures are counted and recorded in the database. After five erroneous entries of the security code the team's data

files are not accessible by the students. The monitor must be contacted to reset the counter. In order to document the access and attempts to access a team database, the routine PROJMG EXEC transmits the database to the monitor's computer disk with the USERID and USERNAME from the disk running PROJMG EXEC.

In the event you dump (exit) out of the program, other files may appear on your disk such as PROJECT MODULE, LOAD MAP, FILE DATAFILE, and FILE DATACODE. Please erase these extra files. The FILEDEFS in Table XXIV will be generated DO NOT ALLOW FILES ON YOUR DISK WHICH ARE NOT PART OF PROJMG TO USE THESE FILE NAMES.

TABLE XXIV
PROJMG FILEDEFS

1. FILE 5	TERMINAL	READ	
2. FILE 6	TERMINAL	PRINT	
3. FILE 9	DISK	DATAFILE	TEAMXX A
4. FILE 11	DISK	DATACODE	TEAMXX A
5. FILE 17	DISK	FT17F001 C	{DATAFILES}
6. FILE 18	DISK	FT18F001 C	{DATAFILES}
7. FILE 19	DISK	FT19F001 C	{DATAFILES}
8. FILE 20	DISK	FT20F001 C	{DATAFILES}

during the execution of PROJMG EXEC .

The following sequence of operations demonstrates the procedures involved in the operation of "PROJMG FORTRAN". Several files are involved in the game operation. They are listed in order to enable the participant to manage his student disk. Problems with file definition conflict should be considered. Do not leave non-game files with the same file name on your disk.

flagging of the program for access to the next DP level. Please leave this file on your disk during your use of the game. Your disk contains your team's security code and number of times it has been incorrectly entered. Once you terminate a run of the simulation, the file is sent to the monitor's who uses it as his copy. On the monitor's PROJMMNG disk, it provides the monitor's copy is the primary copy in order to preclude program abort if DATA CODE cannot be found on your disk. DO NOT TAMPER WITH THESE FILES or difficulty will occur in the program's correlation of your team's security code.

5. DATA CODE

B. I N S T R U C T I O N S

PROJMMNG FORTRAN is available on the Naval Postgraduate School IBM-3033 computer in the 'W. R. CHURCH COMPUTER CENTER'.

The text which follows is a tutorial demonstrating the features of the PROJMMNG package of programs and data files. These features are demonstrated as they would occur on the IBM-3033 using an IBM 3270 terminal. Follow the procedures as they appear in the text. Use your own team number and team security code.

In the following examples, please perform the operations highlighted in the left margin by ==>.

The three column divisions below are arranged with the operator inputs in the left column. Computer responses and the headings for both explanation and for *****EXAMPLES***** are in the center column. The right column contains remarks to aid in understanding the procedures being executed.

ENTER YOUR NEW CODE.

====>Outline

ized to zeros.
PROJMG determines if you have a code of 8
zeros and requires a code input
to be made.
Select a code of up to 8 characters and
enter it.
If you already have a code, the program will
ask you to enter it for a compar-
ative validation.
The program will now proceed to check the
database for previous informa-
tion. If it finds nothing in
the files, it will sequence
through each item from the
participant's decision point
data sheets asking for the
inputs. At the end of each
page it will check the entries
against the input requirements
and ask for corrections as
necessary.

OPERATOR	PROGRAM	REMARKS
====>logon ====>linkproj	BEGIN RECORDING OF TERMINAL SESSION * NOTE1: YOU ARE NOW LINKED TO PROJMG ON YOUR 192 DISK MODE C * NOTE1: PRESS PFO9 TO BREAK THE LINK FILE 'DATAINST DISKNUM A' NOT FOUND. FILE 'DATAFILE * A' NOT FOUND. WHAT IS YOUR TEAM NUMBER? FILE 'DATAFILE TEAM01 C' NOT FOUND. FILE 'DATAFILE TEAM01 A' NOT FOUND. FILE 'DATAFILE TEAM01 C' NOT FOUND. FILE 'DATAFILE TEAM01 A' NOT FOUND. PRESS <ENTER> WHEN YOU ARE READY TO CONTINUE EXECUTION BEGINS... WAIT FOR "EXECUTION TO BEGIN".	PROJMG EXEC checks to see if you are an instructor (monitor). If not, and if it finds no TEAMxx data files, it will create the files needed. You should enter your team's number here. Wait for completion of this process. Determines if a DATAFILE TEAMxx has already been submitted to the monitor. PROJMG EXEC creates all of its needed files on your 'A' disk. There may be a pause of several minutes in completing this process. DO NOT PANIC.
====>1	***** TO TERMINATE THE PROGRAM AT ANY POINT, QUERY TYPE "E"=EXIT IN RESPONSE TO ANY "YES/NO" QUERY. ***** ***CAUTION***IN RESPONSE TO A QUERY, PRESSING <ENTER> WITHOUT PROVIDING DATA WILL DUMP THE PROGRAM. ***** ***YOU ARE ENCOURAGED TO CHANGE YOUR SECURITY CODE.***	All security codes have been initial-

..... P A G E - 1.....

*****INPUT PAGE 1 DP-3 INFO*****.

CONTRACTOR NUMBER,(1 OR 2):?

GUIDANCE APPROACH,(1 OR 2):?

GUIDANCE CONFIGURATION,(1-4;NOT 1 OR 3 AFTER DP-3?

MAINTAINABILITY ENGINEERING FUNDS IN \$M:?

VALUE ENGINEERING FUNDS IN \$M:?

PARALLEL GUIDANCE FUNDS IN \$M.?

.....D A T A C H E C K S.....

====>1
====>1
====>1
====>1.75
====>.75
====>.5

Parallel guidance selections 1 or 3
will be rejected after DP-3.

Query for parallel guidance funds.
If guidance configuration selected is 2
or 4, or after DP-3, parallel
guidance funds will not be
queried. They are zeroed.

Checks of data values will occur
at the end of each page;
Necessary corrections will be
queried.

Sample *****WARNINGS***** of data errors
are demonstrated below.


```

*****INPUT PAGE 2 DP-3 INFO.*****
MOTOR RELIABILITY?
AIRFRAME RELIABILITY?
LAUNCHER/GSE RELIABILITY?
FIRE CONTROL ACCURACY?
GUIDANCE ACCURACY?
MOTOR QUAL TESTS:20-40?
AIRFRAME QUAL TESTS:3-9?
LAUNCHER/GSE QUAL TESTS:2-6?
FIRE CONTROL QUAL TESTS:2-4?
GUIDANCE SYSTEM QUAL TESTS:3-9?
FLIGHT TESTS:10-25?
    
```

.....D A T A C H E C K S.....
C O R R E C T I O N S.....

*****WARNING*****
 GUIDANCE QUALIFICATION TESTS MUST BE GREATER THAN
 3 AND LESS THAN 9. YOUR QUAL TESTS= 1.
 PREVIOUS VALUE 1.00, INPUT A VALUE?

The two warnings below are samples of the
 result of parameter checks at the
 end of this data sheet page.

```

=====90
=====97
=====98.5
=====80
=====70
=====25
=====6
=====3
=====3
=====1
=====7
    
```

====>6

====>17

*****WARNING*****
FLIGHT TESTS MUST BE GREATER THAN 10 AND LESS THAN 25.
YOUR FLIGHT TESTS=7. previous value 7.00, INPUT A VALUE?

.....P A G E - 3.....

*****INPUT PAGE 3 DP-3 INFO.*****

MAXIMUM COST IN \$M?

MINIMUM COST IN \$M?

INCENTIVE FEE % FOR DEVELOPMENT COST?

LATEST WEEK FOR FLIGHT TEST COMPLETION?

EARLIEST WEEK FOR FLIGHT TEST COMPLETION?

INCENTIVE FEE % FOR TEST COMPLETION?

MAXIMUM RELIABILITY % EXPECTED?

MINIMUM RELIABILITY % EXPECTED?

INCENTIVE FEE % FOR RELIABILITY?

====>53
====>47
====>5
====>238
====>202
====>3
====>91
====>92
====>3.5

.....D A T A C H E C K S.....

.....C O R R E C T I O N S.....

*****WARNING*****

MAXIMUM SYSTEM RELIABILITY IS LESS THAN
MINIMUM RELIABILITY; MAX = 91. ; MIN = 92.
FOR MAXIMUM RELIABILITY
PREVIOUS VALUE 91.00, INPUT a VALUE.

FOR MINIMUM RELIABILITY,

====>91

Following are additional examples of
correction check warnings.

The program automatically queries pairs of
values which are for the same
data element.

..... P A G E - 4

*****INPUT PAGE 4 DP-3 INFO.***** MAXIMUM IMPACT ERROR EXPECTED? MINIMUM IMPACT ERROR EXPECTED? INCENTIVE FEE % FOR IMPACT ACCURACY? INPUT DESIRED WEEK FOR COMPLETION OF DEPLOYMENT?	D A T A C H E C K S..... C O R R E C T I O N S.....

====>160

====>140

====>3.5

====>337

FOR 92.00, INPUT a VALUE.

====>85
=====

DATABASE PRINTOUT as SELECTION MENU

NO.	ITEM	CURRENT VALUE @ DP-3	NO.	ITEM	CURRENT VALUE
1.	CONTRACTOR APPROACH	1	16.	GUIDANCE QUAL TESTS	6.00
2.	GUIDANCE CONFIGURATION	1	17.	FLIGHT TESTS	17.00
3.	GUIDANCE RELIABILITY ENG	1.75	18.	MAXIMUM COST	53.00
4.	MAINTAINABILITY ENG	0.75	19.	MINIMUM COST	47.00
5.	VALUE ENG	0.5	20.	MAX COST INCENTIVE	5.00
6.	PARALLEL DEVELOP	90.00	21.	LATEST WEEK	238.00
7.	MOTOR RELIABILITY	97.00	22.	EARLIEST WEEK	202.00
8.	AIRFRAME RELIABILITY	98.50	23.	MAX DELIVER INCENTIVES	3.00
9.	LAUNCHER RELIABILITY	80.00	24.	MAXIMUM RELIABILITY	91.00
10.	FIRE CONTROL ERROR YDS	70.00	25.	MINIMUM RELIABILITY	85.00
11.	GUIDANCE ERROR TESTS	25.00	26.	MAX RELIABIL INCENTIVE	3.50
12.	MOTOR QUAL TESTS	6.00	27.	MAXIMUM ERROR YDS	160.00
13.	AIRFRAME QUAL TESTS	3.00	28.	MINIMUM ERROR YDS	140.00
14.	LAUNCHER QUAL TESTS	3.00	29.	MAX ERROR INCENTIVE	3.50
15.	FIRE CONTROL QUAL TESTS	3.00	30.	WEEK FOR LOT 10	337.00

*****31 INPUT BY PAGE

SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-3

Check each display carefully for erroneous inputs. The results computed are based on the values appearing in this table.
Change by page

*****TYPE IN A LIST OF PAGES YOU DESIRE TO CHANGE:*****
EX:1234 WILL ASK FOR NEW DATA ON ALL PAGES; OR, 23 WILL ASK FOR NEW DATA *****WARNING*****

YOU MUST ENTER AT LEAST ONE DIGIT EVEN IF IT IS A '0'.

Changing pages 1 and 2

=====>31

=====>12

.....R E E N T E R _ P A G E - 1.....

****INPUT PAGE 1 DP-3 INFO*****.

CONTRACTOR NUMBER,(1 OR 2):?

GUIDANCE APPROACH,(1 OR 2):?

GUIDANCE CONFIGURATION,(1-4;NOT 1 OR 3
AFTER DP-3?

MAINTAINABILITY ENGINEERING FUNDS IN M\$:?

VALUE ENGINEERING FUNDS IN M\$:?

====>1

====>1

====>2

====>1.75

====>.75

Parallel guidance selections 1 or 3 will
be rejected after DP-3.

NOTE No query for parallel guidance funds.
If guidance configuration selected is 2
or 4, or after DP-3, parallel
guidance funds will not be
queried. They are zeroed.

Checks of data values will occur at the
end of each page.
Necessary corrections will be queried.
Sample ****WARNING**** of data errors
are demonstrated below.

The value of parallel development funds
at \$0.5 million is not an
acceptable option.

The program now cues the operator to the
problem by asking for a correc-
tion in the guidance config-
uration which had prevented the
desired change in parallel
development funds.

.....D A T A C H E C K S.....

.....C O R R E C T I O N S.....

****WARNING****

PARALLEL DEV FUNDS OF \$ 0.50 M
ARE NOT REQUIRED AND HAVE BEEN RESET TO \$0.0

DO YOU WISH TO CHANGE GUIDANCE CONFIGURATION:
RY=Y OR "N"=NO?

====>N

=====>92
 =====>97
 =====>98.5
 =====>80
 =====>70
 =====>25
 =====>6
 =====>3
 =====>3
 =====>6
 =====>17

.....R E E N T E R _ P A G E - 2.....

*****INPUT PAGE 2 DP-3 INFO.*****

MOTOR RELIABILITY?
 AIRFRAME RELIABILITY?
 LAUNCHER/GSE RELIABILITY?
 FIRE CONTROL ACCURACY?
 GUIDANCE ACCURACY?
 MOTOR QUAL TESTS?
 AIRFRAME QUAL TESTS?
 LAUNCHER/GSE QUAL TESTS?
 FIRE CONTROL QUAL TESTS?
 GUIDANCE SYSTEM QUAL TESTS?
 FLIGHT TESTS?

.....D A T A C H E C K S.....
C O R R E C T J O N S.....

Checks of data values will occur at the
 end of each page.
 Necessary corrections will be queried.
 Sample *****WARNINGS***** of data errors
 are demonstrated below.

SELECTION MENU

```

NO. *ITEM*****CURRENT VALUE @ DP-3 NO. *ITEM*****CURRENT VALUE
1. CONTRACTIONS APPROACH : 1 16. GUIDANCE QUAL TESTS : 6.00
2. GUIDANCE CONFIGURATION : 2 17. FLIGHT TESTS : 17.00
3. GUIDANCE CONFIGURATION : 2 18. MAXIMUM COST : 53.00
4. MAINTAINABILITY ENG MS : 1.75 19. MINIMUM COST : 47.00
5. VALUE ENG MS : 0.75 20. MAX COST INCENTIVE : 5.00
6. PARALLEL DEVELOP MS : 0.0 21. LATEST WEEK : 238.00
7. MOTOR RELIABILITY : 92.00 22. EARLIEST WEEK : 202.00
8. AIRFRAME RELIABILITY : 97.00 23. MAX DELIVER INCENTIVE : 3.00
9. LAUNCHER RELIABILITY : 98.50 24. MAXIMUM RELIABILITY : 91.00
10. FIRE CONTROL ERROR YDS : 80.00 25. MINIMUM RELIABILITY : 85.00
11. GUIDANCE ERROR YDS : 70.00 26. MAX RELIABIL INCENTIVE : 3.50
12. MOTOR QUAL TESTS : 25.00 27. MAXIMUM ERROR YDS : 160.00
13. AIRFRAME QUAL TESTS : 6.00 28. MINIMUM ERROR YDS : 140.00
14. LAUNCHER QUAL TESTS : 3.00 29. MAX ERROR INCENTIVE : 3.50
15. FIRE CONTROL QUAL TESTS : 3.00 30. WEEK FOR LOT 10 : 337.00
*****0. NONE*****31. CHANGE BY PAGE*****
SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-3.

```

====>0

Zero from the Selection Menu sends the program to the Main Menu.

MAIN MENU

DO YOU WISH TO:

1. RECEIVE THE TABLE OF ACHIEVED VALUES.
 *****WARNING*****
 AFTER THE TENTH DP- 3 RUN
 AN ADDITIONAL DEVELOPMENT COST OF \$100,000
 WILL BE INCURRED FOR EACH ADDITIONAL RUN.
 THIS IS RUN 1 OF DP- 3.
2. INPUT SELECTION MENU
3. SUBMIT A CONTRACT PROPOSAL
4. EXIT

====>1

'1' tells the program to proceed with a check of the total fee percentage. It insures the total fee percentage, is 15% of the development contract, is allocated. If the fees do not total 15%, the program queries the team for corrected fee data. If the percentage check is passed, the program calculates achieved values, counts the run number for computation costs and displays the Achieved Values Table below.

ACHIEVED VALUES

DP-3 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 1

INCENTIVE AREA	INCENTIVE PROVISIONS		INCENTIVE ACHIEVEMENTS	
	WORST VALUE	BEST VALUE	ACHIEVED VALUE	FEE EARNED
DEV. COST	\$ 53.0M	\$ 47.0M	\$ 50.64M	0.984M
FLY TST COMPL	238 WK	202 WK	199 WK	1.500M
RELIABILITY	85.0%	91.0%	83.74%	0.0 M
ACCURACY	160YDS	140YDS	143YDS	1.470M
TOTALS				\$ 3.955M
				7.91%
TOTAL CONTRACT PRICE = \$ 54.6M				

IH0001A PAUSE;PRESS <<ENTER>> TO CONTINUE.

====><<ENTER>>

The "PAUSE" causes the screen to remain until the operator indicates he is ready to proceed.
<<ENTER>> symbolizes pressing the enter key which causes the program to start processing again.

MAIN MENU

DO YOU WISH TO:

1. RECEIVE THE TABLE OF ACHIEVED VALUES.
 *****WARNING*****AFTER THE TENTH DP- 3 RUN
 AN ADDITIONAL DEVELOPMENT COST OF \$100,000
 WILL BE INCURRED FOR EACH ADDITIONAL RUN.
 THIS IS RUN 2 OF DP- 3.

2. INPUT SELECTION MENU
 3. SUBMIT A CONTRACT PROPOSAL
 4. EXIT

Sometimes

the display will show "MORE" in the lower right margin of the screen. After several seconds the screen will scroll itself. PAUSE prevents this screen scrolling action. When "MORE" is displayed and the operator wishes to prevent the display from automatically scrolling, the display may be held by pressing <<ENTER>>. "HOLDING" will be displayed. To proceed from a "HOLDING" or "MORE" display, press the <<ALT>> and <<CLEAR>> keys simultaneously.

NOTE the count of calculation runs has changed. The program counts the occurrences of achieved value calculations. After ten calculations, the program adds a charge of \$100,000 for each additional run.

The result of selecting 2 from the Main Menu,
will be to reprint the Selection
Menu from which a data item may be
selected for change.

=====2

SELECTION MENU

```

NO. *ITEM*****CURRENT VALUE @ DP-3 NO. *ITEM*****CURRENT VALUE
1. CONTRACTOR APPROACH : 1 16. GUIDANCE QUAL TESTS : 17.00
2. GUIDANCE CONFIGURATION : 2 17. FLIGHT TESTS : 6.00
3. GUIDANCE CONFIGURATION : 2 18. MAXIMUM COST : 53.00
4. MAINTAINABILITY ENG MS : 1.75 19. MINIMUM COST : 47.00
5. VALUE ENG MS : 0.0 20. MAX COST INCENTIVE : 5.00
6. PARALLEL DEVELOP MS : 92.00 21. LATEST WEEK : 238.00
7. MOTOR RELIABILITY : 97.00 22. EARLIEST WEEK : 202.00
8. AIRFRAME RELIABILITY : 98.50 23. MAX DELIVER INCENTIVE : 3.00
9. LAUNCHER RELIABILITY : 60.00 24. MAXIMUM RELIABILITY : 91.00
10. FIRE CONTROL ERROR YDS : 70.00 25. MINIMUM RELIABILITY : 85.00
11. GUIDANCE ERROR YDS : 25.00 26. MAX RELIABIL INCENTIVE : 3.50
12. MOTOR QUAL TESTS : 6.00 27. MAXIMUM ERROR YDS : 160.00
13. AIRFRAME QUAL TESTS : 3.00 28. MINIMUM ERROR YDS : 140.00
14. LAUNCHER QUAL TESTS : 3.00 29. MAX ERROR INCENTIVE : 3.50
15. FIRE CONTROL QUAL TESTS : 3.00 30. WEEK FOR LOT 10 : 337.00
*****0. NONE*****31. CHANGE BY PAGE*****
      SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-3.
    
```

====>6
====>2

PREVIOUS VALUE 0.0; INPUT A VALUE?

PARALLEL DEV FUNDS OF \$ 2.00M
ARE NOT REQUIRED AND HAVE BEEN RESET TO \$0.0

The procedure for changing any data in the database is accomplished as follows.

To change the parallel development funds, choose item number 6 from the Selection Menu.

Parallel development funds are selected for change.

The new value input for parallel development funds is \$2 million.

The value of parallel development funds at \$2 million is not an acceptable option.

The program now cues the operator to the prob-

DO YOU WISH TO CHANGE GUIDANCE CONFIGURATION:
RY=YES OR "N"=NO?

lem by asking for a correction in the guidance configuration which prevented the desired change in parallel development funds. the After completing the change queries, the program reprints the Selection Menu with the changed values, ready for another selection.

SELECTION MENU

```
*****CURRENT VALUE @ DP-3*****CURRENT VALUE
1. CONTRACTOR : 1
2. GUIDANCE APPROACH : 1
3. GUIDANCE CONFIGURATION : 2
4. MAINTAINABILITY ENG MS: 1.75
5. VALUE ENG MS: 0.75
6. PARALLEL DEVELOP MS: 0.0
7. MOTOR RELIABILITY : 92.00
8. AIRFRAME RELIABILITY : 97.00
9. LAUNCHER RELIABILITY : 98.50
10. FIRE CONTROL ERROR YDS: 80.00
11. GUIDANCE ERROR YDS: 70.00
12. MOTOR QUAL TESTS : 25.00
13. AIRFRAME QUAL TESTS : 6.00
14. LAUNCHER QUAL TESTS : 3.00
15. FIRE CONTROL QUAL TESTS: 3.00
*****0. NONE*****31. CHANGE BY PAGE*****
SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-3.
NO. *ITEM*****CURRENT VALUE
16. GUIDANCE QUAL TESTS : 6.00
17. FLIGHT TESTS : 17.00
18. MAXIMUM COST MS: 53.00
19. MINIMUM COST MS: 47.00
20. MAX COST INCENTIVE %: 5.00<***** Notice the value.
21. LATEST WEEK : 238.00
22. FAULT TEST WEEK : 202.00
23. MAX DELIVER INCENTIVE %: 3.00
24. MAXIMUM RELIABILITY : 91.00
25. MINIMUM RELIABILITY : 85.00
26. MAX RELIABIL INCENTIVE %: 3.50
27. MAXIMUM ERROR YDS: 160.00
28. MINIMUM ERROR YDS: 140.00
29. MAX ERROR INCENTIVE %: 3.50
30. WEEK FOR LOT 10 : 337.00
*****0. NONE*****31. CHANGE BY PAGE*****
```

Choose the item to be changed, the maximum fee for development cost incentive. The previous value was 5%. The desired new value is 4%. The value is changed in the Selection Menu, and the menu is reprinted.

PREVIOUS VALUE 5.00; INPUT A VALUE ?

SELECTION MENU

NO.	*ITEM*****	CURRENT VALUE	@ DP-3	*ITEM*****	CURRENT VALUE
1.	CONTRACTOR	1	1	16.	GUIDANCE QUAL TESTS
2.	GUIDANCE APPROACH	2	1	17.	FLIGHT TESTS
3.	GUIDANCE CONFIGURATION	1.75	2	18.	MAXIMUM COST
4.	MAINTAINABILITY ENG	0.75	0	19.	MINIMUM COST
5.	VALUE ENG DEVELOP	0.0	0	20.	MAX COST INCENTIVE
6.	PARALLEL DEVELOP	97.00	97.00	21.	LATEST WEEK
7.	MOTOR RELIABILITY	97.00	97.00	22.	EARLIEST WEEK
8.	AIRFRAME RELIABILITY	98.50	98.50	23.	MAX DELIVER INCENTIVE
9.	LAUNCHER RELIABILITY	80.00	80.00	24.	MAXIMUM RELIABILITY
10.	FIRE CONTROL ERROR YDS	70.00	70.00	25.	MINIMUM RELIABILITY
11.	GUIDANCE ERROR	25.00	25.00	26.	MAX RELIABIL INCENTIVE
12.	MOTOR QUAL TESTS	6.00	6.00	27.	MAXIMUM ERROR
13.	AIRFRAME QUAL TESTS	3.00	3.00	28.	MINIMUM ERROR
14.	LAUNCHER QUAL TESTS	3.00	3.00	29.	MAX ERROR INCENTIVE
15.	FIRE CONTROL QUAL TESTS	3.00	3.00	30.	WEEK FOR LOT 10
				31.	CHANGE BY PAGE*****

*****31. CHANGE BY PAGE*****

SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-3.

***** Notice the new value.

====>0

WARNINGTOTAL OF FEE PERCENTAGES IS LESS THAN 15%.
THE VALUES YOU PREVIOUSLY SUBMITTED ARE:
COST FEE INCENTIVE= 4.00
DELIVERY DATE FEE INCENTIVE= 3.00
RELIABILITY INCENTIVE FEE= 3.50
ACCURACY INCENTIVE FEE= 3.50
REDO YOUR INPUT.

ENTER THE COST INCENTIVE FEE.
PREVIOUS VALUE 4.00; INPUT A VALUE?

YOU HAVE 11.00% OF THE INCENTIVE FEES REMAINING.
ENTER THE DELIVERY DATE INCENTIVE FEE.
PREVIOUS VALUE 3.00; INPUT A VALUE?

YOU HAVE 7.50% OF THE INCENTIVE FEES REMAINING.
ENTER THE RELIABILITY INCENTIVE FEE.
PREVIOUS VALUE 3.50; INPUT A VALUE?

YOU HAVE 4.00% OF THE INCENTIVE FEES REMAINING.
ENTER THE ACCURACY INCENTIVE FEE.
PREVIOUS VALUE 3.50; INPUT A VALUE?

====>4

====>3.5

====>3.5

====>3.5

The result of selecting 0 from the Selection Menu will be to reprint the Main Menu.
Zero from the Selection Menu sends the program to the Main Menu.
Exit from the Selection Menu routine by entering 0. The program will first evaluate the fee total percentage. Whenever the fee total does not equal 15% the program queries for correction.
These are the old values.

The program keeps track of the fee remainder. It tells how much fee remains to be allocated.

If the ACCURACY FEE entered is not equal to this last remainder, the program assumes a data entry error. It re-loops through the EXIT routine and can return to the percentage routine.

DO YOU WISH TO CONTINUE; "Y"=YES OR "N"=NO?

====>y

WARNINGTOTAL OF FEE PERCENTAGES IS LESS THAN 15%.
THE VALUES YOU PREVIOUSLY SUBMITTED ARE:
COST FEE INCENTIVE= 4.00
DELIVERY DATE FEE INCENTIVE= 3.00
RELIABILITY INCENTIVE FEE= 3.50
ACCURACY INCENTIVE FEE= 3.50
REDO YOUR INPUT.

ENTER THE COST INCENTIVE FEE.
PREVIOUS VALUE 4.00; INPUT A VALUE?

====>4

YOU HAVE 11.00% OF THE INCENTIVE FEES REMAINING.
ENTER THE DELIVERY DATE INCENTIVE FEE.
PREVIOUS VALUE 3.00; INPUT A VALUE?

====>3.5

YOU HAVE 7.50% OF THE INCENTIVE FEES REMAINING.
ENTER THE RELIABILITY INCENTIVE FEE.
PREVIOUS VALUE 3.50; INPUT A VALUE?

====>3.5

YOU HAVE 4.00% OF THE INCENTIVE FEES REMAINING.
ENTER THE ACCURACY INCENTIVE FEE.
PREVIOUS VALUE 3.50; INPUT A VALUE?

====>4.0

Unsuccessful correction of the percentage
fees returns the program to
the percentage correction routine.
These are the old values.

The program keeps track of the fee remainder.
it tells how much fee remains to
be allocated.

Successful completion of the corrections
returns the program to the Main
Menu.

MAIN MENU

DO YOU WISH TO:
1. RECEIVE THE TABLE OF ACHIEVED VALUES.
*****WARNING*****AFTER THE TENTH DP- 3 RUN
AN ADDITIONAL DEVELOPMENT COST OF \$100,000
WILL BE INCURRED FOR EACH ADDITIONAL RUN.
THIS IS RUN 2 OF DP- 3.

2. INPUT SELECTION MENU
3. SUBMIT A CONTRACT PROPOSAL
4. EXIT

====>1

ACHIEVED VALUES

DP-3 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 1

INCENTIVE AREA	INCENTIVE PROVISIONS			INCENTIVE ACHIEVEMENTS		
	WORST VALUE	BEST VALUE	MAX FEE ALLOWED	ACHIEVED VALUE	FEE EARNED	FEE % EARNED
DEV. COST	\$ 53.0M	\$ 47.0M	4.0%	\$ 50.81M	\$ 0.730M	1.46%
FLT TST COMPL	238 WK	202 WK	3.5%	199 WK	1.750M	3.50%
RELIABILITY	85.0%	91.0%	3.5%	83.81%	0.0 M	0.0%
ACCURACY	160YDS	140YDS	4.0%	143YDS	\$ 1.680M	3.36%
TOTALS			15.0%		\$ 4.161M	8.32%
TOTAL CONTRACT PRICE = \$ 55.0M						

The "PAUSE" causes the screen to retain the display until the operator indicates he is ready to continue.

===== <<ENTER>> | H0001A PAUSE ; PRESS <<ENTER>> TO CONTINUE.

MAIN MENU

DO YOU WISH TO:

1. RECEIVE THE TABLE OF ACHIEVED VALUES
*****WARNING*****AFTER THE TENTH DP- 3 RUN
AN ADDITIONAL DEVELOPMENT COST OF \$100,000
WILL BE INCURRED FOR EACH ADDITIONAL RUN.
THIS IS RUN 3 OF DP- 3.
2. INPUT SELECTION MENU
3. SUBMIT A CONTRACT PROPOSAL
4. EXIT

From the Main menu, selection of 3
provides a printout of the database
as the Report Submission Table.

====>3

PROPOSAL SUBMISSION TABLE

```

NO. *ITEM*****CURRENT VALUE @ DP-3 NO. *ITEM*****CURRENT VALUE
1. CONTRACTOR APPROACH : 1 16. GUIDANCE QUAL TESTS : 6.00
2. GUIDANCE CONFIGURATION : 2 17. FLIGHT TESTS : 17.00
3. GUIDANCE CONFIGURATION : 2 18. MAXIMUM COST : 53.00
4. MAINTAINABILITY ENG : 1.75 19. MINIMUM COST : 47.00
5. VALUE ENG : 0.0 20. MAX COST INCENTIVE : 4.00
6. PARALLEL DEVELOP : 0.0 21. LATEST WEEK : 238.00
7. MOTOR RELIABILITY : 92.00 22. EARLIEST WEEK : 202.00
8. AIRFRAME RELIABILITY : 97.00 23. MAX DELIVER INCENTIVE : 3.50
9. LAUNCHER RELIABILITY : 98.50 24. MAXIMUM RELIABILITY : 91.00
10. FIRE CONTROL ERROR YDS : 80.00 25. MINIMUM RELIABILITY : 85.00
11. GUIDANCE ERROR : 70.00 26. MAX RELIABIL INCENTIVE : 3.50
12. MOTOR QUAL TESTS : 25.00 27. MAXIMUM ERROR YDS : 150.00
13. AIRFRAME QUAL TESTS : 6.00 28. MINIMUM ERROR YDS : 140.00
14. LAUNCHER QUAL TESTS : 3.00 29. MAX ERROR INCENTIVE : 4.00
15. FIRE CONTROL QUAL TESTS : 3.00 30. WEEK FOR LOT 10 : 337.00
*****DO YOU WISH TO ENTER THIS DATA AS YOUR PROPOSED DP-3 DECISION?*****
*****CAUTION*****CAUTION*****CAUTION*****
ENTER YOUR TEAM SECURITY CODE IF YOU WISH TO COMMIT TO A DECISION;
OR CONT.

