

1. Report No. CG-D-42-82	2. Government Accession No.	3. Report's Catalog No.
4. Title and Subtitle Development of Measures of Effectiveness for Marine Vehicles for Coast Guard Missions - Volume 2	5. Report Date July 1982	6. Performing Organization Code
7. Author(s) L. C. Tedeschi	8. Performing Organization Report No. CGR&DC 8/82	
9. Performing Organization Name and Address U.S.C.G. R&D Center Analysis & Technology, Inc. Avery Point P.O. Box 220 Groton, CT 06340 No. Stonington, CT 06359	10. Work Unit No. (TRAIL)	11. Contract or Grant No. DTCG-39-81-C-80349
12. Sponsoring Agency Name and Address Department of Transportation U.S. Coast Guard Office of Research and Development Washington, D.C. 20593	13. Type of Report and Period Covered Final Report	14. Sponsoring Agency Code
15. Supplementary Notes		
16. Abstract The U.S. Coast Guard is presently studying candidate advanced marine vehicles (AMVs) to replace its aging cutters. Several programs have been initiated to evaluate marine vehicles that best fulfill required missions. These programs include development of decision aids, test and demonstration of existing AMVs, and development of state-of-the-art point designs. The Research and Development Center has developed measures of effectiveness of AMVs in search and rescue, enforcement of laws and treaties, marine environmental protection, and military readiness missions. The MOEs are based on a probability of mission success and can be calculated based on existing data bases or test data. These MOEs provide a quantitative assessment of a craft's ability to perform a mission under specified operational and environmental constraints. An example is presented of how to use the MOEs in the process of evaluating candidate craft to replace existing cutters or procure new ones. Craft characteristics of AMVs were updated based on state-of-the-art technology. They were then incorporated in the Cutter Resources Effectiveness Evaluation (CREE) computer model, which was converted from FORTRAN IV to BASIC to allow it to run on an HP-9845 interactive desktop computer. Volume II is a user's guide for the program.		
17. Key Words Measures of Effectiveness, Advanced Marine Vehicles, Missions, Computer Model, CREE Model	18. Distribution Statement Document is available to the U.S. public through the National Technical Information Service, Springfield, Virginia 22161	
19. Security Classification (for this report) Unclassified	20. Security Classification (for this paper) Unclassified	21. No. of Pages 22. Price

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol When You Know Multiply By To Find Symbol

LENGTH				
in	inches	* 2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km

AREA				
in ²	square inches	6.5	square centimeters	cm ²
ft ²	square feet	0.09	square meters	m ²
yd ²	square yards	0.8	square meters	m ²
mi ²	square miles	2.6	square kilometers	km ²
	acres	0.4	hectares	ha

MASS (WEIGHT)				
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t

VOLUME				
tsp	teaspoons	5	milliliters	ml
tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft ³	cubic feet	0.03	cubic meters	m ³
yd ³	cubic yards	0.76	cubic meters	m ³

TEMPERATURE (EXACT)				
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

Approximate Conversions from Metric Measures

Symbol When You Know Multiply By To Find Symbol

LENGTH				
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi

AREA				
cm ²	square centimeters	0.16	square inches	in ²
m ²	square meters	1.2	square yards	yd ²
km ²	square kilometers	0.4	square miles	mi ²
ha	hectares (10,000 m ²)	2.5	acres	

MASS (WEIGHT)				
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	

VOLUME				
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	0.125	cups	c
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m ³	cubic meters	35	cubic feet	ft ³
m ³	cubic meters	1.3	cubic yards	yd ³

TEMPERATURE (EXACT)				
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F

* 1 in = 2.54 (exactly). For other exact conversions and more detailed tables, see NBS Misc. Publ. 286, Units of Weights and Measures. Price \$2.25. SD Catalog No. C13.10.286.

Development of Measures of Effectiveness for Marine Vehicles for Coast Guard Missions

Volume 2 - Cutter Resources Effectiveness Evaluation
(CREE) Model Program User's Guide

LIBRARY
RESEARCH REPORTS DIVISION
NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA 93943

LOUIS C. TEDESCHI
Analysis & Technology, Inc.,
P.O. Box 220, North Stonington, CT 06359



July 1982

FINAL REPORT

Document is available to the U.S. public through the
National Technical Information Service,
Springfield, Virginia 22161

Prepared for

DEPARTMENT OF TRANSPORTATION
UNITED STATES COAST GUARD,
OFFICE OF RESEARCH AND DEVELOPMENT
Washington, D.C. 20593

83 - 10 13 041

AD A133623

NOTICE

This document is disseminated under the sponsorship of the Department of Transportation in the interest of information exchange. The United States Government assumes no liability for its contents or use thereof.

The United States Government does not endorse products or manufacturers. Trade or manufacturers' names appear herein solely because they are considered essential to the object of this report.

The contents of this report reflect the views of the Coast Guard Research and Development Center, which is responsible for the facts and accuracy of data presented. This report does not constitute a standard, specification, or regulation.



SAMUEL F. POWEL, III

Technical Director

U.S. Coast Guard Research and Development Center
Avery Point, Groton, Connecticut 06340



TABLE OF CONTENTS

	<u>Page</u>
SECTION 1 -- INTRODUCTION	1
SECTION 2 -- PROGRAM OVERVIEW	2
SECTION 3 -- PROGRAM SETUP	4
3.1 TAPES	4
3.2 SCENARIO GENERATION	5
SECTION 4 -- CREE RUN PROCEDURES	6
APPENDIX A -- CREE PROGRAM LISTING	A-1
APPENDIX B -- SCHAR PROGRAM LISTING	B-1
APPENDIX C -- SPTPOS PROGRAM LISTING	C-1
APPENDIX D -- SPRPOS PROGRAM LISTING	D-1
APPENDIX E -- ALPHA PROGRAM LISTING	E-1

LIST OF ILLUSTRATIONS

<u>Figure</u>		<u>Page</u>
1	File Catalog	4
2	Existing Craft Menu	7
3	New Craft Menu	7
4	Length or Displacement Menu	7
5	Acceptable Craft Length	8
6	Acceptable Craft Displacement	8
7	Acceptable Craft Design Speed	9
8	Acceptable Fuel Fraction	9
9	Visibility Distribution	9
10	Tow Distribution	10
11	Sea State Distribution	10

SECTION 1

INTRODUCTION

The Cutter Resources Effectiveness Evaluation (CREE) model program is presently written for interactive use on a Hewlett Packard 9845B desktop computer. Before beginning any computer program runs, it is suggested that the user read the original user's guide, "A Guide for Users and Analysts," written by the Coast Guard R&D Center, report number CG-D-48-78. The user is advised to pay particular attention to chapter 3, "Program Inputs," for a description of the numbers which the user is required to input. The format descriptions of the numbers are not applicable to this version of the program. Reference to section 3 of Volume I of this report should also prove useful. The method of input is described in detail in sections 3.2 and 4.1 of this manual. Appendices A, B, C and D contain listings for the main program and its three sections. Appendix E contains the listing for "ALPHA," the scenario generation program.

It is also advisable that the user be somewhat familiar with HP-9845B computer operation for ease of use in running this program.

SECTION 2

PROGRAM OVERVIEW

The purpose of the CREE model is to evaluate advanced vehicles in a variety of Coast Guard scenarios. The model is deterministic. The CREE program is broken into four major sections: "MAIN," "SCHAR," "SPTPOS," and "SPRPOS." "MAIN" is the executive program. "SCHAR" is concerned with generating the craft characteristics based on the user inputs. "SPTPOS" calculates the probabilities of success of the various tasks which can be performed by the craft. "SPRPOS" is used to calculate the task completion probabilities of the particular scenario input by the user.

The original CREE program was modularized in this fashion as is the HP 9845 version. In the HP version, the program is also broken into the three major files where each file contains the necessary subroutines and functions for that module to run as a separate entity. This structure is necessary because the maximum number of program lines in memory at any one time is significantly less than the entire program would occupy. It is therefore necessary to have each module contain the routines it needs to operate in the file of the same name and then to overlay the module during execution. This overlay process slows execution time, but is required for CREE to run properly. It would be advantageous to place the program on disk in order to decrease run time if a disk is available.

The following convention is used in this manual:

1. When the user is required to actuate one of the HP command keys, this will be described by saying (e.g.):

Press EXECUTE key.

2. When the user is required to type in a response or command, this will be described as (e.g.):

Type GET "MAIN"

The response must be typed in exactly as it appears, including quote marks, if any. In order to inform the computer that the input is ready to be processed, it will be necessary to press either the CONT key or the EXECUTE key after typing the message. The appropriate key to press will always be noted in the text.

SECTION 3

PROGRAM SETUP

3.1 TAPES

Two data cartridges* are needed to run the program. Tape #1 contains the four program modules and tape #2 contains the necessary data files. In order to ensure that the proper files and tapes are used, a catalog of the files is shown in figure 1.

PROGRAM TAPE (Tape #1)				
NAME T15	PRO TYPE 2	REC/FILE	BYTES/REC	ADDRESS
MAIN	DATA	17	256	5
SCHAR	DATA	162	256	22
SPTPOS	DATA	223	256	184
SPRPOS	DATA	359	256	407
DATA TAPE (Tape #2)				
NAME T15	PRO TYPE 2	REC/FILE	BYTES/REC	ADDRESS
MRB	DATA	1	372	5
PWB	DATA	1	372	7
UTB	DATA	1	372	9
MLB-44	DATA	1	372	11
MLB-52	DATA	1	372	13
ANB-55	DATA	1	372	15
ANB-63	DATA	1	372	17
WPB-82	DATA	1	372	19
WPB-95	DATA	1	372	21
WMEC21	DATA	1	372	23
WMEC27	DATA	1	372	25
WHEC37	DATA	1	372	27
ALPHA	DATA	12	256	29
SCEN1	DATA	1	25000	41

FIGURE 1. FILE CATALOG

*Such as Scotch brand DC 100A.

3.2 SCENARIO GENERATION

In order to generate a scenario for a particular run, it is necessary to run the scenario generation program (ALPHA), which is contained on the data tape (tape #2). The user should consult CG-D-48-78 to obtain the guidance and forms necessary for proper generation of a scenario. Note that some effort must be expended in development of a scenario before the user is ready to implement the scenario in the computer. The description of the kind of inputs which are used can be found in the aforementioned user's guide. The program is run as follows:

1. Place the data tape in the HP-9845B tape drive labeled "T14".
2. Type GET "ALPHA:T14"
3. Press the EXECUTE key.
4. When the program is in memory, the user should press the RUN key on the keyboard.

The first input to be entered is the scenario file name. (This name must be unique or an error will occur.) In that event, press the STOP key on the keyboard and then the RUN key again, this time using a unique file name.

The scenario file generated will be placed on the data tape that was inserted into the tape drive labelled "T14". At the completion of this program, a catalog check of the tape will show the new file name on the tape.

It is important to note that once this scenario file is created, the operator does not have to re-create it to run the same scenario. The file is accessed by the CREE program through user inputs at the proper time.

When you have generated the scenario file, you are now prepared to execute the CREE program.

SECTION 4

CREE RUN PROCEDURES

If it was necessary to create a scenario file and the data tape is not in tape drive "T14", then place the data tape in drive "T14" and the program tape in drive "T15".

Then type the system instruction GET "MAIN" and press the EXECUTE key. When the tape has run and the program is loaded into memory, simply press the RUN key to begin program execution.

The CREE program is "Menu Driven" and the user will be asked to input his choices at each stage.

The first user choice is whether to use an existing Coast Guard craft in the evaluation or to develop a new set of craft characteristics. The first display the user will see will be the question:

DO YOU WANT EXISTING (1) OR NEW (2) COAST GUARD CRAFT?

In response, the user should type the selected response (either 1 or 2), and then press the CONT key.

If the operator elects to use an existing craft, the menu shown in figure 2 is displayed. A choice is made by selecting the appropriate craft code and information is taken from the data tape. When a user chooses an existing craft, empirical data is read from the data tape and the user is not required to input length, displacement, or fuel fraction. The only inputs required in that case are the distribution selections. The distribution menu choices are shown starting with figure 9.

CRAFT-CODE	CRAFT
101	MRB 26'
102	PWB 32'
103	UTB 41'
104	MLB 44'
105	MLB 52'
106	ANB 55'
107	ANB 63'
108	WPB 82'
109	WPB 95'
110	WMEC 210'
111	WMEC 270'
112	WHEC 378'

FIGURE 2. EXISTING CRAFT MENU

If the operator wishes to evaluate a new craft, one not currently in the fleet, the menu in figure 3 is displayed.

CRAFT-CODE	CRAFT
10	HYDROFOIL, SUBMERGED FOIL
11	HYDROFOIL, SURFACE PIERCING
20	ACV - LOW PRESSURE/LENGTH RATIO
21	ACV - HIGH PRESSURE/LENGTH RATIO
30	SES (SURFACE-EFFECT SHIP)
40	PLANING CRAFT
50	CATAMARAN
60	SWATH
70	HYBRID
80	CONVENTIONAL CRAFT

FIGURE 3. NEW CRAFT MENU

A choice of craft is made by selecting the appropriate craft code.

Once a new craft choice has been made, the operator must specify the craft size by selecting either a length or a displacement indicated by the menu shown in figure 4.

FOR PROPOSED NEW CRAFT THE USER CAN CHOOSE EITHER
A LENGTH (1) OR A DISPLACEMENT (2).
INPUT 1 OR 2?

FIGURE 4. LENGTH OR DISPLACEMENT MENU

If the operator chooses to specify craft length, he must input a value for the length within the acceptable ranges shown in figure 5.

CRAFT	ACCEPTABLE LENGTH (FT)
HYDROFOIL, SUBMERGED FOIL	50-250
HYDROFOIL, SURFACE PIERCING	50-250
ACV, LOW PRESSURE/LENGTH RATIO	50-250
ACV, HIGH PRESSURE/LENGTH RATIO	30-350
SES	40-500
PLANING	40-250
CATAMARAN	40-135
SWATH	30-400
HYBRID	40-135
CONVENTIONAL	50-400

FIGURE 5. ACCEPTABLE CRAFT LENGTH

If the operator would rather specify craft displacement, he must make a choice from the acceptable values presented in figure 6.

CRAFT	ACCEPTABLE DISPLACEMENT (TONS)
HYDROFOIL, SUBMERGED FOIL	10-1000
HYDROFOIL SURFACE PIERCING	10-1000
ACV, LOW PRESSURE/LENGTH RATIO	10-1000
ACV, HIGH PRESSURE/LENGTH RATIO	10-1000
SES	10-10000
PLANING	10-1000
CATAMARAN	10-140
SWATH	10-10000
HYBRID	10-140
CONVENTIONAL	30-3500

FIGURE 6. ACCEPTABLE CRAFT DISPLACEMENT

Whether the operator specified displacement or length, he must make the following menu choices.

The operator must select an appropriate maximum design speed for his craft, which are shown in figure 7.

CRAFT	MAXIMUM DESIGN SPEED (KNOTS)
HYDROFOIL, SUBMERGED FOIL	40-50
HYDROFOIL, SURFACE PIERCING	30-40
ACV, LOW PRESSURE/LENGTH RATIO	50-70
ACV, HIGH PRESSURE/LENGTH RATIO	40-60
SES	30-50
PLANING	35-45
CATAMARAN	30-40
SWATH	15-25
HYBRID	30-40
CONVENTIONAL	15-40

FIGURE 7. ACCEPTABLE CRAFT DESIGN SPEED

The operator must then choose the acceptable fuel fraction for this craft (see the original user's guide), as shown in figure 8.

PLEASE INPUT ACCEPTABLE FUEL FRACTION
ACCEPTABLE FUEL FRACTION RANGE IS (.2-.8)

FIGURE 8. ACCEPTABLE FUEL FRACTION

The next choice the operator has is to select the appropriate visibility distribution, tow distribution, and sea state distribution as shown in figures 9, 10, and 11, respectively.

VISIBILITY DISTRIBUTION NUMBER	DISTRIBUTION DESCRIPTION	VISIBILITY		
		GOOD	FAIR	POOR
1	VERY GOOD	0.9	0.1	0.0
2	GOOD	0.7	0.2	0.1
3	GOOD TO FAIR	0.5	0.3	0.2

FIGURE 9. VISIBILITY DISTRIBUTION

TOW DISTRIBUTION NUMBER	CUMULATIVE PROBABILITY OF DISPLACEMENT OF TOWED CRAFT					
	0.0	0.2	0.4	0.6	0.8	1.0
1	0.5	1.0	2.5	7.0	10.0	50.0
2	0.7	2.0	4.0	10.0	30.0	100.0
3	1.0	4.0	7.0	20.0	60.0	500.0
4	2.0	6.0	20.0	50.0	80.0	1000.0
5	10.0	20.0	50.0	100.0	300.0	10,000.0

FIGURE 10. TOW DISTRIBUTION

SEA STATE DISTRIBUTION NUMBER	AVERAGE OF SEA STATE DISTRIBUTION	SEA STATE					
		0-1	1-2	2-3	3-4	4-5	5-6
1	0.5	1.0	0.0	0.0	0.0	0.0	0.0
2	1.0	0.55	0.40	0.05	0.0	0.0	0.0
3	1.5	0.20	0.60	0.15	0.05	0.0	0.0
4	2.0	0.20	0.30	0.35	0.10	0.05	0.0
5	2.5	0.10	0.30	0.30	0.15	0.10	0.05
6	3.0	0.5	0.15	0.25	0.40	0.10	0.05
7	3.5	0.05	0.10	0.15	0.35	0.20	0.15
8	4.0	0.0	0.05	0.15	0.25	0.35	0.20
9	4.5	0.0	0.0	0.05	0.20	0.45	0.30
10	5.0	0.0	0.0	0.0	0.10	0.30	0.60

FIGURE 11. SEA STATE DISTRIBUTION

This completes the input choices for the operator. The model will take approximately 15 minutes to run for the scenario chosen.

During the linking of routine "SPRPOS," one more input is required from the user, which is the scenario file name selected for this run. The program will print the message:

TYPE NAME OF SCENARIO FILE DESIRED

The user should respond by typing in the name of a scenario file developed using the program "ALPHA" as described in section 3.2. A sample scenario is supplied with the data tape (tape #2) and can be accessed for preliminary trials of the program by typing "SCEN1" in response to the above question and pressing the CONT key.

Other miscellaneous inputs will be necessary at selected points of the run. These inputs, for the most part, control whether or not a hard copy of the program outputs will be made using the HP 9845's thermal printer. These do not require special note except for the outputs from the individual sorties. Each sortie will produce a page of output, and, depending on the complexity of the scenario, this may run into a considerable number of pages and require a considerable amount of time. As such, an option is available to suppress the hard copies of all the individual sortie outputs. A note of explanation appears on the screen at the appropriate time.

At the end of a run, the program asks the user:

WOULD YOU CARE TO RUN AGAIN (Y/N)?

Responding with "Y" begins the entire run process again. Responding with "N" terminates the session.

APPENDIX A

```

5      !
10     ! CREE PROGRAM
15     !
20     !
25     ! CUTTER RESOURCE EFFECTIVENESS EVALUATION (CREE)
30     !
260    OPTION BASE 1
265    INTEGER I,K,Infile,Casnum,Ilenq
270    INTEGER Ss1,Ss2,Group,Inst
275    INTEGER Eng
290    INTEGER Visdtb,Towdtb,Taskno1,Rates
285    !
290    DIM Ssprbd(8),Crfnm$(31)
295    DIM Cwspd(4),Sfceng(4),Sfccf(4),Totstfc(4),Sfcgal(4)
300    DIM Hputil(4),Fuelrt(4),Endur(4),Range(4)
305    DIM Fuelr2(4),Eng(4),Ssprob(8,10)
310    DIM Tnrad(4),Motion(4),Engnam$(2),Engnm(2)
315    DIM Cftnam(8)
320    DIM Cc(19),Df(19),Ls(19),Mn(19),Tw(19),Tarry(22,2)
325    DIM Mo(19),Tpos(19),Sk(19),Go(4),Mnacc(4),Mnbkg(4),Mntur(4)
330    DIM Su(4),Lls(4),Xx(4),Aves(10),Towdis(6,5),Vmxuis(3),Cgfr20(12)
335    DIM Cgfr10(12),Visdis(3,3),Gomin(4)
340    !
345    ! DIMENSIONS FOR SPRPOS ROUTINE
350    !
355    DIM Taskno$(25,4)[25],Grpnm$(20)
360    SHORT Nnode(22),Visds1(3,3),Visds2(3,3)
365    SHORT Unajpb(1000),Phpos(1000),Pathtn(1000),Ipthfu(1000)
370    SHORT Beta(20,40)
375    SHORT Tposmx(25)
380    SHORT Timlst(100),Fuelst(100),Prblst(100)
385    SHORT Gpplmx(50,50),Ovcnmx(50,50),Mintim(50),Minfue(50)
390    SHORT Master(25,4),Count(25,4),Totcnt(25,4)
395    SHORT Im(25,4),Imrate(40),Intskn(40),Imptsk(40,40)
400    SHORT Mtask(40)
405    COM Cftnam$,Ssaug,Speed(4),Mfuirt(4),Towspd,Gpdat1(40,2),Gpdat2(40,1
8),Pshlst(100),INTEGER Idisp,Idspd,Ptr
410    !
415    ! THE FOLLOWING IS THE MAIN PROGRAM MODULE WHICH CALLS THE THREE MAJOR
420    ! SUBROUTINES. NAMELY SCHAR,SPTPOS,SPRPOS.
425    !
427    Begin: DISP "LINKING IN SCHAR MODULE"
428            PRINTER IS 16
430            LINK "SCHAR",475,435
435            GOSUB Schar
436            DISP "LINKING IN SPTPOS MODULE"
440            LINK "SPTPOS",475,445
445            GOSUB Sptpos
446            DISP "LINKING IN SPRPOS MODULE"
450            LINK "SPRPOS",475,455
455            GOSUB Sprpos
460            INPUT "WOULD YOU CARE TO RUN AGAIN (Y/N)? ",Ans$
465            IF Ans$="Y" THEN Begin
470            END
500    !
530    ! CONVERSION - THE CREE PROGRAM WAS CONVERTED FROM FORTRAN
535    ! TO HP BASIC (2/82 - 5/82) BY ANALYSIS & TECHNOLOGY
540    ! INC. MAJOR CONTRIBUTORS WERE CHRIS HEIDTMAN AND
545    ! PATTY ROY.
550    !
555    ! MAIN PROGRAM - READS CRAFT INPUT DATA AND CALLS SUBROUTINES
560    ! TO COMPUTE CRAFT CHARACTERISTICS
565    !
570    ! INPUTS ARE: 1. CRAFT TYPE
575    ! 2. CRAFT DISPLACEMENT (IN TONS)
580    ! OR CRAFT LENGTH (IN FEET)

```

585	!	3. DESIGN SPEED (IN KNOTS)
590	!	4. FUEL FRACTION - OF TOTAL PAYLOAD
595	!	5. VISIBILITY, TOW, DEPTH, AND SEA STATE DISTRIBUTION
600	!	NUMBERS
605	!	
610	!	CRAFT AND ENGINE ARE IDENTIFIED BY CODES, AS FOLLOWS:
615	!	CRAFT CODES:
620	!	10. HYDROFOIL-SUBMERGED FOIL
625	!	11. HYDROFOIL-SURFACE PIERCING
630	!	20. AIR CUSHION VEHICLE - LOW P/L
635	!	21. AIR CUSHION VEHICLE - HIGH P/L
640	!	30. SURFACE EFFECT SHIP
645	!	40. PLANING CRAFT
650	!	50. CATAMARAN
655	!	60. SWATH
660	!	70. HYBRID VESSEL
665	!	80. CONVENTIONAL CRAFT
670	!	101. MRB
675	!	102. PWB 32
680	!	103. UTB 41
685	!	104. MLB 44
690	!	105. MLB 52
695	!	106. ANB 55
700	!	107. ANB 63
705	!	108. WPB 82
710	!	109. WPB 95
715	!	110. WMEC 210
720	!	111. WMEC 270
725	!	112. WMEC 378
730	!	
735	!	ENGINE CODES:
740	!	1. GAS TURBINE
745	!	2. DIESEL

APPENDIX B

```

475 ! ***** S C H A R *****
480 !
485 ! THE FOLLOWING IS THE START OF THE CRAFT CHARACTERISTICS PROGRAM
490 ! THE PROGRAM REQUIRES THE USER TO INPUT THE FOLLOWING INFORMATION
495 ! VIA MENU CHOICES: 1. CRAFT TYPE
500 !                   2. CRAFT LENGTH
505 !                   OR 3. CRAFT DISPLACEMENT
510 !                   4. CRAFT SPEED
515 !                   5. VISIBILITY DISTRIBUTION
520 !                   6. TOW DISTRIBUTION
525 !                   7. SEA STATE
530 !                   8. FUEL FRACTION
535 !
540 Schar: ! STARTING POINT
545 DATA 10,128,11,128,20,128,21,128,30,128,40,128,50,128,60,128
550 DATA 70,128,80,128,101,128,102,128,103,128,104,128,105,128
555 DATA 106,128,107,128,108,128,109,128,110,128,111,128,112,128
560 DATA 1.,.35,.20,.20,.10,.05,.05,0.,0.,0.
565 DATA 0.,.40,.60,.30,.30,.15,.10,.05,0.,0.
570 DATA 0.,.05,.15,.35,.30,.25,.15,.15,.05,0.
575 DATA 0.,0.,.05,.10,.15,.40,.35,.25,.20,.10
580 DATA 0.,0.,0.,.05,.10,.10,.20,.35,.45,.30
585 DATA 0.,0.,0.,0.,.05,.05,.15,.20,.30,.60
590 DATA 0.,0.,0.,0.,0.,0.,0.,0.,0.,0.
595 DATA 0.,0.,0.,0.,0.,0.,0.,0.,0.,0.
600 DATA '(GT)', '(DE)'
605 RESTORE
610 MAT READ Tarry
615 MAT READ Ssprob
620 MAT READ Engnam$
625 Menu: PRINT PAGE
630 INPUT "DO YOU WANT EXISTING(1) OR PROPOSED NEW(2) COAST GUARD CRAFT?",Ans
635 IF Ans=1 THEN L12
640 IF Ans=2 THEN L13
645 PRINT "INVALID INPUT, PLEASE TRY AGAIN. PRESS CONT TO GO ON."
650 PAUSE
655 GOTO Menu
660 L13: PRINT " CRAFT-CODE CRAFT"
665 PRINT
670 PRINT " 10 HYDROFOIL,SUBMERGED FOIL"
675 PRINT " 11 HYDROFOIL,SURFACE PIERCING"
680 PRINT " 20 ACV - LOW PRESSURE/LENGTH RATIO"
685 PRINT " 21 ACV - HIGH PRESSURE/LENGTH RATIO"
690 PRINT " 30 SES (SURFACE-EFFECT SHIP)"
695 PRINT " 40 PLANING CRAFT"
700 PRINT " 50 CATAMARAN"
705 PRINT " 60 SWATH"
710 PRINT " 70 HYBRID"
715 PRINT " 80 CONVENTIONAL CRAFT"
720 GOTO L14
725 L12: PRINT PAGE
730 PRINT " CRAFT-CODE CRAFT"
735 PRINT
740 PRINT " 101 MRB 26'"
745 PRINT " 102 PWB 32'"
750 PRINT " 103 UTB 41'"
755 PRINT " 104 MLB 44'"
760 PRINT " 105 MLB 52'"
765 PRINT " 106 ANB 55'"
770 PRINT " 107 ANB 63'"
775 PRINT " 108 WPB 82'"
780 PRINT " 109 WPB 95'"
785 PRINT " 110 WMEC 210'"
790 PRINT " 111 WMEC 270'"
795 PRINT " 112 WMEC 378'"

```

```

800 L14:INPUT "CHOOSE ONE OF THE ABOVE CRAFT BY INPUTTING CRAFT-CODE ",Code
805 !
810 ! NOW SEE IF INPUT VALUE IS A VALID CRAFT TYPE
815 !
820 Fig=0
825 FOR I=1 TO 22
830 IF Code<>Tarry(I,1) THEN L33
835 Fig=1
840 Tarry(I,2)=133
845 L33:NEXT I
850 IF Fig=1 THEN L11
855 PRINT "SORRY INVALID TYPE-CODE, TRY AGAIN! PRESS CONT TO GO ON. "
860 PAUSE
865 GOTO Menu
870 !
875 ! CHECK TO SEE IF CONVENTIONAL CRAFT
880 !
885 L11:IF Code>100 THEN L51
890 L15:PRINT PAGE
895 PRINT " FOR PROPOSED NEW COAST GUARD CRAFT THE USER CAN CHOOSE EITHER"
900 PRINT "A LENGTH (1) OR A DISPLACEMENT (2). "
905 PRINT
910 INPUT "INPUT 1 OR 2? ",Ans
915 IF (Ans=1) OR (Ans=2) THEN L21
920 PRINT " INVALID ENTRY TRY AGAIN. PRESS CONT TO GO ON. "
925 PAUSE
930 GOTO L15
935 !
940 ! SELECT DISP OR LENGTH MENU.
945 !
950 L21:IF Ans=2 THEN Ldisp
955 !
960 ! INPUT LENGTH
965 !
970 L25:PRINT PAGE
975 PRINT USING L26;" CRAFT ACCEPTABLE LENGTH (FT)
"
980 PRINT
985 PRINT USING L26;CHR$(Tarry(1,2)),"HYDROFOIL, SUBMERGED FOIL
50-250"
990 PRINT USING L26;CHR$(Tarry(2,2)),"HYDROFOIL, SURFACE PIERCING
50-250"
995 PRINT USING L26;CHR$(Tarry(3,2)),"ACV, LOW PRESSURE/LENGTH RATIO
50-250"
1000 PRINT USING L26;CHR$(Tarry(4,2)),"ACV, HIGH PRESSURE/LENGTH RATIO
30-350"
1005 PRINT USING L26;CHR$(Tarry(5,2)),"SES
40-500"
1010 PRINT USING L26;CHR$(Tarry(6,2)),"PLANING
40-250"
1015 PRINT USING L26;CHR$(Tarry(7,2)),"CATAMARAN
40-135"
1020 PRINT USING L26;CHR$(Tarry(8,2)),"SWATH
30-400"
1025 PRINT USING L26;CHR$(Tarry(9,2)),"HYBRID
40-135"
1030 PRINT USING L26;CHR$(Tarry(10,2)),"CONVENTIONAL
50-400"
1035 PRINT CHR$(128)
1040 L26:IMAGE K
1045 INPUT " CHOOSE ACCEPTABLE VALUE IN THE RANGE HI-LIGHTED",Leng
1050 IF (Leng<=500) OR (Leng>=30) THEN L31
1055 PRINT "UN-ACCEPTABLE LENGTH INPUT, PLEASE TRY AGAIN. PRESS CONT TO GO
ON"
1060 PAUSE
1065 GOTO L25

```

```

1070 L31:Disp1=0
1075 GOTO L41
1080 !
1085 ! USER HAS CHOSEN TO INPUT DISPLACEMENT
1090 !
1095 Ldisp: PRINT PAGE
1100 PRINT USING L26;CHR$(128),"CRAFT ACCEPTABLE
DISPLACEMENT (TONS)"
1105 PRINT
1110 PRINT USING L26;CHR$(Tarry(1,2)),"HYDROFOIL, SUBMERGED FOIL
10-1000"
1115 PRINT USING L26;CHR$(Tarry(2,2)),"HYDROFOIL SURFACE PIERCING
10-1000"
1120 PRINT USING L26;CHR$(Tarry(3,2)),"ACV, LOW PRESSURE/LENGTH RATIO
10-1000"
1125 PRINT USING L26;CHR$(Tarry(4,2)),"ACV, HIGH PRESSURE/LENGTH RATIO
10-1000"
1130 PRINT USING L26;CHR$(Tarry(5,2)),"SES
10-10000"
1135 PRINT USING L26;CHR$(Tarry(6,2)),"PLANING
10-1000"
1140 PRINT USING L26;CHR$(Tarry(7,2)),"CATAMARAN
10-140"
1145 PRINT USING L26;CHR$(Tarry(8,2)),"SWATH
10-10000"
1150 PRINT USING L26;CHR$(Tarry(9,2)),"HYBRID
10-140"
1155 PRINT USING L26;CHR$(Tarry(10,2)),"CONVENTIONAL
30-3500"
1160 PRINT CHR$(128)
1165 INPUT "CHOOSE ACCEPTABLE VALUE IN RANGE HI-LIGHTED",Disp1
1170 IF (Disp1>=10) AND (Disp1<=3500) THEN L35
1175 PRINT "UN-ACCEPTABLE DISPLACEMENT INPUT. PLEASE TRY AGAIN. "
1180 PRINT "PRESS CONT TO GO ON."
1185 PAUSE
1190 GOTO Ldisp1
1195 L35:Leng=0
1200 L41:PRINT PAGE
1205 PRINT USING L26;CHR$(128)," CRAFT MAXIMUM DESIG
N SPEED (KNOTS)"
1210 PRINT
1215 PRINT USING L26;CHR$(Tarry(1,2)),"HYDROFOIL, SUBMERGED FOIL
40-50"
1220 PRINT USING L26;CHR$(Tarry(2,2)),"HYDROFOIL, SURFACE PIERCING
30-40"
1225 PRINT USING L26;CHR$(Tarry(3,2)),"ACV, LOW PRESSURE/LENGTH RATIO
50-70"
1230 PRINT USING L26;CHR$(Tarry(4,2)),"ACV, HIGH PRESSURE/LENGTH RATIO
40-60"
1235 PRINT USING L26;CHR$(Tarry(5,2)),"SES
30-50"
1240 PRINT USING L26;CHR$(Tarry(6,2)),"PLANING
35-45"
1245 PRINT USING L26;CHR$(Tarry(7,2)),"CATAMARAN
30-40"
1250 PRINT USING L26;CHR$(Tarry(8,2)),"SWATH
15-25"
1255 PRINT USING L26;CHR$(Tarry(9,2)),"HYBRID
30-40"
1260 PRINT USING L26;CHR$(Tarry(10,2)),"CONVENTIONAL
15-40"
1265 PRINT CHR$(128)
1270 INPUT "CHOOSE ACCEPTABLE VALUE IN RANGE HI-LIGHTED ",Dspeed
1275 IF (Dspeed>=15) AND (Dspeed<=60) THEN L45
1280 PRINT "UN-ACCEPTABLE SPEED INPUT. PLEASE TRY AGAIN."
1285 PRINT "PRESS CONT TO GO ON."