```

====>cont

The result of entering "continue" is to
return and reprint the Main Menu.
Continue returns to the Main Menu without
submitting a report.

MAIN MENU

DO YOU WISH TO:

1. RECEIVE THE TABLE OF ACHIEVED VALUES.
*****WARNING*****AFTER THE TENTH DP- 3 RUN
AN ADDITIONAL DEVELOPMENT COST OF \$100,000
WILL BE INCURRED FOR EACH ADDITIONAL RUN.
THIS IS RUN 3 OF DP- 3.
2. INPUT SELECTION MENU
3. SUBMIT A CONTRACT PROPOSAL
4. EXIT

====>3

Provides a printout of the database to
verify the proposed values being
submitted.

PROPOSAL SUBMISSION TABLE

NO.	ITEM	CURRENT VALUE	@ DP-3	NO.	ITEM	CURRENT VALUE
1.	CONTRACTOR	1		16.	GUIDANCE QUAL TESTS	6.00
2.	GUIDANCE APPROACH	1		17.	FLIGHT TESTS	17.00
3.	GUIDANCE CONFIGURATION	2		18.	MAXIMUM COST	53.00
4.	MAINTAINABILITY ENG	1.75		19.	MINIMUM COST	47.00
5.	VALUE ENG	0.0		20.	MAX COST INCENTIVE	4.00
6.	PARALLEL DEVELOP	0.0		21.	LATEST WEEK	238.00
7.	MOTOR RELIABILITY	92.00		22.	EARLIEST WEEK	202.00
8.	AIRFRAME RELIABILITY	97.00		23.	MAX DELIVER INCENTIVE	3.50
9.	LAUNCHER RELIABILITY	98.50		24.	MAXIMUM RELIABILITY	91.00
10.	FIRE CONTROL ERROR YDS	80.00		25.	MINIMUM RELIABILITY	85.00
11.	GUIDANCE ERROR YDS	70.00		26.	MAX RELIABIL INCENTIVE	3.50
12.	MOTOR QUAL TESTS	25.00		27.	MAXIMUM ERROR	160.00
13.	AIRFRAME QUAL TESTS	6.00		28.	MINIMUM ERROR	140.00
14.	LAUNCHER QUAL TESTS	3.00		29.	MAX ERROR INCENTIVE	4.00
15.	FIRE CONTROL QUAL TESTS	3.00		30.	WEEK FOR LOT 10	337.00

*****CAUTION*****
 THIS DATA AS YOUR PROPOSED DP-3 DECISION?
 YOU MAY ONLY INPUT A DECISION ONCE
 ENTER YOUR TEAM SECURITY CODE IF YOU WISH TO COMMIT TO A DECISION;
 OR CONT.

REPORT ACCEPTED VALUES TABLE

```

*****THE FOLLOWING PARAMETERS WERE ACCEPTED*****
AS TEAM 1 DP-3 PROPOSED INPUT TO THE CONTRACTOR.
*****CURRENT VALUE @ DP-3 NO. ITEM*****CURRENT VALUE
1. CONTRACTOR APPROACH 1 17.00
2. GUIDANCE CONFIGURATION 2 53.00
3. GUIDANCE CONFIGURATION 2 47.00
4. MAIN TAINABILITY ENG MS 1.75
5. VALUE ENG MS 0.75
6. PARALLEL DEVELOP MS 92.00
7. MOTOR RELIABILITY 97.00
8. AIRFRAME RELIABILITY 98.50
9. LAUNCHER RELIABILITY 80.00
10. FIRE CONTROL ERROR YDS 70.00
11. GUIDANCE ERROR YDS 25.00
12. MOTOR QUAL TESTS 6.00
13. AIRFRAME QUAL TESTS 3.00
14. LAUNCHER QUAL TESTS 3.00
15. FIRE CONTROL QUAL TESTS 3.00
*****SEE YOUR MONITOR IF YOU HAVE A PROBLEM.*****

16. GUIDANCE QUAL TESTS 6.00
17. FLIGHT TESTS 17.00
18. MAXIMUM COST MS 53.00
19. MINIMUM COST MS 47.00
20. MAX COST INCENTIVE % 4.00
21. LATEST WEEK 238.00
22. EARLIEST WEEK 202.00
23. MAX DELIVER INCENTIVE % 3.50
24. MAXIMUM RELIABILITY 91.00
25. MINIMUM RELIABILITY 85.00
26. MAX RELIABIL INCENTIVE % 3.50
27. MAXIMUM ERROR YDS 160.00
28. MINIMUM ERROR YDS 140.00
29. MAX ERROR INCENTIVE % 4.00
30. WEEK FOR LOT 10 337.00

```

Entering the team's security code will rewrite the file DATABASE TEAMxx. From the Main menu, selection of option 3 directs the program to the report submission routine. This routine determines the DP report last submitted. It then displays the current database selections in a Report Submission table as the offering for the next report. Team approval of the Submission table is made by entering the team's security code. The program can sequence the team through

the stages of contract submission including DP-3 and 4's proposed and final contract reports. Pressing <ENTER> returns the program to the Main Menu display.

====><<ENTER>> IH0001A PAUSE ;PRESS <<ENTER>> TO CONTINUE.

MAIN MENU

DO YOU WISH TO:

1. RECEIVE THE TABLE OF ACHIEVED VALUES.
*****WARNING*****AFTER THE TENTH DP- 3 RUN
AN ADDITIONAL DEVELOPMENT COST OF \$100,000
WILL BE INCURRED FOR EACH ADDITIONAL RUN.
THIS IS RUN 3 OF DP- 3.
2. INPUT SELECTION MENU
3. SUBMIT A CONTRACT PROPOSAL
4. EXIT

====>4

PUN FILE 8092 TO 0543P COPY 001 NOHOLD
PUN FILE 8094 TO 0543P COPY 001 NOHOLD
ANOTHER RUN, 'Y'/'N'?

====>Y

BEGIN RECORDING OF TERMINAL SESSION
* NOTE!: YOU ARE NOW LINKED TO PROJMC ON YOUR 192 DISK
MODE C
* NOTE!: PRESS PF09 TO BREAK THE LINK
FILE 'DATAINST DISKNUM A' NOT FOUND
FILE 'DATAFILE TEAM01 A' NOT FOUND
FILE 'DATAFILE TEAM01 C' NOT FOUND.

WAIT FOR "EXECUTION TO BEGIN"
EXECUTION BEGINS...

Exits from the program.
The data files are now transmitted to the monitor.
The LINKPROJ EXEC program allows the user to reloop to the program start and begin again.

another session.
This time DATAFILE TEAM01 A is found and so is DATAFILE TEAM01 A.
They now exist on your A disk.

SELECTION MENU

NO.	*ITEM*****	CURRENT VALUE	@ DP-3	NO.	*ITEM*****	CURRENT VALUE
1.	CONTRACTOR	1	1	16.	GUIDANCE QUAL TESTS	6.00
2.	GUIDANCE APPROACH	2	2	17.	FLIGHT TESTS	17.00
3.	GUIDANCE CONFIGURATION	1.75	1.75	18.	MAXIMUM COST	53.00
4.	MAINTAINABILITY ENG	0.0	0.0	19.	MINIMUM COST	47.00
5.	VALUE ENG	0.0	0.0	20.	MAX COST INCENTIVE	4.00
6.	PARALLEL DEVELOP	92.00	92.00	21.	LATEST WEEK	238.00
7.	MOTOR RELIABILITY	97.00	97.00	22.	EARLIEST WEEK	202.00
8.	AIRFRAME RELIABILITY	98.50	98.50	23.	MAX DELIVER INCENTIVE	3.50
9.	LAUNCHER RELIABILITY	80.00	80.00	24.	MAXIMUM RELIABILITY	91.00
10.	FIRE CONTROL ERROR YDS	70.00	70.00	25.	MINIMUM RELIABILITY	85.00
11.	GUIDANCE ERROR	25.00	25.00	26.	MAX RELIABL INCENTIVE	3.50
12.	MOTOR QUAL TESTS	6.00	6.00	27.	MAXIMUM ERROR	160.00
13.	AIRFRAME QUAL TESTS	3.00	3.00	28.	MINIMUM ERROR	140.00
14.	LAUNCHER QUAL TESTS	3.00	3.00	29.	MAX ERROR INCENTIVE	4.00
15.	FIRE CONTROL QUAL TESTS	3.00	3.00	30.	WEEK FOR LOT 10	337.00

*****0. NONE*****31. CHANGE BY PAGE*****
 SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-3.

====>0

MAIN MENU

DO YOU WISH TO:

1. RECEIVE THE TABLE OF ACHIEVED VALUES.
*****WARNING*****AFTER THE TENTH DP- 3 RUN
AN ADDITIONAL DEVELOPMENT COST OF \$100,000
WILL BE INCURRED FOR EACH ADDITIONAL RUN.
THIS IS RUN 3 OF DP- 3.

2. INPUT SELECTION MENU
3. SUBMIT A CONTRACT PROPOSAL
4. EXIT

====>3

FINAL SUBMISSION TABLE FOR DP-3

NO.	ITEM	CURRENT VALUE	DP-3	NO.	ITEM	CURRENT VALUE
1.	CONTRACTOR	1		16.	GUIDANCE QUAL TESTS	6.00
2.	GUIDANCE APPROACH	2		17.	FLIGHT TESTS	17.00
3.	GUIDANCE CONFIGURATION	1.75		18.	MAXIMUM COST	53.00
4.	MAINTAINABILITY ENG	0.75		19.	MINIMUM COST	47.00
5.	VALUE ENG	0.0		20.	MAX COST INCENTIVE	4.00
6.	PARALLEL DEVELOP	0.0		21.	LATEST WEEK	238.00
7.	MOTOR RELIABILITY	92.00		22.	EARLIEST WEEK	202.00
8.	AIRFRAME RELIABILITY	97.00		23.	MAX DELIVER INCENTIVE	3.50
9.	LAUNCHER RELIABILITY	98.50		24.	MAXIMUM RELIABILITY	91.00
10.	FIRE CONTROL ERROR YDS	80.00		25.	MINIMUM RELIABILITY	85.00
11.	GUIDANCE ERROR	70.00		26.	MAX RELIABILITY INCENTIVE	3.50
12.	MOTOR QUAL TESTS	25.00		27.	MAXIMUM ERROR	160.00
13.	AIRFRAME QUAL TESTS	6.00		28.	MINIMUM ERROR	140.00
14.	LAUNCHER QUAL TESTS	3.00		29.	MAX ERROR INCENTIVE	4.00
15.	FIRE CONTROL QUAL TESTS	3.00		30.	WEEK FOR LOT 10	337.00

*****IMPORTANT*****
 *****YOU MUST CHECK THE ABOVE ENTRIES WITH THE APPROVED FINAL DP-3 *****
 ENTER YOUR TEAM SECURITY CODE IF YOU WISH TO A DECISION;
 OR "CONT".

The following sequence will demonstrate first the filing of a final DP-3 contract proposal;
 second we will exit from the program to reenter the program at DP-4
 fourth we will review the DP-4 Selection Menu as developed from the current DP-3 data base;
 next the changing of the paired sets of data for expected minimum and maximum reliability;
 sixth DP-4 will be evaluated with

• a table of achieved values; and
 finally, we will exit from the
 program.
 Entering your team's security code will
 submit the final DP-3 report to
 the monitor.

====>01kline

REPORT ACCEPTED VALUES TABLE

```

*****THE FOLLOWING PARAMETERS WERE ACCEPTED*****
AS TEAM 1 DP-3 FINAL*** INPUT TO THE CONTRACTOR.
NO.*ITEM*****CURRENT VALUE @ DP-3 NO.*ITEM*****CURRENT VALUE
1. CONTRACTOR APPROACH : 1 16. GUIDANCE QUAL TESTS : 6.00
2. GUIDANCE CONFIGURATION : 2 17. FLIGHT TESTS : 17.00
3. MAINTAINABILITY ENG MS: 1.75 18. MAXIMUM COST MS: 53.00
4. VALUE ENG MS: 0.75 19. MINIMUM COST MS: 47.00
5. PARALLEL DEVELOP MS: 0.0 20. MAX COST INCENTIVE : 4.00
6. MOTOR RELIABILITY : 92.00 21. LATEST WEEK : 238.00
7. AIRFRAME RELIABILITY : 97.00 22. EARLIEST WEEK : 202.00
8. LAUNCHER RELIABILITY : 98.50 23. MAX DELIVER INCENTIVE : 3.50
9. FIRE CONTROL ERROR YDS: 80.00 24. MAXIMUM RELIABILITY : 91.00
10. GUIDANCE ERROR : 25.00 25. MINIMUM RELIABILITY : 85.00
11. MOTOR QUAL TESTS : 6.00 26. MAX R/LIABIL INCENTIVE : 3.50
12. AIRFRAME QUAL TESTS : 3.00 27. MAXIMUM ERROR YDS: 160.00
13. LAUNCHER QUAL TESTS : 3.00 28. MINIMUM ERROR YDS: 140.00
14. FIRE CONTROL QUAL TESTS: 3.00 29. MAX ERROR INCENTIVE : 4.00
15. *****SEE YOUR MONITOR IF YOU HAVE A PROBLEM. 30. WEEK FOR LOT 10 : 337.00
  
```

IH0001A PAUSE ; PRESS <<ENTER>> TO CONTINUE.

====><<ENTER>>

MAIN MENU

DO YOU WISH TO:
 1. RECEIVE THE TABLE OF ACHIEVED VALUES.
 *****WARNING*****AFTER THE TENTH DP-4 RUN
 AN ADDITIONAL DEVELOPMENT COST OF \$100,000
 WILL BE INCURRED FOR EACH ADDITIONAL RUN
 THIS IS RUN 1 OF DP-4.

2. INPUT SELECTION MENU.
 3. SUBMIT A CONTRACT PROPOSAL
 4. EXIT

NOTE the change in DP- level
 in this block.

====>4

====>Y

This step will stop the program
 and return control to the EXEC,
 PROJMG EXEC. transmitted
 The data files are now transmitted
 to the monitor.
 The LINKPROJ EXEC program allows the user to
 reloop to the program start and
 begin again.

RUN FILE 8092 TO 0543P COPY 001 NOHOLD
 RUN FILE 8094 TO 0543P COPY 001 NOHOLD
 ANOTHER RUN, Y/N?

DP-4

```
BEGIN RECORDING OF TERMINAL SESSION
* NOTE!: YOU ARE NOW LINKED TO PROJMC ON YOUR 192 DISK
MODE C
* NOTE!: PRESS PF09 TO BREAK THE LINK
FILE 'DATAINST DISKNUM A' NOT FOUND
FILE 'DATAFILE TEAM01 C' NOT FOUND
FILE 'DATACODE TEAM01 C' NOT FOUND
PRESS <<ENTER> WHEN YOU ARE READY TO CONTINUE

WAIT FOR "EXECUTION TO BEGIN"
EXECUTION BEGINS...

*****
***** TO TERMINATE THE PROGRAM AT ANY POINT, QUERY
***** TYPE "E"=EXIT IN RESPONSE TO ANY "YES/NO", QUERY
***** *****
*****CAUTION***** IN RESPONSE TO A QUERY, PRESSING <ENTER>
***** WITHOUT PROVIDING DATA WILL DUMP THE PROGRAM.
***** *****

PLEASE ENTER YOUR TEAM SECURITY CODE.
```

====>01kline

The following are DP-4 procedures.
Logon as you did for DP-3.

another session.
This time DATAFILE TEAM01 A is
found; and, so is DATACODE TEAM01 A.
They now exist on your A disk.

The program will now proceed to check the
database for previous information.
It will display the data in a
SELECTION MENU.

SELECTION MENU

```
NO.*ITEM*****CURRENT VALUE @ DP-4 NO.*ITEM*****CURRENT VALUE
1. CONTRACTOR : 1 16. GUIDANCE QUAL TESTS : 6.00
2. GUIDANCE APPROACH : 1 17. FLIGHT TESTS : 17.00
3. GUIDANCE CONFIGURATION : 2 18. MAXIMUM COST : 53.00
4. MAINTAINABILITY ENG MS : 1.75 19. MINIMUM COST : 47.00
5. VALUE ENG MS : 0.0 20. MAX COST INCENTIVE % : 4.00
6. PARALLEL DEVELOP MS : 92.00 21. LATEST WEEK : 238.00
7. MOTOR RELIABILITY : 97.00 22. EARLIEST WEEK : 202.00
8. AIRFRAME RELIABILITY : 98.50 23. MAX DELIVER INCENTIVE % : 3.50
9. LAUNCHER RELIABILITY : 80.00 24. MAXIMUM RELIABILITY : 91.00
10. FIRE CONTROL ERROR YDS : 70.00 25. MINIMUM RELIABILITY : 85.00
11. GUIDANCE ERROR : 25.00 26. MAX RELIABIL INCENTIVE % : 3.50
12. MOTOR QUAL TESTS : 6.00 27. MAXIMUM ERROR YDS : 160.00
13. AIRFRAME QUAL TESTS : 3.00 28. MINIMUM ERROR YDS : 140.00
14. LAUNCHER QUAL TESTS : 3.00 29. MAX ERROR INCENTIVE % : 4.00
15. FIRE CONTROL QUAL TESTS : 3.00 30. WEEK FOR LOT 10 : 337.00
*****31. CHANGE BY PAGE*****
SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-4.
```

We are changing the maximum reliability.

PREVIOUS VALUE 91.00; INPUT A VALUE

DO YOU DESIRE TO CHANGE MINIMUM RELIABILITY, ALSO.

PREVIOUS VALUE 85.00; INPUT A VALUE

The new value for maximum reliability is 85.
The program pairs the maximums and minimums for queries. If one is changed the program will ask if you wish to change the other.

====>24

====>85

====>y

====>78

SELECTION MENU

NO.	*ITEM*****CURRENT VALUE @ DP-4	NO.	*ITEM*****CURRENT VALUE
1.	CONTRACTOR APPROACH	16.	GUIDANCE QUAL TESTS
2.	GUIDANCE CONFIGURATION	17.	FLIGHT TESTS
3.	GUIDANCE CONFIGURATION MS	18.	MAXIMUM COST MS
4.	MAINTAINABILITY ENG MS	19.	MINIMUM COST MS
5.	VALUE ENG	20.	MAX COST INCENTIVE
6.	PARALLEL DEVELOP MS	21.	LATEST WEEK
7.	MOTOR RELIABILITY	22.	EARLIEST WEEK
8.	AIRFRAME RELIABILITY	23.	MAX DELIVER INCENTIVE
9.	LAUNCHER RELIABILITY	24.	MAXIMUM RELIABILITY
10.	FIRE CONTROL ERROR YDS	25.	MINIMUM RELIABILITY
11.	GUIDANCE ERROR	26.	MAX RELIABIL INCENTIVE
12.	MOTOR QUAL TESTS	27.	MAXIMUM ERROR
13.	AIRFRAME QUAL TESTS	28.	MINIMUM ERROR
14.	LAUNCHER QUAL TESTS	29.	MAX ERROR INCENTIVE
15.	FIRE CONTROL QUAL TESTS	30.	WEEK FOR LOT 10
	0. NONE	31.	CHANGE BY PAGE**

SELECT AN ITEM WHICH NEEDS TO BE CHANGED FOR DP-4.

*****Notice the new value.

====>0

MAIN MENU

DO YOU WISH TO: TABLE OF ACHIEVED VALUES.
1. RECEIVE THE WARNING*****AFTER THE TENTH DP-4 RUN
AN ADDITIONAL DEVELOPMENT COST OF \$100,000
WILL BE INCURRED FOR EACH ADDITIONAL RUN.
THIS IS RUN 1 OF DP-4.
2. INPUT SELECTION MENU.
3. SUBMIT A CONTRACT PROPOSAL
4. EXIT

ACHIEVED VALUES

DP-4 *** DEVELOPMENT CONTRACT SUMMARY *** TEAM 1

INCENTIVE AREA	INCENTIVE PROVISIONS			INCENTIVE ACHIEVEMENTS		
	WORST VALUE	BEST VALUE	MAX FEE ALLOWED	ACHIEVED VALUE	FEE EARNED	FEE % EARNED
DEV. COST	\$ 53.0M	\$ 47.0M	4.0%	\$ 50.87M	\$ 0.710M	1.42%
FLT TST COMPL	238 WK	202 WK	3.5%	199 WK	\$ 1.750M	3.50%
RELIABILITY	78.0%	85.0%	3.5%	83.81%	\$ 1.452M	2.90%
ACCURACY	160YDS	140YDS	4.0%	143YDS	\$ 1.681M	3.36%
TOTALS			15.0%		\$ 5.593M	11.19%
TOTAL CONTRACT PRICE = \$ 56.5M						

IH0001A PAUSE ; PRESS <<ENTER>> TO CONTINUE.

====><<ENTER>>

The "PAUSE" causes the screen to retain the display until the operator is ready to continue.

MAIN MENU

DO YOU WISH TO: TABLE OF ACHIEVED VALUES.
1. RECEIVE THE *****WARNING*****AFTER THE TENTH DP-4 RUN
AN ADDITIONAL DEVELOPMENT COST OF \$100,000
WILL BE INCURRED FOR EACH ADDITIONAL RUN.
THIS IS RUN 2 OF DP-4.
2. INPUT SELECTION MENU.
3. SUBMIT A CONTRACT PROPOSAL
4. EXIT

====>3

PROPOSAL SUBMISSION TABLE

```

NO. * ITEM*****CURRENT VALUE @ DP-4 NO. * ITEM*****CURRENT VALUE
1. CONTRACTOR : 1 16. GUIDANCE QUAL TESTS : 6.00
2. GUIDANCE APPROACH : 1 17. FLIGHT TESTS : 17.00
3. GUIDANCE CONFIGURATION : 2 18. MAXIMUM COST : 53.00
4. MAINTAINABILITY ENG : 1.75 19. MINIMUM COST : 47.00
5. VALUE ENG : 0.75 20. MAX COST INCENTIVE : 4.00
6. PARALLEL DEVELOP : 0.0 21. LATEST WEEK : 238.00
7. MOTOR RELIABILITY : 92.00 22. EARLIEST WEEK : 202.00
8. AIRFRAME RELIABILITY : 97.00 23. MAX DELIVER INCENTIVE : 3.50
9. LAUNCHER RELIABILITY : 98.50 24. MAXIMUM RELIABILITY : 85.00
10. FIRE CONTROL ERROR YDS : 80.00 25. MINIMUM RELIABILITY : 78.00
11. GUIDANCE ERROR YDS : 70.00 26. MAX RELIABIL INCENTIVE : 3.50
12. MOTOR QUAL TESTS : 25.00 27. MAXIMUM ERROR YDS : 160.00
13. AIRFRAME QUAL TESTS : 6.00 28. MINIMUM ERROR : 140.00
14. LAUNCHER QUAL TESTS : 3.00 29. MAX ERROR INCENTIVE : 4.00
15. FIRE CONTROL QUAL TESTS : 3.00 30. WEEK FOR LOT 10 : 337.00
*****DO YOU WISH TO ENTER THIS DATA AS YOUR PROPOSED DP-4 DECISION?*****
*****CAUTION*****CAUTION*****

```

ENTER YOUR TEAM SECURITY CODE IF YOU WISH TO COMMIT TO A DECISION;
OR "CONT".

====>01kline

Entering the team's security code
completes the filing of the new
report to the monitor.

```

*****THE FOLLOWING PARAMETERS WERE ACCEPTED*****
AS TEAM 1 DP-4 PROPOSED INPUT TO THE CONTRACTOR.
*****CURRENT VALUE @ DP-4 NO.*ITEM*****CURRENT VALUE
1. CONTRACTOR : 1
2. GUIDANCE APPROACH : 2
3. GUIDANCE CONFIGURATION : 1
4. MAINTAINABILITY ENG MS : 1.75
5. VALUE ENG MS : 0.75
6. PARALLEL DEVELOP MS : 0.0
7. MOTOR RELIABILITY : 92.00
8. AIRFRAME RELIABILITY : 97.00
9. LAUNCHER RELIABILITY : 98.50
10. FIRE CONTROL ERROR YDS : 80.00
11. GUIDANCE ERROR YDS : 70.00
12. MOTOR QUAL TESTS : 25.00
13. AIRFRAME QUAL TESTS : 6.00
14. LAUNCHER QUAL TESTS : 3.00
15. FIRE CONTROL QUAL TESTS : 3.00
*****SEE YOUR MONITOR IF YOU HAVE A PROBLEM.*****

16. GUIDANCE QUAL TESTS : 17.00
17. FLIGHT TESTS MS : 53.00
18. MAXIMUM COST MS : 47.00
19. MINIMUM COST : 4.00
20. MAX COST INCENTIVE : 338.00
21. LATEST WEEK : 202.00
22. EARLIEST WEEK : 3.50
23. MAX DELIVER INCENTIVE : 3.50
24. MAXIMUM RELIABILITY : 85.00
25. MINIMUM RELIABILITY : 78.00
26. MAX RELIABIL INCENTIVE : 3.50
27. MAXIMUM ERROR YDS : 160.00
28. MINIMUM ERROR YDS : 400.00
29. MAX ERROR INCENTIVE : 4.00
30. WEEK FOR LOT 10 : 337.00
*****

```

IH0001A PAUSE ;PRESS <ENTER>> TO CONTINUE.

====><<ENTER>>

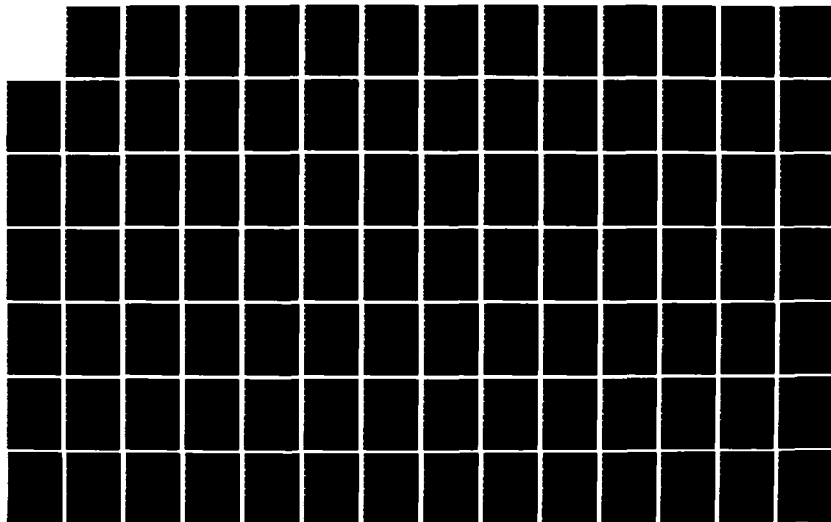
AD-A140 709

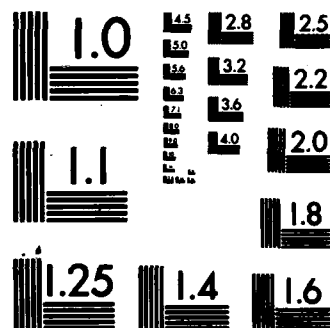
PROJMG FORTRAN: AN INTERACTIVE COMPUTER PROGRAM FOR
USE WITH THE DEFENSE MANAGEMENT SIMULATION EXERCISE(U)
NAVAL POSTGRADUATE SCHOOL MONTEREY CA G W SCHULTZ
MAR 84 F/G 9/2

3/4

UNCLASSIFIED

NL





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

MAIN MENU

DO YOU WISH TO: TABLE OF ACHIEVED VALUES.
 1. RECEIVE THE *****WARNING*****AFTER THE TENTH DP-4 RUN
 AN ADDITIONAL DEVELOPMENT COST OF \$100,000
 WILL BE INCURRED FOR EACH ADDITIONAL RUN.
 THIS IS RUN 2 OF DP-4.
 2. INPUT SELECTION MENU.
 3. SUBMIT A CONTRACT PROPOSAL
 4. EXIT

PUN FILE 8029 TO 0276P COPY 001 NOHOLD
 PUN FILE 8030 TO 0276P COPY 001 NOHOLD
 ANOTHER RUN, 'Y'/'N'?

END RECORDING OF TERMINAL SESSION

====>4

====>n

APPENDIX G
PROJING FORTRAN

```

*****
C      P E O J N N G   P O R T A N
C
C      NAVAL POSTGRADUATE SCHOOL MONTEREY, CALIFORNIA 93943
C
C      MARCH 1984
C
C      Defense Management Simulation
C
C      Simulation and Computer Directorate
C      Industrial College of the Armed Forces
C      Port Lesley J. McNair, Washington, D.C
C
C      1970 April 1
C
C      The following fortran code listing represents the reformatting and
C      aligned version of PROJNNG PCHTRAM for inclusion as Appendix G
C      in a thesis of the same name. The thesis PROJNNG PORTMAN was under-
C      taken in June 1983 to upgrade the computer software support for the
C      DMS exercise. The main effort of the thesis was to make the program
C      user-friendly.
C
C      Requests for this document should be referred to the Superintendent,
C      Naval Postgraduate School, Monterey, California 93943.
C
C      For a list of the variables definitions, see the Glossary,
C      Appendix H
C
C      CCHNOM RESI(48), TIB(20,25,3), CTEST(3), CTOH(3), XNIS(2), YSUBS(48)
C      IFAC(48), ITHN(48), IRC(48), IDDT(48), CHAT(48)
C      IPD(48), ITHN(48), QIV(48), RESB(48), ITYPE(48)
C      CCHNOM RESI(48), ITHN(48), RESD(48), DEC(3), Y(8), RESA(48)
C      PERC(48), TDUE(48), RISC(48), YF(48);
C      IPRC(48);

```



```

410 CALL SELECT(KDE)
C    CALL STCR
C    GOTO 207
C

75 CALL SETUP(KDP)

CONTINUE
      (UPDATE DATABASE BY 'SELECTION' MENU)
      (START FROM LAST COMPLETE DATABASE)

JDP=KDE
CONTINUE
      IF (NDUP(JDP,2) .NE. 0.OR. JEP.LE.3) GOTO 76
JDP=KDE-1
      GOTO 75
CONTINUE

76 CALL SELECT(JDE)
C    PREC AL C U L A T I O N D A T A P R O C E S S I N G
C    .....
C    ..... (CHECK THAT FEE TOTAL PERCENTAGE IS 15%)
C    .....

207 CALL PRCTCK
      CALL SETUP(KDP)

      (STUDENT SPECIFIC PROCESSING TO PROVIDE FURTHER DATA PREPROCESSING
      AND PREVENT UNDC WASTE OF CALCULATION TURN: AFTER 10 TURNS AT A DP
      CCST OF $100,000 PROCESSING FEE IS ASSESSED.)

CCCONTINUE
      (DETERMINE DP, DECISION POINT; AND, PROMPT REPORT SUBMISSIONS)
      IF (MR.NE.0) GOTO 231
      KDP=3
      IF (STAT{3} .EQ. FINAL) KDP=4
      IF (STAT{4} .EQ. FINAL) KDP=5
      IF (STAT{5} .EQ. FINAL.AND. MR.EQ.0) STOP
      IF (STAT{5} .EQ. FINAL.AND. MR.NE.0) GOTO 859
      NEWCP=KDP
      CALL SETUP(KDP)

231

```



```

C      CALL GET (KDP)
      IF (KDP.NE.3) CALL INPUT2

      DPCOST=0.0
      DC 876 IKDP=3, KDP
      IF (ADUMP(IKDP, 28) - 10.) * 100000.
      DPCOST=DPCOST+ (ADUMP(IKDP, 28) - 10.) * 100000.
      IDUMP=ADUMP (KDP, 28) + 1
      (BUFFER DATA ORGANIZATION FOR CALCULATIONS)
      IF (NR.NE.0) GOTO 415
      CALL STORE

      CALL FRTCHS ('CLRSCRN ')
      IQUESH=4
      IQUESH=999
      PRINT 211 KDP, IDUMP, KDP
      *1X, 'RECEIVED THE TABLE OF ACHIEVED V
      *1X, 'AFTER THE TENTH DE- , 11, ' RUN'./
      *1X, 'WILL BE INCURRED FOR EACH ADDITIONAL RUN'./ , 10X,
      *1X, 'IS RUN , 12, ' OF DE- , 11, '
      IF (DPCOST.GT.0.) PRINT 225, DPCOST
      IF (DUNS HAVE COST AN ADDITIONAL $ , F8.0)
      FCNMT (10X, 'DP RUNS HAVE COST AN ADDITIONAL $ , F8.0)
      PRINT 212
      FCNMT (1X, '2. INPUT SELECTION MENU' )
      PRINT 213
      FCNMT (1X, '3. SUBMIT A CONTRACT PROPOSAL' )
      IF (NR.EQ.0) GOTO 216
      IQUESH=4
      IQUESH=5
      PRINT 214, IQUESH
      FCNMT (1X, I1, ' MONITOR'S MENU' )
      PRINT 217, IQUESH
      FCNMT (1X, I1, ' EXIT' )
      READ (IN 208, ERR=240, END=240) ANT
      FCNMT (11)
      IF (ANT.EQ.E) CALL EXITS

      IF (ANT.NE.1) GOTO 226
      IF (STORPL.EQ.999.) GOTO 415
      (ADD THE COST FOR THIS PRINTOUT)

```

```
IF(MR-EQ-0)ADUMP(KDP,28)=AFUMP(KDP,28)+1.
IF(ADUMP(IKDP,28):GT:10.)
DPCDST=DPCOST+100000.
```

```

*
226  GOTO 415
      IF (ANT-EO: IQUESE) STOP 859
      IF (ANT-EO: IQUESE) STOP FINISH
      IF (ANT-EO: 3) CALL 25
      IF (ANT-EO: 3) GOTO 25
      GCNT=GCNT+1
      CCNT=CCNT+1
      GO TO 230
240  CCNT=CCNT+1
      GO TO 230

```

[illegible]

```
CALL ZHW NTEMP(1)=KDP
DO 850 IVER=3,KDP CALL GET(IVER)
```

CALL INPUT3
CALL PRESET
CALL TRADES
CALL DESRES

GOTO(90,100),KKPP

90
100

CALL 34
IF (NTMP(1) - EQ.3. OR. NTIME(1) - EQ. IDP) GOTO 130
ITCT=0

```
DO 110 M=1,INA
  IF (I_T(M)-0.1) 110,110,115
  IF (I_TERM(M)-IDF-1) 110,120,110
  ITC(I_T(M))=I_T(M)
  CONTINUE
  GO TO 140
```

130
IDP=7
ITCT=-1
ITAB(25)=IDP

```

140      CALL MOD9
      IF (IVER.EQ.KDP) GOTO 150
      IF (IVER.NE.KDP-1) GOTO 850
      CALL REHUN
      CONTINUE
      CALL REPCET
      CALL PERF
      CALL PROSUM
      CALL GET (KDP)
      (PRINTOUT THE TABLE OF ACHIEVED VALUES)
      (PREPARE BUFFER DATA FOR RELCOPIING)
      .....
      KDP=KDP+1
      IF (MR.NE.0) GOTO 859
      GOTO 230
      CALL PRTCMS ('CLSCRM ')
      IQUESE=999
      IQUESO=999
      IQUESP=999
      IQUES=999
      IQUESH=999
      IQUESPP=999
      IQUEPE=999
      IQUHR2=999
      CALL GET (KDP)
      PRINT 860
      FORMAT (' MAIN MENU: ',//2X,'1. CHANGE TEAM NUMBER',//2X,
      *2. THE DP SELECTION QUERY',//2X,
      *3. THE INPUT SELECTION MENU',//2X,
      *4. RETURN THE DATA CALCULATION',//2X,
      *5. DO A SENSITIVITY ANALYSIS',//2X,'6. PRINTCUT CCS
      *7.FACTORS')
      IF (DPCOST.GT.0.) PRINT 225,DFCOST
      PRINT 861
      IQUESE=7
      IF (MR.EQ.1) GOTO 862
      IQUHR2=IQUESE
      IQUESE=IQUHR2+1
      PRINT 863,IQUHR2

```

```

863      FORMAT(1X,I2,'. FILE TEAM PROPOSAL')
862
      CKPLT=0.
      CALL PLOT(CKPLT)
      IF(CKPLT.EQ.0) GOTO 885
      PCTPLG=1.0
      IQUES=IQESES
      IQUESP=IQESES+1
      IQUESPP=IQESES+1
      IQUESPP=IQESES+1
      IQUESO=IQESES+1
      IQUESO=IQESES+1
      PRINT 880, IQUES, IQUESP, IQUESPP, IQUESO, KDP
      FCENAT(1X,I2,'. PRINT LAST SENSITIVITY TABLE.//1X,
      *I2,'. PLOT LAST SENSITIVITY ANALYSIS ACHIEVED VALUES.//1X,
      *I2,'. PLOT LAST SENSITIVITY ANALYSIS ACHIEVED VALUES.//1X,
      *I2,'. PLOT WITH SELECTED Y-AXIS SCALES.//1X,
      *I2,'. USE OPTIMUM EI VALUE FROM LAST SENSITIVITY ANALYSIS FOR DP-',I
      *1,' DATA.')
```

```

880      PRINT 890, IQESES
885      FORMAT(1X,I2,'. EXIT')
890      READ(IN,*,ERR=870,END=870)ANS
      REWIND IN
      IF(MR.NE.2) GOTO 877
      IF(ANS.NE.1) GOTO 877
      IF(ANS.NE.1) GOTO 877
      PRINT 27,I TEAM
      PRINT 12,' FILE EE SENT TO THE PR
      READ(IN,28,ERR=870,END=870)ANI
      FORMAT(A1)
      IF(ANT.NE.YES) GOTO 877
      'SPOOL 'PUN 'CONT 'CL
      '0276P '
      'DATAFILE'ITEM)
      'SPOOL 'PUN 'CLOSE 'NOCONT
      IF(ANS.EQ.IQESES) STOP
```

```

27      CALL SETUP(KDP)
      CALL STORE('CE
      CALL FETCHS('TC
      *IY
      CALL FETCHS('PUN
      CALL FETCHS('CE
      *)
      IF(ANS.EQ.1) MR=1
      CONTINUE
```

```

28      IF(ANS.EQ.1) MR=1
      CONTINUE
```

[illegible]


```

RSHIN=ALUMP(JDP,22)
PRSHAX=ADUMP(JDP,23)
ESHAX=ADUMP(JDP,24)
FESHIN=ADUMP(JDP,25)
FESHIN=ADUMP(JDP,26)
DD(13)=ADUMP(JDP,27)

```

```

99 RETURN

```

```

C *****
C SUPEROUTINE SELECT(JDP)
C *****

```

```

C THE MENU GENERATED PROVIDES BOTH A TABLE TO VERIFY THE CURRENT VALUES
C OF INPUT VARIABLES, AND ENABLES CHANGE OF THE DATA BY EITHER TOTAL
C LIST OF DATA, BY PAGE OF DATA, OR BY INDIVIDUAL VARIABLE. THE DATA
C LINEITS ARE ALSO CHECKED FOR VALIDITY WITHIN DESIGN RANGE: AND, ASSISTS
C UPDATES BY PROMPTING RELATED GROUPS OF DATA AND BY SPECIFYING GANE
C RANGES.

```

```

COMMON REST(48),TAB(20,25,3),CTEST(3),CTOH(3),XMIS(2),ISUBS(48)
COMMON IPAC(48),INT(48),IRC(48),QTY(48),IADD(48),CHAT(48)
COMMON RESS(48),ITEM(48),RESU(48),RESB(48),Y(8),ITYPE(48)
COMMON PERC(48),TRES(48),TDESC(48),DEC(3),YF(48),RESA(48)
COMMON IPHRC(48),RRES(48),CL(48),ALN(48),CI(48),CUN(30)
COMMON TSTAR(48),TMIN(48),TMAX(48),IQ(48),AK(30),ISW(3)
COMMON RESHIN(30),NN(8),NN(8),NN(8),NN(8),NN(8),NN(8),NN(8)
COMMON ITAB(25),NN(8),NN(8),NN(8),NN(8),NN(8),NN(8),NN(8)
COMMON RATIO(8),NN(8),NN(8),NN(8),NN(8),NN(8),NN(8),NN(8)
COMMON PER(6),PVR(6),PVR(6),PVR(6),PVR(6),PVR(6),PVR(6),PVR(6)
COMMON CPUNAX,CFUNIN,X,NTMP(25),COV
COMMON IACT(48,2),IPC(48,5),IPG, KDP, NR
COMMON TDS(48,2),JHIN,ITEAM
COMMON NCICOF,GP,KKPP,ITEAM
COMMON RSHAX,RESHIN,FRSHAX,ESHAX,ESHIN,FESHIN,CLMAX,CDMIN,MIS(9)
COMMON RSHAX,TMIN,FTDMIN,QT(25),YC(6),CTC(8),DD(32),MIS(9)
COMMON STORED/STAT(5),NDUMP(5,5),ADUNE(5,28),DECOST,MAXDE
COMMON CONHCHN/PAGES/ IPG(4)

```

```

REAL*8 STAT
INTEGER AMS,IN/5/,E/'E'//,YES/'Y'//,ANO/'N'//
CALL ERASET(218,1,-1,1,1)

CALL GET(JDP)
CALL FRTCHS(CLRSCRN ')
CALL SETUP(KDP)

PCENHAT(15X,*****TEAM .,12., IS IN DP-',11,X,A8,*****),
PRINT 25,KFP,NDUMP(KDP,3),NDUMP(KDP,4),ADUMP(
* KDF,14),NDUMP(KDP,5),ADUMP(KDP,15),ADUMP(KDP,1),
PRINT 30,ADUMP(KDP,2),ADUMP(KDP,17),
ADUMP(KDP,3),ADUMP(KDP,18),
ADUMP(KDP,4),ADUMP(KDP,19),
ADUMP(KDP,5),ADUMP(KDP,20),
ADUMP(KDP,6),ADUMP(KDP,21),
ADUMP(KDP,7),ADUMP(KDP,22),
PRINT 35,ADUMP(KDP,8),ADUMP(KDP,23),24},
ADUMP(KDP,9),ADUMP(KDP,25},
ADUMP(KDP,10),ADUMP(KDP,26},
ADUMP(KDP,11),ADUMP(KDP,27},
ADUMP(KDP,12),ADUMP(KDP,27}
IF FLAG.EQ.2,RETURN
ITEM****CURRENT VALUE @ DP-',I1,4X,NO*ITEM*****
**CURRENT VALUE: 1. CONTRACTOR GUIDANCE APPROACH 16.
**GUIDANCE QUAL TESTS: 2. F6.2, 3. GUIDANCE CONFIGUR
**YALIGHT: 18. MAXIMUM COST 4. MAINT
**ABILITY ENG MS: 5. F6.2, 19. MINIMUM COST MS: F6.2)
PCENHAT(
1. 20. MAX COST INCENTIVE MS: F6.2, 2. F6.2, 4X,
2. 6. MAX PARALLEL DEVELOP MS: F6.2, 4X,
3. 7. LATEST WEEK MS: F6.2, 4X,
4. 7. HOTTEST RELIABILITY MS: F6.2, 4X,
5. 8. AIRFRAME RELIABILITY MS: F6.2, 4X,
6. 8. AIRFRAME INCENTIVE MS: F6.2, 4X,
7. 9. MAX DELIVER RELIABILITY MS: F6.2, 4X,
8. 9. LAUNCHER RELIABILITY MS: F6.2, 4X,
9. 24. MAXIMUM RELIABILITY MS: F6.2, 4X,

```

10
20
15
14

25

30

[illegible]

READ(IN,*,ERR=133,END=133)ANS
 IF(ANS.EQ.0)RETURN
 IF(ANS.GE.1.OR.ANS.LT.32)GO TO 160
 CALL EXITS

133

RETURN

C.....DATA INPUT PROCESSING.....

160

CONTINUE

IF(KDP.GE.4.AND.(ANS.EQ.1.OR.ANS.EQ.2)) IANS=ANS
 *CALL FRCHS('CLRSCRN')
 IF(KDP.GE.4.AND.(ANS.EQ.1.OR.ANS.EQ.2))
 *PRINT 162
 *FORMAT(' AT DP-4 CONTRACTOR AND GUIDANCE APPROACH ARE FROZEN.'
 *// AND, GUIDANCE CONFIGURATION MUST BE SOLELY EITHER "A" OR "B".
 *// 24 (/))

162

*GOTO 10
 IF(ANS.LT.7.AND.KDP.GT.4)GOTO (160,160,401,500,600,701),IANS
 *GOTO (200,300,400,500,600,700,800,900,1000,1100,1200,
 *1300,1400,1500,1600,1700,1800,1900,2000,2100,2200,2300,2400,2500,
 *2600,2700,2800,2900,3000,3100,413200),IANS
 CONTINUE

150

GOTO 10

200

CALL CHANGI(MC)

IF((MC.EQ.1).CR.(MC.EQ.2)) GO TO 10

210

PRINT 230, MC

220

FORMAT('/',CONTRACTOR='I2,' IS INVALID; PLEASE RETYPE CONTRACTOR

230

*NUMBER: EITHER "1" OR "2".)

GO TO 200

300

CALL CHANGI(ICCFG)

305

IF((ICCFG.EQ.1).OR.(ICCFG.EQ.2)) GOTO 10

310

PRINT 310, ICCFG

FORMAT('/',GUIDANCE APPROACH='I2,' IS INVALID; PLEASE RETYPE GUI

310

*DANCE APPROACH: "1" OR "2".)

GO TO 300

```

400 CALL CHANGI(MIS(6))
401 IF (KDP.EQ.3) GOTO 405
    IF (ADUMP(KDP,5).GT.0.) CTC(5)=0.
    IF (MIS(6).EQ.2.CR.MIS(6).EQ.4) GOTO 405
    PRINT 13
    FORMAT(' AT DP-4 FINAL GUIDANCE CONFIGURATION MUST BE SELECIED.')
    * PLEASE SELECT EITHER 2 OR 4 FOR GUIDANCE CONFIGURATION.'
    GOTO 400
131 IF (MIS(6).GE.1) .AND. (MIS(6).LE.4)) GOTO 701
    PRINT 410, MIS(6)
    FORMAT('/', ' GUIDANCE CONFIGURATION=', I2, ' IS INVALID; RETYPE GUIDA
    * NCE CCNFIGURATION: "1", "2", "3", OR "4".')
    GO TO 400
500 CALL CHANGE(CTC(8))
    XC8 = 1.5
505 IF (CTC(8).GE.XC8) GOTO 10
    PRINT 510
    FORMAT(1H0, 'MAINT.')
515 PRINT 515, CTC(8), XC8
    FORMAT(1H+, ' ENG FUNDS OF $', F5.2, ' MINIMUM.')
    * HAVE BEEN RESET TO THE $', F5.2, ' MINIMUM.')
    CTC(8) = XC8
520 PRINT 525
525 FORMAT(1H+, 'MAINT ENG FUNDS = $', F5.2)
    GO TO 500
600 CALL CHANGE(CTC(7))
    XC7 = 0.5
610 IF (CTC(7).GE.XC7) GOTO 10
    PRINT 610
    FORMAT(1H0, 'VALUE')
    PRINT 515, CTC(7), XC7
    CTC(7) = XC7
620 PRINT 620, CTC(7)
    FORMAT(1H+, 'VALUE ENG FUNDS = $', F5.2)
    GO TO 600
700 CALL CHANGE(CTC(5))

```

```

701 IF (NE.EQ.3) GOTO 704
    IF (CTC(5).EQ.0.) GOTO 10
    PRINT 703
703 PCENAT( PARALLEL DEVELOPMENT FUNDS ARE NOT AVAILABLE AFTER DA-3.
    *// PLEASE DO NOT ATTEMPT TO ADD PARALLEL DEVELOPMENT FUNDS. AND.
    *// DC NOT ATTEMPT TO CHANGE GUIDANCE CONFIGURATION TO PARALLEL CC
    *// CONFIGURATIONS. )
    PRINT 702
702 PCENAT(20( ))
    IF (HIS(6).EQ.1-CH.HIS(6).EQ.3) GOTO 401
    GOTO 10
704 IF (HIS(6).EQ.1) .OR. (HIS(6).EQ.3)) GOTO 750
    IF (CTC(5).LE.0.0) GOTO 10
    PRINT 705
705 PCENAT(1H0, PARALLEL DEV FUNDS')
    PRINT 710 CTC(5)
710 PCENAT(1H+, CP $, P5.2,'M')
    PRINT 720
720 PCENAT(1H+, ARE NOT REQUIRED AND HAVE BEEN RESET TO $0.0')
    CTC(5) = 0.0
    PRINT 725
725 PCENAT(1H+, DO YOU WISH TO CHANGE GUIDANCE CONFIGURATION: 'Y'=YES O
    *E 'N'=NO?)
    READ(IN 730,ERR=704,END=704)ANS
730 PCENAT(1H+,
    IF (ANS.EQ.E) CALL EXITS
    IF (ANS.EQ.ANC) GO TO 10
    GO TO 400
750 IF (CTC(5).GT.0.0) GOTO 10
    PRINT 705
755 PCENAT(1H+, ARE REQUIRED.')
    PRINT 760
760 PCENAT(1H+, DO YOU WISH TO CHANGE GUIDANCE CONFIGURATION: 'Y'=YES C
    *E 'N'=NO?)
    READ(IN 765,ERR=750,END=750)ANS
765 PCENAT(1H+,
    IF (ANS.EQ.E) CALL EXITS

```

```

800.      IF (ANS, EQ, ANC) GO TO 700
          GOTO 400
          CALL CHANGE(YC (1))
          IF (YC (1) .GE. 0 .AND. YC (1) .LE. 100.0) GO TO 10
          PRINT 805, YC (1)
          FORMAT(' RELIABILITY EXCEEDED 100%; RELIABILITY=', F3.1, '.,.', ' PLE
805      *SEE REENTER RELIABILITY.')
          GOTO 800
          CALL CHANGE(YC (2))
          IF (YC (2) .GE. 0 .AND. YC (2) .LE. 100.0) GO TO 10
          PRINT 805, YC (2)
          GOTO 900
          CALL CHANGE(YC (3))
          IF (YC (3) .GE. 0 .AND. YC (3) .LE. 100.0) GO TO 10
          PRINT 805, YC (3)
          GOTO 1000
          CALL CHANGE(YC (4))
          IF (YC (4) .GE. 0 .AND. YC (4) .LE. 1000.0) GO TO 10
          PRINT 1105, YC (4)
          FORMAT(' ERROR EXCEEDED 1000YDS; ERROR=', F3.1, '.,.', ' PLEASE REENT
1105      *ER ERROR.')
          GOTO 1100
          CALL CHANGE(YC (5))
          IF (YC (5) .GE. 0 .AND. YC (5) .LE. 1000.0) GO TO 10
          PRINT 1105, YC (5)
          GOTO 1200
          CALL CHANGE(QT (2))
          IF (QT (2) .GE. 20 .AND. QT (2) .LE. 40.) GO TO 10
          PRINT 1305, QT (2)
          FORMAT('X 40. YCUR QT(2) =', F2.0)
          *LESS THAN 40.
          GO TO 1300
          CALL CHANGE(QT (3))
1400

```

```

1405 IF(QT(3).GE.3.-AND.QT(3).LE.9.)GO TO 10
      PRINT 1405 (A)
      CALL CHANGE(CT(3))
      FORMAT(1X,4I)
      YOUR TESTS MUST BE GREATER THAN 3 AND
      * LESS THAN 9. YOUR TESTS =',P1.0)
      GO TO 1400

C1500
1500 IF(CT(6).GE.2.-AND.QT(6).LE.6.)GO TO 10
      PRINT 1500 (A)
      CALL CHANGE(CT(6))
      YOUR TESTS MUST BE GREATER THAN 2 AND
      * LESS THAN 6. YOUR TESTS =',P1.0)
      GO TO 1500

C1600
1600 IF(QT(4).GE.2.-AND.QT(4).LE.4.)GO TO 10
      PRINT 1600 (A)
      CALL CHANGE(QT(4))
      YOUR TESTS MUST BE GREATER THAN 2
      * AND LESS THAN 4. YOUR TESTS =',P1.0)
      GO TO 1600

C1700
1700 IF(QT(5).GE.3.-AND.QT(5).LE.9.)GO TO 10
      PRINT 1700 (A)
      CALL CHANGE(CT(5))
      YOUR TESTS MUST BE GREATER THAN 3 AND
      * LESS THAN 9. YOUR TESTS =',P1.0)
      GO TO 1700

C1800
1800 IF(QT(1).GE.10.-AND.QT(1).LE.25.)GO TO 10
      PRINT 1800 (A)
      CALL CHANGE(QT(1))
      YOUR TESTS MUST BE GREATER THAN 10 AND LESS THAN 25.
      * YOUR TESTS =',P2.0)
      GO TO 1800

C1900
1900 CALL CHANGE(CD MAX)
      PRINT 1900
      FORMAT(1X,10I)
      YOUR DESIRE TO CHANGE MINIMUM DEVELOPMENT COST, ALSO.
      * READ(1X,250)
      IF(ANS.EQ.5)CALL EXIT
      IF(ANS.EQ.4)GO TO 10

```



```

2098          CALL CHANGE(CDQIN)
2099          IF (CDMAX-GE-CDMIN) GOTO 10
2100          PRINT 1995, CDMAX, CDMIN
2101          FORMAT(1X, 'MAX COST IS LESS THAN MIN COST. MAX COST =', F4.1,
2102          *', MIN COST =', F4.1)
2103          GO TO 1900
2104
2105          CALL CHANGE(FCDEIN)
2106          GOTO 10
2107
2108          CALL CHANGE(TDMAX)
2109          PRINT 2210
2110          FCENAT(' DO YOU DESIRE TO CHANGE MINIMUM FLIGHT TEST COMPLETION, A
2111          *LSC. ')
2112          READ(1, 2520, ERR=2200, END=2200) ANS
2113          IF (ANS-EO-E) CALL EXITS
2114          IF (ANS-EO-AMO) GOTO 10
2115          CALL CHANGE(TDMIN)
2116          IF (TDMAX-GE-TDMIN) GOTO 10
2117          PRINT 2296, TDMAX, TDMIN
2118          FORMAT(1X, 'FLIGHT TEST LATEST WEEK IS LESS THAN EARLIEST WEEK: '//
2119          *', LATEST DATE =', F6.0, '; EARLIEST DATE =', F6.0)
2120          GO TO 2200
2121
2122          CALL CHANGE(FTDMIN)
2123          GOTO 10
2124
2125          CALL CHANGE(RSHAX)
2126          PRINT 2510
2127          FCENAT(' DO YOU DESIRE TO CHANGE MINIMUM RELIABILITY, ALSO. ')
2128          READ(1, 2520, ERR=2500, END=2500) ANS
2129          IF (ANS-EO-E) CALL EXITS
2130          IF (ANS-EO-AMO) GOTO 10
2131          CALL CHANGE(RSHIN)
2132          IF (RSHAX-GE-RSHIN) GOTO 10
2133          PRINT 2505, RSHAX, RSHIN
2134          FORMAT(1X, 'MAXIMUM SYSTEM RELIABILITY IS LESS THAN', F6.0, '/', 5X,
2135          *', MINIMUM RELIABILITY: RSHAX =', F6.0, '; RSHIN =', F6.0)
2136          GO TO 2500

```

```

2700 CAIL CHANGE(FRSHAX)
      GOTO 10
2800 CAIL CHANGE(ESNAX)
      GOTO 10
2900 CAIL CHANGE(ESMIN)
      GOTO 10
3000 CAIL CHANGE(FESHIN)
      GOTO 10
3100 CAIL CHANGE(DD (13))
      GOTO 10
3200 RETURN
      INC
C*****
SUBROUTINE SETUE(QDP)
CCOMMON REST(48),TAB(20,25,3),CTEST(3),CTOH(3),XMIS(2),ISUBS(48)
      IFAC(48),ITERM(48),QTY(48),IDDI(48),CHMT(48)
      IFD(48),RESS(48),TDUE(48),RESB(48),Y(8),RESA(48)
      PERC(48),RESC(48),YF(48),ALN(48),CT(48)
      TSTAR(48),TMIN(48),THIN(48),IO(48),CUN(30),ISW(3)
      RESHIN(30),NN(8),NN(8),ADJ(2,8),PDR(6)
      ITAB(25),AN(9),CD,ICB,INA,INF,IDP
      RATIO(6),PVR(6),FCPHIN,X,NTMP(25),COV
      CPUNAX,CPUMIN,CO(18)
      C10,IACI(48,5),IPC(48,5)
      TDS(48,2),JMIN,IC,IFIG,KDP,MR
      NC,ICOFGG,KP,KKPP,TEAM
      RSHAX,ESHIN,FRSHAX,ESNAX,ESMIN,FESHIN,CDHAX,CDMIN,
      EFCMIN,TDHAX,TCHIN,PTDMIN,QT(25),YC(6),CTC(8),DD(32),MIS(9)

```

```

COMMON/STORED/ STAT(5),NDUMP(5,5),ADUMP(5,28),DPCCS1,MAXDP
REAL*8 STAT
INTEGER DATA
NDUMF(ODP,1)=IYEAR
NDUMF(ODP,2)=IYCE
NDUMF(ODP,3)=IYCE
NDUMF(ODP,4)=IYCE
NDUMF(ODP,5)=IYCE
ADUMF(ODP,1)=IYCE
ADUMF(ODP,2)=IYCE
ADUMF(ODP,3)=IYCE
ADUMF(ODP,4)=IYCE
ADUMF(ODP,5)=IYCE
ADUMF(ODP,6)=IYCE
ADUMF(ODP,7)=IYCE
ADUMF(ODP,8)=IYCE
ADUMF(ODP,9)=IYCE
ADUMF(ODP,10)=IYCE
ADUMF(ODP,11)=IYCE
ADUMF(ODP,12)=IYCE
ADUMF(ODP,13)=IYCE
ADUMF(ODP,14)=IYCE
ADUMF(ODP,15)=IYCE
ADUMF(ODP,16)=IYCE
ADUMF(ODP,17)=IYCE
ADUMF(ODP,18)=IYCE
ADUMF(ODP,19)=IYCE
ADUMF(ODP,20)=IYCE
ADUMF(ODP,21)=IYCE
ADUMF(ODP,22)=IYCE
ADUMF(ODP,23)=IYCE
ADUMF(ODP,24)=IYCE
ADUMF(ODP,25)=IYCE
ADUMF(ODP,26)=IYCE
ADUMF(ODP,27)=IYCE
END
C *****
C SUEROUTINE CHANGE(XXX)
C *****

```



```

22      IF (ANS.NE.4) GOTO 877
      IF (MR.NE.2) STOP
27      PRINT 27, ITEAM
      *OMNG DISK: Y/N ?*)
      FORMAT(1X, 'SHOULD THE REWRITTEN TEAM ', I2, ' FILE BE SENT TO THE PR
28      READ (IN, 28, ERR=22, END=22) ANT
      FORMAT(A1)
      IF (ANT.NE.YES) GOTO 877
      CALL SETUP(KDP)
      CALL STORE
      CALL FRTCHS('CE
      *Y
      CALL FRTCHS('TC
      CALL FRTCHS('PUN
      CALL FRTCHS('CF
      **)
      STOP
      CCCONTINUE
877
485      IF (ITEM.EQ.BLK) GOTC 32
      CONTINUE
      IF (ITEAM.EQ.21) GOTO 31
      IF (MR.EQ.1.AND.ITEAMB.EQ.ITEAM) RETURN
      IF (MR.EQ.2.AND.ITEAMB.EQ.ITEAM) RETURN
      CONTINUE
      IF (ITEAMB.NE.999.AND.ITEAMB.NE.ITEAM) PRINT 24, ITEAM
      PCMAT(' RECORDS DO NOT EXIST FOR TEAM ', I2)
      FORMAT(' WHAT TEAM NUMBER?')
      READ (IN, *, ERR=495, END=495) ITEAM
      CONTINUE
      IF (ITEAMB.LE.0. OR ITEAM.GT.21) GOTO 485
      IF (ITEAM.EQ.1) ITEAM=TEAM01
      IF (ITEAM.EQ.2) ITEAM=TEAM02
      IF (ITEAM.EQ.3) ITEAM=TEAM03
      IF (ITEAM.EQ.4) ITEAM=TEAM04
      IF (ITEAM.EQ.5) ITEAM=TEAM05
      IF (ITEAM.EQ.6) ITEAM=TEAM06
      IF (ITEAM.EQ.7) ITEAM=TEAM07
      IF (ITEAM.EQ.8) ITEAM=TEAM08
      IF (ITEAM.EQ.9) ITEAM=TEAM09
      IF (ITEAM.EQ.10) ITEAM=TEAM10