```

```

1290 PAUSE
1295 GOTO L41
1300 L45:PRINT PAGE
1305 PRINT "PLEASE INPUT ACCEPTABLE FUEL FRACTION"
1310 INPUT "ACCEPTABLE FUEL FRACTION RANGE IS (.2-.8)",Fufrc
1315 IF (Fufrc>=.2) AND (Fufrc<=.8) THEN L51
1320 PRINT "UN-ACCEPTABLE FUEL FRACTION INPUT. PLEASE TRY AGAIN."
1325 PRINT "PRESS CONT TO GO ON."
1330 PAUSE
1335 GOTO L45
1340 L51:PRINT PAGE
1345 PRINT "VISIBILITY DISTRIBUTION VISIBILITY"
1350 PRINT "DISTRIBUTION DESCRIPTION"
1355 PRINT "NUMBER GOOD FAIR POOR"
1360 PRINT
1365 PRINT "1 VERY GOOD 0.9 0.1 0.0"
1370 PRINT
1375 PRINT "2 GOOD 0.7 0.2 0.1"
1380 PRINT
1385 PRINT "3 GOOD TO FAIR 0.5 0.3 0.2"
1390 PRINT
1395 PRINT "THREE VISIBILITY-DISTRIBUTIONS ARE AVAILABLE."
1400 PRINT "FOR EXAMPLE, VISIBILITY-DISTRIBUTION NO. 2, CALLED GOOD, IMPLIES"
1405 PRINT "70 PERCENT CHANCE OF GOOD VISIBILITY, 20 PERCENT CHANCE OF FAIR"
1410 PRINT "VISIBILITY AND 10 PERCENT CHANCE OF POOR VISIBILITY"
1415 PRINT
1420 INPUT "CHOOSE AN ACCEPTABLE VISIBILITY-DISTRIBUTION NUMBER (1,2 OR 3)",V
1425 IF (Visdtb=1) OR (Visdtb=2) OR (Visdtb=3) THEN L55
1430 PRINT "UN-ACCEPTABLE VISIBILITY-DISTRIBUTION NUMBER ",Visdtb
1435 PRINT "PRESS CONT TO GO ON"
1440 PAUSE
1445 GOTO L51
1450 L55:PRINT PAGE
1455 PRINT "TOW CUMULATIVE PROBABILITY OF"
1460 PRINT "DISTRIBUTION DISPLACEMENT OF TOWED CRAFT"
1465 PRINT "NUMBER 0.0 0.2 0.4 0.6 0.8 1.0"
1470 PRINT
1475 PRINT "1 0.5 1.0 2.5 7.0 10.0 50.0"
1480 PRINT "2 0.7 2.0 4.0 10.0 30.0 100.0"
1485 PRINT "3 1.0 4.0 7.0 20.0 60.0 500.0"
1490 PRINT "4 2.0 6.0 20.0 50.0 90.0 1000.0"
1495 PRINT "5 10.0 20.0 50.0 100.0 300.0 10,000.0"
1500 PRINT
1505 PRINT "FIVE TOWING-DISTRIBUTION NUMBERS ARE AVAILABLE."
1510 PRINT "FOR EXAMPLE, TOWING DISTRIBUTION NO. 1 INDICATES AN OPERATING REG
ION"
1515 PRINT "WHERE NONE OF THE CRAFT TO BE TOWED ARE LESS THAN 0.5 TON,"
1520 PRINT "20 PERCENT OF THE CRAFT TO BE TOWED ARE LESS THAN 1.0 TON,"
1525 PRINT "40 PERCENT OF THE CRAFT TO BE TOWED ARE LESS THAN 2.5 TONS, ETC."
1530 PRINT
1535 INPUT "CHOOSE AN ACCEPTABLE TOWING-DISTRIBUTION NUMBER (1,2,3,4,OR 5)",T
1540 IF (Towdtb=1) OR (Towdtb=2) OR (Towdtb=3) OR (Towdtb=4) OR (Towdtb=5) TH
EN L61
1545 PRINT "UN-ACCEPTABLE TOWING-DISTRIBUTION NUMBER. PLEAO GO ON."
1550 PRINT "PRESS CONT TO GO ON"
1555 PAUSE
1560 GOTO L55
1565 L61:PRINT PAGE
1570 PRINT "SEA AVERAGE"
1575 PRINT "STATE OF S E A S T A T E"
1580 PRINT "DISTRIB- SEA STATE"
1585 PRINT "UTION DISTRI-"
1590 PRINT "NUMBER BUTION 0-1 1-2 2-3 3-4 4-5 5-6"
1595 PRINT "1 0.5 1.0 0.0 0.0 0.0 0.0"

```

```

1600 PRINT " 2      1.0      0.55 0.40 0.05 0.0 0.0 0.0"
1605 PRINT " 3      1.5      0.20 0.60 0.15 0.05 0.0 0.0"
1610 PRINT " 4      2.0      0.20 0.30 0.35 0.10 0.05 0.0"
1615 PRINT " 5      2.5      0.10 0.30 0.30 0.15 0.10 0.05"
1620 PRINT " 6      3.0      0.5 0.15 0.25 0.40 0.10 0.05"
1625 PRINT " 7      3.5      0.05 0.10 0.15 0.35 0.20 0.15"
1630 PRINT " 8      4.0      0.0 0.05 0.15 0.25 0.35 0.20"
1635 PRINT " 9      4.5      0.0 0.0 0.05 0.20 0.45 0.30"
1640 PRINT " 10     5.0      0.0 0.0 0.0 0.10 0.30 0.60"
1645 PRINT
1650 PRINT "TEN SEA STATES ARE AVAILABLE."
1655 INPUT "CHOOSE AN ACCEPTABLE SEA-STATE DISTRIBUTION NUMBER (1-10)", Sspdtb
1660 IF (Sspdtb=1) OR (Sspdtb=2) OR (Sspdtb=3) OR (Sspdtb=4) OR (Sspdtb=5) TH
EN L65
1665 IF (Sspdtb=6) OR (Sspdtb=7) OR (Sspdtb=8) OR (Sspdtb=9) OR (Sspdtb=10) T
HEN L65
1670 PRINT "UN-ACCEPTABLE SEA-STATE DISTRIBUTION"
1675 PRINT "PRESS CONT TO GO ON"
1680 PAUSE
1685 GOTO L61
1690 L65:Dphdtb=1
1695 !
1700 ! FIND DISCRETE SEA STATE PROBABILITY DISTRIBUTION
1705 !
1710 FOR Ss1=2 TO 8
1715 Ss2=Ss1-1
1720 Ssprbd(Ss1)=.5*Ssprbd(Ss1,Sspdtb)+.5*Ssprbd(Ss2,Sspdtb)
1725 NEXT Ss1
1730 Ssprbd(1)=.5*Ssprbd(1,Sspdtb)
1735 !
1740 ! START OF MAJOR MODULE CRAFT CALCULATIONS.
1745 !
1750 !
1755 ! COMPUTES CRAFT CHARACTERISTICS
1760 !
1765 !
1770 IF Code>=100 THEN L5001
1775 !
1780 ! COMPUTE CHARACTERISTICS FOR HPWC
1785 !
1790 IF Leng=0 THEN Leng=FNFleng(Code,Displ)
1795 IF Displ=0 THEN Displ=FNFDisp(Code,Leng)
1800 Ltob=FNFltob(Code,Leng)
1805 Beam=Leng/Ltob
1810 Dtol=FNFDtol(Code,Leng)
1815 Draf=Dtol*Leng
1820 Deck=FNFdeck(Code,Leng,Beam)
1825 Useid=FNFuseid(Code,Displ)
1830 Fuelcp=Fufrc*Useid
1835 Cangcp=(1-Fufrc)*Useid
1840 Toudsp=FNFTouds(Code,Displ)
1845 Surviv=FNFSurvi(Code,Leng)
1850 Isurv=Surviv
1855 Hpbins=FNHpbins(Code,Displ)
1860 Hpinst=(Dspeed/FNFbsspd(Code))^3*Hpbins
1865 Hppton=Hpinst/Displ
1870 FOR Rate=1 TO 4
1875 Eng(Rate)=FNFeng(Code,Rate)
1880 Cuspd(Rate)=FNFcuspd(Code,Rate,Dspeed)
1885 Fctdsp=Cuspd(Rate)/Dspeed
1890 Fctbsp=Cuspd(Rate)/FNFbsspd(Code)
1895 Hpfctu=FNHpfctu(Code,Rate,Fctdsp,Fctbsp)
1900 IF (Rate=1) OR (Rate=2) THEN Hputil(Rate)=Hpfctu*Hpinst
1905 IF (Rate=3) OR (Rate=4) THEN Hputil(Rate)=Hpfctu*Hpinst
1910 Sfceng(Rate)=FNFSfcen(Eng(Rate),Hpinst)
1915 Sfcctf(Rate)=FNFSfcctf(Eng(Rate),Hpfctu)

```

```

1920      Totalsfc(Rate)=Sfceng(Rate)+Sfccf(Rate)
1925      Sfcgal(Rate)=Totalsfc(Rate)+335/2240
1930      FuelInt(Rate)=Hputil(Rate)+Sfcgal(Rate)
1935      Fueln2(Rate)=FuelInt(Rate)/Cuspd(Rate)
1940      Endur(Rate)=Fuelcp/(FuelInt(Rate)/335)
1945      Range(Rate)=Endur(Rate)*Cuspd(Rate)
1950      Tnrad(Rate)=FNFtnrad(Code,Cuspd(Rate))/3
1955      Motion(Rate)=FNFmutav(Ssprbd(*),Code,Displ,Rate)
1960      NEXT Rate
1965      Hptnkt=Hppton/Cuspd(1)
1970      IF Code<100 THEN L1008
1975      !
1980 L5001: Cqtype=Code-100
1985      IF Code=101 THEN File$="MRB:T14"
1990      IF Code=102 THEN File$="PWB:T14"
1991      IF Code=103 THEN File$="UTB:T14"
1995      IF Code=104 THEN File$="MLB-44:T14"
2000      IF Code=105 THEN File$="MLB-52:T14"
2005      IF Code=106 THEN File$="ANB-55:T14"
2010      IF Code=107 THEN File$="ANB-63:T14"
2015      IF Code=108 THEN File$="WPB-82:T14"
2020      IF Code=109 THEN File$="WPB-95:T14"
2025      IF Code=110 THEN File$="WMEC21:T14"
2030      IF Code=111 THEN File$="WMEC27:T14"
2035      IF Code=112 THEN File$="WMEC37:T14"
2040      !
2045      ASSIGN #1 TO File$
2050      READ #1;Leng,Displ,Dspeed,Fufrc,Ltob,Beam,Dtol,Draf,Deck,Useld,Fuelcp
2055      READ #1;Cargcp,Towdsp,Isurv,Hpinst,Hppton,Hptnkt
2060      MAT READ #1;Cuspd,Eng,Hputil,FuelInt,Fueln2,Endur,Range
2065      ASSIGN * TO #1
2070      FOR I=1 TO 4
2075      Sfceng(I)=0
2080      Sfccf(I)=0
2085      Totalsfc(I)=0
2090      Sfcgal(I)=0
2095      Rate=I
2100      Motion(I)=FNFmutav(Ssprbd(*),Code,Displ,Rate)
2105      IF I<>2 THEN L5002
2110      Tnrad(I)=FNFtnrad(Code,Cuspd(I))/3
2115      GOTO L5000
2120 L5002: Tnrad(I)=0
2125 L5000: NEXT I
2130 L1008: IF Code=10 THEN Crfnm$="HYDROFOIL, SUBMERGED FOIL"
2135      IF Code=11 THEN Crfnm$="HYDROFOIL, SURFACE PIERCING"
2140      IF Code=20 THEN Crfnm$="ACV, LOW PRESSURE/LENGTH RATIO"
2145      IF Code=21 THEN Crfnm$="ACV, HIGH PRESSURE/LENGTH RATIO"
2150      IF Code=30 THEN Crfnm$="SES"
2155      IF Code=40 THEN Crfnm$="PLANING CRAFT"
2160      IF Code=50 THEN Crfnm$="CATAMARAN"
2165      IF Code=60 THEN Crfnm$="SWATH"
2170      IF Code=70 THEN Crfnm$="HYBRID VESSEL"
2175      IF Code=80 THEN Crfnm$="CONVENTIONAL CRAFT"
2180      IF Code=101 THEN Crfnm$="MRB 26'"
2185      IF Code=102 THEN Crfnm$="PWB 32'"
2190      IF Code=103 THEN Crfnm$="UTB 41'"
2195      IF Code=104 THEN Crfnm$="MLB 44'"
2200      IF Code=105 THEN Crfnm$="MLB 52'"
2205      IF Code=106 THEN Crfnm$="ANB 55'"
2210      IF Code=107 THEN Crfnm$="ANB 63'"
2215      IF Code=108 THEN Crfnm$="WPB 82'"
2220      IF Code=109 THEN Crfnm$="WPB 95'"
2225      IF Code=110 THEN Crfnm$="WMEC 210'"
2230      IF Code=111 THEN Crfnm$="WMEC 270'"
2235      IF Code=112 THEN Crfnm$="WMEC 370'"
2240      IF Visdtb=1 THEN Visds2$="VERY GOOD"

```

```

2245 IF Visdtb=2 THEN Visds2$="GOOD"
2250 IF Visdtb=3 THEN Visds2$="GOOD TO FAIR"
2255 PRINT PAGE
2260 INPUT "WOULD YOU LIKE A HARD COPY (Y/N)?",Ans$
2265 IF Ans$="Y" THEN PRINTER IS 0
2270
2275 Outfil=6
2280 Idisp=Displ
2285 Ileng=Leng
2290 Idspd=Speed
2295 PRINT PAGE
2300 PRINT USING L2;"C R A F T C H A R A C T E R I S T I C S"
2305 L2: IMAGE 15X,K
2310 PRINT
2315 PRINT USING L2;" **MODULE INPUTS**"
2320 PRINT
2325 IF Code>100 THEN L4
2330 PRINT USING L3;"CRAFT TYPE",Crfnm$
2335 L3: IMAGE 18X,K,10X,K
2340 GOTO L5
2345 L4: PRINT USING L6;"CRAFT TYPE","COAST GUARD",Crfnm$
2350 L6: IMAGE 18X,K,9X,K,1X,K
2355 L5: PRINT USING L10;"DISPLACEMENT",Idisp,"TONS"
2360 L10: IMAGE 18X,K,4X,6D,1X,K
2365 PRINT USING L20;"LENGTH",Ileng,"FEET"
2370 L20: IMAGE 18X,K,10X,6D,1X,K
2375 PRINT USING L30;"DESIGN SPEED",Idspd,"KNOTS"
2380 L30: IMAGE 18X,K,4X,6D,1X,K
2385 PRINT USING L40;"FUEL FRACTION",Fufrc
2390 L40: IMAGE 18X,K,3X,4D,2D
2395 PRINT
2400 PRINT
2405 PRINT USING L2;" **CRAFT PARTICULARS**"
2410 PRINT
2415 PRINT USING L50;"LENGTH",Leng,"FEET"
2420 L50: IMAGE 13X,K,22X,6D,2X,K
2425 PRINT USING L60;"BEAM",Beam,"FEET"
2430 L60: IMAGE 13X,K,24X,6D,2X,K
2435 PRINT USING L70;"DRAFT",Draf,"FEET"
2440 L70: IMAGE 13X,K,23X,6D,2X,K
2445 PRINT USING L80;"LENGTH/BEAM RATIO",LtoB
2450 L80: IMAGE 13X,K,12X,5D,2D
2455 PRINT USING L90;"DRAFT/LENGTH RATIO",DtoL
2460 L90: IMAGE 13X,K,11X,5D,DD
2465 PRINT USING L100;"DISPLACEMENT",Displ,"TONS"
2470 L100: IMAGE 13X,K,16X,6D,2X,K
2475 PRINT USING L110;"SURVIVABILITY",Isurv,"SEA STATE"
2480 L110: IMAGE 13X,K,14X,7D,4X,K
2485 PRINT USING L120;"TOWS VESSELS UP TO",Towdsp,"TONS"
2490 L120: IMAGE 13X,K,10X,6D,3X,K
2495 PRINT USING L130;"USEABLE DECK AREA",Deck,"SQUARE FEET"
2500 L130: IMAGE 13X,K,11X,6D,3X,K
2505 PRINT USING L140;"CARGO CAPACITY",Cargcp,"TONS"
2510 L140: IMAGE 13X,K,14X,6D,2X,K
2515 PRINT USING L150;"FUEL CAPACITY",Fuelcp,"TONS"
2520 L150: IMAGE 13X,K,15X,6D,2X,K
2525 PRINT USING L160;"USEFUL PAYLOAD",Useid,"TONS"
2530 L160: IMAGE 13X,K,14X,6D,2X,K
2535 PRINT USING L170;"INSTALLED POWER",Hpinst,"HORSE POWER"
2540 L170: IMAGE 13X,K,13X,6D,3X,K
2545 PRINT USING L180;"POWER TO WEIGHT",Hppton,"HP/TON"
2550 L180: IMAGE 13X,K,13X,6D,2X,K
2555 PRINT USING L190;"TRANSPORT EFFICIENCY",Hptnkt,"HP/TON-KNOT"
2560 L190: IMAGE 13X,K,9X,5D,2D,1X,K
2565 PRINT USING L200;"RANGE AT CRUISE SPEED",Range(2),"NAUTICAL MILES"
2570 L200: IMAGE 13X,K,7X,6D,3X,K

```

```

2575      PRINT USING L210;"ENDURANCE AT CRUISE SPEED",Endur(2),"HOURS"
2580 L210:  IMAGE 13X,K,3X,6D.1D,2X,K
2585      PRINT
2590      PRINT
2595      PRINT
2600      PRINT USING L2;"          **CRAFT PERFORMANCE**"
2605      PRINT
2610      PRINT USING L220;"FLANK","CRUISE","REDUCED","ON"
2615 L220:  IMAGE 29X,K,3X,K,2X,K,3X,K
2620      PRINT USING L230;"SPEED","SPEED","SPEED","SCENE"
2625 L230:  IMAGE 29X,K,3X,K,4X,K,3X,K
2630      PRINT
2635      PRINT USING L235;"ENGINE TYPE"
2640 L235:  IMAGE #,10X,K,7X
2645      FOR I=1 TO 4
2650      PRINT USING L236;Engnam$(Eng(I))
2655 L236:  IMAGE #,K,3X
2660      NEXT I
2665      !
2670      PRINT
2675      PRINT "          CALM WATER"
2680      PRINT
2685      PRINT USING L240;"CALM WATER SPEED",Cuspd(*),"KNOTS"
2690 L240:  IMAGE 10X,K,4(6D.1D),4X,K
2695      PRINT USING L250;"SFC (WEIGHT)",Tot$fc(*),"LBS/HP-HR"
2700 L250:  IMAGE 10X,K,5X,4(5D.DD),3X,K
2705      PRINT USING L260;"SFC (VOLUME)",Sfcgal(*),"GAL/HP-HR"
2710 L260:  IMAGE 10X,K,5X,4(5D.2D),3X,K
2715      PRINT USING L270;"HP UTILIZED",Hputil(*),"HP"
2720 L270:  IMAGE 10X,K,5X,4(6D.1D),4X,K
2725      PRINT USING L280;"FUEL CONSUMPTION",Fuelr1(*),"GAL/HR"
2730 L280:  IMAGE 10X,K,4(6D.D),4X,K
2735      PRINT USING L290;"FUEL CONSUMPTION",Fuelr2(*),"GAL/NAUT MI"
2740 L290:  IMAGE 10X,K,4(6D.D),4X,K
2745      PRINT USING L300;"ENDURANCE (FUEL)",Endur(*),"HOURS"
2750 L300:  IMAGE 10X,K,4(6D.1D),4X,K
2755      PRINT USING L310;"RANGE",Range(*),"NAUTICAL MI"
2760 L310:  IMAGE 10X,K,11X,4(6D.1D),4X,K
2765      PRINT USING L320;"TURNING RADIUS",Tnrad(*),"YARDS"
2770 L320:  IMAGE 10X,K,2X,4(6D.D),4X,K
2775      PRINT
2780      PRINT "          IN SELECTED SEAWAY"
2785      PRINT
2790      PRINT USING L330;"CRAFT MOTION",Motion(*),"G"
2795 L330:  IMAGE 10X,K,4X,4(6D.1D),4X,K
2800      !
2805 L2021: ! CONTINUE
2810 L9999: RETURN
2815      !
2820      ! FLENG
2825      !
2830      ! LENGTH(IN FEET)
2835      !
2840      DEF FNFleng(Code,Displ)
2845      IF Code=10 THEN Fleng=23.0*Displ^.342
2850      IF Code=11 THEN Fleng=25.1*Displ^.324
2855      IF Code=20 THEN Fleng=20.8*Displ^.358
2860      IF Code=21 THEN Fleng=12.1*Displ^.447
2865      IF Code=30 THEN Fleng=19.4*Displ^.352
2870      IF Code=40 THEN Fleng=16.9*Displ^.396
2875      IF Code=50 THEN Fleng=.657*Displ+33
2880      IF Code=60 THEN Fleng=14.6*Displ^.357
2885      IF Code=70 THEN Fleng=.657*Displ+33
2890      IF (Code=80) AND (Displ>1000) THEN Fleng=356.4*LGT(Displ)-859.2
2895      IF (Code=80) AND (Displ<=1000) THEN Fleng=110.0*LGT(Displ)-120.0
2900      RETURN Fleng

```

```

2905      FNEND
2910
2915      FDISP
2920
2925      DISPLACEMENT<IN TONS>
2930
2935      DEF FNFdisp<Code,Leng>
2940      IF Code=10 THEN Fdisp=<Leng/23.0>^2.92
2945      IF Code=11 THEN Fdisp=<Leng/23.1>^3.09
2950      IF Code=20 THEN Fdisp=<Leng/20.8>^2.79
2955      IF Code=21 THEN Fdisp=<Leng/12.1>^2.24
2960      IF Code=30 THEN Fdisp=<Leng/19.4>^2.34
2965      IF Code=40 THEN Fdisp=<Leng/16.9>^2.52
2970      IF Code=50 THEN Fdisp=<Leng-33>/.657
2975      IF Code=60 THEN Fdisp=<Leng/14.6>^2.80
2980      IF Code=70 THEN Fdisp=<Leng-33>/.657
2985      IF <Code=80> AND <Leng<=210> THEN Fdisp=10^(<<Leng+120>/110)
2990      IF <Code=80> AND <Leng>210> THEN Fdisp=10^(<<Leng+859.2>/356.4)
2995      RETURN Fdisp
3000      FNEND
3005
3010      FLTOB
3015
3020      LENGTH TO BEAM RATIO
3025
3030      DEF FNFltob<Code,Leng>
3035      IF Code=10 THEN Fltob=4.0
3040      IF Code=11 THEN Fltob=4.5
3045      IF <Code=20> OR <Code=21> THEN Fltob=2.0
3050      IF Code=30 THEN Fltob=3
3055      IF Code=40 THEN Fltob=5.5
3060      IF Code=50 THEN Fltob=2.5
3065      IF Code=60 THEN Fltob=3
3070      IF Code=70 THEN Fltob=3
3075      IF Code=80 THEN Fltob=5
3080      RETURN Fltob
3085      FNEND
3090
3095      FDTOL
3100
3105      DRAFT TO LENGTH RATIO
3110
3115      DEF FNFdtol<Code,Leng>
3120      IF Code=10 THEN Fdtol=.20
3125      IF Code=11 THEN Fdtol=.15
3130      IF <Code=20> OR <Code=21> THEN Fdtol=.01
3135      IF <Code=30> OR <Code=50> THEN Fdtol=.05
3140      IF <Code=40> OR <Code=70> OR <Code=80> THEN Fdtol=.06
3145      IF Code=60 THEN Fdtol=.10
3150      RETURN Fdtol
3155
3160      USEABLE DECK AREA IN SQUARE FEET
3165
3170      DEF FNFdeck<Code,Leng,Beam>
3175      IF <Code=20> OR <Code=21> THEN Da=.50
3180      IF <Code=10> OR <Code=11> OR <Code=40> OR <Code=80> THEN Da=.25
3185      IF Code=30 THEN Da=.75
3190      IF Code=50 THEN Da=.40
3195      IF Code=60 THEN Da=.55
3200      IF Code=70 THEN Da=.30
3205      Fdeck=Leng*Beam*Da
3210      RETURN Fdeck
3215      FNEND
3220
3225      FUSELD
3230

```

```

3235 ! TOTAL USEFUL USELOAD (TONS)
3240 !
3245 DEF FNFuselD(Code,Displ)
3250 IF Code=10 THEN FuselD=FNFF(Displ,20,8,400,122)
3255 IF Code=11 THEN FuselD=FNFF(Displ,50,12,335,98)
3260 IF Code=20 THEN FuselD=FNFF(Displ,15,6,200,68)
3265 IF Code=21 THEN FuselD=FNFF(Displ,20,8,200,38)
3270 IF Code=30 THEN FuselD=FNFF(Displ,90,35,180,70)
3275 IF Code=40 THEN FuselD=.525*Displ-7.5
3280 IF Code=50 THEN FuselD=FNFF(Displ,20,8,200,60)
3285 IF Code=60 THEN FuselD=FNFF(Displ,700,200,4250,1250)
3290 IF Code=70 THEN FuselD=FNFF(Displ,40,12,400,100)
3295 IF Code=80 THEN FuselD=FNFFlglg(Displ,3.5,1,3000,1000)
3300 RETURN FuselD
3305 FNEND
3310 !
3315 ! FHPBIN
3320 !
3325 ! INSTALLED BASE HORSEPOWER
3330 ! (FOR A CRAFT WITH DESIGN SPEED=BASE SPEED)
3335 !
3340 DEF FNFhpbin(Code,Displ)
3345 IF Code=10 THEN Fhpbin=FNFF(Displ,55,2750,300,20000)
3350 IF Code=11 THEN Fhpbin=FNFF(Displ,20,1000,250,14000)
3355 IF Code=20 THEN Fhpbin=FNFF(Displ,8,1570,80,6606)
3360 IF Code=21 THEN Fhpbin=FNFF(Displ,25,2500,150,17000)
3365 IF Code=30 THEN Fhpbin=FNFF(Displ,0,4000,180,10000)
3370 IF Code=40 THEN Fhpbin=FNFF(Displ,35,3000,155,14500)
3375 IF Code=50 THEN Fhpbin=FNFF(Displ,64,3400,74,3600)
3380 IF Code=70 THEN Fhpbin=FNFF(Displ,25,762,200,6845)
3385 IF Code=60 THEN Fhpbin=FNFF3(Displ,200,2000,1000,6800,4000,12800)
3390 IF Code=80 THEN Fhpbin=10^(.659*LGT(Displ)+2.2648)
3395 RETURN Fhpbin
3400 FNEND
3405 !
3410 ! FSURVI
3415 !
3420 ! SURVIVABILITY
3425 !
3430 DEF FNFsurvi(Code,Leng)
3435 Ssur=3.0
3440 IF (Code=20) OR (Code=21) THEN Ssur=2.0
3445 IF Code=50 THEN Ssur=3.5
3450 IF Code=60 THEN Ssur=4.0
3455 Fsurvi=.02*Leng+Ssur
3460 IF Fsurvi>7 THEN Fsurvi=7
3465 RETURN Fsurvi
3470 FNEND
3475 !
3480 ! FENG
3485 !
3490 ! ENGINE TYPE
3495 !
3500 DEF FNFeng(Code,Rate)
3505 OPTION BASE 1
3510 DIM Alldsl(4),Allgt(4),Gt3dsl(4),Gt1dsl(4)
3515 INTEGER Alldsl,Allgt,Gt3dsl,Gt1dsl,Feng
3520 DATA 2,2,2,2
3525 MAT READ Alldsl
3530 DATA 1,1,1,1
3535 MAT READ Allgt
3540 DATA 1,1,1,2
3545 MAT READ Gt3dsl
3550 DATA 1,2,2,2
3555 MAT READ Gt1dsl
3560 !