```

[illegible]


```

C
COMMON REST(48), TAB(20,25,3), CTEST(3), CTOH(3), MIS(2), ISUBS(48)
COMMON IFAC(48), ITIEM(48), QTY(48), IDDI(48), CHAT(48)
COMMON PERC(48), TDUE(48), RESB(48), Y(8), RESA(48)
COMMON IPHC(48), RESC(48), YP(48), CT(48)
COMMON TFIN(48), THIN(48), CL(6), ALN(48), CUM(30), ISW(3)
COMMON RESHIN(30), NN(8), ADJ(2,8), IND(8), PDR(6), IMP, IDP
COMMON ITAB(25), AN(9), PVR(6), CPUMIN, FCPMIN, IMA, C10, INACET, TRADER,
COMMON CPUMAX, RSHIN, RSHMAX, ESHAY, ESMIN, FESMIN, CLMAX, CDMIN,
COMMON DAM1, DAM2, DEPR, SUB, COST, IACT(48,2), ALPHA, PRICE(48,5)
COMMON COSTP2, COSTO, TP, JHIN, TEAM, IC, IFIG, KDP, ME
COMMON TRADER, TDS(48,2), KP, KKPP, ITEAM
COMMON RSHAY, RSHIN, RSHMAX, ESHAY, ESMIN, FESMIN, CLMAX, CDMIN,
COMMON TDMAX, TIMIN, FTDMIN, QT(25), IC(6), CTC(8), DD(32), HIS(9)
COMMON PAGES, TIFG(4)
COMMON ANS, YES, Y, ANO, 'N', 'E', 'B', 'I', 'N', 'S', 'QDP
COMMON VTO(6), ITQ(6)
CALL ERRSET(218,1,-1,1,1)
IF(JFEG(INV).EQ.1) GOTO 15
GOTO 360
11
600
15
25
20
55
700
CALL EXIT
CONTINUE
PRINT 20,KDP
PRINT 1,DP,1,INFO*****'///)
FORMAT(1X,'*****INPUT PAGE 1 PRINT 55
FORMAT(' CONTRACTOR NUMBER, (1 OR 2):')
READ (IN,*,ERR=600,END=600) NC
PRINT (700)
PRINT (' GUIDANCE APPROACH, (1 OR 2):')

```

```

705  FORMAT(' GUIDANCE CONFIGURATION',1-4:NOT 1 OR 3 AFTER DE-3')
      READ (IN,*,ERR=600,END=600) ICOPG
      PRINT 705
710  FORMAT(' MAINTAINABILITY ENGINEERING FUNDS IN $M:')
      READ (IN,*,ERR=600,END=600) C1C(8)
      PRINT 715
715  FORMAT(' VALUE ENGINEERING FUNDS IN $M:')
      READ (IN,*,ERR=600,END=600) C1C(7)
      IF (MIS(6) .EQ. 2) OR MIS(6) .EQ. 4) GO TO 80
720  FORMAT(' PARALLEL GUIDANCE FUNDS IN $M:')
      READ (IN,*,ERR=600,END=600) C1C(5)
80   IF ((NC.EQ.1) .OR. (NC.EQ.2)) GO TO 140
90   PRINT 491
90   PRINT 100, NC
100  FORMAT (//, ' CONTRACTOR= ', I2, ' IS INVALID. ', /, '1' OR '2' .)
      *' PLEASE REENTER CONTRACTOR NUMBER: EITHER '1' OR '2' .)
      READ (IN,*,ERR=130,END=130) NC
130  GO TO 80
130  CALL EXITS
      GO TO 80
140  IF ((ICOPG.EQ.1) .OR. (ICOPG.EQ.2)) GO TO 160
      PRINT 491
145  FORMAT (//, ' GUIDANCE APPROACH= ', I2, ' IS INVALID. ', /,
      *' PLEASE REENTER GUIDANCE APPROACH: '1' OR '2' .)
      READ (IN,*,ERR=150,END=150) ICOPG
      GO TO 140
150  CALL EXITS
      GO TO 140
160  IF ((MIS(6) .GE. 1) .AND. (MIS(6) .LE. 4)) GO TO 180
      PRINT 491
165  PRINT 165, MIS(6)
      *' REENTER GUIDANCE CONFIGURATION= ', I2, ' IS INVALID. ', /
      *' REENTER GUIDANCE CONFIGURATION: '1' OR '2' OR '3' OR '4' .)
      READ (IN,*,ERR=175,END=175) MIS(6)
170

```

```

175 GO TC 160
CALL EXITS
GO TO 160

180 C
YC8 = 1.5
IF (CTC(8).GE.XC8) GOTO 205
PRINT 491
PRINT 185
PCENMAT(1H0,'MAINT')
PRINT 190 CTC(8) XC8
PCENMAT(1H+,'ENG FUNDS OF $',F5.2,' MINIMUM.')
PCENMAT(1H+,'BEEN RESET TO THE $',F5.2,' MINIMUM.')
PCENMAT(1H+,'HAVE BEEN RESET TO THE $',F5.2,' MINIMUM.')
CTC(8) = XC8
CALL CHANGE(CTC(8))
GO TO 180

205 C
XC7 = 0.5
IF (CTC(7).GE.XC7) GOTO 230
PRINT 491
PRINT 210
PCENMAT(1H0,'VALUE')
PRINT 190 CTC(7), XC7
CTC(7) = XC7
PRINT 220 CTC(7)
PCENMAT(1H+,'VALUE ENG FUNDS = $',F5.2)
CALL CHANGE(CTC(7))
GO TO 205

230 C
235 IF (HIS(6).EQ.1) -OR- (HIS(6).EQ.3) GOTO 300
IF (CTC(5).LE.0.0) GOTO 360
PRINT 491
PRINT 240
PCENMAT(1H0,'PARALLEL DEV FUNDS')
PRINT 245 CTC(5)
PCENMAT(1H+,'CP $',F5.2)
PRINT 250
PCENMAT(1H+,'ARE NOT REQUIRED, AND HAVE BEEN RESET TO $0.0')
PCENMAT(1H+,'CTC(5) = 0.0')
PRINT 260
PCENMAT(1H+,'DO YOU WISH TO CHANGE GUIDANCE CONFIGURATION: "Y"=YES G

```

```

54      *R "N"=NO?')
      READ(IN,54,ERR=255,END=255)ANS
      FORMAT(A1)
      IF(ANS.EQ.E)CALL EXITS
      IF(ANS.EQ.AMC)GO TO 360
267     PRINT 270
270     FORMAT(1X,'PLEASE RETYPE GUIDANCE CONFIGURATION: "1", "2", "3" OR "4'
      *R ' ')
      CALL CHANGE(MIS(6))
      GOTO 160

300     IF (CTC(5) .GT. 0.0) GOTO 360
310     PRINT 491
320     PRINT 320
335     PRINT 335
      FORMAT(1X, ' ARE REQUIRED. ')
      *R "N"=NO?')
      READ(IN,54,ERR=300,END=300)ANS
      IF(ANS.EQ.E)CALL EXITS
      IF(ANS.EQ.AMC)GOTO 280
      CALL CHANGI(MIS(6))
      GOTO 160
280     PRINT 290
290     FORMAT(1X, 'INPUT PARALLEL DEVELOPMENT FUNDS: EX: $5,000,000- 5 ')
      READ(IN,*,ERR=300,END=300)CTC(5)
      GO TO 160

360     CALL FRTCHS('CLESCHN ')
      DO 361 INV=1,4
      IF(IPG(INV).EQ.2) GOTO 362
      CCNTINUE
      GOTO 445
610
362     CONTINUE
      PRINT 365,KDP
365     FORMAT(1X, '*****INPUT PAGE 2 DP-',I1,' INFO.*****')
      PRINT 725
725     FORMAT(' MOTOR RELIABILITY. ')

```



```

390 PRINT 390,QT(3)
    FORMAT(1X,VAIRFRAME QUALIFICATION TESTS MUST BE GREATER THAN 3 AND
    * LESS THAN 9,/,YOUR QUAL TESTS=,P2.0)
    CALL CHANGE(QT(3))

400 GO TO 385
    IF(QT(6))GE.2..AND.QT(6).LE.6.)GO TO 410
    PRINT 400
    FORMAT(1X,VAIRFRAME QUALIFICATION TESTS MUST BE GREATER THAN 2 AND
    * LESS THAN 6,/,YOUR QUAL TESTS=,P2.0)
    CALL CHANGE(QT(6))

410 GO TO 400
    IF(QT(4))GE.2..AND.QT(4).LE.4.)GO TO 420
    PRINT 410
    FORMAT(1X,VFIRE CONTROL QUALIFICATION TESTS MUST BE GREATER THAN 2
    * AND LESS THAN 4,/,YOUR QUAL TESTS=,P2.0)
    CALL CHANGE(QT(4))

420 GO TO 410
    IF(QT(5))GE.3..AND.QT(5).LE.9.)GO TO 430
    PRINT 420
    FORMAT(1X,VGUIDANCE QUALIFICATION TESTS MUST BE GREATER THAN 3 AND
    * LESS THAN 9,/,YOUR QUAL TESTS=,P2.0)
    CALL CHANGE(QT(5))

430 GO TO 420
    IF(QT(1))GE.10..AND.QT(1).LE.25.)GO TO 440
    PRINT 430
    FORMAT(1X,FLIGHT TESTS MUST BE GREATER THAN 10 AND LESS THAN 25,/,
    * YOUR QUAL TESTS=,P2.0)
    CALL CHANGE(QT(1))

440 GO TO 430
    CCNTINUE
    CALL FRTCHS('CIESCRN ')
    DC 446 INV=14
    IF(IPG(INV).EQ.3) GOTO 447
    CCNTINUE
    GO TO 440
    CALL EXITS
620

```

```

497      CALL CHANGE(CDMAX)
      PRINT 497
      PCENAT(1X,'FOR MINIMUM COST,')
      CALL CHANGE(CDMIN)
      GO TO 480
490      GE.TDMIN)GOTC 500
      IF(TDMAX:
      PRINT 491
      PCENAT(1X,'*****WARNING*****')
      PRINT 496
      TDMAX,TDMIN
      PCENAT(1X,'PLIGHT TEST LATEST WEEK IS LESS THAN EARLIEST WEEK; LAT
      *EST=',P6.0:
      EARLIEST=',P6.0)
      CALL CHANGE(TDMAX)
498      PRINT 498
      PCENAT(1X,'FOR MINIMUM TIME,')
      CALL CHANGE(TDMIN)
      GO TO 490
500      GE.RSHIN)GOTC 510
      IF(RSHMAX:
      PRINT 491
      PCENAT(1X,'*****WARNING*****')
      PRINT 505
      RSHMAX,RSHIN
      PCENAT(1X,'MAXIMUM SYSTEM RELIABILITY IS LESS THAN MINIMUM RELIABI
      *LITY: MAX=',P6.0:
      MIN=',P6.0)
      PRINT 506
      PCENAT(1X,'FOR MAXIMUM RELIABILITY,')
      CALL CHANGE(RSHAX)
507      PRINT 507
      PCENAT(1X,'FOR MINIMUM RELIABILITY,')
      CALL CHANGE(RSMIN)
      GO TO 500
510      CALL FRTCHS('CIRSCRN ')
      DO 511 INV=1,4
      IF(IPG(INV).EQ.4) GOTO 512
      CCNTINUE
      RETURN
630      CALL EXIT
512      CCNTINUE
      PRINT 520
      KDP
520      PCENAT(1X,'*****INUT PAGE 4 DP-',I1,' INFO.*****')
      PRINT 860
      PCENAT(1X,'MAXIMUM IMPACT ERROR EXPECTED.')
```

```

447 CONTINUE
450 PRINT 450, KDP
450 FORMAT (11,'***INPUT PAGE 3 DP-',I1,' INFO.*****'//)
800 FORMAT(' MAXIMUM COST IN $M.')
805 READ(IN,*,ERR=620,END=620)CDMAX
806 PRINT 805
810 FORMAT(' MINIMUM COST IN $M.')
810 READ(IN,*,ERR=620,END=620)CDMIN
815 PRINT 810
815 FORMAT(' INCENTIVE FEE % FOR DEVELOPMENT COST.')
815 READ(IN,*,ERR=620,END=620)PCDINH
815 IF(PCDINH.LT.0..OR.PCDINH.GT.15.)GOTO 806
815 PRINT 815
815 FORMAT(' LATEST WEEK FOR FLIGHT TEST COMPLETION.')
815 READ(IN,*,ERR=620,END=620)TDMAX
815 PRINT 815
815 FORMAT(' EARLIEST WEEK FOR FLIGHT TEST COMPLETION.')
815 READ(IN,*,ERR=620,END=620)TDMIN
815 IF(TT=15.-PCDINH)
815 PRINT 815
815 FORMAT(' INCENTIVE FEE % FOR FLIGHT TEST COMPLETION')
815 READ(IN,*,ERR=620,END=620)FTDINH
815 IF(FTDINH.LT.0..OR.FTDINH.GT.PERTOT)GOTO 821
815 PRINT 815
815 FORMAT(' MAXIMUM RELIABILITY % EXPECTED.')
815 READ(IN,*,ERR=620,END=620)RSHAX
815 PRINT 815
815 FORMAT(' MINIMUM RELIABILITY % EXPECTED.')
815 READ(IN,*,ERR=620,END=620)RSMIN
815 IF(TT=15.-PCDINH-FTDINH)
815 PRINT 815
815 FORMAT(' INCENTIVE FEE % FOR RELIABILITY.')
815 READ(IN,*,ERR=620,END=620)FRSHAX
815 IF(FRSHAX.LT.0..OR.FRSHAX.GT.PERTOT)GOTO 836
815 PRINT 815
815 FORMAT(' GE.CDINH)GOTO 490
815 PRINT 815
815 FORMAT(' CDMAX,CDMIN
815 PRINT 815
815 FORMAT(' MAXIMUM COST IS LESS THAN MINIMUM COST: MAX =',F4.1,
815 *; MIN

```



```

8 PER(6) PVR(6) CD INP IDP
8 COMCN CPUMAX, CPUMIN, FCPHIN, X, MTEMP(25), COV, C10, TRADET, TRADER,
8 STRADEA, DAH1, DAH2, DEF, SUR, COSTA, COSTD, COSTP1
8 CCCHCN COSTP2, COSTOF, IFCT, ALPHA, PRIOR
8 , TRADER, TP, JMIN, IACT(48,2), IPC(48,5)
8 , TDS(48,2) KP, KKPP, ITEAM
8 COMCN RSMAX, RSMIN, FRSMAX, ESMAX, ESMIN, FESMIN, CDMAX, CDMIN,
8 FCDMIN, TDHAX, TTHIN, OT(25), YC(6), CTC(8), DD(32), MIS(9)
8 COMMON/STORED/ STAT(5), NDUMP(5,5), ADUMP(5,28), DPCOSI, MAXDP
REAL*8 STAT
REAL*8 FINAL, INTEN, PROPOSED,
INTEGER E, C, O, IN/5, ANO, N, ANS, YES, Y, /
DIMENSION HENO(48,2)
- - - - -
JDF=KLP-1
IF(JDP.LT.3) JDF=3
NSTF=17
NTF=5
- - - - - >= DP-4 EDIT' - - - -
C
C
C
930 IF(CTC(8).GE.ADUMP(JDP,1)) GOTO 940
CTC(8)=ADUMP(JDP,1)
PRINT 935, JDP, ADUMP(JDP,1), KDP
935 PCENAT(1), MAINTENANCE ENG FUNDS ARE LESS THAN THOSE AT DP-, I1,
8, 1X, AND HAVE BEEN RESET TO THE DP-, I1, FIGURE: $, F5.2, I1,
*/"N"?
READ(IN, 2896, ERR=2888, END=2888) ANS
FORMAT(A1)
IF(ANS.EC.E) CALL EXITS
IF(ANS.EC.ANO) GO TO 940
IF(ANS.EC.YES) GO TO 2889
CALL EXITS
GO TO 930
2888 CALL CHANGE(CTC(8))
2889 ADUMF(KDE,1)=CTC(8)
GO TO 930
C

```

[illegible]

```

      NC ICOFG KP KKPP ITEAM
      CCHON RSHAX, RSHIN, FESHAX, ESHAX, ESMIN, FESMIN, CDHAX, CDHIN,
      SCDMIN, TDHAX, TDHIN, PTDHIN, QT(25), YC(6), CTC(8), DD(32), MIS(9)
      DIMENSION HEHO(48,2)
      NSTP = 17
      NIF = 5
1002 IC=NC
      IF (NTEMP(1) .EQ.3) GOTO 1919
      IF (ICP .NE. 0) GOTO 1920
C
1919 NTEMP(2) = 3
      IDP = 3
      GOTO (21, 24), NC
21 GOTO (22, 23), ICOFG
22 NSTP = 17
23 GOTO 27
24 NSTP = 18
25 GOTO 27
26 GOTO (25, 26), ICOFG
27 NSTP = 19
      GOTO 27
28 NSTP = 20
      CCNTINUE
      DO 20 K=1, 13
      READ(NSTP, 100) L, (TAB(K, I), I=1, 25) 1X, 9F6.2 / 1X, 9F6.2)
100 FORMAT(5X, I2, 4X, 7F6.2)
20 CONTINUE
      READ(NSTP, 102) (TDS(K, 1), K=1, 13)
102 FORMAT(9X, 13F4.0)
103 READ(NSTP, 103) (ISUBS(I), I=1, 48)
104 READ(NSTP, 105) (IFAC(I), I=1, 48)
105 READ(NSTP, 105) (NT(I), I=1, 48)
106 READ(NSTP, 103) (XP(I), I=1, 48)
107 READ(NSTP, 108) (YF(I), I=1, 48)
108 FORMAT(9X, 9F5.0 / 1X, 13F5.0 / 1X, 13F5.0)
      READ(NSTP, 103) (IRC(I), I=1, 48)
      READ(NSTP, 103) (IDDT(I), I=1, 48)
      READ(NSTP, 105) (IQ(I), I=1, 48)

```

```

110 READ(NSTTP,195) (ITER(I), I=1,48)
    FCIMAT(1,48)
111 READ(NSTTP,108) (QTY(I), I=1,48)
    READ(NSTTP,111) (CMAT(I), I=1,48)
    FORMAT(1,8F6.0 / 1X, 10F6.0 /
    & 1X, 10F6.0 / 1X, 10F6.0)
    READ(NSTTP,108) (RESS(I), I=1,48)
    READ(NSTTP,108) (RESU(I), I=1,48)
    READ(NSTTP,108) (RESB(I), I=1,48)
    DO 500 J=1,5
113 READ(NSTTP,113) (IPC(I,J), I=1,48)
    FORMAT(1,4I3 / 1X, 17I3)
500 CONTINUE
115 READ(NSTTP,103) (ITYPE(I), I=1,48)
    READ(NSTTP,115) (PERC(I), I=1,48)
    FORMAT(1,8F6.4 / 1X, 10F6.4 / 1X, 10F6.4 /
    & 1X, 10F6.4 / 1X, 10F6.4)
    READ(NSTTP,108) (TDUE(I), I=1,48)
117 READ(NSTTP,117) (ALN(I), I=1,48)
    FORMAT(1,14F3.0 / 1X, 17F3.0)
120 READ(NSTTP,165) (IPRC(I), I=1,48)
    READ(NSTTP,120) (CUN(I), I=1,29)
    FORMAT(1,9F5.0 / 1X, 10F5.0 / 1X, 10F5.0)
    READ(NSTTP,120) (RESMAX(I), I=1,29)
    DO 9099 I=1,48
    IF (IFAC(I) .GT. 30) IFAC(I) = 30
9099 CCNTINUE = .0
    CUN(30) = 1.
    RESMIN(30) = 1.
    RESMAX(30) = 1.

```

... CALCULATE COL 1 FROM COL 2 OF TABLES . . .

```

TAE(1,1,1) = 300
TAE(2,1,1) = 400
TAE(3,1,1) = 650
TAE(4,1,1) = 114
TAE(5,1,1) = 080

```

```

TAE(6,1,1) = 090
TAE(7,1,1) = 120
TAE(8,1,1) = 310
TAE(9,1,1) = 110
TAE(10,1,1) = 118
TAE(11,1,1) = 128
TAE(12,1,1) = 150
TAE(13,1,1) = 10
DO 30 K=1,2,3
DO 30 K=1,5,6 =TAB(K,I-1,1) + 10
DO 30 K=1,5,6 =TAB(K,I-1,1) + 1
TDS(1,2,1) = 2
TAE(12,1,1) + 10
TAE(13,1,1) + 2
TAE(14,1,1) + 1
DO 35 K=1,8
TAE(K,I,1) = TAB(K,I-1,1) + 20
DO 41 K=9,11
TAE(K,I,1) = TAB(K,I-1,1) + 2
39 CONTINUE
IDS(1,2) = 2.
INA=48
INF=30
DO 3 I=1,20
DO 4 K=1,25
TAE(I,K,1) = TAB(I,K,1) * 10.0 ** (TDS(I,1) - 5.0)
TAE(I,K,2) = TAB(I,K,2) * 10.0 ** (TDS(I,2) - 5.0)
CCCTI,INUE=0.0
TDS(I,1)=0.0
IDS(I,2)=0.0
3 CCCTI,INUE
IF(NC.NE.-2)GOTO 1920
----- APEX PERFORMANCE BUMP! ----
H=100.0
F=0.05

```

```

C
TAE(14,25,3)=F
DQ 8001 I=1,25
I1=TAB(1,I,1)
I2=TAB(2,I,1)
I3=TAB(3,I,2)
I4=TAB(4,I,2)
I5=TAB(5,I,2)
I6=TAB(6,I,2)
I7=TAB(7,I,2)
I8=TAB(8,I,2)
I9=TAB(9,I,2)
I10=TAB(10,I,2)
I11=TAB(11,I,2)
I12=TAB(12,I,2)
I13=TAB(13,I,2)
I14=TAB(14,I,2)
I15=TAB(15,I,2)
I16=TAB(16,I,2)
I17=TAB(17,I,2)
I18=TAB(18,I,2)
I19=TAB(19,I,2)
I20=TAB(20,I,2)
I21=TAB(21,I,2)
I22=TAB(22,I,2)
I23=TAB(23,I,2)
I24=TAB(24,I,2)
I25=TAB(25,I,2)
I26=TAB(26,I,2)
I27=TAB(27,I,2)
I28=TAB(28,I,2)
I29=TAB(29,I,2)
I30=TAB(30,I,2)
I31=TAB(31,I,2)
I32=TAB(32,I,2)
I33=TAB(33,I,2)
I34=TAB(34,I,2)
I35=TAB(35,I,2)
I36=TAB(36,I,2)
I37=TAB(37,I,2)
I38=TAB(38,I,2)
I39=TAB(39,I,2)
I40=TAB(40,I,2)
I41=TAB(41,I,2)
I42=TAB(42,I,2)
I43=TAB(43,I,2)
I44=TAB(44,I,2)
I45=TAB(45,I,2)
I46=TAB(46,I,2)
I47=TAB(47,I,2)
I48=TAB(48,I,2)
I49=TAB(49,I,2)
I50=TAB(50,I,2)
I51=TAB(51,I,2)
I52=TAB(52,I,2)
I53=TAB(53,I,2)
I54=TAB(54,I,2)
I55=TAB(55,I,2)
I56=TAB(56,I,2)
I57=TAB(57,I,2)
I58=TAB(58,I,2)
I59=TAB(59,I,2)
I60=TAB(60,I,2)
I61=TAB(61,I,2)
I62=TAB(62,I,2)
I63=TAB(63,I,2)
I64=TAB(64,I,2)
I65=TAB(65,I,2)
I66=TAB(66,I,2)
I67=TAB(67,I,2)
I68=TAB(68,I,2)
I69=TAB(69,I,2)
I70=TAB(70,I,2)
I71=TAB(71,I,2)
I72=TAB(72,I,2)
I73=TAB(73,I,2)
I74=TAB(74,I,2)
I75=TAB(75,I,2)
I76=TAB(76,I,2)
I77=TAB(77,I,2)
I78=TAB(78,I,2)
I79=TAB(79,I,2)
I80=TAB(80,I,2)
I81=TAB(81,I,2)
I82=TAB(82,I,2)
I83=TAB(83,I,2)
I84=TAB(84,I,2)
I85=TAB(85,I,2)
I86=TAB(86,I,2)
I87=TAB(87,I,2)
I88=TAB(88,I,2)
I89=TAB(89,I,2)
I90=TAB(90,I,2)
I91=TAB(91,I,2)
I92=TAB(92,I,2)
I93=TAB(93,I,2)
I94=TAB(94,I,2)
I95=TAB(95,I,2)
I96=TAB(96,I,2)
I97=TAB(97,I,2)
I98=TAB(98,I,2)
I99=TAB(99,I,2)
I100=TAB(100,I,2)
I101=TAB(101,I,2)
I102=TAB(102,I,2)
I103=TAB(103,I,2)
I104=TAB(104,I,2)
I105=TAB(105,I,2)
I106=TAB(106,I,2)
I107=TAB(107,I,2)
I108=TAB(108,I,2)
I109=TAB(109,I,2)
I110=TAB(110,I,2)
I111=TAB(111,I,2)
I112=TAB(112,I,2)
I113=TAB(113,I,2)
I114=TAB(114,I,2)
I115=TAB(115,I,2)
I116=TAB(116,I,2)
I117=TAB(117,I,2)
I118=TAB(118,I,2)
I119=TAB(119,I,2)
I120=TAB(120,I,2)
I121=TAB(121,I,2)
I122=TAB(122,I,2)
I123=TAB(123,I,2)
I124=TAB(124,I,2)
I125=TAB(125,I,2)
I126=TAB(126,I,2)
I127=TAB(127,I,2)
I128=TAB(128,I,2)
I129=TAB(129,I,2)
I130=TAB(130,I,2)
I131=TAB(131,I,2)
I132=TAB(132,I,2)
I133=TAB(133,I,2)
I134=TAB(134,I,2)
I135=TAB(135,I,2)
I136=TAB(136,I,2)
I137=TAB(137,I,2)
I138=TAB(138,I,2)
I139=TAB(139,I,2)
I140=TAB(140,I,2)
I141=TAB(141,I,2)
I142=TAB(142,I,2)
I143=TAB(143,I,2)
I144=TAB(144,I,2)
I145=TAB(145,I,2)
I146=TAB(146,I,2)
I147=TAB(147,I,2)
I148=TAB(148,I,2)
I149=TAB(149,I,2)
I150=TAB(150,I,2)
I151=TAB(151,I,2)
I152=TAB(152,I,2)
I153=TAB(153,I,2)
I154=TAB(154,I,2)
I155=TAB(155,I,2)
I156=TAB(156,I,2)
I157=TAB(157,I,2)
I158=TAB(158,I,2)
I159=TAB(159,I,2)
I160=TAB(160,I,2)
I161=TAB(161,I,2)
I162=TAB(162,I,2)
I163=TAB(163,I,2)
I164=TAB(164,I,2)
I165=TAB(165,I,2)
I166=TAB(166,I,2)
I167=TAB(167,I,2)
I168=TAB(168,I,2)
I169=TAB(169,I,2)
I170=TAB(170,I,2)
I171=TAB(171,I,2)
I172=TAB(172,I,2)
I173=TAB(173,I,2)
I174=TAB(174,I,2)
I175=TAB(175,I,2)
I176=TAB(176,I,2)
I177=TAB(177,I,2)
I178=TAB(178,I,2)
I179=TAB(179,I,2)
I180=TAB(180,I,2)
I181=TAB(181,I,2)
I182=TAB(182,I,2)
I183=TAB(183,I,2)
I184=TAB(184,I,2)
I185=TAB(185,I,2)
I186=TAB(186,I,2)
I187=TAB(187,I,2)
I188=TAB(188,I,2)
I189=TAB(189,I,2)
I190=TAB(190,I,2)
I191=TAB(191,I,2)
I192=TAB(192,I,2)
I193=TAB(193,I,2)
I194=TAB(194,I,2)
I195=TAB(195,I,2)
I196=TAB(196,I,2)
I197=TAB(197,I,2)
I198=TAB(198,I,2)
I199=TAB(199,I,2)
I200=TAB(200,I,2)
I201=TAB(201,I,2)
I202=TAB(202,I,2)
I203=TAB(203,I,2)
I204=TAB(204,I,2)
I205=TAB(205,I,2)
I206=TAB(206,I,2)
I207=TAB(207,I,2)
I208=TAB(208,I,2)
I209=TAB(209,I,2)
I210=TAB(210,I,2)
I211=TAB(211,I,2)
I212=TAB(212,I,2)
I213=TAB(213,I,2)
I214=TAB(214,I,2)
I215=TAB(215,I,2)
I216=TAB(216,I,2)
I217=TAB(217,I,2)
I218=TAB(218,I,2)
I219=TAB(219,I,2)
I220=TAB(220,I,2)
I221=TAB(221,I,2)
I222=TAB(222,I,2)
I223=TAB(223,I,2)
I224=TAB(224,I,2)
I225=TAB(225,I,2)
I226=TAB(226,I,2)
I227=TAB(227,I,2)
I228=TAB(228,I,2)
I229=TAB(229,I,2)
I230=TAB(230,I,2)
I231=TAB(231,I,2)
I232=TAB(232,I,2)
I233=TAB(233,I,2)
I234=TAB(234,I,2)
I235=TAB(235,I,2)
I236=TAB(236,I,2)
I237=TAB(237,I,2)
I238=TAB(238,I,2)
I239=TAB(239,I,2)
I240=TAB(240,I,2)
I241=TAB(241,I,2)
I242=TAB(242,I,2)
I243=TAB(243,I,2)
I244=TAB(244,I,2)
I245=TAB(245,I,2)
I246=TAB(246,I,2)
I247=TAB(247,I,2)
I248=TAB(248,I,2)
I249=TAB(249,I,2)
I250=TAB(250,I,2)
I251=TAB(251,I,2)
I252=TAB(252,I,2)
I253=TAB(253,I,2)
I254=TAB(254,I,2)
I255=TAB(255,I,2)
I256=TAB(256,I,2)
I257=TAB(257,I,2)
I258=TAB(258,I,2)
I259=TAB(259,I,2)
I260=TAB(260,I,2)
I261=TAB(261,I,2)
I262=TAB(262,I,2)
I263=TAB(263,I,2)
I264=TAB(264,I,2)
I265=TAB(265,I,2)
I266=TAB(266,I,2)
I267=TAB(267,I,2)
I268=TAB(268,I,2)
I269=TAB(269,I,2)
I270=TAB(270,I,2)
I271=TAB(271,I,2)
I272=TAB(272,I,2)
I273=TAB(273,I,2)
I274=TAB(274,I,2)
I275=TAB(275,I,2)
I276=TAB(276,I,2)
I277=TAB(277,I,2)
I278=TAB(278,I,2)
I279=TAB(279,I,2)
I280=TAB(280,I,2)
I281=TAB(281,I,2)
I282=TAB(282,I,2)
I283=TAB(283,I,2)
I284=TAB(284,I,2)
I285=TAB(285,I,2)
I286=T
```

```

CTC(8)=CTC(8)*1688888.
CTC(7)=CTC(7)*1688888.
CTC(5)=CTC(5)*1600000.
IF (MIS(6)-1) 11,12,11

```

```

12 CTC(6)=CTC(5)
CTC(5)=0.0
11 CONTINUE

```

C

```

COV=12000.
IF (WTEMP(2)-EG-4) GO TO 9999

```

C

```

DO 3200 I=1,INA
HEM=IPRC(I)
IPRC(I)=0
IF (HEM-48) 3070,3070,3060
3051 IF (HEM-48) 3070,3070,3060
3060 IF (HEM-48) 3070,3070,3060
3070 HEM=I
3080 HEMO(HEM,1)=I
ALN(I)=ALN(I)/100.0
TDUE(I)=TDUE(I)/10.0
CHAT(I)=CHAT(I)*1000.0
3200 CCNTINUE

```

C

```

DO 800 3290 M=1,INA
IF (IYPE(M)-5) 3230,3220,3230
3220 RESB(M)=RESB(M)/1.5
GO TO 3231
3230 RESB(M)=RESB(M)