```

```

3565         IF (Code=11) OR (Code=50) OR (Code=70) THEN Feng=Allids1(Rate)
3570         IF (Code=20) OR (Code=21) OR (Code=60) OR (Code=30) THEN Feng=Allgt(
Rate)
3575         IF (Code=10) OR (Code=40) THEN Feng=Gt3ds1(Rate)
3580         IF Code=80 THEN Feng=Gt1ds1(Rate)
3585         RETURN Feng
3590         FNEND
3595
3600 ! FBSSPD
3605 !
3610 ! BASE SPEED
3615 ! (USED FOR BASE CURVES FOR HPINST,ACCEL,BRKG,& TURN)
3620 !
3625         DEF FNFbsspd(Code)
3630         IF (Code=10) OR (Code=20) OR (Code=21) THEN Fbsspd=50
3635         IF (Code=11) OR (Code=70) THEN Fbsspd=40
3640         IF Code=30 THEN Fbsspd=60
3645         IF Code=40 THEN Fbsspd=45
3650         IF Code=50 THEN Fbsspd=30
3655         IF Code=60 THEN Fbsspd=20
3660         IF Code=70 THEN Fbsspd=40
3665         IF Code=80 THEN Fbsspd=25
3670         RETURN Fbsspd
3675         FNEND
3680 !
3685 ! FCWSPD
3690 !
3695 ! CALM WATER SPEED AT GOOD VISIBILITY
3700 !
3705         DEF FNFcwsdp(Code,Rate,Dspeed)
3710         IF Rate=1 THEN Fcwsdp=Dspeed
3715         IF Rate=2 THEN L2
3720         IF Rate=3 THEN Fcwsdp=12
3725         IF Rate=4 THEN Fcwsdp=5
3730         RETURN Fcwsdp
3735 L2: IF Code=10 THEN Fcwsdp=.85*Dspeed
3740         IF Code=11 THEN Fcwsdp=.9*Dspeed
3745         IF (Code=30) OR (Code=40) OR (Code=50) OR (Code=70) THEN Fcwsdp=.875
*Dspeed
3750         IF (Code=20) OR (Code=21) THEN Fcwsdp=.85*Dspeed
3755         IF Code=60 THEN Fcwsdp=.60*Dspeed
3760         IF Code=80 THEN Fcwsdp=.5*Dspeed
3765         RETURN Fcwsdp
3770         FNEND
3775 !
3780 ! FTOWDS
3785 !
3790 ! TOW DISPLACEMENT CAPABILITY IN TONS
3795 !
3800         DEF FNFtowds(Code,Displ)
3805         F=10
3810         IF (Code=20) OR (Code=21) THEN F=2
3815         IF Code=60 THEN F=5
3820         Ftowds=F*Displ*(Displ/100)^.3333
3825         RETURN Ftowds
3830         FNEND
3835 !
3840 ! FSFCEN
3845 !
3850 ! SPECIFIC FUEL CONSUMPTION (LBS PER HORSEPOWER HOUR PER ENGINE)
3855 !
3860         DEF FNFsfcen(Eng,Hpinst)
3865         Hpinst2=Hpinst/2
3870         IF Eng=2 THEN Fsfcen=.35
3875         IF Eng=1 THEN Fsfcen=FNFf3(Hpinst2,400,.7,4000,.48,16000,.40)
3880         RETURN Fsfcen

```

```

3885      FNEND
3890
3895      ! FSFCCF
3900
3905      ! SPECIFIC FUEL CONSUMPTION CORRECTION FACTOR
3910
3915      DEF FNFsfccf(Eng,Hpfctu)
3920      IF (Eng=1) AND (Hpfctu>.5) THEN Fsfccf=-.4*Hpfctu+1.4
3925      IF (Eng=1) AND (Hpfctu>.25) AND (Hpfctu<=.5) THEN Fsfccf=-1.6*Hpfctu
+2
3930      IF (Eng=1) AND (Hpfctu<=.25) THEN Fsfccf=-3.2*Hpfctu+2.4
3935      IF Eng=2 THEN Fsfccf=1
3940      RETURN Fsfccf
3945      FNEND
3950
3955      ! FHPFCT
3960
3965      ! FRACTION OF INSTALLED HORSEPOWER UTILIZED
3970
3975      DEF FNFhpfct(Code,Rate,Fctdsp,Fctbsp)
3980
3985      IF (Rate=1) OR (Rate=2) THEN Pctdsp=100*Fctdsp
3990      IF (Rate=3) OR (Rate=4) OR (Rate=0) THEN Pctdsp=100*Fctbsp
3995
4000      IF (Code<>10) AND (Code<>11) THEN L20
4005      Hppct=FNFF3(Pctdsp,0,5,20,14,40,48,85,75,100,100)
4010      GOTO L99
4015      L20: IF (Code<>20) AND (Code<>21) THEN L30
4020      IF Pctdsp<=10 THEN Hppct=10
4025      IF (Pctdsp<=30) AND (Pctdsp>10) THEN Hppct=2*Pctdsp-10
4030      IF (Pctdsp<=80) AND (Pctdsp>30) THEN Hppct=.20*Pctdsp+44
4035      IF Pctdsp>80 THEN Hppct=2*Pctdsp-100
4040      GOTO L99
4045      L30: IF Code<>30 THEN L40
4050      Hppct=FNFF3(Pctdsp,0,10,80,60,100,100)
4055      GOTO L99
4060      L40: IF Code<>40 THEN L50
4065      IF Pctdsp<=20 THEN Hppct=5
4070      IF (Pctdsp<=40) AND (Pctdsp>20) THEN Hppct=2.75*Pctdsp-50
4075      IF (Pctdsp<=80) AND (Pctdsp>40) THEN Hppct=.25*Pctdsp+50
4080      IF Pctdsp>80 THEN Hppct=1.5*Pctdsp-50
4085      GOTO L99
4090      L50: IF Code<>50 THEN L60
4095      IF Pctdsp<=10 THEN Hppct=.50*Pctdsp+5
4100      IF (Pctdsp<=30) AND (Pctdsp>10) THEN Hppct=2.5*Pctdsp-15
4105      IF (Pctdsp<=80) AND (Pctdsp>30) THEN Hppct=.20*Pctdsp+54
4110      IF Pctdsp>80 THEN Hppct=1.5*Pctdsp-50
4115      GOTO L99
4120      L60: IF Code<>60 THEN L70
4125      Hppct=FNFF4(Pctdsp,0,5,30,12,70,45,100,100)
4130      GOTO L99
4135      L70: IF Code<>70 THEN L80
4140      IF Pctdsp<=10 THEN Hppct=5
4145      IF Pctdsp>10 THEN Hppct=1.05536*Pctdsp-5.5536
4150      GOTO L99
4155      L80: IF Pctdsp<=20 THEN Hppct=10
4160      IF (Pctdsp<=60) AND (Pctdsp>20) THEN Hppct=.25*Pctdsp+5
4165      IF Pctdsp>60 THEN Hppct=2*Pctdsp-100
4170      L99: Fhpct=Hppct/100
4175      RETURN Fhpct
4180      FNEND
4185
4190      ! FTNRAD
4195
4200      DEF FNFtnrad(Code,Cwspd)
4205      IF Code=10 THEN Omega=8

```

```

4210      IF (Code=20) OR (Code=21) THEN Omega=2
4215      IF Code=30 THEN Omega=1.5
4220      IF Code=40 THEN Omega=4
4225      IF (Code=11) OR (Code=50) THEN Omega=3
4230      Ftnrad=1.689*Cwsprd/(3.14159265/180*Omega)
4235      RETURN Ftnrad
4240      FEND
4245      !
4250      ! FMWTAV
4255      !
4260      ! WEIGHTED AVERAGE MOTION OF CRAFT
4265      !
4270      DEF FNFmtau(Ssprbd(*),Code,Disp1,Rate)
4275      OPTION BASE 1
4280      Sam=0
4285      !
4290      ! LAMBDA CONVERTS WAVE HEIGHT FROM CRAFT DISPLACEMENT TO
4295      ! BASE DISPLACEMENT (=100 TONS)
4300      !
4305      Lambda=(100/Disp1)^.333
4310      IF Code=60 THEN Lambda=(1500/Disp1)^.333
4315      FOR Ss1=1 TO 8
4320      Ss=Ss1-1
4325      Wuhtcf=.5*(-1+2.5*EXP(.4*Ss))
4330      Wuhtbs=Lambda*Wuhtcf
4335      Flag=0
4340      Sam=Sam+Ssprbd(Ss1)*FNFmuswh(Code,Rate,Wuhtbs,Flag)
4345      NEXT Ss1
4350      Fmtau=Sam
4355      PRINTER IS 16
4360      RETURN Fmtau
4365      FEND
4370      !
4375      ! FF
4380      !
4385      ! FINDS Y VALUE ON A STRAIGHT LINE, GIVEN X VALUE AND TWO POINTS
4390      ! ON A LINE (ASSUMING LINE EXTENDS INFINITELY)
4395      !
4400      DEF FNFF(X,X1,Y1,X2,Y2)
4405      IF ABS(X2-X1)<.0001 THEN L1
4410      Slope=(Y2-Y1)/(X2-X1)
4415      B=Y1-Slope*X1
4420      Ff=Slope*X+B
4425      RETURN Ff
4430      L1: Ff=(Y1+Y2)/2
4435      RETURN Ff
4440      FEND
4445      !
4450      ! FF3
4455      !
4460      ! FINDS Y VALUE ON BROKEN LINE OF 5 POINTS, GIVEN X VALUE
4465      ! AND THE 5 POINTS
4470      ! (ASSUMING ENDS OF LINE EXTEND INFINITELY)
4475      !
4480      DEF FNFF5(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5)
4485      IF X<=X2 THEN Ff5=FNFF(X,X1,Y1,X2,Y2)
4490      IF (X>X2) AND (X<=X3) THEN Ff5=FNFF(X,X2,Y2,X3,Y3)
4495      IF (X>X3) AND (X<=X4) THEN Ff5=FNFF(X,X3,Y3,X4,Y4)
4500      IF X>X4 THEN Ff5=FNFF(X,X4,Y4,X5,Y5)
4505      RETURN Ff5
4510      FEND
4515      !
4520      ! FF3
4525      !
4530      ! FINDS Y VALUE ON BROKEN LINE OF 3 POINTS, GIVEN X VALUE
4535      ! AND THE 3 POINTS

```

```

4540 !      (ASSUMING ENDS OF LINE EXTEND INFINITELY)
4545 !
4550      DEF FNFF3(X,X1,Y1,X2,Y2,X3,Y3)
4555      IF X<=X2 THEN FF3=FNFF(X,X1,Y1,X2,Y2)
4560      IF X>X2 THEN FF3=FNFF(X,X2,Y2,X3,Y3)
4565      RETURN FF3
4570      FEND
4575 !
4580 !      FFF
4585 !
4590 !      READ CUVRE Y VS X,OR X VS Y DEPENDING UPON FLAG
4595 !      CURVE IS A STRAIGHT LINE
4600 !
4605 !      FLAG = 0 MEANS Y VS X
4610 !      FLAG = 1 MEANS X VS Y (NEGATIVE SLOPE)
4615 !      FLAG = 2 MEANS X VS Y (POSITIVE SLOPE)
4620 !
4625      DEF FNFFF(Xory,Flag,X1,Y1,X2,Y2)
4630      IF Flag=0 THEN Fff=FNFF(Xory,X1,Y1,X2,Y2)
4635      IF Flag=1 THEN Fff=FNFF(Xory,Y2,X2,Y1,X1)
4640      IF Flag=2 THEN Fff=FNFF(Xory,Y1,X1,Y2,X2)
4645      RETURN Fff
4650      FEND
4655 !
4660 !      FFF3
4665 !
4670 !      READ CURVE Y VS X OR X VS Y DEPENDING UPON FLAG
4675 !      CURVE IS A BROKEN LINE OF 3 POINTS
4680 !
4685 !      FLAG = 0 MEANS Y VS X
4690 !      FLAG = 1 MEANS X VS Y (NEGATIVE SLOPE)
4695 !      FLAG = 2 MEANS X VS Y (POSITIVE SLOPE)
4700 !
4705      DEF FNFFF3(Xory,Flag,X1,Y1,X2,Y2,X3,Y3)
4710      IF Flag=0 THEN Fff3=FNFF3(Xory,X1,Y1,X2,Y2,X3,Y3)
4715      IF Flag=1 THEN Fff3=FNFF3(Xory,Y3,X3,Y2,X2,Y1,X1)
4720      IF Flag=2 THEN Fff3=FNFF3(Xory,Y1,X1,Y2,X2,Y3,X3)
4725      RETURN Fff3
4730      FEND
4735 !
4740 !      FMVSWH
4745 !
4750 !      MOTION OF BASE CRAFT VS. WAVE HEIGHT (FOR DISPLACEMENT=100 TONS
4755 !      EXCEPT TYPE 60 DISPLACEMENT=1500 TONS) AND REVERSE
4760 !
4765      DEF FNMvswH(Code,Rate,Wuhtbs,Flag)
4770      In=Wuhtbs
4775      IF Code<>10 THEN L11
4780      IF (Rate=1) OR (Rate=2) THEN Out=FNFFF3(In,Flag,0,0,15,.5,10,1)
4785      IF Rate=3 THEN Out=FNFFF(In,Flag,0,0,3,1)
4790      IF Rate=4 THEN Out=FNFFF(In,Flag,0,0,12,1)
4795      GOTO L999
4800 L11:
4805      IF Code<>11 THEN L20
4810      IF (Rate=1) OR (Rate=2) THEN Out=FNFFF3(In,Flag,0,0,5,.25,8,1)
4815      IF Rate=3 THEN Out=FNFFF(In,Flag,0,0,10,1)
4820      IF Rate=4 THEN Out=FNFFF(In,Flag,0,0,14,1)
4825      GOTO L999
4830 L20:
4835      IF (Code<>20) OR (Code<>21) THEN L40
4840      IF Rate=1 THEN Out=FNFFF3(In,Flag,0,0,4,.2,5,1)
4845      IF Rate=2 THEN Out=FNFFF3(In,Flag,0,0,4,.2,6,1)
4850      IF Rate=3 THEN Out=FNFFF3(In,Flag,0,0,4,.2,7,1)
4855      IF Rate=4 THEN Out=FNFFF3(In,Flag,0,0,4,.2,10,1)
4860      GOTO L999
4865 L40:
4870      IF Code<>30 THEN L50
4875      IF Rate=1 THEN Out=FNFFF3(In,Flag,0,0,5,.2,7,1)
4880      IF Rate=2 THEN Out=FNFFF3(In,Flag,0,0,5,.2,8,1)

```

```

4870      IF Rate=3 THEN Out=FNFFF3(In,Flag,0,0,5,.2,10,1)
4875      IF Rate=4 THEN Out=FNFFF3(In,Flag,0,0,5,.2,12,1)
4880      GOTO L999
4885 L50:  IF Code<>40 THEN L60
4890      IF Rate=1 THEN Out=FNFFF(In,Flag,0,0,3,1)
4895      IF Rate=2 THEN Out=FNFFF(In,Flag,0,0,4,1)
4900      IF Rate=3 THEN Out=FNFFF(In,Flag,0,0,8,1)
4905      IF Rate=4 THEN Out=FNFFF(In,Flag,0,0,12,1)
4910      GOTO L999
4915 L60:  IF (Code<>50) AND (Code<>70) AND (Code<>80) AND (Code<>100) THEN L
80
4920      IF Rate=1 THEN Out=FNFFF(In,Flag,0,0,5,1)
4925      IF Rate=2 THEN Out=FNFFF(In,Flag,0,0,5,1)
4930      IF Rate=3 THEN Out=FNFFF(In,Flag,0,0,8,1)
4935      IF Rate=4 THEN Out=FNFFF(In,Flag,0,0,12,1)
4940      GOTO L999
4945      ! FOR CODE = 60
4950 L80:  Out=FNFFF3(In,Flag,0,0,.17,143,.343,21.01,1)
4955 L999:  Fmusuh=Out
4960      ! IF Fmusuh>1 THEN Fmusuh=1
4965      IF Flag=2 THEN Fmusuh=Out
4970      RETURN Fmusuh
4975      FNEND
4980      !
4985      !
4990      ! FF4
4995      !
5000      ! FINDS Y VALUE ON BROKEN LINE OF 4 POINTS, GIVEN X VALUE
5005      ! AND THE 4 POINTS
5010      ! (ASSUMING ENDS OF LINE EXTEND INFINITELY)
5015      !
5020      DEF FNFF4(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4)
5025      IF X<=X2 THEN FF4=FNFF(X,X1,Y1,X2,Y2)
5030      IF (X>X2) AND (X<=X3) THEN FF4=FNFF(X,X2,Y2,X3,Y3)
5035      IF X>X3 THEN FF4=FNFF(X,X3,Y3,X4,Y4)
5040      RETURN FF4
5045      FNEND
5050      !
5055      !
5060      ! FFLGLG
5065      !
5070      ! FINDS Y VALUE ON A STRAIGHT LINE ON LOG-LOG PAPER,
5075      ! GIVEN X VALUE AND 2 POINTS ON THE LINE (ASSUMING LINE
5080      ! EXTENDS INFINITELY)
5085      !
5090      DEF FNFF1glg(X,X1,Y1,X2,Y2)
5095      IF X1=X2 THEN L1
5100      Slope=(LOG(Y2)-LOG(Y1))/(LOG(X2)-LOG(X1))
5105      B=LOG(Y1)-Slope*LOG(X1)
5110      FF1glg=EXP(Slope*LOG(X)+B)
5115      RETURN FF1glg
5120      FNEND

```

APPENDIX C

```

475 !
480 ! ***** S P T P O S *****
485 !
490 ! SPTPOS SUBROUTINE
495 !
500 ! COMPUTES CRAFT PARAMETERS AND TASK PROBABILITIES OF SUCCESS
505 ! FOR A CRAFT
510 ! TO FIND CRAFT PARAMETERS:
515 !
520 Sptpos:      ! CONVERTED SUBROUTINE SPTPOS
525      DISP " LINK COMPLETE, CALCULATING PERFORMANCE IN SEAWAY"
530      FOR I=1 TO 8
535      Craft$(I)=" "
540      NEXT I
545      DATA .7,.6,.8,.5,.9,.5,.5,1.0,99.,99.,1.0,1.0
550      DATA 99.,1.0,99.,1.0,99.,99.,99.
555      MAT READ Mo
560 !
565      DATA 9999.,9999.,9999.,9999.
570      MAT READ Xx
575      DATA 0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0
580      MAT READ Avesx
585 !
590      DATA .5,.7,1.,2.,10.,1.,2.,4.,6.,20.,2.5,4.,7.,20.,50.
595      DATA 7.,10.,20.,50.,100.,10.,30.,60.,80.,300.
600      DATA 50.,100.,500.,1000.,10000.
605      MAT READ Towdis
610 !
615      DATA .9,.7,.5,.1,.2,.3,.0,.1,.2
620      MAT READ Visdis
625      DATA 99999.,20.,10.
630      MAT READ Vmxvis
635 !
640      DATA 15.,32.5,48.2,9999.,9999.,63.9,9999.,73.6,130.,9999.
645      DATA 9999.,1266.7
650      MAT READ Cgfr20
655      DATA 5.1,9.1,16.0,20.7,21.2,18.6,27.8,23.6,27.9
660      DATA 70.6,89.4,221.4
665      MAT READ Cgfr10
670 !
675      IF Code=10 THEN Typnum=1
680      IF Code=11 THEN Typnum=2
685      IF Code=20 THEN Typnum=3
690      IF Code=21 THEN Typnum=4
695      IF Code=30 THEN Typnum=5
700      IF Code=40 THEN Typnum=6
705      IF Code=50 THEN Typnum=7
710      IF Code=60 THEN Typnum=8
715      IF Code=70 THEN Typnum=9
720      IF Code=80 THEN Typnum=10
725      IF Code>=100 THEN Cgtype=Code-100
730 !
735      FOR Rate=1 TO 4
740      IF Code>=100 THEN Sfceng(Rate)=0
745      IF Code>=100 THEN Sfccf(Rate)=0
750      IF Code>=100 THEN Totstfc(Rate)=0
755      IF Code>=100 THEN Sfcgal(Rate)=0
760      NEXT Rate
765 !
770 ! FIND PARAMETERS
775 !
780 ! ALL PARAMETER PROBABILITIES SET EQUAL TO 1.0 EXCEPT
785 ! WHERE RECALCULATED BELOW
790 !
795      FOR Jtpos=1 TO 19
800      Cc(Jtpos)=1

```

```

805      Df(Jtpos)=1
810      Ls(Jtpos)=1
815      Mn(Jtpos)=1
820      Sk(Jtpos)=1
825      Tw(Jtpos)=1
830      NEXT Jtpos
835      !
840      ! AVERAGE SPEED AND AVERAGE FUEL RATE (IN EXPECTED SEA STATES
845      ! AND VISIBILITIES)
850      !
855      Visfue(1)=99999
860      ! FUEL RATE IN LIMITED VISIBILITY
865      !
870      IF Code>100 THEN L8900
875      FOR Vistyp=2 TO 3 STEP 1
880      Vuis=Vmxvis(Vistyp)
885      Fctbsp=Vuis/FNFbsspd(Code)
890      Rate=0
895      Hpfctu=FNHfpfct(Code,Rate,0,Fctbsp)
900      Zhptil=Hpfctu*FNHfpbin(Code,Disp1)
905      IF Vistyp=2 THEN Zeng=FNFeng(Code,2)
910      IF Vistyp=3 THEN Zeng=FNFeng(Code,3)
915      Zsfcen=FNFsfcen(Zeng,Hpinst)
920      Zsfccf=FNFsfcf(Zeng,Hpfctu)
925      Ztsfc=Zsfcen*Zsfccf
930      Zsfcl=Ztsfc*335/2240
935      Visfue(Vistyp)=Zhptil*Zsfcl
940      NEXT Vistyp
945      GOTO L8910
950      !
955      ! FUEL RATES FOR COAST GUARD CRAFT IN LIMITED VISIBILITY
960      !
965      L8900: Visfue(2)=Cgfr20(Cgtype)
970      Visfue(3)=Cgfr10(Cgtype)
975      L8910: FOR Rate=1 TO 4
980      GOSUB Vutau
985      Speed(Rate)=Vaug
990      Mfulrt(Rate)=Aufurt
995      NEXT Rate
1000      !
1005      ! GO: GO FRACTION (USED IN LIMITING SEA STATE PARAMETER)
1010      !
1015      DATA 15.,8.,5.,0.
1020      MAT READ Gomin
1025      FOR Rate=1 TO 4
1030      IF Dspeed(Gomin(Rate)) THEN L4702
1035      Pcdspd=Gomin(Rate)/Dspeed*100
1040      Flag=1
1045      Ssmx=FNHpdsss(Code,Disp1,Rate,Dspeed,Pcdspd,Flag)
1050      Go(Rate)=FNHcpbss(Ssprob(*),Sspdtb,Ssmx)
1055      GOTO L4701
1060      L4702: Go(Rate)=0
1065      L4701: NEXT Rate
1070      !
1075      ! TW: TOW FRACTION PARAMETER
1080      !
1085      GOSUB Ptud
1090      Tw(12)=Ptoud
1095      Fctdsp=Autuds/Disp1
1100      Towspd=FNHf3(Fctdsp,0,5,.2,10,10,0)
1105      !
1110      ! SK: SEAKINDLINESS PARAMETER (USED IN LIMITING SEA STATE PARAMETER)
1115      !
1120      FOR Jtpos=1 TO 19
1125      IF Mo(Jtpos)=99 THEN L9020
1130      Mtn=Mo(Jtpos)

```

```

1135      IF Jtpos<7 THEN Rate=4
1140      IF (Jtpos>=8) AND (Jtpos<=12) THEN Rate=3
1145      IF (Jtpos>=13) AND (Jtpos<=18) THEN Rate=2
1150      IF Jtpos=19 THEN Rate=1
1155      Flag=2
1160      Wuhtbs=FNFMuswh(Code,Rate,Mtn,Flag)
1165      Lambda=(100/Displ)^.333
1170      IF Type=60 THEN Lambda=(1500/Displ)^.333
1175      Wuhtcf=Wuhtbs/Lambda
1180      Arg=.8*Wuhtcf+.4
1185      Ss=2.5*LOG(Arg)
1190      Sk(Jtpos)=FNFCpbss(Ssprob(*),Sspdtb,Ss)
1195 L9020: NEXT Jtpos
1200      !
1205      ! MN: MANEUVERABILITY PARAMETER
1210      !
1215      Mn(1)=FNFF4(Leng,0,1,50,1,200,.3,99999,.8)
1220      Mn(2)=Mn(1)
1225      Mn(3)=Mn(1)
1230      Mn(4)=Mn(1)
1235      Mn(7)=Mn(1)
1240      Mn(12)=FNFF4(9999999,0,1,500,1,1500,.5,99999,.5)
1245      Mn(14)=FNFF4(Tnrad(2),0,1,500,1,1500,.5,99999,.5)
1250      !
1255      ! SU: SURVIVABILITY (USED IN LIMITING SEA STATE PARAMETER)
1260      !
1265      Su0=FNFCpbss(Ssprob(*),Sspdtb,Isurv)
1270      Su(1)=Su0
1275      Su(2)=Su0
1280      Su(3)=Su0
1285      Su(4)=Su0
1290      !
1295      ! LS: LIMITING SEA STATE PARAMETER
1300      !
1305      FOR Rate=1 TO 4
1310      Lis(Rate)=Go(Rate)
1315      IF Su(Rate)<Go(Rate) THEN Lis(Rate)=Su(Rate)
1320      NEXT Rate
1325      FOR Jtpos=1 TO 19
1330      IF Jtpos<=7 THEN Ls(Jtpos)=MIN(Lis(4),Sk(Jtpos))
1335      IF (Jtpos>=8) AND (Jtpos<=12) THEN Ls(Jtpos)=MIN(Lis(3),Sk(Jtpos))
1340      IF (Jtpos>=13) AND (Jtpos<=18) THEN Ls(Jtpos)=MIN(Lis(2),Sk(Jtpos))
1345      IF Jtpos=19 THEN Ls(Jtpos)=MIN(Lis(1),Sk(Jtpos))
1350      NEXT Jtpos
1355      !
1360      ! CC: CARGO CAPACITY
1365      !
1370      Cc(17)=0 ! ORIGINALLY SET TO 999. CAUSED OUTPUT CONVERSION ERROR IN
FORTRAN VERSION OF THIS PROGRAM.
1375      !
1380      ! DF: DRAFT PARAMETER
1385      !
1390      Df0=1-FNFPdpth(Dphdtb,Draf)
1395      Df(1)=Df0
1400      Df(2)=Df0
1405      Df(3)=Df0
1410      Df(4)=Df0
1415      Df(6)=Df0
1420      Df(7)=Df0
1425      Df(8)=Df0
1430      Df(10)=Df0
1435      Df(11)=Df0
1440      Df(16)=Df0
1445      !
1450      ! PRINT MFULRT,SPEED AND TOWSPD AT END OF CHARACTERISTICS LIST
1455      !

```

```

1460      PRINT USING L3041;"AVG FUEL RATE",Mfuelrc(*),"GAL/HR"
1465 L3041: IMAGE 10X,K,3X,4(6D.D),4X,K
1470      PRINT USING L3042;"AVG SPEED",Speed(*),"KNOTS"
1475 L3042: IMAGE 10X,K,7X,4(6D.D),4X,K
1480      PRINT USING L3043;"TOW SPEED","-","-",Towspd,"-","KNOTS"
1485 L3043: IMAGE 10X,K,13X,K,7X,K,1X,6D.D,6X,K,5X,K
1490      PRINTER IS 16
1495      !
1500      ! PRINT PARAMETER VALUES FOR MASTER TASKS
1505      !
1510 L4795: Ssaug=Avesc(Sspdtb)
1515      Idisp=Displ
1520      Ileng=Leng
1525      Idspd=Dspeed
1530      PRINT
1535      DISP "                END OF CRAFT CHARACTERISTICS, PRESS CONT TO GO ON"
1540      BEEP
1545      PAUSE
1550      DISP
1555      PRINT PAGE
1560      INPUT "WOULD YOU LIKE A HARD COPY LISTING (Y/N)? ",Ans$
1565      IF Ans$="Y" THEN PRINTER IS 0
1570      PRINT PAGE
1575      PRINT USING L4091;"C R A F T P A R A M E T E R S"
1580 L4091: IMAGE 22X,K
1585      PRINT
1590      PRINT
1595      IF Code>100 THEN L502
1600      PRINT USING L4936;"CRAFT TYPE",Crfrm$
1605 L4936: IMAGE 22X,K,5X,K
1610      GOTO L505
1615 L502: PRINT USING L4935;"CRAFT TYPE","COAST GUARD",Crfrm$
1620 L4935: IMAGE 22X,K,5X,K,1X,K
1625 L505: PRINT USING L4937;"DISPLACEMENT",Idisp,"TONS"
1630 L4937: IMAGE 22X,K,6D,2X,K
1635      PRINT USING L4945;"LENGTH",Ileng,"FEET"
1640 L4945: IMAGE 22X,K,6X,6D,2X,K
1645      PRINT USING L5001;"DESIGN SPEED",Idspd,"KNOTS"
1650 L5001: IMAGE 22X,K,6D,2X,K
1655      PRINT USING L5002;"FUEL FRACTION",Fufrc
1660 L5002: IMAGE 22X,K,3D.2D
1665      PRINT
1670      PRINT USING L5006;"VISIBILITY DISTRIBUTION NO.",Visdtb
1675      PRINT USING L5006;"TOW DISTRIBUTION NO.",Towdtb
1680      PRINT USING L5006;"DEPTH DISTRIBUTION NO.",Dphdtb
1685      PRINT USING L5006;"SEA STATE DISTRIBUTION NO.",Sspdtb
1690 L5006: IMAGE 26X,K,2D
1695      PRINT USING L5007;"(AVERAGE SEA STATE =",Avesc(Sspdtb),")"
1700 L5007: IMAGE 26X,K,D.D,K
1705      PRINT
1710      PRINT
1715      PRINT USING L5008;"TASK","CARGO","DRAFT","MANEUVR","SEA","TOW"
1720 L5008: IMAGE 14X,K,2X,K,1X,K,1X,K,2X,K,3X,K
1725      PRINT USING L5009;"CODE","CPCTY","STATE"
1730 L5009: IMAGE 14X,K,2X,K,14X,K
1735      PRINT
1740      PRINT
1745      PRINT USING L5010;"CC","DF","MN","SS","TW"
1750 L5010: IMAGE 22X,K,4X,K,4X,K,4X,K,4X,K
1755      PRINT
1760      PRINT
1765      PRINT USING L5034;"ON SCENE:"
1770      PRINT USING L5012;"ASST","--",Df(1),Mn(1),Ls(1),"--","ASSIST"
1775      PRINT USING L5012;"BOARD","--",Df(2),Mn(2),Ls(2),"--","BOARD"
1780      PRINT USING L5012;"MNAC","--",Df(3),Mn(3),Ls(3),"--","MONITOR ACTIVI
TIES"

```

```

1785      PRINT USING L5012;"RTRV","---",Df(4),Mn(4),Ls(4),"---","RETRIEVE"
1790 L5012: IMAGE 14X,K,4X,K,3X,3(D.2D,2X),1X,K,3X,K
1795      PRINT USING L5031;"WAIT","---","---","---",Ls(5),"---","WAIT"
1800      PRINT USING L5029;"WEQD","---",Df(6),"---",Ls(6),"---","WORK EQUIPMENT
0 DRIFT"
1805      PRINT USING L5018;"WEQP","---",Df(7),Mn(7),Ls(7),"---","WORK EQUIPMENT
0 POSITION"
1810 L5018: IMAGE 14X,K,4X,K,3X,D.2D,2X,D.2D,2X,D.2D,3X,K,3X,K
1815      PRINT
1820      PRINT USING L5034;"REDUCED SPEED:"
1825      PRINT USING L5029;"SDIU","---",Df(8),"---",Ls(8),"---","SEARCH FOR DIST
RESSED UNIT"
1830      PRINT USING L5031;"SESC","---","---","---",Ls(9),"---","SLOW ESCORT"
1835      PRINT USING L5029;"SPAT","---",Df(10),"---",Ls(10),"---","SLOW PATROL"
1840      PRINT USING L5029;"SPEO","---",Df(11),"---",Ls(11),"---","SEARCH FOR PE
OPLE"
1845      PRINT USING L5024;"TOWS","---","---",Mn(12),Ls(12),Tw(12),"TOWS"
1850 L5024: IMAGE 14X,K,4X,K,4X,K,3X,3(D.2D,2X),K
1855      PRINT
1860      PRINT USING L5034;"CRUISE SPEED:"
1865      PRINT USING L5033;"ESCT","---","---","---",Ls(13),"---","ESCORT"
1870      PRINT USING L5027;"IDNT","---","---",Mn(14),Ls(14),"---","IDENTIFY"
1875 L5027: IMAGE 14X,K,4X,K,4X,K,3X,D.2D,2X,D.2D,3X,K,3X,K
1880      PRINT USING L5033;"PATL","---","---","---",Ls(15),"---","PATROL"
1885      PRINT USING L5029;"STGT","---",Df(16),"---",Ls(16),"---","SEARCH FOR TA
RGET"
1890 L5029: IMAGE 14X,K,4X,K,3X,D.2D,3X,K,3X,D.2D,3X,K,3X,K
1895      PRINT USING L5030;"TRPT",Cc(17),"---","---",Ls(17),"---","TRANSPORT"
1900 L5030: IMAGE 14X,K,3X,D.2D,3X,K,4X,K,3X,D.2D,3X,K,3X,K
1905      PRINT USING L5031;"TRST","---","---","---",Ls(18),"---","TRANSIT"
1910 L5031: IMAGE 14X,K,4X,K,4X,K,4X,K,3X,D.2D,3X,K,3X,K
1915      PRINT
1920      PRINT USING L5034;"FLANK SPEED:"
1925      PRINT USING L5031;"RSPD","---","---","---",Ls(19),"---","RESPOND"
1930 L5033: IMAGE 14X,K,4X,K,4X,K,4X,K,3X,D.2D,3X,K,3X,K
1935      PRINT
1940      PRINT
1945      PRINT USING L5034;"*** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AND
WEIGHT OF CARGO)"
1950 L5034: IMAGE 10X,K
1955      PRINT
1960      DISP "                END OF SECOND OUTPUT PAGE, PRESS CONT TO GO ON"
1965      BEEP
1970      PAUSE
1975      DISP
1980      !
1985      ! TO FIND TASK PROBABILITIES OF SUCCESS:
1990      !
1995      FOR Jtpos=1 TO 19
2000      IF Jtpos=17 THEN GOTO L9060
2005      Tpos(Jtpos)=Cc(Jtpos)*Df(Jtpos)*Ls(Jtpos)*Mn(Jtpos)*Tw(Jtpos)
2010      GOTO L9059
2015 L9060: Tpos(Jtpos)=0
2020 L9059: NEXT Jtpos
2025      !
2030      ! PRINT TASK PROBABILITIES OF SUCCESS
2035      !
2040      PRINT PAGE
2045      PRINT USING L6031;"T A S K   P R O B A B I L I T I E S   O F   S U C C
E S S"
2050 L6031: IMAGE 13X,K
2055      PRINT
2060      PRINT
2065      IF Code>100 THEN L306
2070      PRINT USING L6032;"CRAFT TYPE",Crnfms
2075 L6032: IMAGE 22X,K,5X,K