```

C

```

3231 DO 3280 I=1,5
3240 IF (IPC(M,I)) 3280,3280,3250
3250 IF (IPC(M,I)-100) 3270,3270,3260
3260 IPC(M,I)=IPC(M,I)-100
GO TO 3240
3270 KJ=IPC(M,I)
IPC(M,I)=HEMO(KJ,1)
3280 CONTINUE
3290 CCNTINUE

```

C

```

COV=12000.0

```



```

80 CONTINUE
9999 CCONTINUE
      RETURN
      ENCL
C *****
C SUEROUTINE ZRW
C *****
C THIS SUBROUTINE ZEROIZES THE ENTIRE COMMON.
      COMMON IRR(309)
      DO 4 I=1,3506
      4 IRR(I)=0
      RETURN
      ENCL
C *****
C SUEROUTINE PRESET
C *****
C
      COMMON REST(48), TAB(20,25,3), CTEST(3), CTOH(3), XMIS(2), ISUBS(48)
      IFAC(48), INT(48), IRC(49), IDDT(48), CHAT(48)
      IFD(48), ITEM(48), RESB(48), Y(8), RESA(48)
      CCOMMON PERC(48), TDOE(48), DEC(3), YF(48), ALN(48), CT(48)
      PERC(48), RESC(48), CL(6,5), TMIN(48), RESHA(30), AK(30), CUN(30), ISW(3)
      IPERC(48), IPSTAN(48), TPER(48), RESHIN(50), NN(8), NN(8), INDY(8), PDR(6), INF, IDP
      TPER(48), RESHIN(50), NN(8), NN(8), AN(9), ADJ(2,8), CD, INA, INF, IDP, TRADET, TRADER,
      CPUNH, PCBHIN, Y, NTEMP(25), COV, C10, TRADET, TRADER,
      TRALEA, DAN1, DAN2, DEP, SUR, COSTA, COSTD, COSTP1
      COMMON COSTP2, TP, JMIN, WA, IACT(48,2), ALPHA, PRIOR
      TRADER(2), TD(48,2), KP, KNPP, TEAN, IC, IPFG, KDP, HE
      TD(48,2), KP, KNPP, TEAN, IC, IPFG, KDP, HE
      NC, IC, OFG, KP, KNPP, TEAN
      CCOMMON RSHAY, RSHIN, PRSMA, ESMAY, ESMIN, FESMIN, CDMAX, CDMIN,
      ECDHIN, TDHAY, TDHIN, FTDHIN, QT(25), YC(6), CTC(8), DD(32), MIS(9)
      ITAB(25) = IDP
      TOTAL FLIGHT TEST FAB. COSTS
      DO 980 M=1, INA
      980 M=1, INA
      I=IPD(M) - 160

```

```

901 IF (I, 950, 950, 901
902 J=IFAC(H) 902, 902, 950
CL(I, 1)=CL(I, 1)+CUN(J)
CL(I, 2)=CL(I, 2)+CHAT(H)
CL(I, 3)=CL(I, 3)+RESB(H)
CL(I, 5)=ALN(H)
CCNTINUE
950 IF (IRC(H), 980, 980, 949
949 IF (IRC(H)-6) 5, 1, 951, 980
951 IF (ISUBS(H)-8) 5, 2, 980, 980
952 IF (ITYPE(H)-1) 980, 953, 980
953 I=IRC(H)
CL(I, 4)=CL(I, 4)+RESB(H)
980 CONTINUE
C 400 QT(11)=200.0
QT(12)=140.0
402 QT(7)=(QT(11)/100.)*5.
QT(9)=(QT(12)/100.)*5.
C 404 DO 109 M=1, INA
I=IQ(H)
IF (QTY(H), 105, 109, 500
500 IF (I-1) 109, 103, 101, 500
101 IF (I-12) 103, 103, 109
103 X=QT(I)/QTY(H)
IF (X) 97, 97, 105
105 IF (PERC(H)-1.0) 107, 106, 106
106 IF (I-7) 97, 107, 107
97 X=1.0
107 TAE(1, I, 3)=X
QTY(H)=QTY(H)*X
109 CCNTINUE
C 110 DO 119 M=1, INA
I=IQ(H)-106
IF (I-1) 119, 113, 111
111 IF (I-12) 113, 113, 119
113 QTY(H)=QTY(H)*TAB(1, I, 3)

```

```

I=QTY(M)+0.1
CTY(M)=I
119 CONTINUE
C
120 IF (IDP-4) 120,120,250
121 IF (IED(M)-1) 129,121,122
122 IF (MIS(6)-4) 124,123,129
123 QTY(M)=0.0
124 GO TO 129
125 QTY(M)=1.0
126 GO TO 129
127 IF (IED(M)-2) 129,125,129
128 IF (MIS(6)-2) 124,123,124
129 CONTINUE
C
130 SET END DATES
131 DO 269 M=1,INA
132 IF (PERC(M)-1.0) 257,269,269
133 TDUE(M)=900.0
134 IF (IC(M)-1) 258,260,258
135 IF (IC(M)-1) 269,252,269
136 JJ=IFAC(M)
137 X=(RESU(M)-1.0)**(1.0-ALN(M))/RESIN(JJ)
138 TDUE(M)=DD(I)-X-1.0
139 IF (IDMIN) 256,256,255
140 TDUE(M)=TDMIN-1.0
141 IF (TDUE(M)) 251,251,269
142 TDUE(M)=200.0
143 GO TO 269
C
144 DO 268 I=1,10
145 A=I
146 J=I+3
147 JJ=IFAC(M)
148 IF (DD(J)) 261,261,262
149 AJJ=RESU(M)*20.0/RESIN(JJ)
150 DD(J)=DD(I)-((10.0-A)*AJJ)
151 TAE(I,1,3)=DD(J)-(RESU(M)* (20.0 * A - 1.0) **
152 6 ALN(M)) / RESIN(JJ)
153 CONTINUE
268

```

```

C 269 CCNTINUE
C
280 I=QT(11)/20.0+.1
281 I=10
290 DO 299 J=1,I
291 IF (TAB(1,J,3)-TAB(1,1,3)) 291,299,299
299 CONTINUE
C
282 I=3
310 K=I+1
DO 319 J=K,10
IF (TAB(1,J,3)-TAB(1,K,3)) 311,319,319
311 TAB(1,K,3)=TAB(1,J,3)
319 CCNTINUE
C
330 DO 339 M=1,INA
IF (IO(M)-1) 332,331,332
331 TDUE(M)=TAB(1,1,3)
332 IF (IO(M)-12) 339,333,339
333 IF (IDP-4) 334,335,335
334 TAB(1,K,3)=TAB(1,K,3)+(RESU(M)*60.)/RESMIN(JJ)
GO TO 337
335 IF (QT(12)) 339,339,336
336 TAB(1,K,3)=TAB(1,K,3)+(RESU(M)*QT(11))/RESMIN(JJ)
337 TDUE(M)=TAB(1,K,3)
339 CCNTINUE
C
IF (ILE.EQ.4) GOTC 380
371 ALPHA=1.0
380 DO 600 M=1,INA
I=IFD(M)
IF (I-22) 600,605,604
604 IF (I-25) 605,605,600
605 IF (QT(I)) 606,606,610
610 I=IFAC(M)
RESU(M)=QT(I)/CUN(I)
600 CCNTINUE
C

```

```

CCCONTINUE
END
C *****
C SUBROUTINE TRADE
C *****
COMMON REST(48), TAB(20,25,3), CTEST(3), CTOH(3), XNIS(2), ISUBS(48)
COMMON IPAC(48), ITIRH(48), IRC(48), IDDT(48), CHAT(48)
COMMON RESU(48), RESB(48), Y(8), RESA(48)
COMMON PERC(48), TDUE(48), DEC(3), YF(48), CT(48)
COMMON IPER(48), RESC(48), ALN(48), IQ(48), CUN(30), ISW(3)
COMMON TSTAR(48), TTHIN(48), THIN(48), AK(30), IND(8)
COMMON RESHIN(30), NN(8), ADJ(2,8), PDR(6), INF, IDP
COMMON ITAB(25,3), AN(9), CD, NTEMP(25), COV, C10, TRADET, TRADER,
COMMON PVR(6), CPUMIN, FCPHIN, X, COSTH, COSTD, COSTP,
COMMON TRADEA, DAM1, DAM2, DEF, SUB, COSTH, IACT, ALPHA, PRICH
COMMON COSTEP, TP, JMIN, HA, IC, IFIG, KDP, ME
COMMON TDS(48,2), KP, KJPP, TEAM
COMMON RSMAX, RSMIN, FRSMAX, ESHAX, ESNIN, FESMIN, CDHAX, CDMIN,
COMMON ECDHIN, TDMAX, TCHIN, FTDHIN, QT(25), YC(6), CTC(8), DD(32), MIS(9)
DO 40 K=1, IN(201) 40,20,10
IF(IQ(H)-209) 20,20,40
K=IQ(H)-200
JJ=IPAC(M)
N(K)=M
K=NT(M)
DO 30 I=1,24
J=I+1
AAA=ABS(TAB(K,J,2)-TAB(K,I,2))-TAB(K,I,1))/(YF(M)*AAA)
TAB(K,I,3)=CUN(JJ)*(TAB(K,J,1)-TAB(K,I,1))
30 CCCONTINUE
40 CONTINUE
IF(CDHAX.GT.0) GOTO 100

```

```

      TERM1=0.0
      GO TO 110
100  TERM1=(CDMAX-CDMIN)/FCDMIN
110  IF(TDMAX.GT.0) GOTO 120
      TRADET=0.0
      GO TO 130
C ZERNUL WAS ADDED IN MAR 84 TO PREVENT DIVIDE BY ZERO ERRORS.
120  ZERNUL=TDMAX-TDMIN
      IF(ZERNUL.EQ.0.0) ZERNUL=0.00000001
130  TRADET=TERM1*FTDMIN/(ABS(ZERNUL))
      IF(RSHAX.GT.0) GOTO 140
      TRADER=0.0
      GO TO 150
140  ZERNUL=RSHAX-RSMIN
      IF(ZERNUL.EQ.0.0) ZERNUL=0.00000001
150  TRADER=TERM1*FRSHAX/(ABS(ZERNUL))
      IF(ESMAX.GT.0) GOTO 160
      TRADEA=0.0
      GO TO 170
160  ZERNUL=ESMAX-ESMIN
      IF(ZERNUL.EQ.0.0) ZERNUL=0.00000001
170  TRADEA=TERM1*FESMIN/(ABS(ZERNUL))
      IF(CPUMAX.GT.0) GOTO 180
      TRADEP=0.0
      GO TO 190
180  ZERNUL=CPUMAX-CPUMIN
      IF(ZERNUL.EQ.0.0) ZERNUL=0.00000001
      TRADEP=TERM1*PCPHIN/(ABS(ZERNUL))
C 190 CCNTINUE
      RETURN
      END
C *****
C SUERCUTINE DESRES
C
C 73MAR07. REVISED TO DEGRADE TEST VALUES TO
C - EQUIVALENT TABLE VALUES.
C
C COMMON REST(48),TAB(20,25,3),CTEST(3),CTOH(3),XMIS(2),ISUBS(48)
C , IPAC(48), NT(48), IRC(49), IDDT(48)

```

[illegible]

```

      JMIN=K
      NT(36)=K
      11 IF (KDP.GT.4) GOTO 95
      IF (XCTC=CTC(K))
      IF (XCTC.LE.0.0) GOTO 95
      L=IPAC(M)
      XRES=XCTC/CUN(I)
      RESU(M)=XRES-RESS(M)
      REST(M)=XRES
      NN(K)=1
      IF (KDP.GT.4) GOTO 95
      IF ((K.NE.5).AND.(K.NE.6)) GOTO 95
C BACKUP GUIDANCE
      IF ((K.EQ.5).AND.(M6.EQ.3)) GOTO 408
      IF ((K.EQ.6).AND.(M6.EQ.1)) GOTO 408
      GOTO 95
C
      L=NT(M)
      LL=IPAC(M)
      X3=0.3333
      DO 410 J=1,25
      410 TAB(I,J,1)=X3*TAB(I,J,1)
      RESMAX(LL)=X3*RESMAX(LL)
      RESMIN(LL)=X3*RESMIN(LL)
C
      95 CCINUE
      IF (JDE.GE.5) GOTO 100
C DEGRADE INPUT MIN ACCEPT TO EQUIV TAB VALUES
      - FOR INIT SEARCH. DEGRADE INCENT LIMITERS
      - SAME FOR ENDS OF INCENT SEARCHES.
      - THROUGH STAT 103.
C
C RELIABILITY FACTORS
      DQT1 = 1.0 - ((40.0 - QT(2)) * .00162)
      DQT2 = 1.0 - ((5.0 - QT(3)) * .002)
      DQT3 = 1.0 - ((6.0 - QT(6)) * .003)
      DQTR = 1.0 - ((25.0 - QT(1)) * .0065)
C ERROR FACTORS
      DQT4 = 1.0 + ((14.0 - QT(4)) * .019)

```



```

DOT5 = 1.0 * ((5.0 - QT(5)) * .0062)
DOTE = 1.0 * ((25.0 - QT(1)) * .005)
C RELIABILITY TEST VALUES
YDC(1) = YC(1) * DOTT * DFTR
YDC(2) = YC(2) * DOTT * DFTR
YDC(3) = YC(3) * DOTT * DFTR
ERSNAY = RSNAY * DQT1 * DQT2 * DQT3 * DFTR
C ERROR TEST VALUES
YDC(4) = YC(4) * DOTT * DFTE
YDC(5) = YC(5) * DOTT * DFTE
YDC(6) = YC(6) * DOTT * DFTE
LESHIN = ESHIN * DQT4 * DQT5 * DFTE

```

```

C 103 NR=0
NE=0
KA=1
KZ=3

```

```

C MOTCR AIRFRAME, GSE/LAUNCHER

```

```

99 DO 101 K=KA, KZ
ASSIGN 102 TO KGO

```

```

H=NT(K)
J=NT(M)
IF(NN(K) NE.1) GOTO 9
IF(K.LE.3) NR = NR+1
YRESU=RESU(M)
DO 120 I=1,25
IF(XRES6.LE.TAB(J,I,1)) GOTO 130

```

```

120 CONTINUE
I=25
GOTO 130

```

```

C 9 XYCK = YDC(K)
DO 14 I=1,25
IF(K.LE.3) AND: {XYCK.LE.TAB(J,I,2)} } GOTO 130
IF({K.GT.3} AND: {XYCK.GE.TAB(J,I,2)} } GOTO 130
14 CONTINUE
I=25
130 INDX(K)=I
101 CCNTINUE

```

```

      GOTO KGO, (102,56)
C 102 IF (NE-GE-3) GOTO 39
      KA=1
      KZ=3
      ASSIGN 31 TO KGCA
150 DO 30 K=KA,KZ
      H=N(K)
      J=NT(H)
      I=INDX(K)
      RATIO(R)=TAB(J,I,3)
      Y(R)=TAB(J,I,2)
30 CONTINUE
      GOTO KGOA, (31,38)
C 31 XY1=Y {1}
      XY2=Y {2}
      XY3=Y {3}
      IF (XY1 * XY2 * XY3) .GE. DRSHAX) GOTO 159
      DEC {1} = RATIO {1} / {XY2 * XY3}
      DEC {2} = RATIO {2} / {XY1 * XY3}
      DEC {3} = RATIO {3} / {XY1 * XY2}
      GOTO 33
C 159 KA=1
      KZ=3
      ASSIGN 44 TO KGCB
160 DO 35 K=KA,KZ
      H=N(K)
      J=NT(H)
18 TAB(20,K,1)=NT(H)
      IF (ID6-LE-3) GOTO 1450
      I=TAB {20,K,2}
      IF (INDX(K) .GE. I) GOTO 1450
      INDX(K)=I
      GOTO 1470
C 1450 I=INDX(K)
      TAB(20,K,2)=I

```

```

1470      YRESU=TAB(J,I,1)
        RESU(M)=XRESU
        REPT(M)=XRESU+RESS(M)
35      CONTINUE
        GOTO KGOB, (44,80)

C FIND SPALLEST COST/PERFORMANCE FACTOR
33      I=0
DO 34 K=1,3
  IF (NN(K).EQ.1) GOTO 34
C MAX PERFORMANCE NOT REACHED
  YDECK=DEC(K)
  IF (I.GT.0) GOTO 22
  I=1
  DECHIN=YDECK
  HK=K
34      CONTINUE
22      IF (XDECK.LT.DECHIN) HK=K
        IF (XDECK.LT.DECHIN) DECHIN=XDECK
        IF (DEC(HK).GE.HEADER) GOTO 39
C THE SPALLEST IS LESS THAN RELIABILITY TRADE-OFF FACTOR
        K=HK
        M=N(K)
        J=NT(M)
        INDX(K)=INDX(K)+1
        IF (INDX(K).GT.25) INDX(K)=25
        I=INDX(K)
        RATIO(K)=TAB(J,I,3)
        Y(K)=TAB(J,I,2)
        IF (I.LT.25) GOTO 31
C MAX PERFORMANCE REACHED
        RATIO(K)=9.55*10**9
        NN(K)=1
        NR=NR+1
        IF (NR.LT.3) GOTO 31
C
39      KA=1
        KB=3
        ASSIGN 44 TO KGOB
        GOTO 160

```

```

C 44 IF (TRADEA.LE.O) NE=2
C NO INCENTIVE FOR ACCURACY
KA=4
KZ=6
C FIRE CONTROL GUIDANCE A, GUIDANCE B
ASSIGN 56 TC KGO
GOTO 99
56 NE=NE+NN(4)+NN(KP)
IF (NE.GE.2) GOTO 61
KA=4
KZ=6
ASSIGN 38 TC KGOA
GOTO 150
C 38 IF ((Y(4)+Y(KP)).GT.(DESMIN /YF(M))) GO TO 62
61
KA=4
KZ=6
ASSIGN 80 TC KGOB
GOTO 160
C 62 IF (RATIO(4).LE.RATIO(KP)) GOTO 66
IF (NN(KP).EQ.1) GOTO 68
64 EK=KP
GOTO 70
66 IF (NN(4).EQ.1) GOTO 64
68 MK=4
C 70 IF (RATIO(MK).GE.TRADEA) GOTO 61
C
K=MK
H=N(K)
J=NT(H)
INDX(K)=INDX(K)+1
IF (INDX(K).GT.25) INDX(K)=25
I=INDX(K)
RATIO(K)=TAP(J,I,3)
Y(K)=TAB(J,I,2)
IF (I.I.25) GOTO 38
C MAX PERFORMANCE REACHED

```

```

      RATIO(K)=9.59*10**9
      NN(K)=1
      NE=NE+1
      IF (NI.LT.2) GOTO 38
      80   GOTO 61
      KKPP=1
      GOTO 9999
      100  KKPP=2
      9999 RETURN
      END

```

```

C *****
C SUPEROUTINE DESRET
C *****

```

```

COMMON REST(48),TAB(20,25,3),CTEST(3),CTOH(3),XMIS(2),ISUBS(48)
COMMON IPAC(48),INT(48),IRC(49),IDDT(48),CHMT(48)
COMMON PERS(48),ITERH(48),RESB(48),Y(8),RESA(48)
COMMON PERC(48),TDUE(48),REC(3),YF(48),ALN(48),CT(48)
COMMON IPRC(48),RESC(48),CL(6),TMIN(48),IO(48),CUN(30),ISW(3)
COMMON TPTN(48),RESHAX(30),NN(8),ADJ(2,8),ICB,INA,INF,IDE
COMMON RESHIN(50),NN(8),AN(9),CD,PVR(6),CEUHN,ICPHIN,XA,TEMP(25),COV,C10,TRADET,TRADER,
COMMON ITAB(25),RATIO(8),PER(6),CPUMIN,FCOSTA,COSTD,CSTP1
COMMON CPUMAX,CEUHN,DAM1,DAM2,DEF,SUR,COSTA,ICCT,ALPHA,PRIOR
COMMON SRALEA,DAM1,DAM2,DEF,SUR,COSTA,ICCT,ALPHA,PRIOR
COMMON CCMON,TRADEP,TF,JMIN,MA,IC,IFIG,KDP,HK
COMMON TDS(48),NC,COFG,KP,KAPP,ITEAH
COMMON RSMAX,RSMIN,FRSMAX,ESMAX,ESMIN,FESMIN,CDMAX,CDMIN
COMMON TDHAX,TDHIN,FTDHN,QT(25),YC(6),CTC(8),DD(32),MIS(9)
80 K=7
H=N(K)
J=N(K)
ZRSU1=RESU(M)
IF (ILP.GE.5) GOTO 90
IF (ZRSU1.LE.5) TAE(17,11,3) GOTO 2643
88 DO 89 I=1,25

```

```

      IF (ZRSU1-LE.TAE(J,I,1)) GOTO 960
89  CCONTINUE
      I=25
      GOTO 965
C 960 IF (I-EO,1) GOTO 962
      ZTB1=TAB(J,I-1,1)
      ZTB2=TAB(J,I-1,2)
      ZTB3=TAB(J,I-1,3)
      ZTB4=TAB(J,I-1,4)
      ALPHA=ZTB3-(((ZRSU1-ZTB1)/(ZTB2-ZTB1))*(ZTB3-ZTB4))
      GOTO 970
C 962 ZRSU1=TAB(J,I,1)
      ALPHA=TAB(J,I,2)
      TAE(17,11,3)=ZRSU1
      TAE(17,12,3)=ALPHA
      TAE(20,7,1)=J
      TAE(20,7,2)=I
      GOTO 76
C 2643 RESU(M)=TAB(17,11,3)
      ALPHA=TAB(17,12,3)
C 76 C10 = 0
      DO 850 K=1,2
      N=N(K)
      J=N1(H)
      I=INDX(K)
      RESU(H)=TAB(J,I,1)
      JJ=IRC(M)
      AL=CL(JJ,5)
      AL=1+AL
      CTCT = CL(JJ,2) + AL * CL(JJ,1) * CL(JJ,3) *
      & (1.0 + 0.5 * RESU(M) / CL(JJ,4))
      C10=C10+CTCT
850 CONTINUE
C
      K=KP
      N=N(K)

```

```

J=NT(M)
I=INDX(K)
RESU(M)=TAB(J,I,1)
JJ=IRC(M)
AL=CL(JJ,5)
AL=1*AL
CTCT = CL(JJ,2) + AL * CL(JJ,1) * CL(JJ,3) *
      (1.0 + 0.5 * RESU(M) / CL(JJ,4))
C10 = C10 + CTCT
AL=CL(6,5)
AL=1*AL
C10=C10+CL(6,2)+AL*CL(6,1)*CL(6,3)
TAE(20,25,1)=C10
C10 = C10 * ALPHA
C
K=7
M=N(K)
CALL IN VALUE ENGINEERING RESOURCES FROM TABLE
90 XRESU=TAB(17,11,3)
RESU(M)=XRESU
REST(M)=XRESU+RESS(M)
ALPHA=TAB(17,12,3)
C COMPUTE UNIT COST OF PRODUCTION
C10=TAB(20,25,1)*ALPHA
TAE(20,25,1)=C10
C FIND AVAILABILITY FACTOR (ZULU) FROM MAINT. ENG
K=8
M=N(K)
J=NT(M)
ZRSU2=RESU(M)
IF(IIP,GE,5) GOTO 86
IF(ZRSU2.LE.TAE(17,13,3)) GOTO 86
IF(ZRSU2.LE.CURRENT MAINT. ENG. .LE. PREVIOUS, DC NOT SEARCH
DO 85 I=1,25
IF(ZRSU2.LE.TAE(J,I,1)) GOTO 84
CCCONTINUE
I=25
GOTO 975
C
84 IF(I.IQ.1) GOTC 972

```

```
C INTERPOLATE,  
ZTPE=TAB(J,I-1,1)  
ZTIE6=TAB(J,I-1,1)  
ZTIE7=TAB(J,I-1,2)  
ZTIE8=TAB(J,I-1,2)  
ZU10=ZTB - (((ZBSU2-ZTB5)/(ZTB6-ZTB5))*(ZTB7-ZTB8))  
GOTO 980  
C STORE MAINT, ENG., ZU10, + TABLE ID  
ZRSU2=TAB(J,I,i)  
972 ZU10=TAB(J,1,2)  
975 ZU10=TAB(J,1,3)=ZBSU2  
980 TAE{17,14,3}=ZU10  
TAE{17,14,3}=J  
TAE{20,8,2}=I  
GOTO 985  
C 86 RESU(M)=TAB(17,13,3)  
985 REST(M)=RESU(M)+RESS(M)  
C 100 CONTINUE  
RETURN  
END  
***** SUROUTINE SE *****  
C  
COMMON REST(48), TAB(20,25,3), CTEST(3), CTOH(3), XHIS(2), ISUBS(48)  
IPAC(48), ITERM(48), IRC(49), IDDT(48), CHAT(48)  
COMMON PERC(48), TDUE(48), RESB(48), Y(8), ITYPE(48)  
IPRC(48), RESC(48), DEC(3), YF(48), RESA(48)  
TFSTIN(48), THIN(48), CL(6,5), ALN(48), RCT(48)  
CCMCN RESMIN(30), NN(8), RESMAX(30), AK(30), CUN(30) ISW(3)  
ITAB(25), AN(9), ADJ(2,8), INDY(8), PDR(6)  
RATIO(8), PVR(6), AN(9), CD, ICD, INA, INF, IDF  
PER(6), CPUMIN, FCPMIN, MTEMP(25), COV, C10, TRADET, TRADER,  
COMMON CPU MAX, DEF, SUR, COSTM, ICT, ALPHA, PRIC  
STRACEA, DAM1, DAM2, COSTP2, COSTP1
```



```

470 RESU(M) = (RESB(M) + (ADJ(1,J)/ADJ(2,J)) * 0.5 * RESB(M)) * ALPHA
471 REEC(M) = 0.0
472 IF (TY(M) - 0.1) 471, 471, 475
473 PERC(M) = 1.0
474 REEC(M) = RESA(M)
475 IF (TPIN(M)) 1000, 1000, 490
476 TPIN(M) = CD + 1.0
477 GO TO 490
C
475 IF (RESA(M) + 0.01 - REST(M)) 477, 471, 471
476 IF (RESA(M) - REST(M) - RESU(M)) 490, 490, 476
477 TERM1 = ((RESA(M) - REST(M)) / RESU(M)) **
8
478 IF (TERM1) 490, 490, 480
479 IF (TERM1 - 1.0) 481, 481, 471
480 PERC(M) = TERM1
481 CONTINUE
490 MODULE 3
491 SETS DUE DATES
C
590 KTR101=0
DO 585 HINA=1, INA
N = (INA - HINA) + 1
IF (IPRC(M)) 585, 510, 585
510 IF (PERC(M) - 1.0) 520, 585, 585
520 KTR101 = KTR101 + 1
I = IFAC(M)
IF (L.GT.30) L=30
IPRC(M) = -1
AA = TDUE(M) + (REST(M) - REST(M) - RESU(M)) / RESHIN(L)
TERM3 = TDUE(M) - (REST(M) + RESU(M)) / RESHIN(L)
TERM4 = AA - TDUE(M)
C
DO 570 I=1,5
J = IPC(M, I)
IF (J) 580, 580, 530
530 IF (ISUBS(M) - 8) 540, 535, 535
535 IF (ISUBS(M) - ISUBS(J)) 570, 540, 570
1001 PRINT 1001, M, J, K, I
PCFMT(413)

```

```

C      GO TO 2
540 K=IPAC(J)
    IF (K.GT.30) K=30
    IF (J.GT.48) GC TO 1
    2 CCNTINUE
    TERM1=(REST(J)-RESS(J)-RESQ(J))/RESMIN(K)
545 IF (TERM1-TERM4) 545,545,550
    DDI=TERM3
    GC TO 555
C      DDD=TERM3-(TERM1-TERM4)
550 IF (TDUE(J)-DDD) 565,565,560
560 TDUE(J)=DDD
565 IPRC(J)=IPRC(J)-1
C      570 CONTINUE
580 TDUE(M)=AA
C      585 CCNTINUE
    IF (KTR101.GT.0) GOTO 590
    FIX DATA REDUCTION, FLIGHT TEST
    DO 850 M=1,INA
    IF (TDUE(M)-898.0) 850,810,810
810 I=IPC(M)
    IF (I) 850,850,820
820 L=IPAC(M)
    TDUE(M)=TDUE(I)+REST(M)/RESMIN(L)
850 CONTINUE
    HCDULE 5
    SET ACTIVITY POINTERS BY FACILITY
    DO 690 M=1,INA
    IF (PERC(M)-1.0) 610,690,690
610 I=IPAC(M)
611 IF (ISUB5(M)-8) 620,640,640
620 IF (IACT(I)) 641,630,641
630 AK(I)=AK(I)-TRADET
    GO TO 64
C      640 AK(I)=-5000.0

```

```

641 IF (IACT(I,1)) 660,650,660
650 GO TO 690
C
660 MM=IACT(I,1)
670 IF (TDUE(MM)-TLUE(M)) 670,670,680
680 GO TO 690
C
680 IACT(I,2)=MM
690 IACT(I,1)=M
C
690 CCCTINGE
C
690 SET PROCUREMENT DASHPTS - TAB(17,1,3)= BLOCK1, TAB(17
TAE(17,3,3)=0.0
TAE(17,4,3)=0.0
DO 700 M=1,INA
J=3
IF (ISUBS(M)-8) 700,701,702
702 J=J+1
701 I= IFAC(M)
XX=REST(M)
IAM=CHAT(M)*QTY(M)*ALPHA
Y=0.0
IF (CT(M)) 715,715,710
710 Y=CT(M)-TAM
715 XX=REST(M)-RESA(M)
715 Y=Y+TAM+XX+CUN(I)
700 TAE(17,J,3)=TAE(17,J,3)+Y
C
CCCTINGE
C
FETUEN
END
C *****
C SUEROUTINE MOD9
C *****
C
COMMON REST(48), TAB(20,25,3), CTEST(3), CTOH(3), XMIS(2), ISUBS(48)
C
C IFAC(48), INT(48), IRC(49), IDDT(48)
C IFD(48), ITPRM(48), QTY(48), CHAT(48)
C COMMON RESB(48), RESD(48), RESB(48), Y(8)
C RESB(48), TDUE(48), DEC(3), Y(8)

```

[illegible]

```

IF (J,LE.0) GOTC 7
KK = 0
IF (I,LT.5) KK= IPC(M,I+1)
IF (TERM(J)-IDE-1) 10,8,10
IF (PERC(J)-PRICH) 11,12,12
10 PRIOR=PERC(J)
11 IF (TSTAR(J)-X) 13,13,14
12 X=TSTAR(J)
13 IF (KK) 15,7,15
14 IF (PERC(J)-1.0) 16,7,16
15 IF (PERC(H)) 7,7,17
16 JJ=YFAC(J)
17 XX=((PERC(J)-PERC(H))*REST(J))/RESC(JJ)
18 IF (XX-TMIN(J)) 18,7,7
19 CONTINUE
C
IF (X-TSTAR(M)) 19,19,20
20 TSTAR(M)=X
19 IF (PERC(H)-PRIOR) 21,8,8
21 WA=RESMIN(L)
22 IF (AK(L)) 22,27,27
23 CCNTINUE
C
      PRCH MOD 11 IN DHS PROGRAM . . . . .
      T=((REST(M)-RESA(M))/RESMIN(L))+CD
      IF (TDUE(M)-LE-CD) GO TO 83
      IF (TDUE(M)-GE-T) GO TO 82
      TERM1=(REST(M)-RESA(M))/(TDUE(M)-CD)
      CM (A TYPO) CHANGED TO CD 1970 MARCH 25 . . .
      IF (RESMAX(L):GT:TERM1) GO TO 84
      RESC(I)=RESMAX(L)
      GO TO 82
      RESC(L)=TERM1
      CCNTINUE
      IF (RESC(L)-GE:RESMIN(L)) GO TO 26
      RESC(I)=RESMIN(L)
C

```

```

C C C C C
      FROM HERE TO 100 IS EXTRACTED FROM MOD7
      IN DMS PROGRAM . . . . .
26 WA=RESC(L)
27 X=0.0
    I=QTY(M)+0.1
    IP(I-1) 31,50,23
31 X=1.0
    WA=0.0
    REST(M)=0.0
    GO TO 24
C
50 IF (PRIOR-1.0) 8,23,23
23 TERM1=REST(M)-RESA(M)
25 IF (TERM1-0.1-RESC(L)) 25,25,32
    WA=TERM1
    PERC(M)=1.0
    GO TO 100
C
32 IF (RESA(M)-RESS(M)-RESU(M)) 30,30,40
40 X = { ((RESA(M)-RESS(M)+WA)/RESU(M)) ** (1.0/(1.0-ALN(M)
      E
      ) ) }
      QTY(M) }
      X-PRIOH } 24,8,8
30 IF {
24 RESA(M)=RESA(M)+WA
    PERC(M)=X
100 CCNTINUE
C C C C C
      FROM HERE TO 93 PLUS TWO IS EXTRACTED FROM
      MOD8 IN DMS EROGRAM . . . . .
I=0
IP(ISUBS(M)-8) 75,76,77
77 I=I+1
76 I=I+1
75 I=I+1
    IF (RESA(M).NE.WA) GO TO 71
    CTEST(I)=CTEST(I)+(CMAT(M)*QTY(M))*ALPHA
    CT(M)=CT(M)+(CMAT(M)*QTY(M))*ALPHA

```

```

C      TSTAR(M)=CD
C      71 ISW(I) = ISW(I)+1
        X=.0
        IJ=IFAC(M)
        IF(WA-LE. RESMIN(IJ)) GO TO 81
        X= WA-RESMIN(IJ)
C      81 XX=(WA*CUN(IJ)+.5*XX*CUN(IJ))
        CT(M)=CT(M)+XX
        CTST(I)=CTEST(I)+XX
        IF(RESA(M)+.1-IT. REST(M)) GO TO 64
        PERC(M)=1
        RESA(M)=REST(M)
        TFIN(M)=CD
C
C      IA1=IACT{L,1}
C      IA2=IACT{L,2}
C      IF(M-EO-IA1) GOTC 101
C      IF(M-EO-IA2) GOTC 102
C      PRINT 1002
C      1002 FORMAT(' 8**MOD* IACT ERROR ***')
C      STOP
C      101 IACT{L,1}=IA2
C      102 IACT{L,2}=0
C      93 J=ITEFM(M)
C      IF(J-IDP-NE. 1) GO TO 64
C      ITCT=ITCT-1
C      FROM HERE TO 64 IS EXTRACTED FROM MOD7
C      IN DMS PROGRAM .....
C      8 CONTINUE
C      64 RESC(I)=RESMIN(I)
C      FROM HERE TO 64 CONTINUE CHANGED TO ABOVE 1970 MARCH 25 . . .
C      IN DMS PROGRAM .....
C      FROM HERE TO END IS EXTRACTED FROM MOD9
C      IN DMS PROGRAM .....
C      IF (ISW(1) .EQ.0) GO TO 66

```



```

CTCH(1) = CTOH(1) + COV
66 CONTINUE
IF (IIC - GT. 0) GO TO 67
IF (IDP - NE. 7) GO TO 60
DC 52 L=1 INF
IF (IAC(L,1) - GT. 0) GO TO 67
52 CONTINUE
60 CONTINUE
RETURN
END
C *****
C SUROUTINE REPCET
C *****
COMMON REST(48), TAB(20,25,3), CTEST(3), XHIS(2), ISUBS(48)
& , IPAC(48), INT(48), IRC(49), IDDT(48), CHAI(48)
& , ITERM(48), QTY(48), RESB(48), Y(8), IITYPE(48)
& , PERC(48), TDUE(48), DEC(3), YP(48), RESA(48)
& , IPERC(48), RESC(48), ALN(48), RCT(48)
& , TSTAR(48), TCLI(6,5), THIN(48), IQ(48), CUN(30), ISW(3)
& , TFIN(48), RESHAX(30), AK(30), INDY(8)
& , RESHIN(30), NN(8), AN(9), CD, ICD, PDR(6)
& , ITAB(25,3), PVR(6), CFMIN, FCFIN, X, CD, INF, IDP
& , RATIO(8), AN(9), ADJ(2,8), INA, COV, C10, TRADET, TRADER,
& , PER(6), CPUMAX, CFMIN, FCFIN, X, CD, NTEMP(25), COSTD, COSTP1
& , STRADEA, DAM1, DAM2, DEF, SUR, COSTH, IACT, ALPHA, PRICR
& , COMMON, COSTP2, C6STON, TP, JMIN, IC, IFIG, KDP, ME
& , TRADEP, TDS(48,2), NC, ICONF, KP, KKPP, ITEAM
& , COMMON, RSHAX, RSHIN, PRSHAX, ESHAX, ESHIN, FESHIN, CDHAX, CDHIN,
& , FCDMIN, TDHAX, TDHIN, PTDHIN, QT(25), YC(6), CTC(8), DD(32), MIS(9)
& , X1=0,
& , X2=0,
& , X3=0,
ITAB(24)=ICD
ITAB(17,25,3)=CTEST(1)+CTOH(1)
DO 11 I=1,3
IF (CTEST(I)) 12,12,11

```

```

12 CTEST(I)=1.0
11 CCNTINUE
C
X1=(CTEST{1}+CTCH{1})/CTEST{1}
X2=(CTEST{2}+CTCH{2})/CTEST{2}
X3=(CTEST{3}+CTCH{3})/CTEST{3}
DO H=1,INA
IF (ISUBS(H)-8) 2,1,1
2 X=X1
GO TO 5
5 IF (QTY(M)-1-0) 6,6,9
9 IDS(H,1)=PERC(H)
GO TO 10
20 IDS(H,1)=1.
GO TO 10
C
6 IF (REST(M).EQ.0) GO TO 20
TDS(M,1)=RESA(M)/REST(M)
10 IDS(H,2)=CT(H)*X
1 CCNTINUE
C
IDF=7
ITAB(25)=3
CCNTINUE
RETURN
END
C *****
C SUEROUTINE PERF *****
C
COMMON REST(48),TAB(20,25,3),CTEST(3),CTOH(3),XMIS(2),ISUBS(48)
E IFAC(48),INT(48),IRC(49),IDDT(48),CHAT(48)
E IPD(48),ITERN(48),QTY(48),RESB(48),Y(8),ITYPE(48)
E PERC(48),RESU(48),TDUE(48),DEC(3),Y(8),RESA(48)
E IPRC(48),RESC(48),RCL(48),VF(48),ALN(48),RESA(48)
E TSTAR(48),THIN(48),THIN(48),IO(48),CUN(30),CT(48)
E TFIN(48),RESMAX(30),N(8),AK(30),ISW(3)
E RESMIN(30),NN(8),INDX(8),PDR(6)
E ITAB(25),AN(9),ADJ(2,8)
E RATIO(8)
E

```

```

      PVR(6) CPUMIN, FCPHIN, CD, MTEMP(25), COV, C10, TRADET, TRADER,
      STRADEA, DAM1, DAM2, DEF, SUB, COST, IACT, ALPHA, IPC(48,5)
      CCHHCN COSTP2, COSTOP, HA, IC, IPFG, KDP, HR
      , TRADER(2), TP, JMIN, TEAM
      , TDS(48,2) KP, KKP, ITEAM
      , NC, ICONF, RSHAX, ESHAX, ESMIN, PESMIN, CDMAX, CDMIN,
      COMMON RSHAX, RSHIN, PRSHAX, ESHAX, ESMIN, PESMIN, CDMAX, CDMIN,
      EFCDMIN, TDMAX, TIMIN, PTDMIN, QT(25), YC(6), CTC(8), DD(32), HIS(9)
      INTEGER IN/5,
      DATA FNA/0./, PAMAX/0./, PAMIN/0./, PDA/0./, PVA/0./, PEA/0./
      FNA=0.
      FMAX=0.
      FMIN=0.
      PDA=0.
      PVA=0.
      PEA=0.
      KP=JMIN
      DO 118 M=1, INA
      IDS(M)=1
      IF (I(M)-6) 116, 116, 118
      IF (I(M)-1) 118, 400, 117
      I=ISUBES(M)
      AN(I)=1.0+QTY(M)*TDS(M,1)
      AK(I)=1.0+QTY(M)
      GO TO 118
      FTST=QTY(M)*TDS(M,1)
      TTST=QTY(M)
      118 CONTINUE
C 119 DO 130 K=1,6
      M=N(K)
      I=ISUBES(M)
      J=TAB(20,K,1)
      PRIOR=100.0*TAEL(J,2)*(1.0-TAB(L,J,2))-1.0
      IF (K-4) 106, 106, 107
      IF (K-4) 106, 106, 107
      AN(I)=AN(I)+PRICK+FTST
      AK(I)=AK(I)+PRICK+TTST
      GO TO 130

```

```

C 107 IF (K-KP) 110,106,110
110 AN(6) = AN(I) + PRICR + FTEST
AK(6) = AK(I) + PRICR + TTEST
SS = AK(6) - PHOR - 1.0
PEA = TAB(L,J,2)
EVA = EVA* (1.0 - PEA) / (AN(6) + 1.0)
PCA = SORT(PVA)
TEH = EXP(0.1*SS)
PF = 0.915 + 0.314/TEH

C IF (PF - 1.1) 510,510,500
500 PF = 1.0
510 PEA = PEA*PF
520 FAMIN = PEA - PDA
128 IF (FAMIN - 0.1) 128,129,129
129 FAMAX = PEA + PDA

C IF (PAMAX - 0.99) 109,109,430
430 PAMAX = 0.99
PHA = (PEA * AN(6) - 1.0) / (AN(6) - 2.0)
515 FMA = PEA + 0.33* (FAMAX - PEA)

C 109 IF (FMA - FAMIN) 68,70,70
68 FMA = PEA - 0.33* (FMA - FAMIN)
70 SPER = FMA
SFEA = FAMIN
SPVR = FAMAX

C 130 CONTINUE
C DO 200 I = 1,5
C GO TO (171,173,175,177,178),I
C 171 A = 1.067
B = -1.34
C = 0.1

```

C 173 K=1
J1 = 1
J2 = 2
J3 = 3
GO TO 180 AIRFRAME . . .

A=1.029
B=-0.105
C=0.1
K=2

J1 = 4
J2 = 5
J3 = 6
GO TO 180

C 175 LAUNCEER / GROUND SUPPORT EQUIPMENT . . .

A=1.029
B=-0.098
C=0.1
K=3

J1 = 7
J2 = 8
J3 = 9
GO TO 180

C 177 FIRE CONTROL . . .

A=0.908
B=0.306
C=0.1
K=4

J1 = 11
J2 = 10
J3 = 12
GO TO 180

C 178 GUIDANCE . . .

A=0.915
B=0.314
C=0.1
K=KP

J1 = 14
J2 = 13

K=5 CHANGED TO K=KP 1970 MARCH 27 . . .

```

C
C
C
C
C
180
J3 = 15
      THROUGH STATEMENT 191 REVISED 720123 TO CONFORM
      WITH DNS PERFOR CHANGE OF 720530.
      CAUSES FLIGHT TEST RESULTS TO ALWAYS BE COMPUTED
      ON THE EXPECTED NUMBER OF FLIGHT TESTS.
      I=TAB(20,K1)
      LI=TAB(20,K2)
      XPER = TAB(LI,1,2)
      TAELI = XPER
      PRIOR = 100.0 * TABLL * (1.0 - TABLL) - 1.0
      ANIII = AN(I)
      BLOCK II MOST LIKELY
      SS = AK(I) - PRIOR - 1.0
      TEM = EXP(C * SS)
      FF = (A + B / TEM)
      IF (K-GE. 4) GOTO 535
      IF (FF.LT. 0.90) FF = 0.90
      GOTO 550
535 IF (FF * XPER).GT. 0.99) GOTO 91
550 XPER = XPER * FF
      IF (XPER.LE. 0.99) GOTO 195
      91 XPER = 0.99
      195 TAB(14,J1,2) = XPER
      BLOCK II DEVIATION
      PV = XPER * (1.0 - XPER) / (ANIII + 1.0)
      PD = SORT(PV)
      TAB(14,J3,2) = PD
      BLOCK II EXPECTED
      PH = (XPER * ANIII - 1.0) / (ANIII - 2.0)
      PHAX = XPER + FF
      IF (PHAX.GT. 0.99) PHAX = 0.99
      IF (PH.LE. PHAX) GOTO 197
      PH = XPER + 0.33 * (PHAX - XPER)
      197 PHMIN = XPER - FF
      IF (PHMIN.LT. 0.1) PHMIN = 0.1
      IF (PH.GE. PHMIN) GOTO 191

```

```

PM = XPER - 0.33 * (XPER - PHIN)
191 TAF(14,J2) = FN
C . . . ABOVE 10 STATEMENTS INSERTED 1970 APRIL 1 . . .
C 200 CONTINUE
C
TAF(14,16,2) = TAB(14,1,2)*TAB(14,4,2)*TAB(14,7,2)
TAF(14,17,2) = TAB(14,2,2)*TAB(14,5,2)*TAB(14,8,2)
TAF(14,19,2) = TAB(14,10,2)*TAB(14,13,2)
TAF(14,20,2) = TAB(14,11,2)*TAB(14,14,2)
C . . . ABOVE STATEMENT INSERTED 1970 APRIL 1 . . .
C CCNINUE
C FETUEN
C END
C *****
C SUEROUTINE PROSUM
C *****
COMMON REST(48), TAB(20,25,3), CTEST(3), CTOH(3), XMIS(2), ISUBS(48)
& , IFAC(48), NI(48), IRC(49), IDDT(48),
& , IFPD(48), ITERM(48), QTY(48), CHAT(48),
& , PERC(48), TRESU(48), RESB(46), Y(8), IYPE(48)
& , IPRC(48), TRES(48), DEC(3), YF(48), RESA(48)
& , TSTAR(48), RESC(48), ALN(48), CT(30), ISW(3)
& , TFIN(48), THIN(48), CL(6,5), IQ(48), CUN(30)
& , RESHIN(30), NN(8), AK(30), ISW(3)
& , ITAB(25), NN(8), IND(8), PDR(6)
& , RATIO(8), AN(9), ADJ(2,8), INA INF, IDE
& , PER(6), CPUMIN, FCFMIN, X, NTEMP(25), COV, C10, TRADET, TRADER,
COMMON CPUMAX, DAM1, DAM2, DEP, SUR, COSTH, COSTD, COTSTP1
& , TRALEA, DAM1, DAM2, DEP, SUR, COSTH, COSTD, COTSTP1
COMMON COTSTP2, TP, JMIN, HA, IC, IPFG, KDP, PRICR
& , TRADER, TDS(48,2), KP, KPP, ITEAN
& , NC, IC, COFG, RSHIN, PRSHIN, ESHAX, BSMIN, FESMIN, CDMAX, CDMIN,
COMMON RSHAX, RSHIN, PRSHIN, ESHAX, BSMIN, FESMIN, CDMAX, CDMIN,
& , FCDMIN, TDHAX, TDHIN, FTDHIN, OT(25), YC(6), CTC(8), DD(32), MIS(9)
COMMON STORED, STAT(5), NDUIMP(5,5), ADUHP(5,28), DPCCST, MAYDP
& , REAL YFEE(8), FRATIO(8), FYC(5)
& , REAL*8 STAT

```

```

REAL*8 FINAL,'FINAL',INTEN,'PROPOSED',
INTEGER ANS,ANZ,'A',ANO,'N',YES,'Y',E,'E',IN/5/
THIABA=0.
CTEST=CTEST(1)-(CT(19)+CT(20))
FCTOH=CTEST/CT(1)
FCTOH(1)=CTOH(1)*FCTOH
Y(1)=CTEST(1)+CTOH(1)
DO 5 J=1,10
ITAB(J)=TAB(19,J,1)
5 CONTINUE
DO 10 M=1,INA
IF(IQ(M)-1)10,15,10
10 CONTINUE
C
M=INA
15 Y(2)=TPIN(M)
25 Y(3)=TAB(14,16,2)
Y(4)=TAB(14,20,2)
C
30 K=4
M=N(K)
Y(4)=Y(4)*YP(M)
C
PER(1)=CDMAX
AN(1)=CDMIN
FYC(1)=FCDHIN
PER(2)=TDMAX
AN(2)=TDMIN
FYC(2)=PTDMIN
AN(3)=RSMAX
PER(3)=PRSMAX
FYC(3)=ESMAX
AN(4)=ESMIN
PER(4)=FESHIN
FYC(4)=CPUMAX
AN(5)=CPUMIN
FYC(5)=PCPMIN

```

CHANGED TO ROW 20 1970 APRIL 1 . . .


```

SEVA=0.0
FRATIO(8)=0.0
Y(1)=Y(1)/1000000.0
Y(3)=Y(3)*100.0

C THIS SET OF STEPS TO CHECKS THE MOST DESIREABLE VALUES,
C AN(1) VS ACHIEVED VALUES, Y(1), AND LEAST DESIREABLE VALUE, PER(1)
C VS ACHIEVED VALUE IN ORDER TO PLACE THE FEE WITHIN THE LINEAR RATIO
C BOUNDS OF THE DESIRED FEE.
DO 50 I=1,5
  IF(I.EQ.3) GOTO 48
  IF(Y(I)-AN(I)).GE.0. AND PER(I)-Y(I).GE.0. AND PER(I).NE.AN(I))
    *FRATIC(I)=FYC(I)*(ABS((PER(I)-Y(I))/(PER(I)-AN(I))))
  IF(Y(I)-PER(I)).GT.0. FRATIO(I)=0.0
  IF(AN(I)-Y(I)).GE.0. FRATIO(I)=FYC(I)
  GOTO 49
48  IF(Y(3)-PER(3)).GE.0. AND AN(3).NE.PER(3)
    *FRATIC(3)=FYC(3)*(ABS((Y(3)-PER(3))/(AN(3)-PER(3))))
  IF(PER(3)-Y(3)).GT.0. FRATIO(3)=0.0
  IF(Y(3)-AN(3)).GE.0. FRATIO(3)=FYC(3)
  YFEE(I)=FRATIO(I)*(CDMAX+CDMIN)/200.0
  SPVA=SEVA+YFEE(I)
  FRATIC(8)=FRATIO(8)+FRATIO(I)
  CONTINUE
50
C
C TOC=Y(1)+SPVA+LEICOST/1000000.0
C
CALL FRTCMS('CISCRN ')
  IF(MR.NE.0) PRINT 14, ITEAM, MAXDP, STAT(MAXDP)
  FORMAT(12X, '***** TEAM ', I2, ' IS IN DP-', I1, I1, A8, '*****')
  PRINT 1000, NTEME(1), ITEAM
  FCENAT(1000, I1, I1, 'DEVELOPMENT CONTRACT ',
1000 8 'SUMMARY *** TEAM', I3)
  PRINT 1002
1002 FORMAT(1H-18X, 'INCENTIVE PROVISIONS', 11X,
8 'INCENTIVE ACHIEVEMENTS',
8 2X, 'INCENTIVE', 6X, 'WORST', 5X, 'BEST', 4X, 'MAX FEE', 5X, 'ACHIEVED',
8 4X, 'FEE', 5X, 'FEE %')
  PRINT 1010
1010 FORMAT(4X, 'AREA', 9X, 'VALUE', 5X, 'VALUE', 3X, 'ALLCWD', 6X

```

```

C      , 'VALUE', 5X, 'EARNED', 3X, 'EARNED', /)
1020 PRINT 1020, CDMAX, CDMIN, FCDMIN, Y(1), YFEE(1), FRATIO(1)
      FORMAT(1, 'DEV. COST', 6X, '$', F5.1, 'M', F5.1, 'H', F7.1,
C      F7.1, 'X', 5X, '$', F6.2, 'M', F6.3, 'H', F7.2, 'X',
      ITA = TDMAX
      ITI = TDMIN
      IY = Y(2)
1030 PRINT 1030, ITA, ITI, PTDHIN, IY, YFEE(2), FRATIC(2)
      FCEMAT(1H, 'PLT TST COMPL', I6, 'WK', I6, 'WK', F7.1,
C      'X', I10, 'WK', 3X, '$', F6.3, 'M', F7.2, 'X',
      Y(3) = TAB(14, 16, 2) * 100.
1040 PRINT 1040, RSMIN, RSMAX, FRSMAX, Y(3), YFEE(3), FRATIO(3)
      FCRMAT(1, 'RELIABILITY', F16.1, 'X', F8.1, 'X', F7.2, 'X',
C      F12.2, 'X', 3X, '$', F6.3, 'M', F7.2, 'X',
      ITA = ESMAX
      ITI = ESMIN
      IY = TAB(14, 20, 2) * 100.
C      CPUMAX = CPUMAX / 100.
      CPUMIN = CPUMIN / 100.
1050 PRINT 1050, ITA, ITI, FESMIN, IY, YFEE(4), FRATIO(4)
      FORMAT(1, 'ACCURACY', 7X, I4, 'YDS', I6, 'YDS', F7.1, 'X',
C      I10, 'YDS', 3X, '$', F6.3, 'M', F7.2, 'X',
      FEETOT = FCDMIN + FTDHIN + FRSMAX + FESMIN + FCPMIN
1070 PRINT 1070, FEEICI, SPVA, FRATIO(8)
      FORMAT(1H0, 'TOTALS',
C      28X,
C      F4.1, 'X',
C      15X, F7.3, 'M', F7.2, 'X',
      IF (ABS(FEETOT - 15.0) .LT. .0001) GOTO 1090
1080 PRINT 1080
      FORMAT(1H, 'TOT MAX FEE ALLOWED DOES NOT EQUAL
C      15.0% ***)

```



```

264 PRINT 264, 2YX
   FORMAT (1H+, 5X, 'FIRE CONTROL : ', T23, F8.3)
C
C
      LAUNCHER
265 ZYX = ((CT(33) + CT(38)) / QT(6)) / Z3
   PRINT 265, 2YX
   FORMAT (1H+, 6X, ' LAUNCHER : ', T23, F8.3)
C
C
      FLIGHT TESTS
270 ZYX = ((CT(40) + CT(41) + CT(42) + CT(46) + CT(47)))
   / CT(1) / Z3
   PRINT 270, 2YX
   FORMAT (1H+, 3X, ' FLIGHT TESTS : ', T23, F8.3)
   PRINT 111
C
C
      DESIGN FACTCES
310 PRINT 310
   FORMAT (1H0, 'DESIGN FACTORS ACHIEVED')
320 PRINT 320
   FORMAT (8X, 'COMPONENT', T22, 'TABLE', T29, 'ROW', T35, 'FACTOR', T44,
   & 'ACHIEVED', )
C
C
      MOTOR
      KZTB = TAB(20, 1, 1)
      KZRW = TAB(20, 1, 2)
330 PRINT 330, KZTB, KZRW, TAB(KZTB, KZRW, 2), TAB(14, 2, 2)
   FORMAT (12X, 'MOTOR : ', T23, T29, T34, F8.3, T44, F7.3)
C
C
      AIRFRAME
      KZTB = TAB(20, 2, 1)
      KZRW = TAB(20, 2, 2)
331 PRINT 331, KZTB, KZRW, TAB(KZTB, KZRW, 2), TAB(14, 5, 2)
   FORMAT (8X, 'AIR FRAME : ', T23, T29, T34, F8.3, T44, F7.3)
C
C
      GUIDANCE A
      KZTB = TAB(20, 5, 1)
      KZRW = TAB(20, 5, 2)
C TMTABA AND TMTABB WERE FORMED TO ALLOW IF STATEMENT TO GENERATE THE
C FOURTH POSITION WRITE STATEMENT VARIABLES.

```



```

C
8 COMMON CPUMAX, CPUMIN, PVR (6), CD, ICD, INA, INF, IDP
8 C10, CQ (18), IPC (48, 5), JHIN, IC, IFIG, KDP, MR
8 IACT (48, 2), JHIN, IC, IFIG, KDP, MR
8 TDS (48, 2), JHIN, IC, IFIG, KDP, MR
COMMON NC, ICOPG, KP, KKPP, IFTEAM
COMMON RSHAX, RSHIN, PRSHAX, ESHAX, ESMIN, FESMIN, CLMAX, CDMIN,
8 FCDMIN, TDHAX, THIN, PTDHIN, QT (25), YC (6), CTC (8), DD (32), HIS (9)

C
ALEHA=CQ (14)
IDP=4
JHIN = 0
NTEMP (2) = 4

C
DO 1 I=1, 48
IF (I .GT. 3) GC TO 4
DEC (I) = 0
ISW (I) = 0
IF (I .GT. 6) GC TO 2
PVE (I) = 0
PDR (I) = 0
PEE (I) = 0
DO 6 J=1, 5
CL (I, J) = 0.

C
2 IF (I .GT. 8) GC TO 3
CTC (I) = 0
INDX (I) = 0
RATIO (I) = 0
ADJ (1, I) = 0
ADJ (2, I) = 0
NN (I) = 0

C
3 IF (I .GT. 9) GC TO 5
MIS (I) = 0
AN (I) = 0
5 IF (I .GT. 18) GO TO 15
QQ (I) = 0.

C
15 IF (I .GT. 25) GO TO 7

```

```

      Q1(I) = 0.
      ITAB(I) = 0.
7 IF (I.GT. 32) GO TO 8
      DD(I) = 0.
C
C      8 RESC(I) = 0.
C
      DO 10 J = 1, 2
      TDS(I, J) = 0.
10 IACT(I, J) = 0
      1 CONTINUE
C
      DO 20 I = 1, 25
      DO 20 J = 1, 20
      IF (J.NE. 17) GO TO 22
      IF (I.LT. 5) GC TO 20
      22 IAE (J, I, 3) = 0.
      20 CCNTINUE
C
      QC(14) = ALPHA
      RETURN
      END
C *****
C SUERCUTINE STORE *****
C
      INIEGER IN/5/, LATFIL/9/
      COMMON/STORED/ STAT(5), NDUMP(5, 5), ADUMP(5, 28), DPCCST, MAXDP
      REAL*8 STAT
      REAL*8 FINAL/, FINAL/, INTEN/, PROPOSED/, BLK/
      REAL*8 NULL/
      REAL*8 DATFIL
      DO 25 JA = 3, 5
      IF (STAT(JA) .EQ. NULL) STAT(JA) = BLK
      WRITE(DATFIL, 35) (NDUMP(JA, K), K = 1, 5), (ADUMP(JA,
      *K), K = 1, 28), STAT(JA)
      25 CONTINUE
      35 FORMAT(2I2, 1X, 3I1, 2X, 8F7.2, /, 3X, 9F7.2, /, 3X, 6F7.2, /, 3X, 5F7.2, 2X, A8)
      RETURN
      END
C *****

```

```

C
SUEROUTINE FINISH
COMMON REST(48), TAB(20,25,3), CTEST(3), CTOH(3), XMIS(2), ISUBS(48)
& IFAC(48), ITRM(48), QTY(48), IDDT(48), CMAT(48)
& COMMON RESU(48), RESB(48), Y(8), RESA(48)
& PERC(48), TDUE(48), YP(48), ALN(48), CUN(30), ISW(3)
& TSTA(48), CL(6), THIN(48), IQ(48), AK(30), IND(8), PDR(6)
& TFIN(48), RESHIN(30), NN(8), AN(9), CD, NTEMP(25), COV, C10, TRADET, TRADER,
& ITAB(25), PVR(6), FCEHINX, SUR, COST, IACT(48,2), IPC(48,5)
& RATIO(8), PER(6), CFUHN, FCEHINX, ESMAX, ESMIN, FESMIN, CDMAX, CDMIN,
& COMHON CPUMAX, DM2, DEF, C6STOF, HA, IC, JMIN, ITEAM,
& ETRADEA, DAM1, DM2, DEF, SUR, COST, IACT(48,2), IPC(48,5)
& COMHON C6STOF, TP, JMIN, ITEAM,
& TRADEP, TDS(48,2), KP, KKPP, ESMAX, ESMIN, FESMIN, CDMAX, CDMIN,
& NC, ICFG, RSHIN, FTDHIN, OT(25), YC(6), CTC(8), DD(32), MIS(9)
& EFCCHIN, TDHAX, TDMAX, TSTAT(5), NDUMP(5), ADUMP(5,28), DPCCST, MAYDP
& CCMHON, STONED, STAT(5), FLAG, ILEN, STOREFL
& CCMHON, STON/COE, ICODE, FLAG, ILEN, FL21/TEAM21//
& REAL*8 ICODE, FLAG, ILEN, FL21/TEAM21//
& REAL*8 STAT, PRICE
& REAL*8 FINAL, INTEN, PROPOSED, BLK, ' / '
& REAL*8 INCCD, CCNT, ' / '
& INTEGER IN/5, LATFIL/9/

10 PROP=INTEN
IF (STAT(KDP) .EQ. INTEN) PROP=FINAL
IF (STAT(KDP) .EQ. BLK) PROP=INTEN
IF (STAT(KDP) .EQ. NULL) PROP=INTEN
CALL FRICHS('CLRSCRN ')

417 PRINT 415, ITEAM, KDP, STAT(KDP)
415 FORMAT(1X, 'TEAM', I2, ' IN DP-', I1, ' IS IN STATUS ', A8, ' . ')
IF (MR.NE.2) GO TO 410
430 PRINT 430
FORMAT(1X,