```

```

2090      GOTO L507
2095 L506:  PRINT USING L6033;"CRAFT TYPE","COAST GUARD",Crftnm$
2099 L5033: IMAGE 22X,K,5X,K,1X,K
2095 L507:  PRINT USING L4937;"DISPLACEMENT",Idisp,"TONS"
2100      PRINT USING L4945;"LENGTH",Ileng,"FEET"
2105      PRINT USING L5001;"DESIGN SPEED",Idspd,"KNOTS"
2110      PRINT USING L5002;"FUEL FRACTION",Fufrc
2115      PRINT
2120      PRINT USING L6115;"VISIBILITY DISTRIBUTION NO.",Visdtb
2125      PRINT USING L6115;"TOW DISTRIBUTION NO.",Towdtb
2130      PRINT USING L6115;"DEPTH DISTRIBUTION NO.",Dphdtb
2135      PRINT USING L6115;"SEA STATE DISTRIBUTION NO.",Sspdtdb
2140 L6115: IMAGE 26X,K,2D
2145      PRINT USING L6116;"(AVERAGE SEA STATE =",Aves$(Sspdtdb),")"
2150 L6116: IMAGE 26X,K,D.D,K
2155      PRINT
2160      PRINT
2165      PRINT USING L6117;"TASK","TASK PROB","TASK"
2170 L6117: IMAGE 14X,K,3X,K,4X,K
2175      PRINT USING L6001;"CODE","OF SUCCESS"
2180 L6001: IMAGE 14X,K,3X,K
2185      PRINT
2190      PRINT
2195      PRINT USING L6022;"ON SCENE:"
2200      PRINT USING L6013;"ASST",Tpos(1),"ASSIST"
2205      PRINT USING L6013;"BORD",Tpos(2),"BOARD"
2210      PRINT USING L6013;"MNAC",Tpos(3),"MONITOR ACTIVITIES"
2215      PRINT USING L6013;"RTRV",Tpos(4),"RETRIEVE"
2220      PRINT USING L6013;"WAIT",Tpos(5),"WAIT"
2225      PRINT USING L6013;"WEQD",Tpos(6),"WORK EQUIPMENT @ DRIFT"
2230      PRINT USING L6013;"WEQP",Tpos(7),"WORK EQUIPMENT @ POSITION"
2235      PRINT
2240      PRINT USING L6022;"REDUCED SPEED:"
2245      PRINT USING L6012;"SDIU",Tpos(8),"*", "SEARCH FOR DISTRESSED UNIT"
2250      PRINT USING L6013;"SESC",Tpos(9),"SLOW ESCORT"
2255      PRINT USING L6013;"SPAT",Tpos(10),"SLOW PATROL"
2260      PRINT USING L6012;"SPEO",Tpos(11),"*", "SEARCH FOR PEOPLE"
2265      PRINT USING L6013;"TOWS",Tpos(12),"TOWS"
2270      PRINT
2275      PRINT USING L6022;"CRUISE SPEED:"
2280      PRINT USING L6013;"ESCT",Tpos(13),"ESCORT"
2285      PRINT USING L6013;"IDNT",Tpos(14),"IDENTIFY"
2290      PRINT USING L6013;"PATL",Tpos(15),"PATROL"
2295      PRINT USING L6012;"STGT",Tpos(16),"*", "SEARCH FOR TARGET"
2300      PRINT USING L6013;"TRPT",Tpos(17),"TRANSPORT"
2305      PRINT USING L6013;"TRST",Tpos(18),"TRANSIT"
2310      PRINT
2315      PRINT USING L6022;"FLANK SPEED:"
2320 L6022: IMAGE 10X,K
2325      PRINT USING L6013;"RSPD",Tpos(19),"RESPOND"
2330 L6012: IMAGE 14X,K,5X,D.3D,K,4X,K
2335 L6013: IMAGE 14X,K,5X,D.3D,5X,K
2340      PRINT
2345      PRINT
2350      PRINT USING L6024;"*   THIS IS THE P.O.S OF THE ABILITY TO SEARCH CR
AFT' SUCCESS"
2355 L6024: IMAGE 12X,K
2360      PRINT USING L6026;"IN FINDING THE OBJECT OF THE SEARCH IS DEPENDENT
UPON"
2365      PRINT USING L6026;" SCENARIO (E.G., SEARCH AREA)"
2370 L6026: IMAGE 16X,K
2375      PRINT USING L6027;"***** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AN
D WEIGHT OF CARGO)"
2380 L6027: IMAGE 10X,K
2385      PRINT
2390      DISP "          END OF THIRD OUTPUT PAGE. PRESS CONT TO GO ON. "

```

```

2395      BEEP
2400      PAUSE
2405      DISP
2410      !
2415      ! PRINT PARAMETER VALUES FOR EXPANDED TASKS
2420      !
2425      PRINT PAGE
2430      PRINT USING L2031;"C R A F T P A R A M E T E R S"
2435 L2031: IMAGE 24X,K
2440      PRINT
2445      PRINT
2450      IF Code>100 THEN L511
2455      PRINT USING L2032;"CRAFT TYPE",Crfnm$
2460 L2032: IMAGE 24X,K,5X,K
2465      GOTO L512
2470 L511: PRINT USING L2033;"CRAFT TYPE","COAST GUARD",Crfnm$
2475 L2033: IMAGE 24X,K,5X,K,1X,K
2480 L512: PRINT USING L2034;"DISPLACEMENT",Idisp,"TONS"
2485      PRINT USING L2034;"LENGTH",Ileng,"FEET"
2490      PRINT USING L2034;"DESIGN SPEED",Idspd,"KNOTS"
2495 L2034: IMAGE 24X,K,6D,2X,K
2500      PRINT USING L2035;"FUEL FRACTION",Fufrc
2505 L2035: IMAGE 24X,K,3D,2D
2510      PRINT
2515      PRINT USING L2112;"VISIBILITY DISTRIBUTION NO.",Visdtb
2520      PRINT USING L2112;"TOW DISTRIBUTION NO.",Towdtb
2525      PRINT USING L2112;"DEPTH DISTRIBUTION NO.",Dphdtb
2530      PRINT USING L2112;"SEA STATE DISTRIBUTION NO.",Sspdtb
2535 L2112: IMAGE 20X,K,2D
2540      PRINT USING L2113;"(AVERAGE SEA STATE =",Auess(Sspdtb),")"
2545 L2113: IMAGE 20X,K,D,D,K
2550      PRINT
2555      PRINT
2560      PRINT USING L2114;"TASK","CARGO","DRAFT","MANEUVR","SEA","TOW"
2565 L2114: IMAGE 12X,K,2X,K,1X,K,1X,K,2X,K,2X,K
2570      PRINT USING L2115;"CODE","CPCTY","STATE"
2575 L2115: IMAGE 12X,K,2X,K,14X,K
2580      PRINT
2585      PRINT USING L2116;"CC","DF","MN","SS","TW"
2590 L2116: IMAGE 20X,K,4X,K,4X,K,4X,K,3X,K
2595      PRINT
2600      PRINT
2605      PRINT USING L2117;"ON SCENE:"
2610 L2117: IMAGE 8X,K
2615      PRINT USING L2118;"BRD","---",Df(2),Mn(2),Ls(2),"---","BOARD"
2620      PRINT USING L2118;"FFF","---",Df(7),Mn(7),Ls(7),"---","FIGHT FIRE FROM
CG VESSEL"
2625      PRINT USING L2220;"FFO","---",Df(5),Ls(5),"---","FIGHT FIRE ON AND
THER VESSEL"
2630 L2220: IMAGE 12X,K,4X,K,4X,K,4X,K,3X,D,DD,3X,K,3X,K
2635      PRINT USING L2118;"GAS","---",Df(1),Mn(1),Ls(1),"---","GENERAL ASSISTA
NCE"
2640      PRINT USING L2222;"INS","---",Df(5),Ls(5),"---","INSPECTION"
2645      PRINT USING L2118;"LEQ","---",Df(7),Mn(7),Ls(7),"---","LOAD EQUIPMENT"
2650      PRINT USING L2222;"LOI","---",Df(5),Ls(5),"---","LOITER"
2655      PRINT USING L2118;"LSB","---",Df(7),Mn(7),Ls(7),"---","LAUNCH SMALL BO
AT"
2660      PRINT USING L2118;"MAC","---",Df(3),Mn(3),Ls(3),"---","MONITOR ACTIVIT
IES"
2665      PRINT USING L2118;"MOS","---",Df(3),Mn(3),Ls(3),"---","MONITOR OIL SPI
LL"
2670      PRINT USING L2222;"OBA","---",Df(5),Ls(5),"---","ON BOARD ASSISTAN
CE"
2675      PRINT USING L2222;"OSC","---",Df(5),Ls(5),"---","ON SCENE COMMANDE
R(GENERAL)"
2680      PRINT USING L2118;"RBP","---",Df(2),Mn(2),Ls(2),"---","RETRIEVE BOARDI

```

```

NG PARTY"
2685      PRINT USING L2118;"ROB","---",Df(4),Mn(4),Ls(4),"---","RETRIEVE OBJECT
S"
2690      PRINT USING L2118;"RPE","---",Df(4),Mn(4),Ls(4),"---","RESCUE PEOPLE"
2695      PRINT USING L2118;"RSB","---",Df(7),Mn(7),Ls(7),"---","RETRIEVE SMALL
BOAT"
2700      PRINT USING L2118;"SSI","---",Df(3),Mn(3),Ls(3),"---","STAKEOUT SPECIA
L INTEREST VESSEL"
2705      PRINT USING L2222;"SZE","---","---","---",Ls(5),"---","SEIZE"
2710      PRINT USING L2118;"TWS","---",Df(7),Mn(7),Ls(7),"---","TAKE WATER SAMP
LE"
2715      PRINT USING L2118;"ULQ","---",Df(7),Mn(7),Ls(7),"---","UNLOAD EQUIPMEN
T"
2720      PRINT USING L2222;"WQB","---","---","---",Ls(5),"---","WORK EQUIPMENT FR
OM SMALL BOAT"
2725 L2222: IMAGE 12X,K,4X,K,4X,K,4X,K,3X,D.DD,3X,K,3X,K
2730      PRINT USING L2223;"WQD","---",Df(6),"---",Ls(6),"---","WORK EQUIPMENT @
DRIFT"
2735 L2223: IMAGE 12X,K,4X,K,3X,D.DD,3X,K,3X,D.DD,3X,K,3X,K
2740      PRINT USING L2118;"WQF","---",Df(7),Mn(7),Ls(7),"---","WORK EQUIPMEN @
FIXED POSITION"
2745 L2118: IMAGE 12X,K,4X,K,3X,D.DD,2X,D.DD,2X,D.DD,3X,K,3X,K
2750      PRINT
2755      DISP "          END OF FOURTH OUTPUT PAGE. PRESS CONT TO GO ON. "
2760      BEEP
2765      PAUSE
2770      DISP
2775      PRINT PAGE
2780      PRINT USING L3031;"C R A F T   P A R A M E T E R S"
2785 L3031: IMAGE 26X,K
2790      PRINT
2795      PRINT
2800      IF Code>100 THEN L306
2805      PRINT USING L3032;"CRAFT TYPE",Crfrm$
2810 L3032: IMAGE 26X,K,5X,K
2815      GOTO L307
2820 L306:  PRINT USING L3033;"CRAFT TYPE","COAST GUARD",Crfrm$
2825 L3033: IMAGE 26X,K,5X,K,1X,K
2830      PRINT
2835 L307:  PRINT USING L3036;"DISPLACEMENT",Idisp,"TONS"
2840      PRINT USING L3035;"LENGTH",Ileng,"FEET"
2845 L3035: IMAGE 26X,K,6X,6D,2X,K
2850      PRINT USING L3036;"DESIGN SPEED",Idspd,"KNOTS"
2855 L3036: IMAGE 26X,K,6D,2X,K
2860      PRINT USING L3037;"FUEL FRACTION",Fufrc
2865 L3037: IMAGE 26X,K,3D,D
2870      PRINT
2875      PRINT USING L3038;"VISIBILITY DISTRIBUTION NO.",Visdtdb
2880      PRINT USING L3038;"TOW DISTRIBUTION NO.",Towdtdb
2885      PRINT USING L3038;"DEPTH DISTRIBUTION NO.",Dphdtdb
2890      PRINT USING L3038;"SEA STATE DISTRIBUTION NO.",Sspdtdb
2895 L3038: IMAGE 30X,K,2D
2900      PRINT USING L3039;"(AVERAGE SEA STATE=",Aves3(Sspdtdb),")"
2905 L3039: IMAGE 30X,K,D.D,K
2910      PRINT
2915      PRINT
2920      PRINT USING L3001;"TASK","CARGO","DRAFT","MANEUV","SEA","TOW"
2925 L3001: IMAGE 14X,K,2X,K,1X,K,1X,K,2X,K,3X,K
2930      PRINT USING L3002;"CODE","CPCTY","STATE"
2935 L3002: IMAGE 14X,K,2X,K,14X,K
2940      PRINT
2945      PRINT USING L3003;"CC","DF","MN","SS","TW"
2950 L3003: IMAGE 22X,K,4X,K,4X,K,4X,K,4X,K
2955      PRINT
2960      PRINT
2965      PRINT USING L3015;"REDUCED SPEED:"

```

```

2970      PRINT USING L3005;"SDU","---",Df(8),"---",Ls(8),"---","SEARCH FOR DISTR
ESSED UNIT"
2975      PRINT USING L3006;"SES","---","---","---",Ls(9),"---","SLOW ESCORT"
2980 L3006: IMAGE 14X,K,4X,K,4X,K,4X,K,3X,D.2D,3X,K,3X,K
2985      PRINT USING L3005;"SPE","---",Df(11),"---",Ls(11),"---","SEARCH FOR PEO
PLE"
2990      PRINT USING L3005;"SPT","---",Df(10),"---",Ls(10),"---","SLOW PATROL"
2995 L3005: IMAGE 14X,K,4X,K,3X,D.2D,3X,K,3X,D.2D,3X,K,3X,K
3000      PRINT USING L3007;"TOW","---","---",Mn(12),Ls(12),Tw(12),"TOW"
3005 L3007: IMAGE 14X,K,4X,K,4X,K,3X,D.2D,2X,D.2D,2X,D.2D,2X,K
3010      PRINT
3015      PRINT USING L3015;"CRUISE SPEED:"
3020      PRINT USING L3009;"ESC","---","---","---",Ls(13),"---","ESCORT"
3025 L3009: IMAGE 14X,K,4X,K,4X,K,4X,K,3X,D.2D,3X,K,3X,K
3030      PRINT USING L3010;"IDC","---","---",Mn(14),Ls(14),"---","IDENTIFY CRAFT"
3035      PRINT USING L3010;"IDF","---","---",Mn(14),Ls(14),"---","IDENTIFY"
3040 L3010: IMAGE 14X,K,4X,K,4X,K,3X,D.2D,2X,D.2D,3X,K,3X,K
3045      PRINT USING L3011;"PAT","---","---","---",Ls(15),"---","PATROL"
3050      PRINT USING L3011;"SFL","---","---","---",Ls(18),"---","SEARCH FOR FLEET"
3055 L3011: IMAGE 14X,K,4X,K,4X,K,4X,K,3X,D.2D,3X,K,3X,K
3060      PRINT USING L3012;"SSH","---",Df(16),"---",Ls(16),"---","SEARCH FOR SHI
P"
3065 L3012: IMAGE 14X,K,4X,K,3X,D.2D,3X,K,3X,D.2D,3X,K,3X,K
3070      PRINT USING L3013;"TEQ",Cc(17),"---","---",Ls(17),"---","TRANSPORT"
3075 L3013: IMAGE 14X,K,3X,D.2D,3X,K,4X,K,3X,D.2D,3X,K,3X,K
3080      PRINT USING L3014;"TPE","---","---","---",Ls(18),"---","TRANSPORT PEOPLE"
3085      PRINT USING L3014;"TRA","---","---","---",Ls(18),"---","TRANSIT"
3090      PRINT
3095      PRINT USING L3015;"FLANK SPEED:"
3100 L3015: IMAGE 10X,K
3105      PRINT USING L3014;"DSH","---","---","---",Ls(19),"---","DASH"
3110      PRINT USING L3014;"INT","---","---","---",Ls(19),"INTERDICT"
3115 L3014: IMAGE 14X,K,4X,K,4X,K,4X,K,3X,D.2D,3X,K,3X,K
3120      PRINT
3125      PRINT
3130      PRINT USING L3016;"*** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AND
WEIGHT OF CARGO"
3135 L3016: IMAGE 10X,K
3140      PRINT
3145      DISP "                END OF FIFTH PAGE OF OUTPUT. PRESS CONT TO GO ON."
3150      BEEP
3155      PAUSE
3160      DISP
3165      !
3170      ! PRINT EXPANDED TASK PROBABILITIES OF SUCCESS
3175      !
3180      ! TPOS(1) =ASST=GAS
3185      ! TPOS(2) =BORD=BRD=RBP
3190      ! TPOS(3) =MNAC=SSI=MAC=MOS
3195      ! TPOS(4) =RTRV=ROB=RPE
3200      ! TPOS(5) =WAIT=FFO=INS=LOI=OBR=SZE=OSC=WQB
3205      ! TPOS(6) =WEQD=WQD
3210      ! TPOS(7) =WEQP=FFF=LEQ=LSB=RSB=TWS=ULQ=WQF
3215      ! TPOS(8) =SDIU=SDU
3220      ! TPOS(9) =SESC=SES
3225      ! TPOS(10)=SPAT=SPT
3230      ! TPOS(11)=SPEO=SPE
3235      ! TPOS(12)=TOWS=TOW
3240      ! TPOS(13)=ESCT=ESC
3245      ! TPOS(14)=IDNT=IDC=IDF
3250      ! TPOS(15)=PATL=PAT
3255      ! TPOS(16)=STGT=SSH
3260      ! TPOS(17)=TRPT=TEQ=****

```

```

3265 ! TPOS(19)=TRST=SFL=TPE=TRA
3270 ! TPOS(19)=RSPD=INT=DSH
3275 !
3280 PRINT PAGE
3285 PRINT USING L7031;"T A S K P R O B A B I L I T I E S O F S U C C
E S S"
3290 L7031: IMAGE 13X,K
3295 PRINT
3300 PRINT
3305 IF Code>100 THEN L706
3310 PRINT USING L7032;"CRAFT TYPE",Crfrm$
3315 L7032: IMAGE 17X,K,5X,K
3320 GOTO L707
3325 L706: PRINT USING L7033;"CRAFT TYPE","COAST GUARD",Crfrm$
3330 L7033: IMAGE 17X,K,5X,K,1X,K
3335 L707: PRINT USING L7001;"DISPLACEMENT",Idisp,"TONS"
3340 L7001: IMAGE 17X,K,6D,2X,K
3345 PRINT USING L7002;"LENGTH",Ileng,"FEET"
3350 L7002: IMAGE 17X,K,6X,6D,2X,K
3355 PRINT USING L7003;"DESIGN SPEED",Idspd,"KNOTS"
3360 L7003: IMAGE 17X,K,6D,2X,K
3365 PRINT USING L7004;"FUEL FRACTION",Fufrc
3370 L7004: IMAGE 17X,K,3D,2D
3375 PRINT
3380 PRINT USING L7112;"VISIBILITY DISTRIBUTION NO.",Visdtb
3385 PRINT USING L7112;"TOW DISTRIBUTION NO.",Towdtb
3390 PRINT USING L7112;"DEPTH DISTRIBUTION NO.",Dphdtb
3395 PRINT USING L7112;"SEA STATE DISTRIBUTION NO.",Sspdtb
3400 L7112: IMAGE 21X,K,2D
3405 PRINT USING L7113;"(AVERAGE SEA STATE=",Aves$Sspdtb,")"
3410 L7113: IMAGE 21X,K,D,D,K
3415 PRINT
3420 PRINT
3425 PRINT USING L7005;"TASK","TASK PROB.","TASK"
3430 L7005: IMAGE 14X,K,3X,K,4X,K
3435 PRINT USING L7006;"CODE","OF SUCCESS"
3440 L7006: IMAGE 14X,K,3X,K
3445 PRINT
3450 PRINT
3455 PRINT USING L7007;"ON SCENE:"
3460 L7007: IMAGE 10X,K
3465 PRINT USING L7008;"BRD",Tpos(2),"BOARD"
3470 PRINT USING L7008;"FFF",Tpos(7),"FIGHT FIRE FROM CG VESSEL"
3475 PRINT USING L7008;"FFO",Tpos(5),"FIGHT FIRE ON ANOTHER VESSEL"
3480 PRINT USING L7008;"GAS",Tpos(1),"GENERAL ASSISTANCE"
3485 PRINT USING L7008;"INS",Tpos(3),"INSPECTION"
3490 PRINT USING L7008;"LEQ",Tpos(7),"LOAD EQUIPMENT"
3495 PRINT USING L7008;"LOI",Tpos(5),"LOITER"
3500 PRINT USING L7008;"LSB",Tpos(7),"LAUNCH SMALL BOAT"
3505 PRINT USING L7008;"MAC",Tpos(3),"MONITOR ACTIVITIES"
3510 PRINT USING L7008;"MOS",Tpos(3),"MONITOR OIL SPILL"
3515 PRINT USING L7008;"OBA",Tpos(5),"ON BOARD ASSISTANCE"
3520 PRINT USING L7008;"OSC",Tpos(5),"ON SCENE COMMANDER(GENERAL)"
3525 PRINT USING L7008;"RBP",Tpos(2),"RETRIEVE BOARDING PARTY"
3530 PRINT USING L7008;"ROB",Tpos(4),"RETRIEVE OBJECTS"
3535 PRINT USING L7008;"RPE",Tpos(4),"RESCUE PEOPLE"
3540 PRINT USING L7008;"RSB",Tpos(7),"RETRIEVE SMALL BOAT"
3545 PRINT USING L7008;"SSI",Tpos(3),"STAKEOUT SPECIAL INTEREST VESSEL"
3550 PRINT USING L7008;"SZE",Tpos(5),"SEIZE"
3555 PRINT USING L7008;"TWS",Tpos(7),"TAKE WATER SAMPLE"
3560 PRINT USING L7008;"ULQ",Tpos(7),"UNLOAD EQUIPMENT"
3565 PRINT USING L7008;"WQB",Tpos(5),"WORK EQUIPMENT FROM SMALL BOAT"
3570 PRINT USING L7008;"WQD",Tpos(6),"WORK EQUIPMENT @ DRIFT"
3575 PRINT USING L7008;"WQF",Tpos(7),"WORK EQUIPMENT @ FIXED POSITION"
3580 L7008: IMAGE 14X,K,6X,D,3D,5X,K
3585 PRINT

```

```

3590      DISP "                END OF SIXTH PAGE OF OUTPUT. PRESS CONT TO GO ON"
3595      BEEP
3600      PAUSE
3605      DISP
3610      PRINT PAGE
3615      PRINT USING L8031;"T A S K   P R O B A B I L I T I E S   O F   S U C C
E S S"
3620 L8031: IMAGE 13X,K
3625      PRINT
3630      PRINT
3635      IF Code>100 THEN L806
3640      PRINT USING L8032;"CRAFT TYPE",Crfnm$
3645 L8032: IMAGE 17X,K,5X,K
3650      GOTO L807
3655 L806:  PRINT USING L8033;"CRAFT TYPE","COAST GUARD",Crfnm$
3660 L8033: IMAGE 17X,K,5X,K,1X,K
3665 L807:  PRINT USING L8034;"DISPLACEMENT",Idisp,"TONS"
3670 L8034: IMAGE 17X,K,6D,2X,K
3675      PRINT USING L8035;"LENGTH",Ileng,"FEET"
3680 L8035: IMAGE 17X,K,6X,6D,2X,K
3685      PRINT USING L8036;"DESIGN SPEED",Idspd,"KNOTS"
3690 L8036: IMAGE 17X,K,6D,2X,K
3695      PRINT USING L8037;"FUEL FRACTION",Fufrc
3700 L8037: IMAGE 17X,K,3D,2D
3705      PRINT
3710      PRINT USING L8112;"VISIBILITY DISTRIBUTION NO.",Visdtb
3715      PRINT USING L8112;"TOW DISTRIBUTION NO.",Towdtb
3720      PRINT USING L8112;"DEPTH DISTRIBUTION NO.",Dphdtb
3725      PRINT USING L8112;"SEA STATE DISTRIBUTION NO.",Sspdtb
3730 L8112: IMAGE 21X,K,2D
3735      PRINT USING L8113;"(AVERAGE SEA STATE=",Avesc(Sspdtb),")"
3740 L8113: IMAGE 21X,K,D.D,K
3745      PRINT
3750      PRINT
3755      PRINT USING L8001;"TASK","TASK PROB","TASK"
3760 L8001: IMAGE 14X,K,3X,K,4X,K
3765      PRINT USING L8002;"CODE","OF SUCCESS"
3770 L8002: IMAGE 14X,K,3X,K
3775      PRINT
3780      PRINT
3785      PRINT USING L8003;"REDUCED SPEED:"
3790 L8003: IMAGE 10X,K
3795      PRINT USING L8004;"SDU",Tpos(8),"*", "SEARCH FOR DISTRESSED UNIT"
3800      PRINT USING L8005;"SES",Tpos(9),"SLOW ESCORT"
3805      PRINT USING L8004;"SPE",Tpos(11),"*", "SEARCH FOR PEOPLE"
3810      PRINT USING L8005;"SPT",Tpos(10),"SLOW PATROL"
3815      PRINT USING L8005;"TOW",Tpos(12),"TOW"
3820      PRINT
3825      PRINT USING L8003;"CRUISE SPEED:"
3830      PRINT USING L8005;"ESC",Tpos(13),"ESCORT"
3835      PRINT USING L8005;"IDC",Tpos(14),"IDENTIFY CRAFT"
3840      PRINT USING L8005;"IDF",Tpos(14),"IDENTIFY FLEET"
3845      PRINT USING L8005;"PAT",Tpos(15),"PATROL"
3850      PRINT USING L8005;"SFL",Tpos(18),"SEARCH FOR FLEET"
3855      PRINT USING L8004;"SSH",Tpos(16),"*", "SEARCH FOR SHIP"
3860      PRINT USING L8005;"TEQ",Tpos(17),"TRANSPORT EQUIPMENT"
3865      PRINT USING L8005;"TPE",Tpos(18),"TRANSPORT PEOPLE"
3870      PRINT USING L8005;"TRA",Tpos(18),"TRANSIT"
3875      PRINT
3880      PRINT USING L8003;"FLANK SPEED:"
3885      PRINT USING L8005;"DSH",Tpos(19),"DASH"
3890      PRINT USING L8005;"INT",Tpos(19),"INTERDICT"
3895 L8004: IMAGE 14X,K,6X,D.3D,K,4X,K
3900 L8005: IMAGE 14X,K,6X,D.3D,5X,K
3905      PRINT
3910      PRINT

```

```

3915      PRINT USING L3006;"*   THIS IS THE P.O.S. OF THE ABILITY TO SEARCH.
CRAFT'S SUCCESS"
3920 L3006: IMAGE 12X,K
3925      PRINT USING L3007;"IN FINDING THE OBJECT OF THE SEARCH IS DEPENDENT
UPON"
3930      PRINT USING L3007;"SCENARIO (E.G. SEARCH AREA)"
3935 L3007: IMAGE 16X,K
3940      PRINT
3945      PRINT USING L3008;"***** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AN
D WEIGHT OF CARGO)"
3950 L3008: IMAGE 10X,K
3955      PRINT
3960      DISP "                END OF SPTPOS OUTPUT. PRESS CONT TO GO ON "
3965      PAUSE
3970      DISP
3975      PRINTER IS 16
3980      RETURN
3985 !
3990 Vmtav:      ! CONVERTED SUBROUTINE VMTAV
3995      Vaug=0
4000      Avfurl=0
4005      FOR I=1 TO 3
4010      Sso=I-1
4015      FOR J=1 TO 9
4020      Ss=Sso+J/10
4025      FOR K=1 TO 3
4030      Flag=0
4035      Pvinss=FNFpdsss(Code,Displ,Rate,Dspeed,Ss,Flag)
4040      Vinss=Dspeed*Pvinss/100
4045      Vvuis=MIN(Vinss,Vmxuis(K))
4050      Pofss=Ssprbd(I)/9
4055      Pofuis=Visdis(K,Visdtb)
4060      Vaug=Vaug+Vvuis*Pofss*Pofuis
4065      Fueuse=Fuelrt(Rate)
4070      IF Vmxuis(K)<Vinss THEN Fueuse=Visfue(K)
4075      Avfurl=Avfurl+Fueuse*Pofss*Pofuis
4080      NEXT K
4085      NEXT J
4090      NEXT I
4095      RETURN
4100 !
4105 !      PTWD
4110 !
4115 !      TOW DISPLACEMENT CUMULATIVE PROBABILITY DISRTIBUTION.
4120 !
4125 !      PTOWD(D) = PROBABILITY THAT CRAFT TO BE TOWED HAS DISPLACEMENT < D
4130 !      AVTWDS = AVERAGE DISPLACEMENT VALUE THAT CAN BE TOWED
4135 !
4140 Ptwd: ! CONVERTED SUBROUTINE PTWD
4145      In=Towdsp
4150      A=Towdis(6,Towdtb)
4155      Out=FNFf8(In,0,0,Towdis(1,Towdtb),0,Towdis(2,Towdtb),.2,Towdis(3,To
wdtb),.4,Towdis(4,Towdtb),.6,Towdis(5,Towdtb),.8,A,1,999999,1)
4160      Ptwd=Out
4165      In=Ptwd/2
4170      Out=FNFf8(In,0,0,0,Towdis(1,Towdtb),.2,Towdis(2,Towdtb),.4,Towdis(3
,Towdtb),.6,Towdis(4,Towdtb),.8,Towdis(5,Towdtb),1,A,1,999999)
4175      Avtuds=Out
4180      RETURN
4185 !
4190 ! *****
4195 !      SUBPROGRAMS
4200 ! *****
4205 !
4210 ! FUNCTION FBSSPD
4215 !