```



```

416 * WHAT DP AND WHAT STATUS DO YOU WANT: EX: 3 FINAL: 4 PROPOSED '?'
      READ(IN,416,ERR=420)KDP,PROP
      FCEMAT(11,1X,1A8)
      IF (KDP.EQ.EASY) RETURN
      IF (PROP.EQ.FINAL) STAT(KDP)=INTEN
      IF (KDP.LT.3.OR.KDP.GT.5) GOTO 417
      IF (INTEN.OR.PROP.EQ.FINAL) CALL FRICHS('CLRSCRN ')
      IF (PROP.EQ.INTEN.OR.PROP.EQ.FINAL) GOTO 410
      GOTO 417
      RETURN

420 CONTINUE
410 PRINT 85,KDP,NDUNE(KDP,3),ADUMP(KDP,14)
      ADUMP(KDP,13),ADUMP(KDP,4),ADUMP(KDP,15),ADUMP(KDP,1)
      ADUMP(KDP,16)
      PRINT 90,ADUMP(KDP,2),ADUMP(KDP,17),
      ADUMP(KDP,3),ADUMP(KDP,18),
      ADUMP(KDP,4),ADUMP(KDP,19),
      ADUMP(KDP,5),ADUMP(KDP,20),
      ADUMP(KDP,6),ADUMP(KDP,21),
      ADUMP(KDP,7),ADUMP(KDP,22),
      PRINT 110,ADUMP(KDP,8),ADUMP(KDP,23),
      ADUMP(KDP,9),ADUMP(KDP,24),
      ADUMP(KDP,10),ADUMP(KDP,25),
      ADUMP(KDP,11),ADUMP(KDP,26),
      ADUMP(KDP,12),ADUMP(KDP,27),
      IF (STAT(KDP).EQ.BLK.OR.STAT(KDP).EQ.NULL) PRINT 36,INTEN,KDP
      FORMAT('***DC YOU WISH TO ENTER THIS DATA AS YOUR ',A8,' DP-',
      *12,' DECISION?***')
      IF (STAT(KDP).EQ.INTEN) PRINT 37,KDP
      FCEMAT(1X,32(' '),IMPORTANT:32(' '),1X5(' '),
      *YOU MUST CHECK THE ABOVE ENTRIES WITH THE APPROVED FINAL DP-',1)
      *.,5(' '),
      PRINT 38
      FCEMAT(4X,'ENTER YOUR TEAM SECURITY CODE IF YOU WISH TO COMMIT
      *10 A DECISION: '//,30X,'OR "CONT."')
      READ(IN,125,ERR=10,END=10) INCD
      FCEMAT(A8)
      IF (INCD.EQ.CCNT) RETURN
125

```

[illegible]

```

** 26. MAX BELIAEIL INCENTIVM: , P6.2
** 27. 12. MOTOR QUAL TESTS : , P6.2, 4X,
** 28. MAXIMUM ERROR YDS: , P6.2
** 29. 13. AIRFRAME QUAL TESTS : , P6.2, 4X,
** 30. MINIMUM ERROR YDS: , P6.2
** 31. 14. LAUNCHER QUAL TESTS : , P6.2, 4X,
** 32. MAX ERROR INCENTIVE : , P6.2
** 33. 15. FIRE CCNTROL QUAL TSTS: , P6.2, 4X,
** 34. WEEK FOR IOT 10 : , P6.2
** 35. PCENAT(14X) : , P6.2
** 36. 14X, AS TEAM : 12, 1X, A8, INPUT TO THE CONTRACTOR. }
** 37. IF (NR.EQ.0) PRINT 425
** 38. PCENAT(
** 39. 14X, *****SEE YOUR MONITOR IF YOU HAVE A PROBLEM.*****
** 40. IF (STAT(KDE) .EQ. FINAL) KDE=KDE+1
** 41. PROP=INTEN
** 42. IF (STAT(KDE) .EQ. FINAL) KDE=KDE+1
** 43. PRINT 305
** 44. FORMAT( ' *****WARNING*****YOUR CODE WAS NOT ENTERED CORRECTLY. ' )
** 45. PAUSE ' :PRESS <ENTER>>
** 46. RETURN
** 47. END
** 48. SUEROUTINE PLTSC
** 49. *****
** 50. CCOMMON/PLOTVA/YEASE,YCURVE
** 51. DIMENSION XBASE(12), YCURVE(12,12)
** 52. REAL RBLK, CHARA(12), J,J',K,K',M,M',N,N',
** 53. INTEGER HORIZ, C,C',D,D',E,E',F,F',G,G',H,H',I,I',
** 54. INTEGER QUES, L,L',BLK, PRD, QUOTE, LINE(55,22), VERT,
** 55. CCM, AST, PLUS, DASH, GREAT, COLON, SEMICO,
** 56. PPTPCT, AT, ANFER, T, T',
** 57. INTEGER SPRIT, CHARKY(20)
** 58. REAL*8 DV COST, DEV COST,
** 59. REAL*8 TSTCMP, TESTCOMP,
** 60. REAL*8 RLIBIL, RELIABIL,

```



```

C
      XMIN=0.0
      LINE(51,21)=VERT
      IF(YMX-NE.99999.9)GOTO 62
C DETERMINE X AND Y SET SIZE, HIN, AND MAX FOR SCALE GENERATION
      YMAX=0.0
      YMIN=0.0
      IF(PCTFLG-NE.0.0)GOTO 63
      DO 60 INCRE=1,11
      DO 60 JNCRE=1,11
      *
      *
      *
      *
      IF(YCURVE(JNCRE,INCRE).GT.YMAX)
      YMAX=YCURVE(JNCRE,INCRE)
      IF(YCURVE(JNCRE,INCRE).LT.YMIN)
      YMIN=YCURVE(JNCRE,INCRE)
      IF(XBASE(JNCRE).GT.XMAX)
      XMAX=XBASE(JNCRE)
      IF(XBASE(JNCRE).LT.XMIN)
      XMIN=XBASE(JNCRE)
      *
      *
      *
      *
      CONTINUE
      GOTO 61
C THE NEXT TWO LINES SET THE Y-AXIS VALUES FOR THE SCALED ROUTINE
62
      IF(YMX-NE.99999.9)YMIN=YMX
      IF(YMX-NE.99999.9)YMAX=YMX
C APPROXIMATE THE AXIS SCALES.
61
      YSIZE=ABS(YMAX-YMIN)
      IF(YSIZE.EQ.0.0)YSIZE=1.0
      YSCALE=(YSIZE/20.0)
      YSIZE=ABS(YMAX-YMIN)
      IF(XSIZE.EQ.0.0)XSIZE=1.0
      XSCALE=(XSIZE/20.0)
      GOTO 63
C ESTABLISH THE X AND Y SCALES AS STANDARD MODULUS.
C64
      YLOG=ALOG10(YSCALE)
      YLOG=YLOG
      DYLOG=ABS(YLOG-XYLOG)
      DO 74 JITIG=1,20
      BIT=(21.0-JITIG)/2.0
      SIG=YLOG/(ABS(YLOG))
      IF(DYLOG.GE.AICG10(BIT))GO TO 76
      CCNTINUE
C74

```

```

C76      YSCALE=10**((IYLOG+A LOG10(BIT))*SIG)
C      IF(YSCALE.LT.(0.0005)) YSCALE=0.0005
C      MAKE THE AXIS MIN AND MAX INTEGER MULTIPLES OF THE SCALE
C      YMIN=SIGN (INT (ABS (YMIN/YSCALE)) +1.0,YMIN)
C      YMAX=SIGN (INT (ABS (YMAX/YSCALE)) +1.0,YMAX)
C63      IF (PCTPLG.NE.0.0) YSCALE=0.5
C      DEVIOP LINE AND COLUMN MATRIX OF CHARACTERS
C      'CHARA' CHARACTERS REPRESENT DATA POINTS:
C      = COST DATA
C      T = FLIGHT TEST COMPLETION DATE DATA
C      R = RELIABILITY DATA
C      Q = ACCURACY DATA
C      % = TOTAL FEE PERCENTAGE
C      $ = TOTAL CCST
C      * = REFLECTIVENESS INDEX
C      & = MULTIPLE CURVE INTERSECTION
C      CHARA {1} =CENT
C      CHARA {2} =CENT
C      CHARA {3} =T
C      CHARA {4} =T
C      CHARA {5} =R
C      CHARA {6} =R
C      CHARA {7} =AT
C      CHARA {8} =AT
C      CHARA {9} =PTPCT
C      CHARA {10} =DCL
C      CHARA {11} =AST
C      CHARA {12} =AMPER
C      YFAC=(21-INERW)*YSCALE+YMIN
C      INROW=INERW+1
C      DO 70 IYNM=1,12
C      DO 70 IYNM=1,11
C      DO 70 INERW=1,21
C      DO 70 LINECL=1,51
C      IF (AES(YCURVE(IYNM,IXNM))-YFAC).LE.(0.5*YSCALE).AND.(IXNM-1)*5.EQ.(
C      *LINECL-1).AND.(LINE(IYNM,IXNM).NE.BLK.OR.LINE(LINECL,INROW).NE.
C      *HORIZ.OR.LINE(LINECL,INROW).NE.VERT))
C      *LINE(LINECL,INROW)=AMPER
C      IF (AES(YCURVE(IYNM,IXNM))-YFAC).LE.0.5*YSCALE.AND.
C      *(IYNM-1)*5.EQ.(LINECL-1).AND.LINE(LINECL,INROW).NE.AMPER)

```

```

70 *LINE(LINECL,INFCW)=CHARA(IYNN)
   CCNTINUE
      CHARKY(4)=CENT
      CHARKY(6)=T
      CHARKY(8)=B
      CHARKY(10)=AT
      CHARKY(12)=PRTPCT
      CHARKY(14)=DOL
      CHARKY(16)=AST
      CHARKY(18)=AMPER
      CALL FRTCHS('CLRSCRN')
      DO 80 JNROW=1,10
      JNROW=(JNROW-1)*2
      KROW=JROW+1
      YFACT=(21-KROW)*YSCALE+YMIN
      PRINT 85,YFACT,(LINE(INCREH,KROW),INCREH=1,51)
85 FCEMAT(F8.2,T10,51A1)
      KROW=KROW+1
      IF (KRCW-GE.4-AND-KROW-LE.18) PRINT 89,(LINE(INCREM,KROW),INCREM=1,5
      *1) CHARKY(KROW),CHARDI(KROW)
      FCEMAT(T10,51A1,2X,A1,A8)
89 IF (KRCW-LT.4-OR-KROW-GT.18) PRINT 90,(LINE(INCREM,KRCW),INCREM=1,51
      *)
      FCEMAT(T10,51A1)
80 CCNTINUE
      PRINT 95,YMIN,(LINE(KNCREM,21),KNCREM=1,51)
      IF (XBASE(11).GE.100.0) GOTO 147
      IF (XBASE(11).GE.10.0) GOTO 148
      PRINT 145,(XBASE(INCRE),INCRE=1,11)
      FCEMAT(T8,1F5.2,/)
      PAUSE ':PRESS <<ENTER>> TO CONTINUE.'
      RETURN
147 PRINT 149,(XBASE(INCRE),INCRE=1,11)
149 FCEMAT(T6,1F5.0,/)
      PAUSE ':PRESS <<ENTER>> TO CONTINUE.'
      RETURN
148 PRINT 150,(XBASE(INCRE),INCRE=1,11)
150 FCEMAT(T7,1F5.1,/)
      PAUSE ':PRESS <<ENTER>> TO CONTINUE.'

```

```

C.....GENERATE ENTRY CONDITIONS FOR THE FOLLOWING 'ENTRY ROUTINES'
ENTRY PLTSCL(YMN, YMX)
RETURN

```

PCTFLG=0.0

GOTO 10
ENTRY PLTCT(PTCTPLG)

$$Y_{MN} = 0.0$$

GOTC 10

END

SUBROUTINE FSEFF(TOC, FINDEX)

[illegible]
$$\text{EXPECTED AVAILABILITY} = \frac{\text{THEORETICAL AVAIL}}{(\text{1} + \text{AVAILABILITY FACTOR})}$$

$$\text{AVAILABILITY FACTOR}(\text{MAINTAINABILITY FUNDS}) = .29 * \text{EXP}(-.44 * \text{FUNDS})$$


```

C      THE AVFCTR FORMULA IS FROM AN EXPONENTIAL CURVE FITTING SOLU-
C      TION OBTAINED ON AN HP-41CV STATISTICS PACK MODULE. THE
C      REGRESSION (R**2) COEFFICIENT ACHIEVED WAS 1.00.
C      AVFCTR=0.29*(1/(EXP(0.44*Y(1))))
C      AVAIL=THAVL/(1+AVFCTR)
C      IF(AVAIL.LT.0.) AVAIL=0.
C      IF(AVAIL.GT.100.) AVAIL=100.
C      SSK = .333*(1+AVAIL*(1.8523*(1/(EXP(.00634*Y(4))))))
C      IF(SSK.GT.1.) SSK=1.
C      START DATE FOR PRODUCTION OF BLOCK 1 LOTS IS COMPLETION OF THE
C      FLIGHT TESTS, Y(2), IN FLY-BEFORE-BUY. IT IS DELAYED BY 3
C      MONTHS FOR
C      DELAY=8.0*(1.0+SIN(MTDOL-1.1)*3.141593/2.0)
C      START1=(IN1((Y(2)+DELAY)/13.0)+1.0)*13.0
C      FSE=100.0*(100.0-SSPK)*2.0/100.0
C      IF(FSE.GT.85.) FSE=85.
C      CALL SCHEDU(FSE,AFSE,START1)
C      FINDEX = 53.3016+1.44635*AFSE-0.953836*(TOC+40.)
C      FINDEX = (0.993-A LOG((40.+TOC)/AFSE))/0.01066
C      RETURN
C      END
C      SUBROUTINE SCHEDU(FSE,AFSE,START1)
C      REAL Bigger,AFSE,SUMFSE,ZFSE,XRAY(20),ZEBRA(20),FSE
C      DATA XRAY/50.0,50.0,50.0,50.0,50.0,50.0,50.0,50.0,50.0,50.0,
C      DO 50 I=1,20
C      ZEBRA(I)=0.0
C      ZPSE=0.0
C      NEEK = 183 + 26 * I
C      IF(START1.GT.NEEK) GOTO 40
C      DO 10 J=1,10
C      ZWEEK = START1 +13.0* (J-1)
C      IF(ZWEEK.LE.NEEK) ZFSE = ZFSE + FSE/10.0
C      CONTINUE
C      CONTINUE
C      IF(ZFSE.GT.80.0) ZFSE = 80.0
C      ZEBRA(I) = ZFSE
C      CCNTINUE
C      10
C      40
C      50

```

```

C
      SUMFSE = 0.0
      DO 80 I=1,20
        BIGGER = XRAY(I)
        IF (XRAY(I) - LT.ZEBRA(I))
          BIGGER = ZEBRA(I)
        SUMFSE = SUMFSE + BIGGER
      80 CCNTINUE
      AFSE = SUMFSE / 20.0
      RETURN
      ENF
C*****
C
      SUBROUTINE SNSIIV
      COMMON REST(48), TAB(20,25,3), CTEST(3), CTOH(3), XMIS(2), ISUBS(48)
      IFAC(48), IPD(48), INT(48), ITERM(48), IDDI(48), CCMAT(48)
      COMMON BERS(48), TRESU(48), RESB(48), QTY(48), IITYPE(48)
      PERC(48), TDUE(48), DEFC(3), Y(8), RESA(48),
      IPRC(48), RESC(48), YF(48), CT(48),
      TSTAR(48), RCL(6,5), ALN(48), CUN(30), ISW(3)
      COMMON TFIN(48), TMIN(48), THIN(48), RESMAX(30), AK(30),
      ITAB(25,30), NN(8), N(8), INDX(8)
      RATIO(2), AN(9), ADJ(2,8), PDR(6),
      PER(6), PVR(6), FCFIN,X,NTEMP(25), COV
      COMMON CPUHAX, CEUMIN, FCFIN,X,NTEMP(25), COV
      IACT(48,2), IPC(48,5), IFIG, KDP, MF
      TDS(48,2), JMIN,TEAM
      COMMON NC,ICOF(6,KP,KKPP,I,TEAM
      COMMON RSMAX, RSHIN,FRSMPX,ESMAX,ESMIN,FESMIN,CDMAX,CDMIN,
      EFCMIN,TDMAX,TTCMIN,FTDMIN,QT(25),YC(6),CTC(8),DD(32),MIS(9)
      COMMON PLOTVA/SAVSEN, SAVDC
      COMMON STORED/STAT(5), ADUMP(5,28), DPCCST,MAXDP
      REAL XFAC(11),RBLK/,YFEE(8),FRATIO(8),FYC(5)
      FOLLOWING SETS,CF,REAL,DATA CREATE LABELS FOR THE TABLES.
      REAL#8 AREAL,AREF2,AREAF3
C

```

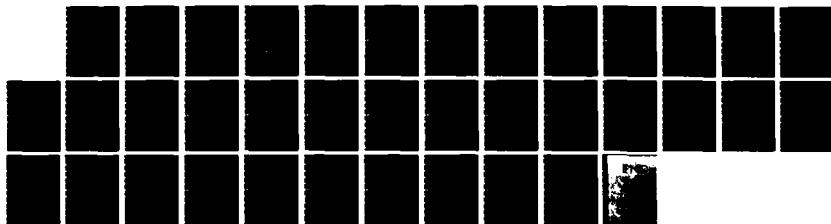
AD-A140 709

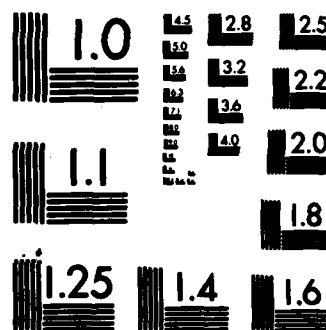
PROJMG FORTRAN: AN INTERACTIVE COMPUTER PROGRAM FOR
USE WITH THE DEFENSE MANAGEMENT SIMULATION EXERCISE(U)
NAVAL POSTGRADUATE SCHOOL MONTEREY CA G W SCHULTZ
MAR 84 F/G 9/2

4/4

UNCLASSIFIED

NL





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


```

C
** 10. AIRFRAME QUAL TESTS
** 11. LAUNCHER/USE QUAL TESTS
** 12. FIRE CONTROL QUAL TESTS
*201,1,*****0. NONE*****
22: IMPACT ACCURACY INCENTIVE'././
23: DEPLOYMENT DATE RANGE'././
READ(IN,ERR=20,END=20) SENSTY
IF(SENSTY.EQ.0) RETURN
CALL PRTCHS('CLRSCHN ')
RATIO2=1.
CALL ZHW
DO 2600 ISEN=1,11
  WTEMP(1)=NDP
  DO 150 IVER2=3,NDP
    CALL GET(IVER2)
    GOTO(200,300,400,500,600,700,800,900,1000,1100,1200,1300,1400,
    *1500,1600,1700,1800,1900,2000,2100,2200,2300,2400),
    *SENSTY
  CONTINUE
  GOTO 20
  IF( ISEN.GT.1) GOTO 220
  AREA1=MAINT
  AREA2=ENGIN
  AREA3=FUNDS
  PRINT 210
  FORMAT(' INPUT THE VALUES TO VARY MAINT ENG FUNDS BETWEEN.',
  *./, 'WHAT IS THE LOWER VALUE: >1.5 ?')
  READ(IN,ERR=200,END=200) LOWER
  IF(LOWER.LT.1.5) LOWER=1.5
  PRINT 215
  FORMAT(' WHAT IS THE UPPER VALUE: <4.0 ?')
  READ(IN,ERR=200,END=200) UPPER
  IF(UPPER.GT.4.0) UPPER=4.0
  IF(LOWER.GT. UPPER) GOTO 200
  RANGE=(ABS(UPPER-LOWER))*1
  CTC(8)=LOWER+RANGE*(ISEN-1)
  IF(CTC(8).GT.UPPER) GOTO 2620
  SENVAR=CTC(8)
  GOTO 2440
  CCCTINUE
  IF( ISEN.GT.1) GOTO 320

```

```

310 AREA1=VALUE
    AREA2=ENG
    AREA3=FUNDS
    PRINT
    310 THE VALUES TO VARY VALUE ENG FUNDS BETWEEN.
    PCHMAT(1) IS THE LOWER VALUE. MUST BE 1.75 TO 2.2 MAY (?)
    *//. WHAT IS THE LOWER VALUE. MUST BE 1.75 TO 2.2 MAY (?)
    READ(1)ERR=3.00,END=300,LOWER
    PRINT(1)ERR=3.00,END=300,LOWER=0.5
    315 FORMAT(1) IS THE UPPER VALUE. MUST BE LESS THAN 2.2(?)
    C HEAD(1)ERR=3.00,END=300,UPPER
    IF(UPPER.GT.3.0)UPPER=3.0
    IF(LOWER.GT.UPPER)GOTO 300
    RANGE=(ABS(UPPER-LOWER))*0.1
    320 CIC(7)=LOWER+RANGE*(ISEN-1)
    IF(CIC(7).GT.UPPER)GOTO 2620
    SENVAR=CIC(7)
    GOTO 2440
    400 CONTINUE
    IF(KDP.GI.3.OR.HIS(6).EQ.2.OR.HIS(6).EQ.4)GOTO 20
    IF(ISEN.GI.1)GOTO 420
    AREA1=PARAL
    AREA2=DEVEL
    AREA3=FUNDS
    PRINT
    410 PCHMAT(1) IS THE LOWER VALUE?
    *//. WHAT IS THE LOWER VALUE?
    READ(1)ERR=400,END=400,LOWER
    IF(LOWER.LE.0)LOWER=0.1
    415 PRINT(1)ERR=400,END=400,LOWER=0.1
    PCHMAT(1) IS THE UPPER VALUE?
    HEAD(1)ERR=400,END=400,UPPER
    IF(UPPER.GT.400)UPPER=400
    RANGE=(ABS(UPPER-LOWER))*0.1
    420 CIC(5)=LOWER+RANGE*(ISEN-1)
    IF(CIC(5).GT.UPPER)GOTO 2620
    SENVAR=CIC(5)
    GOTO 2440
    500 CONTINUE
    IF(ISEN.GI.1)GOTO 520

```

```

510 AREA1=MOTOR
    AREA2=HELIA
    AREA3=UNDRI
    PRINT 510 THE VALUES TO VARY MOTOR RELIABILITY BETWEEN.
    *// * WHAT IS THE LOWER VALUE?
    READ (IN,*,ERR=500,IND=500) LOWER
    PRINT 515
    PCENAI(, WHAT IS THE UPPER VALUE?)
    READ (IN,*,ERR=500,IND=500) UPPER
    IF (UPPER.GT.100.) UPPER=100.
    IF (LOWER.GT.UPPER) GOTO 500
    RANGE=(ABS(UPPER-LOWER))*.1
    IF (LOWER+RANGE*(ISEN-1)
    YC(1)=LOWER+RANGE*(ISEN-1)
    IF (YC(1).GT.UPPER) GOTO 2620
    SENVAR=YC(1)
    GOTO 2440
    CONTINUE
    IF (ISEN.GT.1) GOTO 620
    AREA1=AIRFR
    AREA2=HELIA
    AREA3=UNDRI
    PRINT 610
    PCENAI(, INPUT THE VALUES TO VARY AIRFRAME RELIABILITY BETWEEN.
    *// * WHAT IS THE LOWER VALUE?
    READ (IN,*,ERR=600,IND=600) LOWER
    PRINT 615
    PCENAI(, WHAT IS THE UPPER VALUE?
    READ (IN,*,ERR=600,IND=600) UPPER
    IF (UPPER.GT.100.) UPPER=100.
    IF (LOWER.GT.UPPER) GOTO 600
    RANGE=(ABS(UPPER-LOWER))*.1
    IF (LOWER+RANGE*(ISEN-1)
    YC(2)=LOWER+RANGE*(ISEN-1)
    IF (YC(2).GT.UPPER) GOTO 2620
    SENVAR=YC(2)
    GOTO 2440
    CONTINUE
700

```



```

IF (ISEN.GT.1) GOTO 720
ARIA1=LAUNC
ARIA2=RELI
ARIA3=UNDELT
PRINT 710
FORMAT(' INPUT THE VALUES TO VARY LAUNCHER RELIABILITY BETWEEN.
*./.' WHAT IS THE LOWER VALUE?')
READ (IN,*,ERR=700,END=700) LOWER
IF (LOWER.LE.0) LOWER=.0
PRINT 715
FORMAT(' WHAT IS THE UPPER VALUE?')
READ (IN,*,ERR=700,END=700) UPPER
IF (UPPER.GT.100) UPPER=100.
IF (LOWER.GT.UPPER) GOTO 700
RANGE=(ABS(UPPER-LOWER))
YC(3)=LOWER+RANGE*.1*(ISEN-1)
IF (YC(3).GT.UPPER) GOTO 2620
IF (YC(3).LE.0) GOTO 2620
SENVAR=YC(3)
GOTO 2440
CONTINUE
IF (ISEN.GT.1) GOTO 820
ARIA1=PIREC
ARIA2=ACCUR
ARIA3=UNDELT
PRINT 810
FORMAT(' INPUT THE VALUES TO VARY FIRE CONTRCL IMPACT BETWEEN.
*./.' WHAT IS THE LOWER VALUE MUST BE GREATER THAN 0?')
READ (IN,*,ERR=800,END=800) LOWER
IF (LOWER.LE.0) LOWER=.0
PRINT 815
FORMAT(' WHAT IS THE UPPER VALUE MUST BE LESS THAN 100?')
READ (IN,*,ERR=800,END=800) UPPER
IF (UPPER.GT.250) UPPER=250.
IF (LOWER.GT.UPPER) GOTO 800
RANGE=(ABS(UPPER-LOWER))*.1
YC(4)=RANGE*(ISEN-1)
IF (YC(4).GT.UPPER) GOTO 2620
IF (YC(4).LE.0) GOTO 2620
SENVAR=YC(4)

```

```

900      GOTO 2440
        CCNTINUE
        IF (ISEN.GT.1)GOTO 920
        AREA1=GUIDE
        AREA2=ACCU
        AREA3=UNDEL
        PRINT 910
        FORMAT(' INPUT THE VALUES TO VARY GUIDANCE IMPACT ERROR BETWEEN.
        *./.' WHAT IS THE LOWER VALUE?')
        READ(IN,*,ERR=900,END=900) LOWER
        READ(IN,*,ERR=900,END=900) LOWER
        PRINT 915
        FORMAT(' WHAT IS THE UPPER VALUE?')
        READ(IN,*,ERR=900,END=900) UPPER
        IF (UPPER.GT.250) UPPER=250.
        IF (LOWER.GT.UPPER) GOTO 900
        RANGE=(ABS((UPPER-LOWER)*.1))
        YC(5)=LOWER+(RANGE*(ISEN-1))
        IF (YC(5).GT.UPPER) GOTO 2620
        SENVAR=YC(5)
        GOTO 2440
        CCNTINUE
        IF (ISEN.GT.1)GOTO 1020
        AREA1=MOTOR
        AREA2=QUAL
        AREA3=TEST
        PRINT 1010
        FORMAT(' INPUT THE VALUES TO VARY MOTOR QUAL TESTS BETWEEN.
        *./.' WHAT IS THE LOWER NUMBER OF TESTS:
        READ(IN,*,ERR=1000,END=1000) LOWER
        READ(IN,*,ERR=1000,END=1000) LOWER
        PRINT 1015
        FORMAT(' WHAT IS THE UPPER NUMBER: <40 ?')
        READ(IN,*,ERR=1000,END=1000) UPPER
        IF (UPPER.GT.40) UPPER=40.
        IF (LOWER.GT.UPPER) GOTO 1000
        RANGE=(INT((UPPER-LOWER)*.1))
        RANGE=INT(RANGE*.1)
        QT(2)=LOWER+INT(RANGE*(ISEN-1))
        IF (QT(2).GT.UPPER) GOTO 2620

```

```

1100      SENVAR=QT(2)
      GOTO 2440
      CONTINUE
      IF (ISEN.GT.1) GOTO 1120
      AREA1=ALINE
      AREA2=QUAL
      AREA3=TEST
      PRINT 1110
      FORMAT(' INPUT THE VALUES TO VARY AIRFRAME QUAL TESTS BETWEEN.',
      *./.' WHAT IS THE LOWER NUMBER OF TESTS: >3 ?')
      READ(LINE,*) ERR=1100,END=1100,LOWER
      IF (LOWER.GT.1.3.) LOWER=3.
      PRINT 1115
      FORMAT(' WHAT IS THE UPPER NUMBER: <9 ?')
      READ(LINE,*) ERR=1100,END=1100,UPPER
      IF (UPPER.GT.9.) UPPER=9.
      IF (LOWER.GT.UPPER) GOTO 1100
      RANGE=(INT((UPPER-LOWER)*.1))
      RANGE=LI:1..RANGE=1.
      QT(3)=LOWER+(INT(RANGE)*(ISEN-1))
      1120      GOTO 2620
      SENVAR=QT(3)
      GOTO 2440
      CONTINUE
      IF (ISEN.GT.1) GOTO 1220
      AREA1=LAUNC
      AREA2=QUAL
      AREA3=TEST
      PRINT 1210
      FORMAT(' INPUT THE VALUES TO VARY LAUNCHER QUAL TESTS BETWEEN.',
      *./.' WHAT IS THE LOWER NUMBER OF TESTS: >2 ?')
      READ(LINE,*) ERR=1200,END=1200,LOWER
      IF (LOWER.GT.1.2.) LOWER=2.
      PRINT 1215
      FORMAT(' WHAT IS THE UPPER NUMBER: <6 ?')
      READ(LINE,*) ERR=1200,END=1200,UPPER
      IF (UPPER.GT.6.) UPPER=6.
      IF (LOWER.GT.UPPER) GOTO 1200
      RANGE=(INT((UPPER-LOWER)*.1))
      RANGE=LI:1..RANGE=1.