```

```

4220 ! BASE SPEED
4225 ! (USED FOR BASE CURVES FOR HPINST,ACCEL,BRKG,& TURN)
4230 !
4235 DEF FNBsspd<Code>
4240 IF <Code=10> OR <Code=20> OR <Code=21> THEN Fbsspd=50
4245 IF <Code=11> OR <Code=70> THEN Fbsspd=40
4250 IF Code=30 THEN Fbsspd=60
4255 IF Code=40 THEN Fbsspd=45
4260 IF Code=50 THEN Fbsspd=30
4265 IF Code=60 THEN Fbsspd=20
4270 IF Code=70 THEN Fbsspd=40
4275 IF Code=80 THEN Fbsspd=25
4280 RETURN Fbsspd
4285 FNEND
4290 !
4295 ! FHPFCT
4300 !
4305 ! FRACTION OF INSTALLED HORSEPOWER UTILIZED
4310 !
4315 DEF FNFhpfct<Code,Rate,Fctdsp,Fctbsp>
4320 IF <Rate=1> OR <Rate=2> THEN Pctdsp=100*Fctdsp
4325 IF <Rate=3> OR <Rate=4> OR <Rate=0> THEN Pctdsp=100*Fctbsp
4330 IF <Code<>10> AND <Code<>11> THEN L20
4335 Hppct=FNFF3<Pctdsp,0,5,20,14,40,48,85,75,100,100>
4340 GOTO L99
4345 L20: IF <Code<>20> AND <Code<>21> THEN L30
4350 IF Pctdsp<=10 THEN Hppct=10
4355 IF <Pctdsp<=30> AND <Pctdsp>10> THEN Hppct=2*Pctdsp-10
4360 IF <Pctdsp<=80> AND <Pctdsp>30> THEN Hppct=.20*Pctdsp+44
4365 IF Pctdsp>80 THEN Hppct=2*Pctdsp-100
4370 GOTO L99
4375 L30: IF Code<>30 THEN L40
4380 Hppct=FNFF3<Pctdsp,0,10,80,60,100,100>
4385 GOTO L99
4390 L40: IF Code<>40 THEN L50
4395 IF Pctdsp<=20 THEN Hppct=5
4400 IF <Pctdsp<=40> AND <Pctdsp>20> THEN Hppct=2.75*Pctdsp-50
4405 IF <Pctdsp<=80> AND <Pctdsp>40> THEN Hppct=.25*Pctdsp+50
4410 IF Pctdsp>80 THEN Hppct=1.5*Pctdsp-50
4415 GOTO L99
4420 L50: IF Code<>50 THEN L60
4425 IF Pctdsp<=10 THEN Hppct=.50*Pctdsp+5
4430 IF <Pctdsp<=30> AND <Pctdsp>10> THEN Hppct=2.5*Pctdsp-15
4435 IF <Pctdsp<=80> AND <Pctdsp>30> THEN Hppct=.20*Pctdsp+54
4440 IF Pctdsp>80 THEN Hppct=1.5*Pctdsp-50
4445 GOTO L99
4450 L60: IF Code<>60 THEN L70
4455 Hppct=FNFF4<Pctdsp,0,5,30,12,70,45,100,100>
4460 GOTO L99
4465 L70: IF Code<>70 THEN L80
4470 IF Pctdsp<=10 THEN Hppct=5
4475 IF Pctdsp>10 THEN Hppct=1.05556*Pctdsp-5.55556
4480 GOTO L99
4485 L80: IF Pctdsp<=20 THEN Hppct=10
4490 IF <Pctdsp<=50> AND <Pctdsp>20> THEN Hppct=.25*Pctdsp+5
4495 IF Pctdsp>50 THEN Hppct=2*Pctdsp-100
4500 L99: Fhpfct=Hppct/100
4505 RETURN Fhpfct
4510 FNEND
4515 !
4520 ! FENG
4525 !
4530 ! ENGINE TYPE
4535 !
4540 DEF FNFeng<Code,Rate>
4545 OPTION BASE 1

```

```

4550      DIM All1ds1(4),All1gt(4),Gt3ds1(4),Gt1ds1(4)
4555      INTEGER All1ds1,All1gt,Gt3ds1,Gt1ds1,Feng
4560      DATA 2,2,2,2
4565      MAT READ All1ds1
4570      DATA 1,1,1,1
4575      MAT READ All1gt
4580      DATA 1,1,1,2
4585      MAT READ Gt3ds1
4590      DATA 1,2,2,2
4595      MAT READ Gt1ds1
4600      IF (Code=11) OR (Code=50) OR (Code=70) THEN Feng=All1ds1(Rate)
4605      IF (Code=20) OR (Code=21) OR (Code=60) OR (Code=30) THEN Feng=All1gt(
Rate)
4610      IF (Code=10) OR (Code=40) THEN Feng=Gt3ds1(Rate)
4615      IF Code=80 THEN Feng=Gt1ds1(Rate)
4620      RETURN Feng
4625      FNEED
4630      !
4635      ! FMVSWH
4640      !
4645      ! MOTION OF BASE CRAFT VS. WAVE HEIGHT (FOR DISPLACEMENT=100 TONS
4650      ! EXCEPT TYPE 60 DISPLACEMENT=1500 TONS) AND REVERSE
4655      !
4660      DEF FNFmvswh(Code,Rate,Wghtbs,Flag)
4665      !
4670      In=Wghtbs
4675      !
4680      IF Code<>10 THEN L11
4685      IF (Rate=1) OR (Rate=2) THEN Out=FNFff3(In,Flag,0,0,15,.5,18,
1)
4690      IF Rate=3 THEN Out=FNFff3(In,Flag,0,0,8,1)
4695      IF Rate=4 THEN Out=FNFff3(In,Flag,0,0,12,1)
4700      GOTO L999
4705      L11: IF Code<>11 THEN L20
4710      IF (Rate=1) OR (Rate=2) THEN Out=FNFff3(In,Flag,0,0,5,.25,8,1
)
4715      IF Rate=3 THEN Out=FNFff3(In,Flag,0,0,10,1)
4720      IF Rate=4 THEN Out=FNFff3(In,Flag,0,0,14,1)
4725      GOTO L999
4730      L20: IF (Code<>20) OR (Code<>21) THEN L40
4735      IF Rate=1 THEN Out=FNFff3(In,Flag,0,0,4,.2,5,1)
4740      IF Rate=2 THEN Out=FNFff3(In,Flag,0,0,4,.2,6,1)
4745      IF Rate=3 THEN Out=FNFff3(In,Flag,0,0,4,.2,7,1)
4750      IF Rate=4 THEN Out=FNFff3(In,Flag,0,0,4,.2,10,1)
4755      GOTO L999
4760      L40: IF Code<>30 THEN L50
4765      IF Rate=1 THEN Out=FNFff3(In,Flag,0,0,5,.2,7,1)
4770      IF Rate=2 THEN Out=FNFff3(In,Flag,0,0,5,.2,8,1)
4775      IF Rate=3 THEN Out=FNFff3(In,Flag,0,0,5,.2,10,1)
4780      IF Rate=4 THEN Out=FNFff3(In,Flag,0,0,5,.2,12,1)
4785      GOTO L999
4790      L50: IF Code<>40 THEN L60
4795      IF Rate=1 THEN Out=FNFff3(In,Flag,0,0,3,1)
4800      IF Rate=2 THEN Out=FNFff3(In,Flag,0,0,4,1)
4805      IF Rate=3 THEN Out=FNFff3(In,Flag,0,0,8,1)
4810      IF Rate=4 THEN Out=FNFff3(In,Flag,0,0,12,1)
4815      GOTO L999
4820      L60: IF (Code<>50) OR (Code<>70) OR (Code<>80) OR (Code<>100) THEN
L80
4825      IF Rate=1 THEN Out=FNFff3(In,Flag,0,0,5,1)
4830      IF Rate=2 THEN Out=FNFff3(In,Flag,0,0,6,1)
4835      IF Rate=3 THEN Out=FNFff3(In,Flag,0,0,8,1)
4840      IF Rate=4 THEN Out=FNFff3(In,Flag,0,0,12,1)
4845      GOTO L999
4850      ! FOR CODE = 60
4855      L80: Out=FNFff3(In,Flag,0,0,17,143,.343,21.01,1)

```

```

4860 L999:      Fmuswh=Out
4865 !          IF Fmuswh>1 THEN Fmuswh=1
4870          IF Flag=2 THEN Fmuswh=Out
4875          RETURN Fmuswh
4880          FEND
4885 !
4890 !      FCPBSS
4895 !
4900 !      CUMULATIVE PROBABILITY OF SEA STATE
4905 !      FCPBSS(SS)= PROB THAT SEA STATE < OR = SS
4910 !
4915          DEF FNFcpbss(Ssprob(*),Sspdtb,Ss)
4920          OPTION BASE 1
4925          INTEGER Iss
4930          IF Ss<8 THEN L50
4935          Fcpbss=1
4940          RETURN Fcpbss
4945 L50:      IF Ss<=0 THEN Ss=4
4950          Iss=Ss-.5
4955          Intrp=Ss-Iss
4960          Prbsum=0
4965          FOR J=1 TO Iss
4970          Prbsum=Prbsum+Ssprob(J,Sspdtb)
4975          NEXT J
4980          Intss=Iss+1
4985          Fcpbss=Prbsum+Intrp*Ssprob(Intss,Sspdtb)
4990          RETURN Fcpbss
4995          FEND
5000 !
5005 !      FF4
5010 !
5015 !      FINDS Y VALUE ON A BROKEN LINE OF FIVE POINTS, GIVEN X VALUE
5020 !      AND THE FIVE POINTS. (ASSUMING LINE EXTENDS INFINITELY)
5025 !
5030          DEF FNFf4(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4)
5035          IF X<=X2 THEN Ff4=FNFF(X,X1,Y1,X2,Y2)
5040          IF (X>X2) AND (X<=X3) THEN Ff4=FNFF(X,X2,Y2,X3,Y3)
5045          IF X>X3 THEN Ff4=FNFF(X,X3,Y3,X4,Y4)
5050          RETURN Ff4
5055          FEND
5060 !
5065 !      FF
5070 !
5075 !      FINDS Y VALUE ON A STRAIGHT LINE GIVEN X VALUE AND TWO POINTS
5080 !      ON THE LINE. (ASSUMING LINE EXTENDS INFINITELY).
5085 !
5090          DEF FNFF(X,X1,Y1,X2,Y2)
5095          IF ABS(X2-X1)<.0001 THEN L1
5100          Slope=(Y2-Y1)/(X2-X1)
5105          IF ABS(Y2-Y1)<.0001 THEN Slope=0
5110          Ff=Slope*X+Y1-Slope*X1
5115          RETURN Ff
5120 L1:      Ff=(Y1+Y2)/2
5125          RETURN Ff
5130          FEND
5135 !
5140 !      FPDSSS
5145 !
5150 !      PERCENT DESIGN SPEED VS SEA STATE AND
5155 !      SEA STATE VS PERCENT DESIGN SPEED
5160 !
5165 !      FOR CRUISE : MAXIMUM IS LIMITED BY LINE PARALLEL TO INITIAL
5170 !      FLANK LINE AND STARTING FROM PERCENT DESIGN SPEED AXIS
5175 !      AT CWSPD(2)
5180 !
5185 !      FOR REDUCED SPEED: MAXIMUM IS LIMITED BY LINE PARALLEL TO

```

```

5190 ! SEA STATE AXIS AT
5195 ! PERCENT DESIGN SPEED = CWSPD(3)/DESIGN SPEED*100
5200 !
5205 DEF FNFpdsss(Code,Displ,Rate,Dspeed,ss,Flag)
5210 REAL In,Disp,Pcdsmx,Out
5215 In=ss
5220 IF Code=10 THEN L10
5225 IF Code=11 THEN L11
5230 IF (Code=20) OR (Code=21) THEN L20
5235 IF (Code=30) OR (Code=70) THEN L30
5240 IF (Code=80) OR (Code=102) OR (Code=103) THEN L30
5245 IF (Code=106) OR (Code=107) OR (Code=108) THEN L30
5250 IF (Code=109) OR (Code=110) OR (Code=111) OR (Code=112) THEN L30
5255 IF Code=50 THEN L50
5260 IF Code=60 THEN L60
5265 IF Code=40 THEN L40
5270 IF (Code=101) OR (Code=104) OR (Code=105) THEN L101
5275 L10: IF Rate=2 THEN L1002
5280 IF Displ<=100 THEN Out=FNFff4(In,Flag,0,100,5,91.7,5,20,7,0)
5285 IF (Displ>100) AND (Displ<=200) THEN Out=FNFff4(In,Flag,0,100,5.5,90
,0,5.5,20,7.5,0)
5290 IF Displ>200 THEN Out=FNFff4(In,Flag,0,100,6,90,6,20,8,0)
5295 GOTO L991
5300 ! FOR RATE = 2 AND CODE = 10
5305 L1002: IF Displ<=100 THEN Out=FNFff4(In,Flag,0,85,5,76.7,5,20,7,0)
5310 IF (Displ>100) AND (Displ<=200) THEN Out=FNFff4(In,Flag,0,85,5.5,75.
0,5.5,20,7.5,0)
5315 IF Displ>200 THEN Out=FNFff4(In,Flag,0,85,6,75,6,20,18,0)
5320 GOTO L999
5325 !
5330 L11: IF Rate=2 THEN L1102
5335 IF Displ<=100 THEN Out=FNFff4(In,Flag,0,100,4.5,62.5,5.18,20,7,0)
5340 IF (Displ>100) AND (Displ<=200) THEN Out=FNFff4(In,Flag,0,100,5,58.3
,5.66,20,7.5,0)
5345 IF Displ>200 THEN Out=FNFff4(In,Flag,0,100,5.5,54.2,6.3,20,8,0)
5350 GOTO L991
5355 !
5360 L1102: IF Displ<=100 THEN Out=FNFff4(In,Flag,0,90,4.72,50.5,5.18,20,7,0)
5365 IF (Displ>100) AND (Displ<=200) THEN Out=FNFff4(In,Flag,0,90,5.25,46
,5.66,20,7.5,0)
5370 IF Displ>200 THEN Out=FNFff4(In,Flag,0,90,5.8,41.5,6.13,20,8,0)
5375 GOTO L999
5380 !
5385 L20: IF Rate=2 THEN L1020
5390 IF Displ<=20 THEN Out=FNFff4(In,Flag,0,100,3,0)
5395 IF (Displ>20) AND (Displ<=50) THEN Out=FNFff4(In,Flag,0,100,1.5,100,
2.5,80,4,0)
5400 IF (Displ>50) AND (Displ<=100) THEN Out=FNFff4(In,Flag,0,100,2,100,3
,80,4.5,0)
5405 IF (Displ>100) AND (Displ<=150) THEN Out=FNFff4(In,Flag,0,100,2.5,10
0,3.5,80,5,0)
5410 IF (Displ>150) AND (Displ<=200) THEN Out=FNFff4(In,Flag,0,100,3,100,
4,30,5.5,0)
5415 IF Displ>200 THEN Out=FNFff4(In,Flag,0,100,3.5,100,4.5,80,6,0)
5420 GOTO L991
5425 L1020: IF Displ<=20 THEN Out=FNFff4(In,Flag,0,85,.45,85,3,0)
5430 IF (Displ>20) AND (Displ<=50) THEN Out=FNFff4(In,Flag,0,85,2.25,85,2
.5,80,4,0)
5435 IF (Displ>50) AND (Displ<=100) THEN Out=FNFff4(In,Flag,0,85,2.75,85,
3,80,4.5,0)
5440 IF (Displ>100) AND (Displ<=150) THEN Out=FNFff4(In,Flag,0,85,3.25,85
,3.5,80,5,0)
5445 IF (Displ>150) AND (Displ<=200) THEN Out=FNFff4(In,Flag,0,85,3.75,85
,4,80,5.5,0)
5450 IF Displ>200 THEN Out=FNFff4(In,Flag,0,85,4.25,85,4.5,80,6,0)
5455 GOTO L999

```

```

5460 !
5465 L30: IF Rate=2 THEN L1030
5470 IF Displ<=150 THEN Out=FNFff3(In,Flag,0,100,.5,100,5.5,0)
5475 IF Displ>150 THEN Out=FNFff3(In,Flag,0,100,1,100,6,0)
5480 GOTO L991
5485 L1030: IF Displ<=150 THEN Out=FNFff3(In,Flag,0,87.5,1.125,87.5,5.5,0)
5490 IF Displ>150 THEN Out=FNFff3(In,Flag,0,87.5,1.625,87.5,6,0)
5495 GOTO L999
5500 !
5505 L40: IF Rate=2 THEN L1040
5510 IF Displ<=5 THEN Out=FNFff3(In,Flag,0,100,1.72,92,5,0)
5515 IF Displ<=20 THEN Out=FNFff3(In,Flag,0,100,2.28,89,5.5,0)
5520 IF Displ<=50 THEN Out=FNFff3(In,Flag,0,100,2.75,85.5,6,0)
5525 IF Displ<=100 THEN Out=FNFff3(In,Flag,0,100,3.5,82.5,6.5,0)
5530 IF Displ<=200 THEN Out=FNFff3(In,Flag,0,100,4.1,79.8,7,0)
5535 IF Displ>200 THEN Out=FNFff3(In,Flag,0,100,4.65,77.5,7.5,0)
5540 GOTO L991
5545 !
5550 L1040: IF Displ<=5 THEN Out=FNFff3(In,Flag,0,87.5,2.2,76.5,5,0)
5555 IF Displ<=20 THEN Out=FNFff3(In,Flag,0,87.5,2.8,73.5,5.5,0)
5560 IF Displ<=50 THEN Out=FNFff3(In,Flag,0,87.5,3.4,71.0,6,0)
5565 IF Displ<=100 THEN Out=FNFff3(In,Flag,0,87.5,4.05,67,6.5,0)
5570 IF Displ<=200 THEN Out=FNFff3(In,Flag,0,87.5,4.65,64,7,0)
5575 IF Displ>200 THEN Out=FNFff3(In,Flag,0,87.5,5.15,61.5,7.5,0)
5580 GOTO L999
5585 !
5590 L50: IF Rate=2 THEN L1050
5595 IF Displ<=100 THEN Out=FNFff3(In,Flag,0,100,4,74,6,0)
5600 IF Displ>100 THEN Out=FNFff3(In,Flag,0,100,5.3,65.5,7,0)
5605 GOTO L991
5610 L1050: IF Displ<=100 THEN Out=FNFff3(In,Flag,0,87.5,4.4,59,6,0)
5615 IF Displ>100 THEN Out=FNFff3(In,Flag,0,87.5,5.7,50.5,7,0)
5620 GOTO L999
5625 !
5630 L60: IF Rate=2 THEN L1060
5635 IF Displ<=500 THEN Out=FNFff4(In,Flag,0,100,3,100,5,90,8,0)
5640 IF Displ>500 THEN Out=FNFff4(In,Flag,0,100,4,100,6,90,9,0)
5645 GOTO L991
5650 L1060: IF Displ<=500 THEN Out=FNFff3(In,Flag,0,60,6,60,8,0)
5655 IF Displ>500 THEN Out=FNFff3(In,Flag,0,60,7,60,9,0)
5660 GOTO L999
5665 !
5670 L80: IF Rate=2 THEN L1080
5675 IF Displ<=5 THEN Out=FNFff3(In,Flag,0,100,2,83,5,0)
5680 IF (Displ>5) AND (Displ<=20) THEN Out=FNFff3(In,Flag,0,100,2.67,77,5
.5,0)
5685 IF (Displ>20) AND (Displ<=50) THEN Out=FNFff3(In,Flag,0,100,3.34,72,
6,0)
5690 IF (Displ>50) AND (Displ<=100) THEN Out=FNFff3(In,Flag,0,100,4,65,6.
5,0)
5695 IF (Displ>100) AND (Displ<=200) THEN Out=FNFff3(In,Flag,0,100,4.67,6
0,7,0)
5700 IF Displ>200 THEN Out=FNFff3(In,Flag,0,100,5.34,54,7.5,0)
5705 GOTO L991
5710 L1080: IF Displ<=5 THEN Out=FNFff3(In,Flag,0,60,4.1,25.6,5,0)
5715 IF (Displ>5) AND (Displ<=20) THEN Out=FNFff3(In,Flag,0,60,4.85,19.5,
5.5,0)
5720 IF (Displ>20) AND (Displ<=50) THEN Out=FNFff3(In,Flag,0,60,5.58,13.0
,6,0)
5725 IF (Displ>50) AND (Displ<=100) THEN Out=FNFff3(In,Flag,0,60,6.3,6.4,
6.5,0)
5730 IF (Displ>100) AND (Displ<=200) THEN Out=FNFff3(In,Flag,0,60,7,0)
5735 IF Displ>200 THEN Out=FNFff3(In,Flag,0,60,7,0,7.5,0)
5740 GOTO L999
5745 L101: IF Rate=2 THEN L1101
5750 IF Displ<=10 THEN Out=FNFff3(In,Flag,0,100,1,96.7,5,0)

```

```

5755      IF (Displ>10) AND (Displ<=25) THEN Out=FNFFF3(In,Flag,0,100,2,93.3,6
,0)
5760      IF Displ>25 THEN Out=FNFFF3(In,Flag,0,100,3,90,7,0)
5765      GOTO L991
5770 L1101: IF Displ<=10 THEN Out=FNFFF3(In,Flag,0,70,2.45,62,5,0)
5775      IF (Displ>10) AND (Displ<=25) THEN Out=FNFFF3(In,Flag,0,85,2.75,76,6
,0)
5780      IF Displ>25 THEN Out=FNFFF3(In,Flag,0,100,3,90,6.5,0)
5785      GOTO L999
5790      !
5795 L991:  IF Rate=1 THEN L999
5800      IF Rate=3 THEN L993
5805      IF Rate=4 THEN L994
5810 L993:  Pcdsmx=12/Dspeed*100
5815 L995:  IF (Flag=0) AND (Out>Pcdsmx) THEN Out=Pcdsmx
5820      IF (Flag=1) AND (In>Pcdsmx) THEN Out=0
5825      GOTO L999
5830 L994:  Pcdsmx=5/Dspeed*100
5835      GOTO L995
5840      !
5845      !      IF CALCULATED OUTPUT OF SEA STATE VS PERCENT DESIGN SPEED
5850      !      IS LESS THAN ZERO ,SET VALUE EQUAL TO ZERO
5855      !
5860 L999:  IF Out<0 THEN Out=0
5865      Fpdsss=Out
5870      RETURN Fpdsss
5875      FEND
5880      !
5885      !      FF5
5890      !
5895      !      FINDS Y VALUE ON BROKEN LINE OF 5 POINTS, GIVEN X VALUE
5900      !      AND THE 5 POINTS
5905      !      (ASSUMING ENDS OF LINE EXTEND INFINITELY)
5910      !
5915      DEF FNFF5(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5)
5920      IF X=X2 THEN FF5=FNFF(X,X1,Y1,X2,Y2)
5925      IF (X>X2) AND (X<=X3) THEN FF5=FNFF(X,X2,Y2,X3,Y3)
5930      IF (X>X3) AND (X<=X4) THEN FF5=FNFF(X,X3,Y3,X4,Y4)
5935      IF X>X4 THEN FF5=FNFF(X,X4,Y4,X5,Y5)
5940      RETURN FF5
5945      FEND
5950      !
5955      !      FHPBIN
5960      !
5965      !      INSTALLED BASE HORSEPOWER
5970      !      (FOR A CRAFT WITH DESIGN SPEED=BASE SPEED)
5975      !
5980      DEF FNHpbm(Code,Displ)
5985      IF Code=10 THEN Fhpbin=FNFF(Displ,55,2756,300,20000)
5990      IF Code=11 THEN Fhpbin=FNFF(Displ,20,1000,250,14000)
5995      IF Code=20 THEN Fhpbin=FNFF(Displ,3,1570,30,6606)
6000      IF Code=21 THEN Fhpbin=FNFF(Displ,25,2500,150,17000)
6005      IF Code=30 THEN Fhpbin=FNFF(Displ,0,4000,100,10000)
6010      IF Code=40 THEN Fhpbin=FNFF(Displ,35,3000,155,14500)
6015      IF Code=50 THEN Fhpbin=FNFF(Displ,64,3400,74,3600)
6020      IF Code=70 THEN Fhpbin=FNFF(Displ,25,752,200,6845)
6025      IF Code=60 THEN Fhpbin=FNFF3(Displ,200,2000,1000,6800,4000,12800)
6030      IF Code=80 THEN Fhpbin=10^(.659*LGT(Displ)+2.2648)
6035      RETURN Fhpbin
6040      !
6045      !      FSFCEN
6050      !
6055      !      SPECIFIC FUEL CONSUMPTION (LBS PER HORSEPOWER HOUR PER ENGINE)
6060      !
6065      DEF FNFsfcen(Eng,Hpinst)
6070      hpins2=Hpinst/2

```

```

6075      IF Eng=2 THEN Fsfcen=.35
6080      IF Eng=1 THEN Fsfcen=FNFF3(Hpins2,400,.7,4000,.48,15000,.4)
6085      RETURN Fsfcen
6090      FNFEND
6095      !
6100      ! FFF3
6105      !
6110      ! READ CURVE Y VS X OR X VS Y DEPENDING UPON FLAG
6115      ! CURVE IS A BROKEN LINE OF 3 POINTS
6120      !
6125      ! FLAG = 0 MEANS Y VS X
6130      ! FLAG = 1 MEANS X VS Y (NEGATIVE SLOPE)
6135      ! FLAG = 2 MEANS X VS Y (POSITIVE SLOPE)
6140      !
6145      DEF FNFFf3(Xory,Flag,X1,Y1,X2,Y2,X3,Y3)
6150      IF Flag=0 THEN Fff3=FNFFf3(Xory,X1,Y1,X2,Y2,X3,Y3)
6155      IF Flag=1 THEN Fff3=FNFFf3(Xory,Y3,X3,Y2,X2,Y1,X1)
6160      IF Flag=2 THEN Fff3=FNFFf3(Xory,Y1,X1,Y2,X2,Y3,X3)
6165      RETURN Fff3
6170      FNFEND
6175      !
6180      ! FF3
6185      !
6190      ! FINDS Y VALUE ON BROKEN LINE OF 3 POINTS, GIVEN X VALUE
6195      ! AND THE 3 POINTS
6200      ! (ASSUMING ENDS OF LINE EXTEND INFINITELY)
6205      !
6210      DEF FNFFf3(X,X1,Y1,X2,Y2,X3,Y3)
6215      IF X<=X2 THEN Ff3=FNFFf3(X,X1,Y1,X2,Y2)
6220      IF X>X2 THEN Ff3=FNFFf3(X,X2,Y2,X3,Y3)
6225      RETURN Ff3
6230      FNFEND
6235      !
6240      ! FSFCCF
6245      !
6250      ! SPECIFIC FUEL CONSUMPTION CORRECTION FACTOR
6255      !
6260      DEF FNFsfccf(Eng,Hpfctu)
6265      IF (Eng=1) AND (Hpfctu>.5) THEN Fsfccf=-.4*Hpfctu+1.4
6270      IF (Eng=1) AND (Hpfctu>.25) AND (Hpfctu<=.5) THEN Fsfccf=-1.6*Hpf
ctu+2
6275      IF (Eng=1) AND (Hpfctu<=.25) THEN Fsfccf=-3.2*Hpfctu+2.4
6280      IF Eng=2 THEN Fsfccf=1
6285      RETURN Fsfccf
6290      FNFEND
6295      !
6300      ! FFF
6305      !
6310      ! READ CURVE Y VS X, OR X VS Y DEPENDING UPON FLAG
6315      ! CURVE IS A STRAIGHT LINE
6320      !
6325      ! FLAG = 0 MEANS Y VS X
6330      ! FLAG = 1 MEANS X VS Y (NEGATIVE SLOPE)
6335      ! FLAG = 2 MEANS X VS Y (POSITIVE SLOPE)
6340      !
6345      DEF FNFFf(Xory,Flag,X1,Y1,X2,Y2)
6350      IF Flag=0 THEN Fff=FNFFf(Xory,X1,Y1,X2,Y2)
6355      IF Flag=1 THEN Fff=FNFFf(Xory,Y2,X2,Y1,X1)
6360      IF Flag=2 THEN Fff=FNFFf(Xory,Y1,X1,Y2,X2)
6365      RETURN Fff
6370      FNFEND
6375      !
6380      ! FF8
6385      !
6390      ! FINDS Y VALUE ON BROKEN LINE OF 3 POINTS, GIVEN X VALUES AND
6395      ! THE EIGHT POINTS.

```

```

6400      !
6405      !
6410      ! DEF FNFF8(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5,X6,Y6,X7,Y7,X8,Y8)
6415      ! IF X<X2 THEN FF8=FNFF(X,X1,Y1,X2,Y2)
6420      ! IF (X>X2) AND (X<=X3) THEN FF8=FNFF(X,X2,Y2,X3,Y3)
6425      ! IF (X>X3) AND (X<=X4) THEN FF8=FNFF(X,X3,Y3,X4,Y4)
6430      ! IF (X>X4) AND (X<=X5) THEN FF8=FNFF(X,X4,Y4,X5,Y5)
6435      ! IF (X>X5) AND (X<=X6) THEN FF8=FNFF(X,X5,Y5,X6,Y6)
6440      ! IF (X>X6) AND (X<=X7) THEN FF8=FNFF(X,X6,Y6,X7,Y7)
6445      ! IF X>X7 THEN FF8=FNFF(X,X7,Y7,X8,Y8)
6450      ! RETURN FF8
6455      ! FNEND
6460      !
6465      ! FFF4
6470      !
6475      ! READ CURVE Y VS X OR X VS Y DEPENDING UPON FLAG
6480      ! CURVE IS A BROKEN LINE OF 4 POINTS
6485      !
6490      ! FLAG = 0 MEANS Y VS X
6495      ! FLAG = 1 MEANS X VS Y (NEGATIVE SLOPE)
6500      ! FLAG = 2 MEANS X VS Y (POSITIVE SLOPE)
6505      !
6510      ! DEF FNFFF4(Xory,Flag,X1,Y1,X2,Y2,X3,Y3,X4,Y4)
6515      ! IF Flag=0 THEN Fff4=FNFF4(Xory,X1,Y1,X2,Y2,X3,Y3,X4,Y4)
6520      ! IF Flag=1 THEN Fff4=FNFF4(Xory,Y4,X4,Y3,X3,Y2,X2,Y1,X1)
6525      ! IF Flag=2 THEN Fff4=FNFF4(Xory,Y1,X1,Y2,X2,Y3,X3,Y4,X4)
6530      ! RETURN Fff4
6535      ! FNEND
6540      !
6545      ! FPDPTH
6550      !
6555      ! DEPTH CUMULATIVE PROBABILITY DISTRIBUTION
6560      !
6565      ! FPDPTH(D)=PROBABILITY THAT DEPTH < D
6570      !
6575      ! DEF FNFpdpth(Dphdtb,Depth)
6580      ! IF Dphdtb=1 THEN Fpdpth=0
6585      ! RETURN Fpdth
        ! FNEND

```

APPENDIX D

```

475 Sprpos:  GLOBAL SUBROUTINE SPRPOS
480          Nmstsk=19
485          DATA .9,.7,.5,.1,.2,.3,0.,.1,.2
490          MAT READ Vids1
495          Vids2$(1)="VERY GOOD"
500          Vids2$(2)="GOOD"
505          Vids2$(3)="GOOD TO FAIR"
510          MAT Qucnmx=ZER
515          Totprb=0
520          DATA 7,3,4,3,5,5,3,3,4,4,2,3,2
525          DATA 2,4,3,4,5,0,0,0,0
530          MAT READ Nnode
535          DATA 19,13,8,2,19,14,9,7,0,14,11,5
540          DATA 0,15,10,1,0,18,12,5,0,16,0,7
545          DATA 0,17,0,5,0,18,0,7,0,18,0,3
550          DATA 0,0,0,3,0,0,0,5,0,0,5,0,0,2
555          DATA 0,0,0,4,0,0,0,4,0,0,7,0,0,3
560          DATA 0,0,0,5,0,0,0,7,0,0,7,0,0,5
565          DATA 0,0,0,6,0,0,0,7,0,0,0,0,0,0
570          MAT READ Master
575          ! ARRAY VARIABLE MASTER AND NNODE ARE POSSIBLE INCORRECT
580          MAT Beta=ZER
585          Tasknos(1,1)="DASH"
590          Tasknos(2,1)="INTERDICT"
595          FOR I=3 TO 25 STEP 1
600            Tasknos(I,1)=" "
605          NEXT I
610          Tasknos(1,2)="ESCORT"
615          Tasknos(2,2)="IDENTIFY CRAFT"
620          Tasknos(3,2)="IDENTIFY FLEET"
625          Tasknos(4,2)="PATROL"
630          Tasknos(5,2)="SEARCH FOR FLEET"
635          Tasknos(6,2)="SEARCH FOR SHIP: FOUND"
640          Tasknos(7,2)="TRANSPORT EQUIPMENT"
645          Tasknos(8,2)="TRANSPORT PEOPLE"
650          Tasknos(9,2)="TRANSIT"
655          FOR I=10 TO 25 STEP 1
660            Tasknos(I,2)=" "
665          NEXT I
670          Tasknos(1,3)="SEARCH DSTR UNIT: FOUND"
675          Tasknos(2,3)="SLOW ESCORT"
680          Tasknos(3,3)="SEARCH FOR PEOP: FOUND"
685          Tasknos(4,3)="SLOW PATROL"
690          Tasknos(5,3)="TOW"
695          FOR I=6 TO 25 STEP 1
700            Tasknos(I,3)=" "
705          NEXT I
710          Tasknos(1,4)="BOARD"
715          Tasknos(2,4)="FIGHT FIRE FROM CG VESSEL"
720          Tasknos(3,4)="FGHT FIRE ON OTHER VESSEL"
725          Tasknos(4,4)="GENERAL ASSISTANCE"
730          Tasknos(5,4)="INSPECTION"
735          Tasknos(6,4)="LOAD EQUIPMENT"
740          Tasknos(7,4)="LOITER"
745          Tasknos(8,4)="LAUNCH SMALL BOAT"
750          Tasknos(9,4)="MONITER ACTIVITIES"
755          Tasknos(10,4)="MONITER OIL SPILLS"
760          Tasknos(11,4)="ON BOARD ASSISTANCE"
765          Tasknos(12,4)="ON SCENE COMMANDER"
770          Tasknos(13,4)="RETRIEVE BOARDING PARTY"
775          Tasknos(14,4)="RETRIEVE OBJECTS"
780          Tasknos(15,4)="RESCUE PEOPLE"
785          Tasknos(16,4)="RETRIEVE SMALL BOAT"
790          Tasknos(17,4)="STAKE OUT SPEC INT VESSEL"
795          Tasknos(18,4)="SEIZE"
800          Tasknos(19,4)="TAKE WATER SAMPLE"

```

```

805 Taskno$(20,4)="UNLOAD EQUIPMENT"
810 Taskno$(21,4)="WORK EQUIP FROM SM BOAT"
815 Taskno$(22,4)="WORK EQUIPMENT @ DRIFT"
820 Taskno$(23,4)="WORK EQUIP @ FIXED POS"
825 Taskno$(24,4)=" "
830 Taskno$(25,4)=" "
835 Grpnm$(1)="ASSIST"
840 Grpnm$(2)="ESCORT"
845 Grpnm$(3)="FIGHT FIRE"
850 Grpnm$(4)="IDENTIFY"
855 Grpnm$(5)="INSPECT"
860 Grpnm$(6)="MONITER"
865 Grpnm$(7)="PATROL"
870 Grpnm$(8)="RESCUE"
875 Grpnm$(9)="RESCUE RETURN"
880 Grpnm$(10)="SAR SEARCH"
885 Grpnm$(11)="SEARCH FLEET"
890 Grpnm$(12)="SEIZE"
895 Grpnm$(13)="SENSOR SEARCH"
900 Grpnm$(14)="STANDBY"
905 Grpnm$(15)="STEAM"
910 Grpnm$(16)="TRANSFER EQUIP"
915 Grpnm$(17)="TRANSPORT EQUIP"
920 Grpnm$(18)="WORK EQUIPMENT"
925 Grpnm$(19)=" "
930 Grpnm$(20)=" "
935 ! GET PATHS
940 PRINT PAGE
945 INPUT "TYPE NAME OF SCENARIO FILE DESIRED ",Scen$
950 Scen$=Scen$&"T14"
955 ASSIGN #2 TO Scen$
960 PRINTER IS 16
965 PRINT PAGE
970 INPUT "WOULD YOU LIKE A HARD COPY OF SCENARIO DATA (Y/N)? ",Ans$
975 IF Ans$="Y" THEN PRINTER IS 0
980 ! READ CG PROGRAM AND SCENARIO NUMBER
985 READ #2;Progrm$
990 READ #2;Scenno
995 PRINT PAGE
1000 PRINT USING L131;"** SCENARIO DATA **"
1005 L131: IMAGE K
1010 PRINT
1015 PRINT USING L132;"CG PROGRAM=",Progrm$
1020 L132: IMAGE 1X,K,K
1025 PRINT USING L133;"SCENARIO NO. =",Scenno
1030 L133: IMAGE 1X,K,2D
1035 ! READ MAXIMUM TIME
1040 READ #2;Mxtime
1045 PRINT USING L127;"MAXIMUM TIME=",Mxtime
1050 L127: IMAGE 1X,K,4D.D
1055 ! READ FRACTION OF RANGE THAT CAN BE USED
1060 READ #2;Rangfr
1065 PRINT USING L137;"RANGE FRACTION=",Rangfr
1070 L137: IMAGE 1X,K,D.2D
1075 ! READ NUMBER OF DAYS IN OPERATION
1080 READ #2;Ndays
1085 PRINT USING L506;" NO. DAYS OF OPERATION=",Ndays
1090 L506: IMAGE 1X,K,4D
1095 ! READ THE NUMBER OF IMPORTANT TASKS AND THE IMPORTANT TASKS
1100 ! R=RATE OF TASK,T=TASK NUMBER; USER CAN INPUT IMPORTANT
1105 ! TASKS (10/LINE) AFTER HE INPUTS THE NUMBER OF IMPORTANT TASKS
1110 READ #2;Nmimtk
1115 FOR I=1 TO Nmimtk
1120 READ #2;Imrate(I),Imtskn(I)
1125 NEXT I
1130 FOR I=1 TO Nmimtk

```

```

1135      PRINT Imrate(I),Imtskn(I)
1140 L520:  Mtask(I)=Imrate(I)+100+Imtskn(I)
1145      NEXT I
1150      PRINT USING L508;" NUMBER OF IMPORTANT TASKS=",Nmimtk
1155 L508:  IMAGE K,2D
1160      FOR I=1 TO Nmimtk
1165      PRINT USING L802;Mtask(I)
1170 L802:  IMAGE #,1X,3D
1175      NEXT I
1180      PRINT
1185      FOR I=1 TO Nmimtk
1190      IF Imrate(I)=0 THEN L510
1195      Imptsk(Imtskn(I),Imrate(I))=1
1200 L509:  NEXT I
1205 L510:  CONTINUE
1210 ! READ NUMBER OF NODES
1215      READ #2;Nnodov
1220      PRINT USING L126;" NODES=",Nnodov
1225 L126:  IMAGE K,2D
1230 ! READ CONNECTION MATRIX
1235 ! THE NUMBER OF PROBABILITIES PER LINE = THE NUMBER OF NODES.
1240 ! IF >15, USE TWO LINES WITH 15 PROBABILITIES ON THE FIRST LINE.
1245      PRINT USING L120;" CONNECTION MATRIX="
1250 L120:  IMAGE K
1255      FOR I=1 TO Nnodov
1260      FOR J=1 TO Nnodov
1265      READ #2;Ovcnm(I,J)
1270      PRINT USING L118;Ovcnm(I,J)
1275 L118:  IMAGE #,DDD.D,2X
1280 L108:  NEXT J
1285      PRINT
1290      NEXT I
1295 ! READ GROUP PLACEMENT MATRIX
1300 ! THE NUMBER OF GROUPS PER LINE = THE NUMBER OF NODES.
1305 ! IF >15, USE TWO LINES WITH 15 GROUPS ON THE FIRST LINE.
1310      PRINT USING L121;" GROUP PLACEMENT MATRIX="
1315 L121:  IMAGE K
1320      FOR I=1 TO Nnodov
1325      FOR J=1 TO Nnodov
1330      READ #2;Gpplm(I,J)
1335      PRINT USING L117;Gpplm(I,J)
1340      NEXT J
1345      PRINT
1350      NEXT I
1355 L117:  IMAGE #,DDDD,2X
1360 ! READ GROUP DATA
1365      READ #2;Iright,I down
1370 ! NOW READ GROUP DATA MATRIX
1375      FOR I=1 TO I down
1380      FOR J=1 TO Iright
1385      READ #2;Beta(J,I)
1390      NEXT J
1395      NEXT I
1400 ! CONTINUE ON AND READ THE REST OF THE DATA
1405      FOR I=1 TO I down
1410      FOR J=1 TO Iright-2
1415      Gdat2(I,J)=Beta(J+2,I)
1420      NEXT J
1425      NEXT I
1430      FOR I=1 TO 25
1435      FOR J=1 TO 2
1440      Gdat1(I,J)=Beta(J,I)
1445      NEXT J
1450      NEXT I
1455      PRINT USING L122;" &GROUP DATA="
1460 L122:  IMAGE K

```

```

1465      FOR I=1 TO Idown
1470      PRINT USING L109;Gpdat1(I,1),Gpdat1(I,2)
1475 L109:  IMAGE #,1X,DDD.DD,1X,DDD.DD
1480      FOR J=1 TO Irigh-2
1485      PRINT USING L110;Gpdat2(I,J)
1490 L110:  IMAGE #,DDD.DD
1495      NEXT J
1500      PRINT
1505      NEXT I
1510      PRINT USING L105;" &END"
1515 L105:  IMAGE K
1520 ! READ NUMBER OF PRINTOUTS
1525      READ #2;Nptout
1530      PRINT USING L141;" NUMBER OF PRINTOUTS= ",Nptout
1535 L141:  IMAGE K,DD
1540 ! READ OUTPUT FORMAT
1545 ! FLAG=1 FOR FULL OUTPUT
1550 ! FLAG=2 FOR PARTIAL OUTPUT:NO SORTIES PRINTED
1555      READ #2;Flag
1560      PRINT USING L143;" OUTPUT FORMAT=",Flag
1565 L143:  IMAGE K,D
1570 PRINTER IS 16
1575      PRINT
1580      PRINT "NOTE: THERE MAY BE MANY PAGES OF OUTPUT IF THE HARDCOPY"
1585      PRINT "      OPTION IS SELECTED FOR THE INDIVIDUAL SORTIES"
1590 ! CONVERT FUEL(TONS) TO FUEL(GALLONS)
1595      Mxfuel=Fuelcp
1600      Mxgals=Mxfuel*335*Rangfr
1605 ! CALCULATE TPOS(TASK PROBABILITY OF SUCCESS)
1610      FOR I=1 TO Nmstsk
1615      Tposmx(I)=Cc(I)*Df(I)*Mn(I)*Ls(I)*Tw(I)
1620      NEXT I
1625 ! INITIALIZE
1630      Npath=0
1635 ! ZERO OUT TIME, FUEL AND TASK COUNTERS
1640      PRINTER IS 16
1645      INPUT "WOULD YOU LIKE A HARD COPY OF INDIVIDUAL SORTIE OUTPUTS (Y/N)
)? ",Ans$
1650      IF Ans$="Y" THEN PRINTER IS 0
1655      Totim=0
1660      Totfue=0
1665      FOR I=1 TO 25
1670      FOR J=1 TO 4
1675      Totcnt(I,J)=0
1680      NEXT J
1685      NEXT I
1690      Pthtim=0
1695      Pthfue=0
1700      Pthprb=1
1705      Ptr=0
1710      Totprb=0
1715      FOR I=1 TO 100
1720      Pshist(I)=0
1725      Tlmlst(I)=0
1730      Fuelst(I)=0
1735      Prblst(I)=0
1740      NEXT I
1745      FOR I=1 TO 3
1750      FOR J=1 TO 3
1755      Visdis(I,J)=Visds1(I,J)
1760      NEXT J
1765      NEXT I
1770 ! GET MINIMUM FUEL AND TIME PATHS
1775      N=Nnodov
1780      CALL Fminph(N,Ovchmx(*),Gpplmx(*),Mintim(*),Minfue(*))
1785 ! FIND PATHS (SORTIES) THROUGH THE FLOWCHART

```

```

1790 ! START AT OVERALL NODE 1
1795   Pround=1
1800   Lsoug=0
1805   Entry=1
1810   Lktime=0
1815   Lkfue1=0
1820   Lkprob=1
1825   GOSUB Fpush
1830 ! GET NEXT OVERALL NODE
1835 L10: IF Mintim(Pround)+Pthtim>Mxtime THEN L90
1840       IF Minfue(Pround)+Pthfue>Mxgals THEN L90
1845       Begin=Lsoug+1
1850       IF Begin>Nnodov THEN L90
1855       FOR J=Begin TO Nnodov
1860           Ouprob=Ovcnmx(Pround,J)
1865           IF Ouprob>0 THEN L24
1870       NEXT J
1875       Ouprob=1
1880       GOTO L90
1885 ! FOUND AN OVERALL NODE TO GO TO
1890 L24: Nxound=J
1895 ! GET THIS LINK'S GROUP (IF ANY) AND START AT GROUPNODE 1
1900 L20: Lnksgp=Gpplmx(Pround,Nxound)
1905       IF Lnksgp=0 THEN L40
1910       Prgpnd=100*Lnksgp+1
1915       Lsgpgo=0
1920       Entry=Prgpnd
1925       Lktime=0
1930       Lkfue1=0
1935       Lkprob=Ouprob
1940       GOSUB Fpush
1945       Pthprb=Pthprb*Ouprob
1950       Ouprob=1
1955 ! GET NEXT GROUP NODE IN PRESENT GROUP
1960 L60: Begin=FNfxnode(Lsgpgo)+1
1965       Group=FNfgroup(Prgpnd)
1970       Inst=FNfinst(Prgpnd)
1975       Node=FNfxnode(Prgpnd)
1980       IF Group=90 THEN L890
1985       Numnds=Nnode(Group)
1990       GOTO L891
1995 L890: Numnds=9
2000 L891: IF Begin>Numnds THEN L80
2005       FOR J=Begin TO Numnds
2010           CALL Flkdat(Group,Inst,Node,J,Lkprob,Lktime,Lkfue1)
2015 ! CHECK FOR TIME AND FUEL
2020           IF (Lkprob>0) AND (Pthtim+Lktime<=Mxtime) AND (Pthfue+Lkfue1<=Mxgals)
2025           THEN L62
2030           NEXT J
2035           GOTO L80
2035 ! FOUND A GROUP NODE TO GO TO IN PRESENT GROUP
2040 L62: Node=J
2045       Nxgpnd=FNfpack(Node,Group,Inst)
2050       Entry=Nxgpnd
2055       GOSUB Fpush
2060       Pthtim=Pthtim+Lktime
2065       Pthfue=Pthfue+Lkfue1
2070       Pthprb=Pthprb*Lkprob
2075 ! TEST IF END OF PATH IN GROUP
2080       IF (Node=2) OR (Node=9) THEN L40
2085       Prgpnd=Nxgpnd
2090       Lsgpgo=0
2095       GOTO L60
2100 ! NO GROUP FOR THIS OVERALL LINK -OR- FINISHED THIS LINK
2105 L40: IF Nxound=2 THEN L42
2110       Lsoug=0

```

```

2115      Pround=Nxound
2120      Entry=Nxound
2125      Lktime=0
2130      Lkfue1=0
2135      Lkprob=0uprob
2140      GOSUB Fpush
2145      Pthprb=Pthprb*0uprob
2150      GOTO L10
2155 ! FOUND A COMPLETE PATH
2160 L42:   Entry=2
2165      Lktime=0
2170      Lkfue1=0
2175      Lkprob=0uprob
2180      GOSUB Fpush
2185      Pthprb=Pthprb*0uprob
2190      GOTO L200
2195 L201:  GOSUB Fpop
2200      Pthprb=Pthprb/Popprb
2205 L44:   IF Pshlst(Ptr)>=10000 THEN L92
2210 ! WENT BACK TO OVERALL NODE
2215      Lsoug0=Nxound
2220      GOTO L10
2225 ! WENT BACK TO GROUP NODE.
2230 L92:   Pround=FNHfouvnd(Dummy)
2235      GOTO L80
2240 ! TOP IS A GROUP NODE, WANT TO POP IT
2245 L90:   GOSUB Fpop
2250      Pthtim=Pthtim-Poptim
2255      Pthfue=Pthfue-Popfue
2260      Pthprb=Pthprb/Popprb
2265      IF FNHfxnode(Popped)=1 THEN L44
2270      Lsgpgo=Popped.
2275      Prgpnd=Pshlst(Ptr)
2280      GOTO L60
2285 ! AT OVERALL NODE WITH NO OVERALL NODES TO GO TO -OR-
2290 ! WITH NOT ENOUGH TIME OR FUEL LEFT: REMOVE TOP OVERALL NODE
2295 L90:   GOSUB Fpop
2300      Pthprb=Pthprb/Popprb
2305      IF Popped=1 THEN L9999
2310      Lsoug0=Popped
2315      Nxound=Popped
2320      IF Pshlst(Ptr)>=10000 THEN L92
2325      Pround=Pshlst(Ptr)
2330      GOTO L10
2335 ! FOUND A PATH: FIND PATH DATA AND PRINT IT OUT
2340 L200:  Npath=Npath+1
2345      Iphfue=Pthfue
2350      Unadjp=1
2355      FOR I=1 TO Ptr
2360      Unadjp=Unadjp+Prblst(I)
2365      NEXT I
2370      Unajpb(Npath)=Unadjp
2375      Totprb=Totprb+Unadjp
2380 ! STORE SORTIE TIME AND FUEL IN ARRAYS
2385      Pathtm(Npath)=Pthtim
2390      Ipthfu(Npath)=Iphfue
2395      FOR Nprtd=1 TO Nptous
2400 ! WRITE HEADING
2405      PRINT
2410      DISP "SORTIE NUMBER",Npath," PRINTING"
2415 L221:  PRINT PAGE
2420      PRINT
2425      PRINT
2430      PRINT
2435      IF Flag=1 THEN PRINT USING L209;Program$," SCENARIO ",Scenno
2440 L209:  IMAGE 30X,K,K,2D

```

```

2445      IF Flag=1 THEN PRINT USING L210;"SORTIE NUMBER ",Npath
2450 L210:  IMAGE 29X,K,4D
2455      PRINT
2460      PRINT
2465      IF Flag=1 THEN PRINT USING L211;"  OPERATIONAL REQUIREMENTS:","SELEC
TED CRAFT"
2470 L211:  IMAGE 13X,K,10X,K
2475      PRINT
2480      IF Flag=1 THEN PRINT USING L212;"  MAXIMUM DURATION ",Mxtime," HOURS
",Crftms
2485 L212:  IMAGE 13X,K,4D.D,K,6X,K
2490      IF Flag=1 THEN PRINT USING L213;"  RANGE FRACTION ",Rangfr,"DISPLACE
MENT",Idisp," TONS"
2495 L213:  IMAGE 13X,K,D.2D,16X,K,5D,K
2500      IF Flag=1 THEN PRINT USING L214;"  VISIBILITY ",Visds2$,"DESIGN SPEE
D",Idspd," KNOTS"
2505 L214:  IMAGE 13X,K,12A,12X,K,1X,2D,K
2510      IF Flag=1 THEN PRINT USING L215;"  AVERAGE SEA STATE ",Ssavg,"FUEL F
RACTION",Fufrc
2515 L215:  IMAGE 13X,K,D.D,14X,K,D.2D
2520      IF Flag=1 THEN PRINT
2525      IF Flag=1 THEN PRINT
2530      IF Flag=1 THEN PRINT USING L202;" GROUP","TASK","LOCATION","TASK","T
ASK","TASK"
2535 L202:  IMAGE 10X,K,5X,K,18X,K,2X,K,4X,K,4X,K
2540      IF Flag=1 THEN PRINT USING L203;"NAME","NAME","CODE","TIME","FUEL","
POS"
2545 L203:  IMAGE 11X,K,6X,K,20X,K,4X,K,4X,K,5X,K
2550      IF Flag=1 THEN PRINT USING L217;"(HRS)","(GALS)"
2555 L217:  IMAGE 53X,K,2X,K
2560      IF Flag=1 THEN PRINT
2565      IF Flag=1 THEN PRINT
2570      Posprd=1
2575      Ccmin=9999
2580      Dfmin=9999
2585      Mnmin=9999
2590      Lsmin=9999
2595      Twmin=9999
2600 ! ZERO OUT TASK COUNTER AFTER A CRAFT HAS
2605 ! COMPLETED A SORTIE
2610      FOR I=1 TO 25
2615      FOR J=1 TO 4
2620      Count(I,J)=0
2625      NEXT J
2630      NEXT I
2635      Ntask=0
2640      FOR I=1 TO Ptr
2645      Locatn=Pshlst(I)
2650      Inst=FNFinst(Locatn)
2655      Group=FNFGgroup(Locatn)
2660      IF Group=0 THEN L204
2665      Node=FNFXnode(Locatn)
2670      IF Node=1 THEN L205
2675      Node1=Node2
2680      Node2=Node
2685      CALL Ftask(Group,Node1,Node2,Taskno1,Rate6)
2690      Rate=Rate6
2695      Taskno=Taskno1
2700      Taskn1=Taskno
2705      IF Group>=90 THEN Taskn1=Taskn1+19
2710      Rate1=Rate
2715      IF Group>=90 THEN Rate1=1
2720 ! NOTE: COUNTER FOR THE 3 SEARCH FAILURES ARE STORED
2725 ! IN RATE 1 AND TASKNOS 20,22 AND 25
2730      IF Taskno=0 THEN L204
2735 ! TASK IS PERFORMED INCREMENT TASK COUNTER

```

```

2740      Count(Taskn1,Rate1)=Count(Taskn1,Rate1)+1
2745 L98:   Mastsk=Master(Taskno,Rate)
2750 ! FIND CC FOR MASTER TASK 17
2755 ! FIND MINIMUM VALUE OF EACH PARAMETER: CC,DF,MN,LS,TW
2760      Cc0=Cc(Mastsk)
2765      IF Mastsk=17 THEN GOSUB Fcc17
2770      IF Cc0<=Ccmin THEN Ccmin=Cc0
2775      IF Df(Mastsk)<=Dfmin THEN Dfmin=Df(Mastsk)
2780      IF Mn(Mastsk)<=Mnmin THEN Mnmin=Mn(Mastsk)
2785      IF Ls(Mastsk)<=Lsmin THEN Lsmin=Ls(Mastsk)
2790      IF Tw(Mastsk)<=Twmin THEN Twmin=Tw(Mastsk)
2795      Tpos=Tposmx(Mastsk)
2800      IF Mastsk=17 THEN Tpos=Cc0*Df(17)*Mn(17)*Ls(17)*Tw(17)
2805      Ifulst=Fuelst(I)
2810      IF Group>=90 THEN L896
2815      IF Flag=1 THEN PRINT USING L48;"*",Tasknos(Taskno,Rate),Locatn,Timlst(I),Ifulst,Tpos
2820 L48:   IMAGE 17X,K,24A,2X,6D,2X,3D.D,2X,6D,3X,2D.2D
2825      GOTO L825
2830 ! TASK FAILURE
2835 L896:  IF Flag=1 THEN PRINT USING L893;"*",Tasknos(Taskno,Rate),": FAILED",Locatn,Timlst(I),Ifulst,Tpos
2840 L893:  IMAGE 17X,K,16A,K,2X,6D,2X,3D.D,2X,6D,3X,2D.2D
2845 L825:  Ntask=Ntask+1
2850      GOTO L47
2855 L205:  Basgrp=Group
2860      IF Group>=90 THEN Basgrp=Group-80
2865      IF Flag=1 THEN PRINT USING L206;Grpnm$(Basgrp),Locatn
2870 L206:  IMAGE 11X,15A,18X,6D
2875      Node2=1
2880      GOTO L47
2885 L204:  IF Flag=1 THEN PRINT USING L207;Locatn
2890 L207:  IMAGE 44X,6D
2895 L47:  NEXT I
2900 ! CALCULATE PATH PROBABILITY OF SUCCESS
2905      Pthpos=Ccmin*Dfmin*Mnmin*Lsmin*Twmin
2910      IF Flag=1 THEN PRINT
2915      IF Flag=1 THEN PRINT USING L208;"TIME TO COMPLETE SORTIE (HRS)",Pthtim
2920 L208:  IMAGE 11X,K,12X,3D.D
2925      IF Flag=1 THEN PRINT USING L401;"FUEL CONSUMED IN SORTIE (GALS)",Iphfue
2930 L401:  IMAGE 11X,K,18X,6D
2935      IF Flag=1 THEN PRINT
2940      IF Flag=1 THEN PRINT USING L402;"SORTIE PROBABILITY OF SUCCESS",Pthpos
2945 L402:  IMAGE 18X,K,22X,D.4D
2950      IF Flag=1 THEN PRINT USING L403;"SORTIE FREQUENCY OF OCCURRENCE",Unajpb(Npath)
2955 L403:  IMAGE 18X,K,21X,D.4D
2960      Pt=Pthpos*Unajpb(Npath)
2965      Totim=Totim+Pthtim*Pt
2970      Totfue=Totfue+Pt*Pthfue
2975 ! LOOP TO INCREMENT TOTAL COUNTER FOR TASKS
2980      FOR I=1 TO 25
2985      FOR J=1 TO 4
2990      Totcnt(I,J)=Totcnt(I,J)+Pt*Count(I,J)
2995      NEXT J
3000      NEXT I
3005      Phpos(Npath)=Pthpos
3010      NEXT Nprted
3015      DISP "SORTIE NUMBER",Npath," COMPLETED"
3020      GOTO L201
3025 ! PRINT SORTIE SUMMARY
3030 L9999: ! CONTINUE
3035      PRINTER IS 16

```

```

3040      PRINT
3045      DISP Npath," PAGES OF SORTIE OUTPUT COMPLETED. PRESS CONT TO GO ON
"
3050 L219:  IMAGE K,DD,K
3055      BEEP
3060      PAUSE
3065      DISP
3070      FOR Nprted=1 TO Nptout
3075      PRINTER IS 16
3080      PRINT PAGE
3085      INPUT "WOULD YOU LIKE A HARD COPY OF SORTIE SUMMARY, (Y/N)? ",Ans$
3090      IF Ans$="Y" THEN PRINTER IS 0
3095      PRINT PAGE
3100      PRINT
3105      PRINT
3110      PRINT USING L9991;"***** SORTIE SUMMARY *****"
3115 L9991:  IMAGE 18X,K
3120      PRINT
3125      PRINT USING L9992;Progrm$," SCENARIO ",Scenno
3130 L9992:  IMAGE 30X,K,K,2D
3135      PRINT
3140      PRINT
3145      PRINT USING L211;" OPERATIONAL REQUIREMENTS:","SELECTED CRAFT"
3150      PRINT
3155      PRINT USING L212;" MAXIMUM DURATION ",Mxtime," HOURS",Crfrm$
3160      PRINT USING L213;" RANGE FRACTION ",Rangfr,"DISPLACEMENT",Idisp," T
OMS"
3165      PRINT USING L215;" AVERAGE SEA STATE ",Ssavg,"FUEL FRACTION",Fufrc
3170      IF Npath=0 THEN L9994
3175      PRINT
3180      PRINT
3185      PRINT USING L9997;"FRACTION OF SCENARIO COMPLETED ",Totprb
3190 L9997:  IMAGE 23X,K,D,4D
3195      PRINT
3200      PRINT USING L9993;"SORTIE","SORTIE","SORTIE","FREQUENCY","SORTIE","S
ORTIE"
3205 L9993:  IMAGE 13X,K,2X,K,2X,K,2X,K,8X,K,8X,K
3210      PRINT USING L9994;"NO.", "TIME", "FUEL", "OF", "PROBABILITY", "SUCCESSFUL
"
3215 L9994:  IMAGE 15X,K,4X,K,4X,K,7X,K,8X,K,4X,K
3220      PRINT USING L9995;"(HRS)","(GALS)","OCCURRENCE", "OF", "SUCCESS"
3225 L9995:  IMAGE 22X,K,2X,K,2X,K,4X,K,2X,K,4X,K
3230      PRINT
3235      PRINT
3240      Propos=0
3245      FOR I=1 TO Npath
3250      Adjprb=Unajpb(I)
3255      Pthpos=Phpos(I)
3260      Contrb=Pthpos*Adjprb
3265      Propos=Propos+Contrb
3270      PRINT USING L9998;I,Pathtm(I),Ipthfu(I),Adjprb,Pthpos,Contrb
3275 L9998:  IMAGE 13X,4D,4X,3D,D,3X,6D,5X,D,4D,8X,D,4D,8X,D,4D
3280      NEXT I
3285      ! CALCULATE TIME TO COMPLETE AVERAGE SORTIE AND
3290      ! FUEL CONSUMED IN AVERAGE SORTIE
3295      AveTime=Totim/Propos
3300      AveFue=Totfue/Propos
3305      PRINTER IS 16
3310      PRINT
3315      DISP "          END OF SORTIE SUMMARY. PRESS CONT TO GO ON."
3320      BEEP
3325      PAUSE
3330      DISP
3335      ! PRINT OVERALL RESULTS
3340      PRINT PAGE
3345      INPUT "WOULD YOU LIKE A HARD COPY OF SCENARIO OVERALL RESULTS (Y/N)?

```

```

3345      INPUT "WOULD YOU LIKE A HARD COPY OF SCENARIO OVERALL RESULTS (Y/N)?
",Ans$
3350      IF Ans$="Y" THEN PRINTER IS 0
3355      PRINT PAGE
3360      PRINT
3365      PRINT USING L480;"***** SCENARIO OVERALL RESULTS *****
**"
3370 L480:  IMAGE 13X,K
3375      PRINT
3380      PRINT USING L481;Program$," SCENARIO ",Scenno
3385 L481:  IMAGE 30X,K,K,2D
3390      PRINT
3395      PRINT
3400      PRINT USING L211;"  OPERATIONAL REQUIREMENTS:","SELECTED CRAFT"
3405      PRINT
3410      PRINT USING L212;"  MAXIMUM DURATION ","Mxtime"," HOURS",Crfnm$
3415      PRINT USING L213;"  RANGE FRACTION ","Rangfr","DISPLACEMENT",Idisp," T
ONS"
3420      PRINT USING L214;"  VISIBILITY ","Visds2$","DESIGN SPEED",Idspd," KNOT
S"
3425      PRINT USING L215;"  AVERAGE SEA STATE ",Ssavg,"FUEL FRACTION",Fufrc
3430      Perprb=Totprb*100
3435      PRINT
3440      PRINT
3445      PRINT USING L405;" PERCENT OF SCENARIO COMPLETED ",Perprb
3450 L405:  IMAGE 23X,K,3D.D
3455      PRINT
3460      PRINT USING L406;"PROBABILITY OF SUCCESSFULLY COMPLETING SCENARIO ",
Propos
3465 L406:  IMAGE 15X,K,2X,D.2D
3470      PRINT
3475      PRINT
3480      PRINT USING L407;"SPECIFICATIONS OF THE AVERAGE SORTIE:"
3485 L407:  IMAGE 15X,K
3490      PRINT
3495      PRINT USING L408;"TIME TO COMPLETE AVERAGE SORTIE",Avetim," HRS"
3500 L408:  IMAGE 23X,K,6D.D,K
3505      PRINT
3510      PRINT USING L409;"FUEL CONSUMED IN AVERAGE SORTIE",Avefue,"GALS"
3515 L409:  IMAGE 23X,K,6D.D,K
3520      PRINT
3525      PRINT
3530      PRINT USING L410;"TASK COMPOSITION IN AVERAGE SORTIE:"
3535 L410:  IMAGE 15X,K
3540      PRINT
3545      PRINT USING L522;"TASK","TIMES","TASK"
3550 L522:  IMAGE 23X,K,6X,K,5X,K
3555      PRINT USING L523;"CODE","COMPLETED","NAME"
3560 L523:  IMAGE 23X,K,4X,K,3X,K
3565      PRINT
3570      PRINT USING L7002;"ON SCENE:"
3575 L7002:  IMAGE 19X,K
3580      IF Totcnt<1,4>>0 THEN PRINT USING L7003;"BRD",Totcnt<1,4>,"BOARD"
3585 L7003:  IMAGE 23X,K,6X,2D.2D,5X,K
3590      IF Totcnt<2,4>>0 THEN PRINT USING L7003;"FFF",Totcnt<2,4>,"FIGHT FIR
E FROM CG VESSEL"
3595      IF Totcnt<3,4>>0 THEN PRINT USING L7003;"FFO",Totcnt<3,4>,"FIGHT FIR
E ON ANOTHER VESSEL"
3600      IF Totcnt<4,4>>0 THEN PRINT USING L7003;"GAS",Totcnt<4,4>,"GENERAL A
SSISTANCE"
3605      IF Totcnt<5,4>>0 THEN PRINT USING L7003;"INS",Totcnt<5,4>,"INSPECTIO
N"
3610      IF Totcnt<6,4>>0 THEN PRINT USING L7003;"LEQ",Totcnt<6,4>,"LOAD EQUI
PMENT"
3615      IF Totcnt<7,4>>0 THEN PRINT USING L7003;"LOI",Totcnt<7,4>,"LOITER"
3620      IF Totcnt<8,4>>0 THEN PRINT USING L7003;"LSB",Totcnt<8,4>,"LAUNCH SM