```

295

```

1420      RANGE=(INT((UPPER-LOWER)*.1))
          IF (RANGE.LT.1.) RANGE=1.
          QT(5)=LOWER+INT(RANGE*(ISEN-1))
          IF (QT(5).GT.UPPER) GOTO 2626
          SENVAR=QT(5)
          GOTO 2440
1500      CONTINUE
          IF (ISEN.GT.1) GOTO 1520
          AREA1=PLIGHT
          AREA2=TEST
          AREA3=UNDEL
          PRINT 1510
1510      FORMAT(' INPUT THE VALUES TO VARY FLIGHT TESTS BETWEEN'
          *//, ' WHAT IS THE LOWER NUMBER OF TESTS: >10 ?')
          READ(IN,*ERR=1500,END=1500) LOWER
          IF (LOWER.LT.10.) LOWER=10.
          PRINT 1515
1515      FORMAT(' WHAT IS THE UPPER NUMBER OF TESTS: <25 ?')
          READ(IN,*ERR=1500,END=1500) UPPER
          IF (UPPER.GT.25.) UPPER=25.
          RANGE=(ABS(UPPER-LOWER))*.1
          IF (LOWER.GT.UPPER) GOTO 1500
          QT(1)=LOWER+INT(RANGE*(ISEN-1))
          IF (QT(1).GT.UPPER) GOTO 2626
          SENVAR=QT(1)
          GOTO 2440
1600      CONTINUE
          IF (ISEN.GT.1) GOTO 1620
          AREA1=DEVEL
          AREA2=COST
          PRINT 1625
1625      FORMAT(' DO YOU WANT TO VARY THE MAXIMUM OR MINIMUM DEVELOPMENT CC
          *ST?',' ENTER EITHER MAXIMUM OR MINIMUM.')
          READ(IN,1630,ERR=1600,END=1600) LABEL
          AREA3=LABEL
          PRINT 1630
1630      FORMAT(' AS')
          PRINT 1610
1610      FORMAT(' INPUT THE LOWER VALUE FOR 'AB,' DEVELOPMENT COST. ')
          READ(IN,*ERR=1600,END=1600) LOWER

```

```

1615      IF (LOWER.LE.0) LOWER=.01
          PRINT 1615, 'THE UPPER COST FOR ', A8, '?'
          READ (IN, *ERR=1600, END=1600) UPPER
          IF (UPPER.GT.125.) UPPER=125.
          IF (LOWER.GT.UPPER) GOTO 1600
          RANGE=(ABS(UPPER-LOWER))
          SENVAR=LOWER+RANGE*.1*(ISEN-1)
          IF (ISEN.EQ.1.AND.SENVAR.EQ.0.) PRINT 2535
          IF (SENVAR.EQ.0.) GOTO 2600
          IF (LABEL.EQ.HAX) CDHAX=SENVAR
          IF (LABEL.EQ.MIN) CDHIN=SENVAR
          GOTO 2440
1700      CCNTINUE
          IF (ISEN.NE.1) GOTO 1720
          AREA1=DEVEL
          AREA2=COST
          AREA3=INCENT
          PRINT 1710, 'THE RANGE TO VARY INCENTIVE % OF DEVELOPMENT COST?'
          READ (IN, *ERR=1700, END=1700) LOWER
          IF (LOWER.LE.0) LOWER=.01
          PRINT 1715, 'THE UPPER VALUE?'
          READ (IN, *ERR=1700, END=1700) UPPER
          IF (UPPER.GT.100.) UPPER=100.
          IF (LOWER.GT.UPPER) GOTO 1700
          FCDHIN=LOWER+(ABS(UPPER-LOWER))*.1*(ISEN-1)
          IF (FCDHIN.LE.0.) PRINT 2535
          IF (FCDHIN.LE.0.) GOTO 2600
          SENVAR=FCDHIN+FTDIN+FRSHAX+FESMIN
          PRINT 1720, 'THE UPPER VALUE?'
          IF (DELPER.LE.15.) GOTO 2440
          RATIO2=(15.-FCDHIN)/(FTDIN+FRSHAX+FESMIN)
          FTDIN=FRSHAX+RATIO2
          FESMIN=FRSHAX+RATIO2
          GOTO 2440
1800      CCNTINUE

```

```

1825 IF (ISEN.GT.1)GOTO 1820
      AREA1=FLIGHT
      AREA2=TESTC
      PRINT 1825
      FORMAT(' DO YOU WANT TO VARY THE EARLIEST OR LATEST FLIGHT TEST'
            *//.' COMPLETION DATE? ENTER EITHER LATE OR EARLY.')
      READ(IN,1630,ERR=1800,END=1800) LABEL
      AREA3=LABEL
      PRINT 1810, LABEL
      FORMAT(' INPUT THE RANGE TO CHANGE ',A8,' COMPLETION DATE.'
            *//.' WHAT IS THE LOWER VALUE?',1800) LOWER
      READ(IN,1630,ERR=1800,END=1800) LOWER
      PRINT 1815, LABEL
      FORMAT(' WHAT IS THE UPPER VALUE FOR ',A8,' COMPLETION?')
      READ(IN,1630,ERR=1800,END=1800) UPPER
      IF (UPPER.GT.300.)UPPER=300.
      IF (LOWER.GT.UPPER)GOTO 1800
      RANGE=(ABS(UPPER-LOWER))*1
      IF (RANGE.LT.1.)RANGE=1
      SENVAR=LOWER+(INT(RANGE*(UPPER-LOWER)))
      IF (SENVAR.GT.UPPER)GOTO 2620
      IF (LABEL.EQ.LATE)TDHAI=SENVAR
      IF (LABEL.EQ.EARLY)TDHMIN=SENVAR
      GOTO 2440
1900 CCNTINUE
      IF (ISEN.NE.1)GOTO 1920
      AREA1=FLIGHT
      AREA2=TESTC
      AREA3=INCEN
      PRINT 1910
      FORMAT(' INPUT THE UPPER VALUE FOR DELIVERY INCENTIVE.')
      READ(IN,1630,ERR=1900,END=1900) UPPER
      IF (UPPER.GT.100.)UPPER=100.
      PRINT 1915
      FORMAT(' WHAT IS THE LOWER VALUE FOR DELIVERY INCENTIVE?')
      READ(IN,1630,ERR=1900,END=1900) LOWER
      IF (LOWER.LT.0)LOWER=0
      IF (LOWER.GT.UPPER)GOTO 1900
      FIDMIN=LOWER+(ABS(UPPER-LOWER))*1*(ISEN-1)
1920

```

```

2000 IF (ISEN.EQ.1.AND.FTDMIN.LE.0.) PRINT 2535
      IF (FTDMIN.LE.0.) GOTO 2600
      DELPER=PCDMIN+FTDMIN+FRSHAX+FESMIN
      SENVAR=FTDMIN
      IF (DELPER.LE.15.) GOTO 2440
      RATIO2=(15.-FTDMIN)/(PCDMIN+FRSHAX+FESMIN)
      PCDMIN=PCDMIN*RATIO2
      FRSHAX=FRSHAX*RATIO2
      FESMIN=FESMIN*RATIO2
      GOTO 2440
2005 CONTINUE
      IF (ISEN.NE.1) GOTO 2020
      AREA1=RELIA
      PRINT 2005
      FORMAT(' DO YOU WANT TO VARY THE UPPER OR LOWER LEVEL OF RELIABILT
        *Y?',' INPUT HIGH OR LOW?')
      READ(IN,1630,ERR=2000,END=2000) LABEL
      AREA2=LABEL
      AREA3=UNDEL
      PRINT 2010,LABEL
      IF (WHAT IS THE UPPER VALUE FOR 'A8,' RELIABILITY?')
      READ(IN,*,ERR=2000,END=2000) UPPER
      IF (UPPER.GT.100.) UPPER=100.
      PRINT 2015,LABEL
      IF (WHAT IS THE LOWER VALUE FOR 'A8,' RELIABILITY?')
      READ(IN,*,ERR=2000,END=2000) LOWER
      IF (LOWER.LE.0) LOWER=.01
      IF (LOWER.GT.UPPER) GOTO 2000
      SENVAR=LOWER+(ABS(UPPER-LOWER))*1*(ISEN-1)
      IF (ISEN.EQ.1.AND.(SENVAR.GT.100..OR.SENVAR.LT.0.)) PRINT 2535
      IF (SENVAR.GT.100..OR.SENVAR.LT.0.) GOTO 2440
      IF (LABEL.EQ.HIGH) RSHIN=SENVAR
      IF (LABEL.EQ.LOW) RSHIN=SENVAR
      GOTO 2440
2100 CONTINUE
      IF (ISEN.NE.1) GOTO 2120
      AREA1=RELIA
      AREA2=INCEM
      AREA3=UNDEL
      PRINT 2110

```



```

2110 FORMAT(' INPUT THE UPPER VALUE OF INCENTIVE % FOR RELIABILITY.')
      READ (IN,*,ERR=2100,END=2100) UPPER
      IF (UPPER.GT.100.) UPPER=100.
      PRINT 2115
      FCENAT(' WHAT IS THE LOWER VALUE?')
      READ (IN,*,ERR=2100,END=2100) LOWER
      IF (LOWER.LE.0) LOWER=0
      IF (LOWER.GT.UPPER) GOTO 2100
      PRSMAX=LOWER+(ABS(UPPER-LOWER))*1*(ISEN-1)
      DEUPER=PCDHIN+FTDHIN+FRSHAX+FESHIN
      SENVAR=PRSMAX
      IF (FRSHAX.LE.0.) PRINT 2535
      IF (FRSHAX.LE.0.) GOTO 2600
      IF (DEUPER.LE.15.) GOTO 2440
      RATIO2=(15.-FRSHAX)/(PCDHIN+FRSMAX+FESHIN)
      FCDHIN=PCDHIN*RATIO2
      FTDHIN=FTDHIN*RATIO2
      FESHIN=FESHIN*RATIO2
      GOTO 2440
2200 CCNTINUE
      IF (ISEN.NE.1) GOTO 2220
      AREA1=PIREC
      PRINT 2210
      FCENAT(' DO WANT TO CHANGE THE GREATER OR LESSER ACCURACY?')
      * INPUT EITHER GREATER OR LESSER.
      READ (IN,1630,ERR=2200,END=2200)
      AREA2=LABEL
      AREA3=UNDEL
      PRINT 2215,LABEL
      FCENAT(' WHAT IS THE UPPER VALUE FOR 'A8,' ACCURACY?')
      READ (IN,*,ERR=2200,END=2200) UPPER
      IF (UPPER.GT.300.) UPPER=300.
      PRINT 2216
      FCENAT(' WHAT IS THE LOWER VALUE FOR 'A8,' ACCURACY?')
      READ (IN,*,ERR=2200,END=2200) LOWER
      IF (LOWER.LE.0) LOWER=0
      IF (LOWER.GT.UPPER) GOTO 2200
      SENVAR=LOWER+(ABS(UPPER-LOWER))*1*(ISEN-1)
      IF (SENVAR.LE.0.) GOTO 2620
      IF (LABEL.EQ.GREAT) ESMAX=SENVAR

```

```

2300      IF (LABEL.EQ.LOW) ESHIN=SENVAR
          GO TO 2440
          CC CONTINUE
          IP (ISEN.NE.1) GO TO 2320
          AREA1=PIREC
          AREA2=INCEN
          AREA3=UNDR1
          PRINT 2310
2310      *1* WHAT IS THE RANGE TO VARY INCENTIVE % FOR ACCURACY .',./,
          *1* WHAT IS THE UPPER VALUE?',
          *1* ERR=2300, END=2300, UPPER
          READ (IN, *1* UPPER.GI.100.) UPPER=100.
          PRINT 2315
2315      FCENAT(' WHAT IS THE LOWER VALUE?')
          READ (IN, *1* ERR=2300, END=2300) LOWER
          IF (LOWER.LE.0) LOWER=.01
          IF (LOWER.GT.100.) LOWER=100.
          ESHIN=LOWER+(ABS(UPPER-LOWER))*1*(ISEN-1)
          DELPER=FCENHIN+FTDNIN+FRSHAX+FESHIN
          SINVAR=FESHIN
          IF (ISEN.EQ.1.AND.FESHIN.LE.0) PRINT 2535
          IF (FESHIN.LE.0) GO TO 2600
          IF (DELPER.LE.15.) GO TO 2440
          RATIO2=(15-FESHIN)/(FCENHIN+FRSHAX+FESHIN)
          FCDHIN=FCENHIN+RATIO2
          PTDHIN=FTDNIN+RATIO2
          PRSHAX=FRSHAX+RATIO2
          GO TO 2440
2400      CC CONTINUE
          IF (ISEN.NE.1) GO TO 2420
          AREA1=DELI
          AREA2=BLK
          AREA3=UNDR1
          PRINT 2410
2410      *1* WHAT IS THE RANGE TO VARY DELIVERY DATE.',./, WHAT IS THE E
          *1* WHAT IS THE LATER DATE?',
          *1* ERR=2400, END=2400) LOWER
          READ (IN, *1* LOWER.LE.0) LOWER=.01
          PRINT 2415
2415      FCENAT(' WHAT IS THE LATER DATE?')

```

```

2420 READ (IN,*)ERR=2400,END=2400,UPPER
      IF (LOWER:GT:UPPER)GOTO 2400
      RANGE={ABS(UPPER-LOWER)}+1
      IF (RANGE:LI:1.)RANGE=1.
      DD(13)=LOWER+INT(RANGE*(ISRN-1))
      IF (DD(13):GT:UPPER)GOTO 2620
      IF (DD(13):LI:0.)AND (ISRN:EQ:1)PRINT 2535
      SENVAR=DD(13)
      CCNTINUE
2440 CALL INPUT3
      CALL PRESET
      CALL TRADES
      CALL DESRES
      CALL DESRES
      CALL SE
      IF (NTEMP(1) -EQ. 3 -OR. NTEMP(1) -EQ. IDP) GO TO 130
      ITCT=0
      DO 110 N=1,INA
        IF (QTY(H)-0.1) 110,110,115
        IF (ITERH(H)-IDP-1) 110,120,110
        ITCT=ITCT+1
      CCNTINUE
      GO TO 140
      IDP=7
      ITCT=-1
      ITAB(25)=IDP
      CALL MOD9
      CALL REEUN
      CCNTINUE
      CALL REPORT
      CALL PERP
      CALL GET (KDP)
      IF (SENSTY-EQ.1-0 } CTC (8) =SENVAR
      IF (SENSTY-EQ.2-0 } CTC (7) =SENVAR
      IF (SENSTY-EQ.3-0 } CTC (5) =SENVAR
      IF (IVER2-EQ.KDP)GOTO 150
      IF (IVER2-NE.KDP-1)GOTO 155

```

[illegible]

```

C
3315  N=INA
25    Y(2)=FTH(N) 16 2)
      Y(3)=TAB(14 2)
      Y(4)=TAB(14 2)
C 30  . . . ROW 19 CHANGED TO ROW 20 1970 APRIL 1 . . .
      K=4
      H=N(K)
      Y(4)=Y(4)+YF(N)
      Y(5)=C10
C
      PER(1)=CDHAX
      AN(1)=CDHIN
      FYC(1)=FCDHIN
      PER(2)=TDHAX
      AN(2)=TDHIN
      FYC(2)=PTDIN
      AN(3)=RSHAX
      PER(3)=RSHIN
      FYC(3)=PRSHAX
      AN(4)=ESHAX
      PER(4)=ESHIN
      FYC(4)=FESHIN
      PER(5)=CPUMAX
      AN(5)=CPUMIN
      FYC(5)=PCPIN
      SEVA=0.0
      PRATIC(8)=0.0
      Y(1)=Y(1)/100000.0
      Y(3)=Y(3)/100.0
C
C THIS SET OF STEPS IC LABEL 50 CHECKS THE MOST DESIREABLE VALUES
C AN(1) VS ACHIEVED VALUES Y(1), AND LEAST DESIREABLE VALUE PER(1)
C VS ACHIEVED VALUE IN ORDER 10, PLACE THE FEE WITHIN THE LINEAR RATIO
C BOUNDS OF THE DESIRED PER.
      DO 50 I=1,5
      IF(I-10.3) GOTO 48
      IF(Y(I)-AN(I)).GE.0. AND PER(I)-Y(I).GE.0. AND PER(I).NE.AN(I))
      *PRATIO(I)=FYC(I)*(ABS((PER(I)-Y(I))/(PER(I)-AN(I))))
      IF(Y(I)-PER(I).GE.0.) PRATIO(I)=0.0

```


[illegible]

```

3995 *J111G) CONTINUE
                                PAUSE ':PRESS <ENTER>> TO CONTINUE.'
                                RETURN
ENTRY OPTIN
4005 GOTO (4010, 4020, 4030, 4040, 4050, 4060, 4070, 4080, 4090, 4100, 4110, 4120,
    *SINSEI)
4010 CONTINUE
    CIC(8) = SAVSIN(SLX)
    GOTO 4400
4020 CONTINUE
    CIC(7) = SAVSIN(SLX)
    GOTO 4400
4030 CONTINUE
    CIC(5) = SAVSIN(SLX)
    GOTO 4400
4040 CONTINUE
    IC(1) = SAVSIN(SLX)
    GOTO 4400
4050 CONTINUE
    IC(2) = SAVSIN(SLX)
    GOTO 4400
4060 CONTINUE
    IC(3) = SAVSIN(SLX)
    GOTO 4400
4070 CONTINUE
    IC(4) = SAVSIN(SLX)
    GOTO 4400
4080 CONTINUE
    IC(5) = SAVSIN(SLX)
    GOTO 4400
4090 CONTINUE
    CI(2) = SAVSIN(SLX)
    GOTO 4400
4100 CONTINUE
    CI(3) = SAVSIN(SLX)
    GOTO 4400

```


308

```

4230      CONTINUE      DL(13)=SAVSEN(SELX)
4400      CONTINUE      CALL SETUP(KDP)

      ENTRY PLOTST(ISEN)
      RETURN
      END
*****
C      SUECUTINE PRICK
C
      COMMON REST(48),TAB(20,25,3),CTEST(3),CTOH(3),XNIS(2),ISUBS(48)
      IPAC(48),ITIRH(48),IRC(48),IDDT(48),CHAT(48)
      IPD(48),ITIRH(48),RESB(48),Y(8),RESA(48)
      PERC(48),TDRS(48),DEC(3),YF(48),CT(48)
      IPER(48),TDRS(48),ALN(48),CUM(30),ISW(3)
      TSTAN(48),TMIN(48),IQ(48),K(30),INDX(8),PDR(6)
      TPRIN(48),RSHAX(30),NN(8),AM(9),CDN(6),INF,IDP
      ITAB(25),RATIO(8),PVR(6),CPMIN,X,INA,C10,TRADET,TRADER,
      PER(6),CFUMIN,PCPMIN,Y,CDN,TEMP(25),COV,C10,TRADET,TRADER,
      STRADEA,DAH1,DAH2,DEF,SUR,COSTH,COSTP1
      CCHMCHN,COSTP2,C6STON,IACT,ALPHA,PRIOR
      TRADERP,2,TP,JMIN,IC,IACT(48,2),KDP,ME
      TDS(48,FG),KPKPP,TEAM
      NCICOFG,KPKPP,TEAM
      COMMON RSHAX,RSHIN,PRSHAX,ESHAX,ESHIN,FESHIN,CDHAX,CDHIN,(9)
      CFCDMIN,TDNAX,TMIN,PTDIN,QT(25),YC(6),CTC(8),DD(32),NIS(9)
      INTEGER IN/5,
      REAL*4 ANS,CHG,'CHG','E','E','YES','Y','/
      REAL*4 NORH,'NORH','MORE','MORE','/
      REAL*8 GREAT,'GREAT','LESS','LESS','/,'SIZE
      PERTOT=PCDMIN+PTDIN+PRSHAX+FESHIN
      IP(PERTOT,GT,14.999999).AND.PERTOT.LT.15.000001)RETURN
      CALL FETCHS('CIESCHN')
      IP(PERTOT,GT,15.)SIZE=MORE

```

```

5      IF PERTOT-15.1 SIZE=LESS
      PRINT 5 SIZE,PCDHIN,PTDINH,FRSHAX,FRSHIN
      PORNAT(1) *****WARNING*****TOTAL OF FEE PERCENTAGES IS :.A8. THAM
      *15%
      * YOU PREVIOUSLY SUBMITTED ARE: .P5.2. COST FEE INCENTIVE=.P5.2. THAM
      * DELIVERY DATE FEE INCENTIVE=.P5.2. RELIABILITY INCENTIVE FEE
      * .P5.2. ACCURACY INCENTIVE FEE=.P5.2. REDO YOUR INPUT.//)
      PRINT 510
500     PORNAT(1) ENTER THE COST INCENTIVE FEE.)
510     CALL CHANGE(PCDHIN)
      PERTCT=PCDHIN
      IF PERTOT-15.1 GOTO 600
      PRINT 605 PERTCT
605     PORNAT(1) TOTAL OF INCENTIVE FEES (.P5.2.) EXCEEDS 15%.
      CALL EXIT
      GOTO 600
600     DIFF=15.-PERTOT
      PRINT 615,DIFF
615     PORNAT(1) YOU HAVE .P5.2.% OF THE INCENTIVE FEES REMAINING.)
515     PORNAT(1) ENTER THE DELIVERY DATE INCENTIVE FEE.)
      CALL CHANGE(PTDINH)
      PERTCT=PCDHIN+PTDINH
      IF PERTOT-15.1 GOTO 620
      PRINT 605 PERTCT
      CALL EXIT
      GOTO 600
620     DIFF=15.-PERTOT
      PRINT 615,DIFF
520     PORNAT(1) ENTER THE RELIABILITY INCENTIVE FEE.)
      CALL CHANGE(FRSHAX)
      PERTOT=PCDHIN+PTDINH+FRSHAX
      IF PERTOT-15.1 GOTO 625
      PRINT 605 PERTCT
      CALL EXIT
      GOTO 620
625     DIFF=15.-PERTOT
      PRINT 615,DIFF
      PRINT 525

```

525

FORMAT(' ENTER THE ACCURACY INCENTIVE FEE. ')
CALL CHANGE(PESHIN)
PERTOT=PCDHN+ETDHN+PRSNAX+PESHIN
IF(PERTOT.EQ.15.)RETURN
CALL EXIT
GCIO 400
END

APPENDIX H PROJING PROGRAM VARIABLES GLOSSARY

AA	Character string for 'accuracy'
AAA	Recursive TDU sum
ACCUR	Array variable for data file fields 6 through 33
ADJ	Average force
ADUMP	Structure Effectiveness
APSE	Character string for 'airframe'
AIRFR	
AJJ	
AK	
ALN	
ALPHA	
AMPEH	
AN	Character for curve plot:8
ANO	Variable for answer to Yes/No questions
ANS	Negative answer to Yes/No questions
ANT	Variable for answer to Yes/No questions
AREA1	Variable for answer to Yes/No questions
AREA2	Character buffer for creating labels in Sensitivity Table
AREA3	Character buffer for creating labels in Sensitivity Table
AT	Character buffer for creating labels in Sensitivity Table
BIGGER	Character string used in Sensitivity Table label
BLK	Character string for '00000000'
CD	Development cost
CDMAX	Maximum development cost
CDMIN	Minimum development cost
CENT	Character for Curve Plot:4
CHANGE	Subprogram for changing real variables
CHANGI	Subprogram for changing imaginary variables
CHARA	Character string buffer to put characters in plot lines
CHARKY	Array for character key definitions
CHARKDF	

CKPIT

CLAI
CODE
COLON
COMID
COSID
COSIN
COSICT
COSTP1
COSTP2
COV
CPUNAX
CPUHIN
CT
CTC
CTEST
CTOH
CUN
C10
D
DAM1
DAM2
DASHOD
DATPIL
DD
DEC
DECK
DECmin
DEF
DELAY
DELIV
DELPER
DESHES
DESRIT
DEVEL
DYTE
DPTH
D
DOL
DPCOST
DQT1

Subroutine to check the availability
of plot data

Common buffer label
Character for plot curve label;:
Character for plot curve label;:
Cost of development
Cost of manufacture
Cost total
Cost parallel contractor a
Cost parallel contractor b
Cost of production maximum
Cost of production minimum
Cost of tests total
Input desired engineering costs
Cost of system tests
Cost of tests overhead

Character for curve plot;:-
File for LATACODE TEAMXX
File for DATAFILE TEAMXX
Input desired Delivery Date
Decrease for smallest performance factor
Impact
Decrease for smallest performance
factor impact
Decrease for smallest performance
factor impact

Character string for 'delivery' in
sensitivity table

Subroutine to decrease resources
Subroutine to decrease resources
because of tests
Character string for 'development'
in Sensitivity Table

Character \$
Decision Pcnt cost
Differential value of flight tests

DOT2	Differential value of motor qual tests
DOT3	Differential value of airframe qual te
DOT4	Differential value of fire control qua
DOT5	Differential value of guidance qual te
DOT6	Differential value of launcher qual te
DRSHAX	Differential value applied to maximum reliability
EARLY	Character string for determining
FI	sensitivity analysis selection
FIOP1	Effectiveness index
ENGIN	Optimum effectiveness index
ERRSET	Character string for Sensitivity Table
ESMAX	'engineering
ESMIN	Macro Subroutine for resetting error
EXIIS	functions on the IBM
EXP	Accuracy error maximum
PCDMIN	Accuracy error minimum
PCPHIN	Subroutine to choose to stop or reloop
PCION	back into the program
PEPCT	Fee for development cost incentive
FESHIN	Fee for production cost incentive
FINAI	Total of fees
FINISH	Fee for accuracy incentive
FIREC	Label for team data files to identify
FLAG	final of proposal
FLIGHT	Subroutine for submitting final and
FL21	proposal contract proposal
PRSNAX	Character string for Sensitivity Table
PRICHS	'fire control, for Sensitivity Table
PSE	Flag for generating variation to the
FSEFF	Selection Table
PTDMIN	Character string for the Sensitivity
PUNDS	Table
G	Flag for the monitors sessions
GET	Fee for reliability incentive
GREAT	Macro subroutine for clearing the
	screen in the IBM
	Force structure effectiveness
	Subroutine for force structure effectiv
	Fee for test completion date incentive
	Character string for Sensitivity Table
	label 'funds
	Subroutine to place data file buffer data
	in array buffer
	Sensitivity selection label for para-
	meter choice

IRC
 ISAVL2
 ISAVL4
 ISEN
 ISUBS
 ISW
 ITAB
 ITCT
 ITEN
 ITENHB
 ITENH
 ITERR
 ITYPE
 IVER
 IVER2
 IXBASE
 IXNE
 IYNN
 IY2
 IY4
 J
 J1
 J2
 J3
 JA
 JDP
 JITIG
 JIIGA
 JJ
 JMIN
 KA
 KB
 KDP
 KGO
 KGOA
 KGOB
 KIO
 KKPP
 KP
 KTB
 KTR101
 KZ
 KZRN
 L
 LATE

save sensitivity data for flight test
 date
 save sensitivity data for accuracy
 for do loop of 10 iterations in
 Sensitivity Table

team number
 team number
 team number

DP loops of the main program
 DP loops of the Sensitivity achieved
 value loop
 Integer value for plot x-axis
 Integer x-axis minimum value
 Integer y-axis minimum value
 Integer for flight test completion date
 Integer for accuracy
 Integer for do loops

Integer for do loops
 INTEGER FOR DP
 Integer for do loops
 Integer for do loops
 Integer for do loops
 Integer for do loop finding minimum axis
 values
 Integer dc loop variable

Current DF
 Assigned goto
 Assigned goto
 Assigned goto
 Flag for data input at current dp
 Flag for data input at DP-3

Sensitivity selection parameter choice

LAUNC
LLESS
LINEE
LINECL
LINCHN
LL
LOWR
H
HAINI
HEHO
HPTDCL
HINA
HISCT
HK
HOD9
HORE
HOTOR
HR
H6
H
NC
NE
NDUNE
NEHDP
NN
NORM
NR
NSTIP
NT
NTEMP
NULL
OPTIN
P
PAGEIN
PAGES
PANAY
PANIN
PARAI
PD
PDA
PCTFIG
PDR
PERC
PEREC
PERICT

Character string for Sensitivity Table
Character <
Plot line array
Plot line array column
Integer increment

Sensitivity selection parameter choice
Character string for Sensitivity Table
Character string for Sensitivity Table

MINIMUM FLIGHT TEST COST

Subroutine for achieved value calculation
Sensitivity selection parameter choice
Character string for Sensitivity Table
Monitor mode selected

Contractor choice

Team data file first five fields
Newly selected DP level

Table number from which to read data

Temporary buffer for DP

Character string for Sensitivity Table
'Optimum'

Subroutine for page input
Variable for pages to be input

Flag for per plotting percentage plot

Total percentage

GUIDE	Character string for Sensitivity Table
HIGH	guidance
HORIZ	Sensitivity selection label for parameter
I	Choice
IAC	Character for plot lines:
IC	Integer variable for incrementing do loops
ICD	incremental costs overall
ICOLE	incremental development costs
ICOPG	Security code buffer
IDDI	Integer cost of guidance
IDP	Integer delivery date
IDUNE	Initial DP computed for
	Integer variable array for first five
	fields of team data file
IPAC	Count for security code violations
IPAUIT	
IPD	
IPIG	
IKDE	
IN	Integer for do loop variable referring
INA	to DP as stored MPP
INCEN	Integer for do loop incrementing
	Initial accuracy from tables
INCOD	Character string for Sensitivity Table
	incentive
INDEX	Security code input from the team
INP	violations
INP012	Index for selecting table variables
INP013	Subroutine for input of data to
INTEN	Current DP
IPC	Subroutine for input of data from
IPG	table files
IPRC	Proposal status
IQ	
IQUEFP	Integer for question response for
IQUESF	percentage plot
IQUESH	question response: exit
	question response: monitor's main
	menu
IQUESO	question response: Use optimum ei
	for parameter value
IQUESP	question response: sensitivity table
	plot
IQUESS	question response: sensitivity table
	print
IQUSPP	question response: variable axis plot

SE
SELECT
SELX
SENSICO
SENSIY
SENVAR
SETUF
SIZE
SNSIIV
SNSFRT
SPBA
SPERT
SPRIT
SPRII
STAT
STORE
STORID
STORPL
SUBPSE
SUR
T1
T10
T11
T12
T13
T14
T15
T16
T19
TAB
TDHAX
TDMIN
TDS
TDUE
TEAMIN
TEAM01
TEAM02
TEAM03
TEAM04
TEAM05
TEAM06
TEAM07
TEAM08
TEAM09

Subroutine to change Selection Menu.
Buffer for sensitivity parameter level w opt ei
Character string for curve plot: ;
Choice from Sensitivity Menu
Sensitivity Table parameter value
Subroutine to return data values to
buffer
Plot y-axis choice
Subroutine for sensitivity analysis
Entry routine to reprint Sensitivity
Table

Date for start of block 1 production
Status of the team DP level
Subroutine to write data files onto the
disk
Common buffer for database array
Stored value of flag
Sum of force structure effectiveness

Array of data from the data tables
Flight test completion date desired max
Flight test completion date desired min

Subroutine to input team number and
verify security code
Team 1
Team 2
Team 3
Team 4
Team 5
Team 6
Team 7
Team 8
Team 9

YIC
 YICURVE
 YDC
 YES
 YF
 YFAC
 YFAC1
 YHAI
 YHIN
 YPRE
 YHNI
 YSCALE
 ZBRRA
 ZBRNUL
 ZERC
 ZRSU1
 ZRSU2
 ZRW
 ZTB1
 ZTB2
 ZTB3
 ZTB4
 ZTB5
 ZTB6
 ZTB7
 ZTB8
 ZUIO
 ZWEER

Curve data arrays
 Answer to Yes/No questions
 Y-axis scale
 Y-axis scale
 Y-axis maximum value
 Y-axis minimum value
 Achieved value fee in dollars
 Y-axis minimum
 Y-axis maximum
 Y-axis scale
 Contractor name
 Character string for zero or null
 (blank)
 Subroutine to zeroize all non-data base
 equivalent values

Final lot deployment week

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2. Defense Logistics Studies Information Exchange U.S. Army Logistics Management Center Fort Lee, Virginia 23801	1
3. Academic Group Chairman, Code 39 Command, Control and Communications Group Naval Postgraduate School Monterey, California 93943	1
4. Professor M. Kline, Code 54KX Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943	10
5. Professor G. Bradley, Code 52BZ Department of Computer Science Naval Postgraduate School Monterey, California 93943	5
6. LCDR J. Ferris, SC, USN, Code 54Fj Department of Administrative Sciences Naval Postgraduate School Monterey, California 93943	1
7. Dr. A. Singer Director Weapons Acquisition Simulation National Defense University Industrial College of the Armed Forces Fort McNair, Washington, D.C. 20319	2
8. Dr. Hans Ebenfelt System AB Kungsgatan 8 S-111 43 Stockholm, Sweden	1
9. Colonel Lui Pao Chen Director, Special Projects Organization Ministry of Defence Departure Building Paya Lekar Airport Singapore 1953	1
10. Lieutenant Colonel Ralph Shain Army R&D Division Headquarters, Ministry of Defence Tel Aviv, Israel	1

11. Mr. Robert Butler 1
The Assessment Group
710 Wilshire Boulevard
Santa Monica, California 90401
12. Library, Code 0142 2
Naval Postgraduate School
Monterey, California 93943
13. LCDR G. Schultz, USN 2
1153 Sandwood Place
San Pedro, CA 90731

END

FILMED

6-84

DTIC