```

```

ALL BOATS"
3625 IF Totcnt(9,4)>0 THEN PRINT USING L7003;"MAC",Totcnt(9,4),"MONITOR A
CTIVITIES"
3630 IF Totcnt(10,4)>0 THEN PRINT USING L7003;"MOS",Totcnt(10,4),"MONITOR
OIL SPILL"
3635 IF Totcnt(11,4)>0 THEN PRINT USING L7003;"QBA",Totcnt(11,4),"ON BOAR
D ASSISTANCE"
3640 IF Totcnt(12,4)>0 THEN PRINT USING L7003;"OSC",Totcnt(12,4),"ON SCEN
E COMMANDER(GENERAL)"
3645 IF Totcnt(13,4)>0 THEN PRINT USING L7003;"RBP",Totcnt(13,4),"RETRIEV
E BOARDING PARTY"
3650 IF Totcnt(14,4)>0 THEN PRINT USING L7003;"ROB",Totcnt(14,4),"RETRIEV
E OBJECTS"
3655 IF Totcnt(15,4)>0 THEN PRINT USING L7003;"RPE",Totcnt(15,4),"RESCUE
PEOPLE"
3660 IF Totcnt(16,4)>0 THEN PRINT USING L7003;"RSB",Totcnt(16,4),"RETRIEV
E SMALL BOAT"
3665 IF Totcnt(17,4)>0 THEN PRINT USING L7003;"SSI",Totcnt(17,4),"STAKEOU
T SPECIAL INTEREST VESSEL"
3670 IF Totcnt(18,4)>0 THEN PRINT USING L7003;"SZE",Totcnt(18,4),"SEIZE"
3675 IF Totcnt(19,4)>0 THEN PRINT USING L7003;"TWS",Totcnt(19,4),"TAKE WA
TER SAMPLE"
3680 IF Totcnt(20,4)>0 THEN PRINT USING L7003;"ULQ",Totcnt(20,4),"UNLOAD
EQUIPMENT"
3685 IF Totcnt(21,4)>0 THEN PRINT USING L7003;"WQB",Totcnt(21,4),"WORK EQ
UIPMENT FROM SMALL BOAT"
3690 IF Totcnt(22,4)>0 THEN PRINT USING L7003;"WQD",Totcnt(22,4),"WORK EQ
UIPMENT @ DRIFT"
3695 IF Totcnt(23,4)>0 THEN PRINT USING L7003;"WQF",Totcnt(23,4),"WORK EQ
UIPMENT @ FIXED POSITION"
3700 FOR I=1 TO 25
3705 IF Totcnt(I,4)>0 THEN L494
3710 NEXT I
3715 PRINT USING L7027;"NO TASKS"
3720 L7027: IMAGE 23X,K
3725 L494: PRINT
3730 PRINT USING L8002;"REDUCED SPEED:"
3735 L8002: IMAGE 19X,K
3740 IF Totcnt(1,3)>0 THEN PRINT USING L7003;"SDU",Totcnt(1,3),"SEARCH FO
R DISTRESSED UNIT: FOUND"
3745 IF Totcnt(20,1)>0 THEN PRINT USING L7003;"SDU",Totcnt(20,1),"SEARCH
FOR DISTRESSED UNIT: FAILED"
3750 IF Totcnt(2,3)>0 THEN PRINT USING L7003;"SES",Totcnt(2,3),"SLOW ESCO
RT"
3755 IF Totcnt(3,3)>0 THEN PRINT USING L7003;"SPE",Totcnt(3,3),"SEARCH FO
R PEOPLE: FOUND"
3760 IF Totcnt(22,1)>0 THEN PRINT USING L7003;"SPE",Totcnt(22,1),"SEARCH
FOR PEOPLE: FAILED"
3765 IF Totcnt(4,3)>0 THEN PRINT USING L7003;"SPT",Totcnt(4,3),"SLOW PATR
OL"
3770 IF Totcnt(5,3)>0 THEN PRINT USING L7003;"TOW",Totcnt(5,3),"TOW"
3775 FOR I=1 TO 5
3780 IF Totcnt(I,3)>0 THEN L495
3785 NEXT I
3790 PRINT USING L7027;"NO TASKS"
3795 L495: PRINT
3800 PRINT USING L8008;"CRUISE SPEED"
3805 L8008: IMAGE 19X,K
3810 IF Totcnt(1,2)>0 THEN PRINT USING L7003;"ESC",Totcnt(1,2),"ESCORT"
3815 IF Totcnt(2,2)>0 THEN PRINT USING L7003;"IDC",Totcnt(2,2),"IDENTIFY
CRAFT"
3820 IF Totcnt(3,2)>0 THEN PRINT USING L7003;"IDF",Totcnt(3,2),"IDENTIFY
FLEET"
3825 IF Totcnt(4,2)>0 THEN PRINT USING L7003;"PAT",Totcnt(4,2),"PATROL"
3830 IF Totcnt(5,2)>0 THEN PRINT USING L7003;"SFL",Totcnt(5,2),"SEARCH FO
R FLEET"

```

```

3835      IF Totcnt(6,2)>0 THEN PRINT USING L7003;"SSH",Totcnt(6,2),"SEARCH FO
R SHIP: FOUND"
3840      IF Totcnt(25,1)>0 THEN PRINT USING L7003;"SSH",Totcnt(25,1),"SEARCH
FOR SHIP: FAILED"
3845      IF Totcnt(7,2)>0 THEN PRINT USING L7003;"TEQ",Totcnt(7,2),"TRANSPORT
EQUIPMENT"
3850      IF Totcnt(8,2)>0 THEN PRINT USING L7003;"TPE",Totcnt(8,2),"TRANSPORT
PEOPLE"
3855      IF Totcnt(9,2)>0 THEN PRINT USING L7003;"TRA",Totcnt(9,2),"TRANSIT"
3860      FOR I=1 TO 9
3865      IF Totcnt(I,2)>0 THEN L497
3870      NEXT I
3875      PRINT USING L7027;"NO TASKS"
3880 L497: PRINT
3885      PRINT USING L8000;"FLANK SPEED:"
3890      IF Totcnt(1,1)>0 THEN PRINT USING L7003;"DSH",Totcnt(1,1),"DASH"
3895      IF Totcnt(2,1)>0 THEN PRINT USING L7003;"INT",Totcnt(2,1),"INTERDICT
"
3900      FOR I=1 TO 2
3905      IF Totcnt(I,1)>0 THEN L502
3910      NEXT I
3915      PRINT USING L7027;"NO TASKS"
3920 L502: PRINT
3925      PRINTER IS 16
3930      DISP "          END OF OVERALL RESULTS. PRESS CONT TO GO ON."
3935      BEEP
3940      PAUSE
3945      DISP
3950      PRINT PAGE
3955      INPUT "WOULD YOU LIKE A HARD COPY OF SCENARIO EVALUATION (Y/N)? ",An
s$
3960      IF Ans$="Y" THEN PRINTER IS 0
3965      PRINT PAGE
3970      PRINT
3975      PRINT USING L482;"***** SCENARIO EVALUATION *****"
3980 L482: IMAGE 16X,K
3985      PRINT USING L481;Progrm$," SCENARIO ",Scenno
3990      PRINT
3995      PRINT
4000      PRINT USING L211;" OPERATIONAL REQUIREMENTS:","SELECTED CRAFT"
4005      PRINT
4010      PRINT USING L212;" MAXIMUM DURATION ",Mxtime," HOURS",Crfrm$
4015      PRINT USING L213;" RANGE FRACTION ",Rangfr,"DISPLACEMENT",Idisp," T
ONS"
4020      PRINT USING L214;" VISIBILITY ",Visds2$,"DESIGN SPEED",Idspd," KNOT
S"
4025      FOR I=1 TO 25
4030      FOR J=1 TO 4
4035      Im(I,J)=Totcnt(I,J)*Ndays
4040      NEXT J
4045      NEXT I
4050      PRINT
4055      PRINT
4060      PRINT USING L511;"IMPORTANT TASKS COMPLETED IN ",Ndays," DAYS OF OPE
RATION"
4065 L511: IMAGE 18X,K,4D,K
4070      PRINT
4075      PRINT
4080      PRINT USING L512;"TASK","TIMES","TASK"
4085 L512: IMAGE 23X,K,6X,K,5X,K
4090      PRINT USING L513;"CODE","COMPLETED","NAME"
4095 L513: IMAGE 23X,K,4X,K,3X,K
4100      PRINT
4105      PRINT USING L2002;"ON SCENE:"
4110 L2002: IMAGE 19X,K
4115      ! IF AN IMPORTANT TASK IS NOT PERFORMED, ITS TASK CODE

```

```

4120 ! AND TASK NAME WILL STILL APPEAR IN THE OUTPUT, WITH
4125 ! THE NUMBER OF TIMES COMPLETED BEING 0.0. IF IT IS
4130 ! DESIRED AN IMPORTANT TASK NOT PERFORMED WILL NOT APPEAR
4135 ! IN THE OUTPUT.
4140 IF (Im(1,4))=0) AND (Imptsk(1,4)=1) THEN PRINT USING L2003;"BRD",Im(
1,4),"BOARD"
4145 L2003: IMAGE 23X,K,6X,5D,5X,K
4150 IF (Im(2,4))=0) AND (Imptsk(2,4)=1) THEN PRINT USING L2003;"FFF",Im(
2,4),"FIGHT FIRE FROM CG VESSEL"
4155 IF (Im(3,4))=0) AND (Imptsk(3,4)=1) THEN PRINT USING L2003;"FF0",Im(
3,4),"FIGHT FIRE ON ANOTHER VESSEL"
4160 IF (Im(4,4))=0) AND (Imptsk(4,4)=1) THEN PRINT USING L2003;"GAS",Im(
4,4),"GENERAL ASSISTANCE"
4165 IF (Im(5,4))=0) AND (Imptsk(5,4)=1) THEN PRINT USING L2003;"INS",Im(
5,4),"INSPECTION"
4170 IF (Im(6,4))=0) AND (Imptsk(6,4)=1) THEN PRINT USING L2003;"LEQ",Im(
6,4),"LOAD EQUIPMENT"
4175 IF (Im(7,4))=0) AND (Imptsk(7,4)=1) THEN PRINT USING L2003;"LOI",Im(
7,4),"LOITER"
4180 IF (Im(8,4))=0) AND (Imptsk(8,4)=1) THEN PRINT USING L2003;"LSB",Im(
8,4),"LAUNCH SMALL BOAT"
4185 IF (Im(9,4))=0) AND (Imptsk(9,4)=1) THEN PRINT USING L2003;"MAC",Im(
9,4),"MONITOR ACTIVITIES"
4190 IF (Im(10,4))=0) AND (Imptsk(10,4)=1) THEN PRINT USING L2003;"MOS",I
m(10,4),"MONITOR OIL SPILL"
4195 IF (Im(11,4))=0) AND (Imptsk(11,4)=1) THEN PRINT USING L2003;"OBA",I
m(11,4),"ON BOARD ASSISTANCE"
4200 IF (Im(12,4))=0) AND (Imptsk(12,4)=1) THEN PRINT USING L2003;"OSC",I
m(12,4),"ON SCENE COMMANDER(GENERAL)"
4205 IF (Im(13,4))=0) AND (Imptsk(13,4)=1) THEN PRINT USING L2003;"RBP",I
m(13,4),"RETRIEVE BOARDING PARTY"
4210 IF (Im(14,4))=0) AND (Imptsk(14,4)=1) THEN PRINT USING L2003;"ROB",I
m(14,4),"RETRIEVE OBJECTS"
4215 IF (Im(15,4))=0) AND (Imptsk(15,4)=1) THEN PRINT USING L2003;"RPE",I
m(15,4),"RESCUE PEOPLE"
4220 IF (Im(16,4))=0) AND (Imptsk(16,4)=1) THEN PRINT USING L2003;"RSB",I
m(16,4),"RETRIEVE SMALL BOAT"
4225 IF (Im(17,4))=0) AND (Imptsk(17,4)=1) THEN PRINT USING L2003;"SSI",I
m(17,4),"STAKEOUT SPECIAL INTEREST VESSEL"
4230 IF (Im(18,4))=0) AND (Imptsk(18,4)=1) THEN PRINT USING L2003;"SZE",I
m(18,4),"SEIZE"
4235 IF (Im(19,4))=0) AND (Imptsk(19,4)=1) THEN PRINT USING L2003;"TWS",I
m(19,4),"TAKE WATER SAMPLE"
4240 IF (Im(20,4))=0) AND (Imptsk(20,4)=1) THEN PRINT USING L2003;"ULQ",I
m(20,4),"UNLOAD EQUIPMENT"
4245 IF (Im(21,4))=0) AND (Imptsk(21,4)=1) THEN PRINT USING L2003;"WQB",I
m(21,4),"WORK EQUIPMENT FROM SMALL BOAT"
4250 IF (Im(22,4))=0) AND (Imptsk(22,4)=1) THEN PRINT USING L2003;"WQD",I
m(22,4),"WORK EQUIPMENT @ DRIFT"
4255 IF (Im(23,4))=0) AND (Imptsk(23,4)=1) THEN PRINT USING L2003;"WQF",I
m(23,4),"WORK EQUIPMENT @ FIXED POSITION"
4260 FOR I=1 TO 25
4265 IF (Im(I,4))=0) AND (Imptsk(I,4)=1) THEN L694
4270 NEXT I
4275 PRINT USING L2027;"NO IMPORTANT TASKS SPECIFIED"
4280 L694: PRINT
4285 PRINT USING L3001;"REDUCED SPEED:"
4290 L3001: IMAGE 19X,K
4295 IF (Im(1,3))=0) AND (Imptsk(1,3)=1) THEN PRINT USING L2003;"SDU",Im(
1,3),"SEARCH FOR DISTRESSED UNIT: FOUND"
4300 IF (Im(1,3))=0) AND (Imptsk(1,3)=1) THEN PRINT USING L2003;"SDU",Im(
20,1),"SEARCH FOR DISTRESSED UNIT: FAILED"
4305 IF (Im(2,3))=0) AND (Imptsk(2,3)=1) THEN PRINT USING L2003;"SES",Im(
2,3),"SLOW ESCORT"
4310 IF (Im(3,3))=0) AND (Imptsk(3,3)=1) THEN PRINT USING L2003;"SPE",Im(
3,3),"SEARCH FOR PEOPLE: FOUND"

```

```

4315         IF (Im(3,3)>=0) AND (Imptsk(3,3)=1) THEN PRINT USING L2003;"SPE",Im(
22,1),"SEARCH FOR PEOPLE: FAILED"
4320         IF (Im(4,3)>=0) AND (Imptsk(4,3)=1) THEN PRINT USING L2003;"SPT",Im(
4,3),"SLOW PATROL"
4325         IF (Im(5,3)>=0) AND (Imptsk(5,3)=1) THEN PRINT USING L2003;"TOW",Im(
5,3),"TOW"
4330         FOR I=1 TO 5
4335         IF (Im(I,3)>=0) AND (Imptsk(I,3)=1) THEN L695
4340         NEXT I
4345         PRINT USING L2027;"NO IMPORTANT TASKS SPECIFIED"
4350 L695: PRINT
4355         PRINT USING L3008;"CRUISE SPEED:"
4360 L3008: IMAGE 19X,K
4365         IF (Im(1,2)>=0) AND (Imptsk(1,2)=1) THEN PRINT USING L2003;"ESC",Im(
1,2),"ESCORT"
4370         IF (Im(2,2)>=0) AND (Imptsk(2,2)=1) THEN PRINT USING L2003;"IDC",Im(
2,2),"IDENTIFY CRAFT"
4375         IF (Im(3,2)>=0) AND (Imptsk(3,2)=1) THEN PRINT USING L2003;"IDF",Im(
3,2),"IDENTIFY FLEET"
4380         IF (Im(4,2)>=0) AND (Imptsk(4,2)=1) THEN PRINT USING L2003;"PAT",Im(
4,2),"PATROL"
4385         IF (Im(5,2)>=0) AND (Imptsk(5,2)=1) THEN PRINT USING L2003;"SFL",Im(
5,2),"SEARCH FOR FLEET"
4390         IF (Im(6,2)>=0) AND (Imptsk(6,2)=1) THEN PRINT USING L2003;"SSH",Im(
6,2),"SEARCH FOR SHIP: FOUND"
4395         IF (Im(6,2)>=0) AND (Imptsk(6,2)=1) THEN PRINT USING L2003;"SSH",Im(
25,1),"SEARCH FOR SHIP: FAILED"
4400         IF (Im(7,2)>=0) AND (Imptsk(7,2)=1) THEN PRINT USING L2003;"TEQ",Im(
7,2),"TRANSPORT EQUIPMENT"
4405         IF (Im(8,2)>=0) AND (Imptsk(8,2)=1) THEN PRINT USING L2003;"TPE",Im(
8,2),"TRANSPORT PEOPLE"
4410         IF (Im(9,2)>=0) AND (Imptsk(9,2)=1) THEN PRINT USING L2003;"TRA",Im(
9,2),"TRANSIT"
4415         FOR I=1 TO 9
4420         IF (Im(I,2)>=0) AND (Imptsk(I,2)=1) THEN L697
4425         NEXT I
4430         PRINT USING L7027;"NO IMPORTANT TASKS SPECIFIED"
4435 L697: PRINT
4440         PRINT USING L3013;"FLANK SPEED:"
4445 L3013: IMAGE 19X,K
4450         IF (Im(1,1)>=0) AND (Imptsk(1,1)=1) THEN PRINT USING L2003;"DSH",Im(
1,1),"DASH"
4455         IF (Im(2,1)>=0) AND (Imptsk(2,1)=1) THEN PRINT USING L2003;"INT",Im(
2,1),"INTERDICT"
4460         FOR I=1 TO 2
4465         IF (Im(I,1)>=0) AND (Imptsk(I,1)=1) THEN L702
4470         NEXT I
4475         PRINT USING L2027;"NO IMPORTANT TASKS SPECIFIED"
4480 L2027: IMAGE 23X,K
4485 L702: GOTO L9990
4490 L9994: PRINT
4495         PRINT
4500         PRINT USING L9993;"NO SORTIES CAN BE COMPLETED"
4505 L9993: IMAGE 24X,K
4510 L9990: NEXT Npnted
4515         PRINT
4520         DISP "                END OF SPRPOS OUTPUT."
4525         PRINTER IS 16
4530         RETURN
4535 ! FSSHP
4540 ! CALCULATES PROBABILITY AND TIME FOR SEARCH
4545 ! FOR SHIP TASK
4550 Fsshp: ! CONVERTED SUBROUTINE FSSHP
4555         IF (Sw=>0) OR (Tmax=>0) THEN Lh99
4560         Deltat=.1
4565         Delth=Deltat/2

```

```

4570      GOTO Lh90
4575 Lh99:  Ps=0
4580      Pf=1
4585      Ts=0
4590      Tf=0
4595      RETURN
4600      ! INITIALIZE COUNTERS
4605 Lh90:  Fprod=1
4610      Tnum=0
4615      Tdenom=0
4620      Psdelt=1
4625      ! START AT TIME ZERO
4630      T=0
4635      ! FIND TARGET'S AREA THAT IS STILL UNSEARCHED
4640 Lh20:  Atgtot=PI*(E+Vtar*(Tbef+T+Delth))^2
4645      Asrchd=Sw*Speed(2)*(T+Delth)
4650      Atguns=Atgtot-Asrchd
4655      ! TEST AREA UNSEARCHED; IF .LE. ASRCHD, THEN PS=1, ELSE CALCULATE
4660      ! PROBABILITY OF SUCCESS IN NEXT DELTA T
4665      IF Atguns<=Asrchd THEN Lh100
4670      Psdelt=Sw*Speed(2)*Deltat/Atguns
4675      Fprod=Fprod*(1-Psdelt)
4680      Tnum=Tnum+(T+Delth)*Psdelt
4685      Tdenom=Tdenom+Psdelt
4690      T=T+Deltat
4695      ! TEST FOR TIME
4700      IF T>Tmax THEN Lh101
4705      GOTO 4895
4710 Lh100: Ps=1
4715      Pf=0
4720      Tnum=Tnum+(T+Delth)*Psdelt
4725      Tdenom=Tdenom+Psdelt
4730      Ts=Tnum/Tdenom
4735      Tf=Tmax
4740      RETURN
4745      ! TIME IS TMAX
4750 Lh101: Ps=1-Fprod
4755      Pf=Fprod
4760      Ts=Tnum/Tdenom
4765      Tf=Tmax
4770      RETURN
4775      ! PTWD
4780      ! TOW DISPLACEMENT CUMULATIVE PROBABILITY DISTRIBUTION
4785      ! PTOWD(D)= PROBABILITY THAT CRAFT TO BE TOWED HAS DISPLACEMENT < D
4790      ! AVTWDs = AVERAGE DISPLACEMENT VALUE THAT CAN BE TOWED
4795 Ptwd:  ! CONVERTED SUBROUTINE PTWD
4800      In=Towdsp
4805 L10:   Out=FNFF8(In,0,0,Towdis(1,Towdtb),0,Towdis(2,Towdtb),.2,Towdis(3,Towdtb),.4,Towdis(4,Towdtb),.6,Towdis(5,Towdtb),.8,Towdis(6,Towdtb),1,999999,1)
4810      Ptwd=Out
4815      In=Ptwd/2
4820      Out=FNFF8(In,0,0,0,Towdis(1,Towdtb),.2,Towdis(2,Towdtb),.4,Towdis(3,Towdtb),.6,Towdis(4,Towdtb),.8,Towdis(5,Towdtb),1,Towdis(6,Towdtb),1,999999)
4825 L60:   Avtwd=Out
4830 L100:   RETURN
4835      ! PUSHDOWN LIST SUBROUTINES
4840      ! FPUSH
4845      ! PUT A NUMBER ON THE PUSHDOWN LIST
4850 Fpush:  ! CONVERTED SUBROUTINE FPUSH
4855      INTEGER Pshlst,Ptr,I,Entry
4860      Ptr=Ptr+1
4865      Pshlst(Ptr)=Entry
4870      Tmlst(Ptr)=Lktime
4875      Fuelst(Ptr)=Lkfuel
4880      Prblst(Ptr)=Lkprob
4885      RETURN

```

```

4890 !
4895 ! FPOP
4900 !
4905 ! POP A NUMBER OFF THE PUSHDOWN LIST
4910 !
4915 Fpop: ! CONVERTED SUBROUTINE FPOP
4920     IF Ptr<0 THEN L9001
4925     Popped=Pshlst(Ptr)
4930     Pshlst(Ptr)=0
4935     Poptim=Timlst(Ptr)
4940     Timlst(Ptr)=0
4945     Popfue=Fuelst(Ptr)
4950     Fuelst(Ptr)=0
4955     Popprb=Prblst(Ptr)
4960     Prblst(Ptr)=0
4965     Ptr=Ptr-1
4970     RETURN
4975 L9001: Popped=-1
4980     Ptr=0
4985     RETURN
4990 !
4995 ! FCC17
5000 !
5005 ! FINDS THE CARGO CARRYING PARAMETER FOR THE
5010 ! TRANSPORT MASTER TASK
5015 !
5020 Fcc17: ! CONVERTED SUBROUTINE FCC17
5025     Group1=Group
5030     IF Group>=90 THEN Group1=Group-80
5035 !
5040 ! GET ROW OF GROUP AND INSTANCE
5045 !
5050     FOR Irow=1 TO 100
5055     IF (Gpdat1(Irow,1)=Group1) AND (Gpdat1(Irow,2)=Inst) THEN L1712
5060     NEXT Irow
5065 L1712: IF (Node1<>1) AND (Node2<>2) THEN L1734
5070     Area=Gpdat2(Irow,9)
5075     Wght=Gpdat2(Irow,10)
5080     GOTO 2910
5085 L1734: IF (Node1<>3) AND (Node2<>4) THEN L1799
5090     Area=Gpdat2(Irow,5)
5095     Wght=Gpdat2(Irow,6)
5100 L1700: Cc0=0
5105     IF (Area<=Deck) AND (Wght<=Cargcp) THEN Cc0=1
5110 L1799: RETURN
5115 !
5120 ! FF
5125 ! FINDS Y VALUE ON A STRAIGHT LINE, GIVEN X VALUE AND TWO POINTS
5130 ! ON THE LINE (ASSUMING LINE EXTENDS INFINITELY)
5135 !
5140     DEF FNFF(X,X1,Y1,X2,Y2)
5145     IF ABS(X2-X1)<.0001 THEN L1
5150     Slope=(Y2-Y1)/(X2-X1)
5155     B=Y1-Slope*X1
5160     Ff=Slope*X+B
5165     RETURN Ff
5170 L1: Ff=(Y1+Y2)/2
5175     RETURN Ff
5180     FNEND
5185 ! FFS
5190 ! FINDS Y VALUE ON BROKEN LINE OF 5, GIVEN X VALUE
5195 ! AND THE 5 POINTS
5200 ! (ASSUMING ENDS OF LINE EXTEND INFINITELY)
5205     DEF FNFF5(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5)
5210     IF X<=X2 THEN FFS=FNFF(X,X1,Y1,X2,Y2)
5215     IF (X>X2) AND (X<=X3) THEN FFS=FNFF(X,X2,Y2,X3,Y3)

```

```

5220      IF (X>X3) AND (X<=X4) THEN Ff5=FNFF(X,X3,Y3,X4,Y4)
5225      IF X>X4 THEN Ff5=FNFF(X,X4,Y4,X5,Y5)
5230      RETURN Ff5
5235      FNEND
5240      ! FF3
5245      ! FINDS Y VALUE ON BROKEN LINE OF 3 POINTS, GIVEN X VALUE
5250      ! AND THE 3 POINTS
5255      ! (ASSUMING ENDS OF LINE EXTEND INFINITELY)
5260      DEF FNFF3(X,X1,Y1,X2,Y2,X3,Y3)
5265      IF X<=X2 THEN Ff3=FNFF(X,X1,Y1,X2,Y2)
5270      IF X>X2 THEN Ff3=FNFF(X,X2,Y2,X3,Y3)
5275      RETURN Ff3
5280      FNEND
5285      ! FF4
5290      ! FINDS Y VALUE ON BROKEN LINE OF 4 POINTS, GIVEN X VALUE
5295      ! AND THE 4 POINTS
5300      ! (ASSUMING ENDS OF LINE EXTEND INFINITELY)
5305      DEF FNFF4(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4)
5310      IF X<=X2 THEN Ff4=FNFF(X,X1,Y1,X2,Y2)
5315      IF (X>X2) AND (X<=X3) THEN Ff4=FNFF(X,X2,Y2,X3,Y3)
5320      IF X>X3 THEN Ff4=FNFF(X,X3,Y3,X4,Y4)
5325      RETURN Ff4
5330      ! FPACK
5335      ! PACKS THE TWO-DIGIT GROUP, INSTANCE, AND NODE INTO ONE
5340      ! SIX-DIGIT NUMBER OF THE FORM: 'GGIINN'.
5345      DEF FNFPack(Node,INTEGER Group,Inst)
5350      Fpack=10000*Group+100*Inst+Node
5355      RETURN Fpack
5360      FNEND
5365      ! FGROUPO
5370      DEF FNFGrouPO(Z)
5375      INTEGER Temp
5380      Temp=Z/10000
5385      Fgroup=Temp
5390      RETURN Fgroup
5395      FNEND
5400      ! FINST
5405      DEF FNFinst(Z)
5410      INTEGER Temp,Temp1,Temp2
5415      Temp1=Z/100
5420      Temp2=Z/10000
5425      Temp=Temp1-Temp2*100
5430      Finst=Temp
5435      RETURN Finst
5440      FNEND
5445      ! FXNODE
5450      DEF FNFXnode(Z)
5455      INTEGER Temp,Temp1
5460      Temp=Z/100
5465      Temp1=Z-Temp*100
5470      Fxnode=Temp1
5475      RETURN Fxnode
5480      FNEND
5485      ! FHOVND
5490      ! FIND HIGHEST (CLOSEST TO TOP) OVERALL NODE ON PUSHDOWN LIST.
5495      ! TOP OF PUSHDOWN LIST IS ASSUMED TO BE A GROUP NODE.
5500      DEF FNFHound(Dummy)
5505      OPTION BASE 1
5510      COM Cftnams,Ssaug,Speed(4),MfuInt(4),Towspd,Gpdat1(40,2),Gpdat2(40,19)
5515      ,Pshlst(100),INTEGER Idisp,Icspd,Ptr
5520      INTEGER Temp
5525      FOR I=1 TO 100
5530      IF Pshlst(Ptr-I)<10000 THEN L20
5535      L20: Temp=Pshlst(Ptr-I)
5540      Fhound=Temp

```

```

5545     RETURN Fhound
5550     FNEND
5555     ! FSKTIM
5560     DEF FNFsktim(Mastsk,Motion)
5565     IF (Mastsk=1) OR (Mastsk=2) OR (Mastsk=4) OR (Mastsk=6) OR (Mastsk=7)
        THEN L1
5570         Fsktim=1
5575     RETURN Fsktim
5580 L1:   Fsktim=FNFF3(Motion(4),0,1,.5,1,1,2)
5585     RETURN Fsktim
5590     FNEND
5595     ! FMNTIM
5600     DEF FNFmntim(Mastsk,Leng)
5605     IF (Mastsk=2) OR (Mastsk=4) OR (Mastsk=7) THEN L1
5610         Fmntim=1
5615     RETURN Fmntim
5620 L1:   Fmntim=FNFF4(Leng,0,1,100,1,400,3,1000,3)
5625     RETURN Fmntim
5630     FNEND
5635     ! FVZTIM
5640     DEF FNFvztim(Mastsk,Visdis,Visdtb)
5645     IF Mastsk=14 THEN L14
5650         Fvztim=1
5655     RETURN Fvztim
5660 L14:  Fvztim=Visdis(1,Visdtb)+Visdis(2,Visdtb)+2*Visdis(3,Visdtb)
5665     RETURN Fvztim
5670     FNEND
5675     ! FPOD
5680     ! CALCULATES PROBABILITY OF DETECTION USING CURVES FROM
5685     ! SAR MANUAL, FIGURE 8-65.
5690     DEF FNFpod(Cf,Nrsrch)
5695     IF Nrsrch=0 THEN Fpod=0
5700     IF Nrsrch=1 THEN Fpod=FNFF5(Cf,0,0,.6,.55,1.1,.83,1.5,.95,1.8,.97)
5705     IF Nrsrch=2 THEN Fpod=FNFF5(Cf,0,0,.39,.6,.7,.85,1,.96,1.3,.99)
5710     IF Nrsrch=3 THEN Fpod=FNFF5(Cf,0,0,.21,.5,.4,.76,.6,.9,.99,.99)
5715     IF Nrsrch=4 THEN Fpod=FNFF5(Cf,0,0,.21,.6,.4,.85,.6,.95,.8,1)
5720     IF Nrsrch=5 THEN Fpod=FNFF5(Cf,0,0,.22,.7,.4,.9,.5,.96,.75,1)
5725     IF Fpod>1 THEN Fpod=1
5730     RETURN Fpod
5735     FNEND
5740     ! FPDPTH
5745     ! DEPTH CUMULATIVE PROBABILITY DISTRIBUTION
5750     !
5755     ! FPDPTH(D)=PROBABILITY THAT DEPTH < D
5760     DEF FNFpdpth(Dphdtb,Depth)
5765     IF Dphdtb=1 THEN Fpdpth=0
5770     RETURN Fpdth
5775     FNEND
5780     ! FPRVTM
5785     ! FINDS TIME OF THE PREVIOUS (CLOSEST TO TOP) TASK ON PUSHDOWN LIST.
5790     DEF FNFprvtm(Dummy)
5795     FOR I=1 TO 100
5800     IF Ptr-I<=0 THEN L30
5805     IF Timlst(Ptr-I)>0 THEN L20
5810     NEXT I
5815 L20:  Fprvtm=Timlst(Ptr-I)
5820     RETURN Fprvtm
5825 L30:  Fprvtm=0
5830     RETURN Fprvtm
5835     FNEND
5840     ! FSPEDU
5845     ! CALCULATES PROBABILITY AND TIME FOR SEARCH
5850     ! FOR PEOPLE AND SEARCH FOR DIST. UNIT TASKS
5855     SUB Fspedu(Speed(+),Sw,A,Nmax,Cf,Tmax,Ps,Pf,Ts,Tf)
5860     OPTION BASE 1
5865     DIM Alpha(5)

```

```

5870      DATA 1.00,2.11,3.31,4.37,5.16
5875      MAT READ Alpha
5880      IF (Sw<=0) OR (A<=0) OR (Nmax<=0) OR (Cf<=0) OR (Tmax<=0) THEN Ls99
5885      GOTO Ls300
5890 Ls99:  Ps=0
5895      Pf=1
5900      Ts=0
5905      Tf=0
5910      SUBEXIT
5915 Ls300: Nfulls=Nmax
5920      Sfrac=0
5925      ! TEST IF NUMBER OF FULL SEARCHES TAKES TOO LONG
5930 Ls301: Tf=0
5935      FOR I=1 TO Nfulls
5940      Tnext=Cf*A*Alpha(I)/(Speed(3)*Sw)
5945      Tf=Tf+Tnext
5950      IF Tf>Tmax THEN Ls32
5955      NEXT I
5960      GOTO Ls33
5965      ! IF EXCEEDS TMAX
5970 Ls32:  Nfulls=I-1
5975      Tfulls=Tf-Tnext
5980      Tlast=Tmax-Tfulls
5985      Sfrac=Tlast/Tnext
5990      Tf=Tmax
5995      ! FINDS PS AND PF
6000 Ls33:  Podn=FNFPod(Cf,Nfulls)
6005      Podn1=FNFPod(Cf,Nfulls+1)
6010      Podl=Sfrac*(Podn1-Podn)
6015      Ps=Podn+Podl
6020      Pf=1-Ps
6025      ! CALCULATE TIME OF SUCCESSFUL SEARCH
6030      Xsum=0
6035      Tmbefj=0
6040      FOR I=1 TO Nfulls
6045      Podj=FNFPod(Cf,I)-FNFPod(Cf,I-1)
6050      IF I<=1 THEN Ls92
6055      Tmbefj=Tmbefj+Timj
6060 Ls92:  Timj=Cf*A*Alpha(I)/(Speed(3)*Sw)
6065      Avtmj=Timj/2
6070      Tottmj=Avtmj+Tmbefj
6075      Xsum=Xsum+Tottmj*Podj
6080      NEXT I
6085      IF Sfrac=0 THEN Ls95
6090      Tottml=Tfulls+Tlast/2
6095      Xsum=Xsum+Tottml*Podl
6100 Ls95:  Ts=Xsum/Ps
6105      SUBEND
6110      ! FLKDAT
6115      ! LINK DATA
6120      ! FOR A GROUP LINK: RETURNS PROBABILITY, TIME, FUEL USED.
6125      SUB Flkdat(INTEGER Group,Inst,REAL Node1,Node2,Lkprob,Lktime,Lkfuel)
6130      OPTION BASE 1
6135      SHORT Gp1pb(7,7),Gp2pb(3,3),Gp3pb(4,4),Gp4pb(3,3),Gp5pb(6,5)
6140      SHORT Gp6pb(5,5),Gp7pb(3,3),Gp8pb(3,3),Gp9pb(4,4),Gp10pb(4,4)
6145      SHORT Gp11pb(2,2),Gp12pb(3,3),Gp13pb(2,2),Gp14pb(2,2),Gp15pb(4,4)
6150      SHORT Gp16pb(3,3),Gp17pb(4,4),Gp18pb(6,6),Gp90pb(9,9),Gp93pb(3,9)
6155      COM Cf,thms,Ssaug,Speed(4),Mfulint(4),Towspd,Gpdat1(40,2),Gpdat2(40,1)
6160      ! GET ROW OF GROUP AND INSTANCE
6165      Group1=Group
6170      IF Group=90 THEN Group1=10
6175      IF Group=93 THEN Group1=13
6180      FOR Irow=1 TO 100
6185      IF (Gpdat1(Irow,1)=Group1) AND (Gpdat1(Irow,2)=Inst) THEN L200
6190      NEXT Irow

```

```

6195      !
6200 L200:  Lktime=0
6205      Lkfuel=0
6210      IF Group=90 THEN L90
6215      IF Group=93 THEN L93
6220      ON Group GOTO L1,L2,L3,L4,L5,L6,L7,L8,L9,L10,L11,L12,L13,L14,L15,L16,
L17,L18
6225      !
6230      ! 1. ASSIST GROUP
6235      !
6240 L1:    DATA 0.,92.,91.,0.,93.,94.,0.,0.,0.,0.,0.,0.,0.,0.,0.,1.,0.,0.,
0.
6245      DATA 0.,1.,0.,0.,0.,0.,0.,0.,1.,0.,0.,0.,0.,0.,0.,0.,0.,1.
6250      DATA 0.,1.,0.,0.,0.,0.,0.
6255      RESTORE L1
6260      MAT READ Gpipb
6265      Gpipb(1,2)=Gpdat2(Irow,2)
6270      Gpipb(1,3)=Gpdat2(Irow,1)
6275      Gpipb(1,5)=Gpdat2(Irow,3)
6280      Gpipb(1,6)=Gpdat2(Irow,4)
6285      Lkprob=Gpipb(Node1,Node2)
6290      IF Lkprob=0 THEN SUBEXIT
6295 L112:  IF (Node1<>1) OR (Node2<>2) THEN L113
6300      T4=Gpdat2(Irow,8)
6305      Lktime=T4*FNFsktim(1,Motion)
6310      Lkfuel=Lktime*Mfulrt(4)
6315      SUBEXIT
6320 L113:  IF (Node1<>1) OR (Node2<>3) THEN L115
6325      T1=Gpdat2(Irow,5)
6330      Lktime=T1*FNFsktim(2,Motion)*FNFmntim(2,Leng)
6335      Lkfuel=Lktime*Mfulrt(4)
6340      SUBEXIT
6345 L115:  IF (Node1<>1) OR (Node2<>5) THEN L116
6350      T5=Gpdat2(Irow,9)
6355      Lktime=T5*FNFsktim(4,Motion)*FNFmntim(4,Leng)
6360      Lkfuel=Lktime*Mfulrt(4)
6365      SUBEXIT
6370 L116:  IF (Node1<>1) OR (Node2<>6) THEN L134
6375      T6=Gpdat2(Irow,10)
6380      Lktime=T6*FNFsktim(7,Motion)*FNFmntim(7,Leng)
6385      Lkfuel=Lktime*Mfulrt(4)
6390      SUBEXIT
6395 L134:  IF (Node1<>3) OR (Node2<>4) THEN L142
6400      T2=Gpdat2(Irow,6)
6405      Lktime=T2
6410      Lkfuel=Lktime*Mfulrt(4)
6415      SUBEXIT
6420 L142:  IF (Node1<>4) OR (Node2<>2) THEN L167
6425      T3=Gpdat2(Irow,7)
6430      Lktime=T3*FNFsktim(2,Motion)*FNFmntim(2,Leng)
6435      Lkfuel=Lktime*Mfulrt(4)
6440      SUBEXIT
6445 L167:  IF (Node1<>6) OR (Node2<>7) THEN L172
6450      T7=Gpdat2(Irow,11)
6455      Lktime=T7
6460      Lkfuel=Lktime*Mfulrt(4)
6465      SUBEXIT
6470 L172:  IF (Node1<>7) OR (Node2<>2) THEN L199
6475      T8=Gpdat2(Irow,12)
6480      Lktime=T8*FNFsktim(7,Motion)*FNFmntim(7,Leng)
6485      Lkfuel=Lktime*Mfulrt(4)
6490 L199:  SUBEXIT
6495      !
6500      ! 2. ESCORT GROUP
6505      !
6510 L2:    DATA 0.,91.,92.,0.,0.,0.,0.,1.,0.

```

```

6515      RESTORE L2
6520      MAT READ Gp2pb
6525      Gp2pb(1,2)=Gpdat2(Irow,1)
6530      Gp2pb(1,3)=Gpdat2(Irow,2)
6535      Lkprob=Gp2pb(Node1,Node2)
6540      IF Lkprob=0 THEN SUBEXIT
6545 L212: IF (Node1<>1) OR (Node2<>2) THEN L213
6550      D1=Gpdat2(Irow,3)
6555      V1=Gpdat2(Irow,4)
6560      Lktime=D1/V1
6565      Lkfuel=Lktime*Mfulrt(3)
6570      SUBEXIT
6575 L213: IF (Node1<>1) OR (Node2<>3) THEN L299
6580      D2=Gpdat2(Irow,5)
6585      Lktime=D2/Speed(2)
6590      Lkfuel=Lktime*Mfulrt(2)
6595 L299: SUBEXIT
6600      !
6605      ! 3. FIGHT FIRE GROUP
6610      !
6615 L3:   DATA 0.,92.,91.,0.,0.,0.,0.,0.,0.,0.,0.,1.,0.,1.,0.,0.
6620      RESTORE L3
6625      MAT READ Gp3pb
6630      Gp3pb(1,2)=Gpdat2(Irow,2)
6635      Gp3pb(1,3)=Gpdat2(Irow,1)
6640      Lkprob=Gp3pb(Node1,Node2)
6645      IF Lkprob=0 THEN SUBEXIT
6650      IF (Node1<>1) OR (Node2<>2) THEN L313
6655      T4=Gpdat2(Irow,6)
6660      Lktime=T4*FNFSktim(7,Motion)*FNFmntim(7,Leng)
6665      Lkfuel=Lktime*Mfulrt(4)
6670      SUBEXIT
6675 L313: IF (Node1<>1) OR (Node2<>3) THEN L334
6680      T1=Gpdat2(Irow,3)
6685      Lktime=T1*FNFSktim(2,Motion)*FNFmntim(2,Leng)
6690      Lkfuel=Lktime*Mfulrt(4)
6695      SUBEXIT
6700 L334: IF (Node1<>3) OR (Node2<>4) THEN L342
6705      T2=Gpdat2(Irow,4)
6710      Lktime=T2
6715      Lkfuel=Lktime*Mfulrt(4)
6720      SUBEXIT
6725 L342: IF (Node1<>4) OR (Node2<>2) THEN L399
6730      T3=Gpdat2(Irow,5)
6735      Lktime=T3*FNFSktim(2,Motion)*FNFmntim(2,Leng)
6740      Lkfuel=Lktime*Mfulrt(4)
6745 L399: SUBEXIT
6750      !
6755      ! 4. IDENTIFY GROUP
6760      !
6765 L4:   DATA 0.,91.,92.,0.,0.,0.,0.,0.,1.,0.
6770      RESTORE L4
6775      MAT READ Gp4pb
6780      Gp4pb(1,2)=Gpdat2(Irow,1)
6785      Gp4pb(1,3)=Gpdat2(Irow,2)
6790      Lkprob=Gp4pb(Node1,Node2)
6795      IF Lkprob=0 THEN SUBEXIT
6800 L412: IF (Node1<>1) OR (Node2<>2) THEN L413
6805      Tship1=Gpdat2(Irow,3)
6810      Dship1=Gpdat2(Irow,4)
6815      Nship1=Gpdat2(Irow,5)
6820      Idtime=Nship1*Tship1*FNFuztim(14,Visdis,Visdtb)
6825      Travel=(Nship1-1)*Dship1/Speed(2)
6830      Lktime=Idtime+Travel
6835      Lkfuel=Lktime*Mfulrt(2)
6840      SUBEXIT

```

```

6845 L413: IF (Node1<>1) OR (Node2<>3) THEN L499
6850 T2=Gpdat2(Irow,5)
6855 Lktime=T2*FNFSktim(14,Motion)*FNFWztim(14,Viedis,Viedto)
6860 Lkfuel=Lktime*Mfulrt(2)
6865 L499: SUBEXIT
6870 !
6875 ! 5. INSPECT GROUP
6880 !
6885 L5: DATA 0.,0.,91.,0.,92.,0.,0.,0.,0.,0.,0.,0.,0.,1.,0.,0.
6890 DATA 0.,1.,0.,0.,0.,0.,0.,0.,0.,0.,1.,0.,1.,0.,0.,0.
6895 RESTORE L5
6900 MAT READ Gp5pb
6905 Gp5pb(1,3)=Gpdat2(Irow,1)
6910 Gp5pb(1,5)=Gpdat2(Irow,2)
6915 Lkprob=Gp5pb(Node1,Node2)
6920 IF Lkprob=0 THEN SUBEXIT
6925 L513: IF (Node1<>1) OR (Node2<>3) THEN L515
6930 T1=Gpdat2(Irow,3)
6935 Lktime=T1*FNFSktim(7,Motion)*FNFMntim(7,Leng)
6940 Lkfuel=Lktime*Mfulrt(4)
6945 SUBEXIT
6950 L515: IF (Node1<>1) OR (Node2<>5) THEN L534
6955 T4=Gpdat2(Irow,6)
6960 Lktime=T4*FNFSktim(2,Motion)*FNFMntim(2,Leng)
6965 Lkfuel=Lktime*Mfulrt(4)
6970 SUBEXIT
6975 L534: IF (Node1<>3) OR (Node2<>4) THEN L542
6980 T2=Gpdat2(Irow,4)
6985 Lktime=T2
6990 Lkfuel=Lktime*Mfulrt(4)
6995 SUBEXIT
7000 L542: IF (Node1<>4) OR (Node2<>2) THEN L556
7005 T3=Gpdat2(Irow,5)
7010 Lktime=T3*FNFSktim(7,Motion)*FNFMntim(7,Leng)
7015 Lkfuel=Lktime*Mfulrt(4)
7020 SUBEXIT
7025 L556: IF (Node1<>5) OR (Node2<>6) THEN L562
7030 T5=Gpdat2(Irow,7)
7035 Lktime=T5
7040 Lkfuel=Lktime*Mfulrt(4)
7045 SUBEXIT
7050 L562: IF (Node1<>6) OR (Node2<>2) THEN L599
7055 T6=Gpdat2(Irow,8)
7060 Lktime=T6*FNFSktim(2,Motion)*FNFMntim(2,Leng)
7065 Lkfuel=Lktime*Mfulrt(4)
7070 L599: SUBEXIT
7075 !
7080 ! 6. MONITOR GROUP
7085 !
7090 L6: DATA 0.,91.,92.,93.,94.,0.,0.,0.,0.,0.,0.,1.,0.,0.,0.
7095 DATA 0.,1.,0.,0.,0.,0.,1.,0.,0.,0.
7100 RESTORE L6
7105 MAT READ Gp6pb
7110 Gp6pb(1,2)=Gpdat2(Irow,1)
7115 Gp6pb(1,3)=Gpdat2(Irow,2)
7120 Gp6pb(1,4)=Gpdat2(Irow,3)
7125 Gp6pb(1,5)=Gpdat2(Irow,4)
7130 Lkprob=Gp6pb(Node1,Node2)
7135 IF Lkprob=0 THEN SUBEXIT
7140 L612: IF (Node1<>1) OR (Node2<>2) THEN L613
7145 T1=Gpdat2(Irow,5)
7150 Lktime=T1
7155 Lkfuel=Lktime*Mfulrt(4)
7160 SUBEXIT
7165 L613: IF (Node1<>1) OR (Node2<>3) THEN L614
7170 T2=Gpdat2(Irow,6)

```

```

7175      Lktime=T2
7180      Lkfuel=Lktime*Mfulnt(4)
7185      SUBEXIT
7190 L614: IF (Node1<>1) OR (Node2<>4) THEN L615
7195      T3=Gpdat2(Irow,7)
7200      Lktime=T3
7205      Lkfuel=Lktime*Mfulnt(4)
7210      SUBEXIT
7215 L615: IF (Node1<>1) OR (Node2<>5) THEN L699
7220      T4=Gpdat2(Irow,8)
7225      Lktime=T4
7230      Lkfuel=Lktime*Mfulnt(4)
7235 L699: SUBEXIT
7240      !
7245      ! 7. PATROL GROUP
7250      !
7255 L7:   DATA 0.,91.,92.,0.,0.,0.,0.,1.,0.
7260      RESTORE L7
7265      MAT READ Gp7pb
7270      Gp7pb(1,2)=Gpdat2(Irow,1)
7275      Gp7pb(1,3)=Gpdat2(Irow,2)
7280      Lkprob=Gp7pb(Node1,Node2)
7285      IF Lkprob=0 THEN SUBEXIT
7290 L712: IF (Node1<>1) OR (Node2<>2) THEN L713
7295      D1=Gpdat2(Irow,3)
7300      V1=Gpdat2(Irow,4)
7305      Lktime=D1/V1
7310      Lkfuel=Lktime*Mfulnt(3)
7315      SUBEXIT
7320 L713: IF (Node1<>1) OR (Node2<>3) THEN L799
7325      D2=Gpdat2(Irow,5)
7330      Lktime=D2/Speed(2)
7335      Lkfuel=Lktime*Mfulnt(2)
7340 L799: SUBEXIT
7345      !
7350      ! 8. RESCUE GROUP
7355      !
7360 L8:   DATA 0.,91.,92.,0.,0.,0.,0.,1.,0.
7365      RESTORE L8
7370      MAT READ Gp8pb
7375      Gp8pb(1,2)=Gpdat2(Irow,1)
7380      Gp8pb(1,3)=Gpdat2(Irow,2)
7385      Lkprob=Gp8pb(Node1,Node2)
7390      IF Lkprob=0 THEN SUBEXIT
7395 L812: IF (Node1<>1) OR (Node2<>2) THEN L813
7400      T1=Gpdat2(Irow,3)
7405      Lktime=T1*FNFsktim(4,Motion)*FNFmntim(4,Leng)
7410      Lkfuel=Lktime*Mfulnt(4)
7415      SUBEXIT
7420 L813: IF (Node1<>1) OR (Node2<>3) THEN L899
7425      T2=Gpdat2(Irow,4)
7430      Lktime=T2*FNFsktim(4,Motion)*FNFmntim(4,Leng)
7435      Lkfuel=Lktime*Mfulnt(4)
7440 L899: SUBEXIT
7445      !
7450      ! 9. RESCUE RETURN GROUP
7455      !
7460 L9:   DATA 0.,91.,92.,93.,0.,0.,0.,0.,0.,1.,0.,0.,0.,1.,0.,0.
7465      RESTORE L9
7470      MAT READ Gp9pb
7475      Gp9pb(1,2)=Gpdat2(Irow,1)
7480      Gp9pb(1,3)=Gpdat2(Irow,2)
7485      Gp9pb(1,4)=Gpdat2(Irow,3)
7490      Lkprob=Gp9pb(Node1,Node2)
7495      IF Lkprob=0 THEN SUBEXIT
7500 L912: IF (Node1<>1) OR (Node2<>2) THEN L913

```

```

7505      D1=Gpdat2(Irow,4)
7510      Lktime=D1/Towspd
7515      Lkfue1=Lktime*Mfulnt(3)
7520      SUBEXIT
7525 L913: IF (Node1<>1) OR (Node2<>3) THEN L914
7530      D2=Gpdat2(Irow,5)
7535      V2=Gpdat2(Irow,6)
7540      Lktime=D2/V2
7545      Lkfue1=Lktime*Mfulnt(3)
7550      SUBEXIT
7555 L914: IF (Node1<>1) OR (Node2<>4) THEN L999
7560      D3=Gpdat2(Irow,7)
7565      Lktime=D3/Speed(2)
7570      Lkfue1=Lktime*Mfulnt(2)
7575 L999: SUBEXIT
7580      !
7585      ! 10. SAR SEARCH GROUP
7590      ! SUCCESS
7595      !
7600 L10:  DATA 0.,0.,91.,92.,0.,0.,0.,0.,0.,915.,0.,0.,0.,925.,0.,0.
7605      RESTORE L10
7610      MAT READ Gp10pb
7615      Gp10pb(1,3)=Gpdat2(Irow,1)
7620      Gp10pb(1,4)=Gpdat2(Irow,2)
7625      Sw1=Gpdat2(Irow,3)
7630      A1=Gpdat2(Irow,4)
7635      Nsear1=Gpdat2(Irow,5)
7640      Cf1=Gpdat2(Irow,6)
7645      Tmax1=Gpdat2(Irow,7)
7650      Sw2=Gpdat2(Irow,8)
7655      A2=Gpdat2(Irow,9)
7660      Nsear2=Gpdat2(Irow,10)
7665      Cf2=Gpdat2(Irow,11)
7670      Tmax2=Gpdat2(Irow,12)
7675      CALL Fspedu(Speed(*),Sw1,A1,Nsear1,Cf1,Tmax1,Ps1,Pf1,Ts1,Tf1)
7680      CALL Fspedu(Speed(*),Sw2,A2,Nsear2,Cf2,Tmax2,Ps2,Pf2,Ts2,Tf2)
7685      Gp10pb(3,2)=Ps1
7690      Gp10pb(4,2)=Ps2
7695      Lkprob=Gp10pb(Node1,Node2)
7700      IF Lkprob=0 THEN SUBEXIT
7705 L1032: IF (Node1<>3) OR (Node2<>2) THEN L1042
7710      Lktime=Ts1
7715      Lkfue1=Lktime*Mfulnt(3)
7720      SUBEXIT
7725 L1042: IF (Node1<>4) OR (Node2<>2) THEN L1099
7730      Lktime=Ts2
7735      Lkfue1=Lktime*Mfulnt(3)
7740 L1099: SUBEXIT
7745      !
7750      ! 11. SEARCH FLEET GROUP
7755      !
7760 L11:  DATA 0.,1.,0.,0.
7765      RESTORE L11
7770      MAT READ Gp11pb
7775      Lkprob=Gp11pb(Node1,Node2)
7780      IF Lkprob=0 THEN SUBEXIT
7785 L1112: IF (Node1<>1) OR (Node2<>2) THEN L1199
7790      D1=Gpdat2(Irow,1)
7795      Lktime=D1/Speed(2)
7800      Lkfue1=Lktime*Mfulnt(2)
7805 L1199: SUBEXIT
7810      !
7815      ! 12. SEIZE GROUP
7820      !
7825 L12:  DATA 0.,0.,1.,0.,0.,0.,0.,1.,0.
7830      RESTORE L12

```

```

7835      MAT READ Gp12pb
7840      Lkprob=Gp12pb(Node1,Node2)
7845      IF Lkprob=0 THEN SUBEXIT
7850 L1213: IF (Node1<>1) OR (Node2<>3) THEN L1232
7855      T1=Gpdat2(Irow,1)
7860      Lktime=T1
7865      Lkfuel=Lktime*Mfulrt(4)
7870      SUBEXIT
7875 L1232: IF (Node1<>3) OR (Node2<>2) THEN L1299
7880      D2=Gpdat2(Irow,2)
7885      Lktime=D2/Speed(2)
7890      Lkfuel=Lktime*Mfulrt(2)
7895 L1299: SUBEXIT
7900      !
7905      ! 13. SENSOR SEARCH GROUP
7910      ! THIS GROUP MUST ALWAYS FOLLOW A STEAM GROUP.
7915      ! SUCCESS
7920      !
7925 L13:  DATA 0.,95.,0.,0.
7930      RESTORE L13
7935      MAT READ Gp13pb
7940      Su=Gpdat2(Irow,1)
7945      E=Gpdat2(Irow,2)
7950      Vtar=Gpdat2(Irow,3)
7955      Tmax=Gpdat2(Irow,4)
7960      Tbef=FNFprvta(0)
7965      GOSUB Fsshp
7970      Gp13pb(1,2)=Ps
7975      Lkprob=Gp13pb(Node1,Node2)
7980      IF Lkprob=0 THEN SUBEXIT
7985      Lktime=Ts
7990      Lkfuel=Lktime*Mfulrt(2)
7995 L1399: SUBEXIT
8000      !
8005      ! 14. STANDBY GROUP
8010      !
8015 L14:  DATA 0.,1.,0.,0.
8020      RESTORE L14
8025      MAT READ Gp14pb
8030      Lkprob=Gp14pb(Node1,Node2)
8035      IF Lkprob=0 THEN SUBEXIT
8040 L1412: IF (Node1<>1) OR (Node2<>2) THEN L1499
8045      T1=Gpdat2(Irow,1)
8050      Lktime=T1
8055      Lkfuel=Lktime*Mfulrt(4)
8060 L1499: SUBEXIT
8065      !
8070      ! 15. STEAM GROUP
8075      !
8080 L15:  DATA 0.,91.,92.,93.,0.,0.,0.,0.,0.,1.,0.,0.,0.,1.,0.,0.
8085      RESTORE L15
8090      MAT READ Gp15pb
8095      Gp15pb(1,2)=Gpdat2(Irow,1)
8100      Gp15pb(1,3)=Gpdat2(Irow,2)
8105      Gp15pb(1,4)=Gpdat2(Irow,3)
8110      Lkprob=Gp15pb(Node1,Node2)
8115      IF Lkprob=0 THEN SUBEXIT
8120 L1512: IF (Node1<>1) OR (Node2<>2) THEN L1513
8125      D1=Gpdat2(Irow,4)
8130      Lktime=D1/Speed(2)
8135      Lkfuel=Lktime*Mfulrt(2)
8140      SUBEXIT
8145 L1513: IF (Node1<>1) OR (Node2<>3) THEN L1514
8150      D2=Gpdat2(Irow,5)
8155      Lktime=D2/Speed(1)
8160      Lkfuel=Lktime*Mfulrt(1)

```

```

8165      SUBEXIT
8170 L1514: IF (Node1<>1) OR (Node2<>4) THEN L1599
8175      D3=Gpdat2(Irow,5)
8180      Lktime=D3/Speed(1)
8185      Lkfuel=Lktime*Mfulrt(1)
8190 L1599: SUBEXIT
8195      !
8200      ! 16. TRANSFER EQUIPMENT GROUP
8205      !
8210 L16:   DATA 0.,91.,92.,0.,0.,0.,0.,1.,0.
8215      RESTORE L16
8220      MAT READ Gp16pb
8225      Gp16pb(1,2)=Gpdat2(Irow,1)
8230      Gp16pb(1,3)=Gpdat2(Irow,2)
8235      Lkprob=Gp16pb(Node1,Node2)
8240      IF Lkprob=0 THEN SUBEXIT
8245 L1612: IF (Node1<>1) OR (Node2<>2) THEN L1613
8250      T1=Gpdat2(Irow,3)
8255      Lktime=T1*FNFsktim(7,Motion)*FNFmntim(7,Leng)
8260      Lkfuel=Lktime*Mfulrt(4)
8265      SUBEXIT
8270 L1613: IF (Node1<>1) OR (Node2<>3) THEN L1699
8275      T2=Gpdat2(Irow,4)
8280      Lktime=T2*FNFsktim(7,Motion)*FNFmntim(7,Leng)
8285      Lkfuel=Lktime*Mfulrt(4)
8290 L1699: SUBEXIT
8295      !
8300      ! 17. TRANSPORT EQUIPMENT GROUP
8305      !
8310 L17:   DATA 0.,92.,91.,0.,0.,0.,0.,0.,0.,0.,0.,1.,0.,1.,0.,0.
8315      RESTORE L17
8320      MAT READ Gp17pb
8325      Gp17pb(1,2)=Gpdat2(Irow,2)
8330      Gp17pb(1,3)=Gpdat2(Irow,1)
8335      Lkprob=Gp17pb(Node1,Node2)
8340      IF Lkprob=0 THEN SUBEXIT
8345 L1712: IF (Node1<>1) OR (Node2<>2) THEN L1713
8350      D4=Gpdat2(Irow,8)
8355      A4=Gpdat2(Irow,9)
8360      W4=Gpdat2(Irow,10)
8365      Lktime=D4/Speed(2)
8370      Lkfuel=Lktime*Mfulrt(2)
8375      SUBEXIT
8380 L1713: IF (Node1<>1) OR (Node2<>3) THEN L1734
8385      T1=Gpdat2(Irow,3)
8390      Lktime=T1*FNFsktim(7,Motion)*FNFmntim(7,Leng)
8395      Lkfuel=Lktime*Mfulrt(4)
8400      SUBEXIT
8405 L1734: IF (Node1<>3) OR (Node2<>4) THEN L1742
8410      D2=Gpdat2(Irow,4)
8415      A2=Gpdat2(Irow,5)
8420      W2=Gpdat2(Irow,6)
8425      Lktime=D2/Speed(2)
8430      Lkfuel=Lktime*Mfulrt(2)
8435      SUBEXIT
8440 L1742: IF (Node1<>4) OR (Node2<>2) THEN L1799
8445      T3=Gpdat2(Irow,7)
8450      Lktime=T3*FNFsktim(7,Motion)*FNFmntim(7,Leng)
8455      Lkfuel=Lktime*Mfulrt(4)
8460 L1799: SUBEXIT
8465      !
8470      ! 18. WORK EQUIPMENT GROUP
8475      !
8480 L18:   DATA 0.,92.,91.,0.,93.,94.,0.,0.,0.,0.,0.,0.,0.,1.,0.,0.
8485      DATA 0.,1.,0.,0.,0.,0.,0.,1.,0.,0.,0.,0.,1.,0.,0.,0.
8490      RESTORE L18

```



```

8825      Lktime=Tf1
8830      Lkfuel=Lktime*MfuInt(3)
8835      SUBEXIT
8840 L9049: IF (Node1<>4) OR (Node2<>9) THEN L9099
8845      Lktime=Tf2
8850      Lkfuel=Lktime*MfuInt(3)
8855 L9099: SUBEXIT
8860      !
8865      ! 93. SENSOR SEARCH GROUP
8870      ! THIS GROUP MUST ALWAYS FOLLOW A STEAM GROUP
8875      ! FAILURE
8880      !
8885 L93:  MAT Gp93pb=ZER
8890      Gp93pb(1,9)=97
8895      Su=Gpdat2(Irow,1)
8900      E=Gpdat2(Irow,2)
8905      Vtar=Gpdat2(Irow,3)
8910      Tmax=Gpdat2(Irow,4)
8915      Tbef=FNFPprvm(0)
8920      GOSUB Fsshp
8925      Gp93pb(1,9)=Pf
8930      Lkprob=Gp93pb(Node1,Node2)
8935      IF Lkprob=0 THEN SUBEXIT
8940      Lktime=Tf
8945      Lkfuel=Lktime*MfuInt(2)
8950 L9399: SUBEND
8955      !
8960      ! FTASK
8965      !
8970      SUB Ftask(INTEGER Group,REAL Node1,Node2,INTEGER Taskno,Rate)
8975      OPTION BASE 1
8980      DIM Gp1tk(7,7),Gp2tk(3,3),Gp3tk(4,4),Gp4tk(3,3)
8985      DIM Gp5tk(6,6),Gp6tk(5,5),Gp7tk(3,3),Gp8tk(3,3)
8990      DIM Gp9tk(4,4),Gp10tk(4,4),Gp11tk(2,2),Gp12tk(3,3)
8995      DIM Gp13tk(2,2),Gp14tk(2,2),Gp15tk(4,4),Gp16tk(3,3)
9000      DIM Gp17tk(4,4),Gp18tk(6,6),Gp90tk(9,9),Gp93tk(9,9)
9005      IF Group=90 THEN Lt90
9010      IF Group=93 THEN Lt93
9015      ON Group GOTO Lt1,Lt2,Lt3,Lt4,Lt5,Lt6,Lt7,Lt8,Lt9,Lt10,Lt11,Lt12,Lt1
3,Lt14,Lt15,Lt16,Lt17,Lt18
9020      !
9025      ! 1. ASSIST GROUP
9030      !
9035 Lt1:  DATA 0,404,401,0,414,408,0,0,0,0,0,0,0,0,0,0,411,0,0,0
9040      DATA 0,413,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9045      DATA 0,0,0,0,0,0,411,0,416,0,0,0,0,0,0,0,0,0,0
9050      RESTORE Lt1
9055      MAT READ Gp1tk
9060      Rate=Gp1tk(Node1,Node2)/100
9065      Taskno=Gp1tk(Node1,Node2)-Rate*100
9070      SUBEXIT
9075      !
9080      ! 2. ESCORT GROUP
9085      !
9090 Lt2:  DATA 0,302,201,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9095      RESTORE Lt2
9100      MAT READ Gp2tk
9105      Rate=Gp2tk(Node1,Node2)/100
9110      Taskno=Gp2tk(Node1,Node2)-Rate*100
9115      SUBEXIT
9120      !
9125      ! 3. FIGHT FIRE GROUP
9130      !
9135 Lt3:  DATA 0,402,401,0,0,0,0,0,0,0,0,0,0,403,0,413,0,0,0
9140      RESTORE Lt3
9145      MAT READ Gp3tk

```

```

9150      Rate=Gp3tk(Node1,Node2)/100
9155      Taskno=Gp3tk(Node1,Node2)-Rate*100
9160      SUBEXIT
9165      !
9170      ! 4. IDENTIFY GROUP
9175      !
9180 Lt4:   DATA 0,203,202,0,0,0,0,0
9185      RESTORE Lt4
9190      MAT READ Gp4tk
9195      Rate=Gp4tk(Node1,Node2)/100
9200      Taskno=Gp4tk(Node1,Node2)-Rate*100
9205      SUBEXIT
9210      !
9215      ! 5. INSPECT GROUP
9220      !
9225 Lt5:   DATA 0,0,408,0,401,0,0,0,0,0,0,0,0,405,0,0
9230      DATA 0,416,0,0,0,0,0,0,0,0,405,0,413,0,0,0,0
9235      RESTORE Lt5
9240      MAT READ Gp5tk
9245      Rate=Gp5tk(Node1,Node2)/100
9250      Taskno=Gp5tk(Node1,Node2)-Rate*100
9255      SUBEXIT
9260      !
9265      ! 6. MONITOR GROUP
9270      !
9275 Lt6:   DATA 0,409,410,412,417,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9280      RESTORE Lt6
9285      MAT READ Gp6tk
9290      Rate=Gp6tk(Node1,Node2)/100
9295      Taskno=Gp6tk(Node1,Node2)-Rate*100
9300      SUBEXIT
9305      !
9310      ! 7. PATROL GROUP
9315      !
9320 Lt7:   DATA 0,304,204,0,0,0,0,0
9325      RESTORE Lt7
9330      MAT READ Gp7tk
9335      Rate=Gp7tk(Node1,Node2)/100
9340      Taskno=Gp7tk(Node1,Node2)-Rate*100
9345      SUBEXIT
9350      !
9355      ! 8. RESCUE GROUP
9360      !
9365 Lt8:   DATA 0,415,414,0,0,0,0,0
9370      RESTORE Lt8
9375      MAT READ Gp8tk
9380      Rate=Gp8tk(Node1,Node2)/100
9385      Taskno=Gp8tk(Node1,Node2)-Rate*100
9390      SUBEXIT
9395      !
9400      ! 9. RESCUE RETURN GROUP
9405      !
9410 Lt9:   DATA 0,305,302,208,0,0,0,0,0,0,0,0,0,0,0,0
9415      RESTORE Lt9
9420      MAT READ Gp9tk
9425      Rate=Gp9tk(Node1,Node2)/100
9430      Taskno=Gp9tk(Node1,Node2)-Rate*100
9435      SUBEXIT
9440      !
9445      ! 10. SAR SEARCH GROUP
9450      ! SUCCESS
9455 Lt10:  DATA 0,0,0,0,0,0,0,0,0,0,303,0,0,0,301,0,0
9460      RESTORE Lt10
9465      MAT READ Gp10tk
9470      Rate=Gp10tk(Node1,Node2)/100
9475      Taskno=Gp10tk(Node1,Node2)-Rate*100

```



```

9810 !
9815 Lt13: DATA 0,422,408,0,419,423,0,0,0,0,0,0,0,0,421,0,0
9820 DATA 0,416,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
9825 RESTORE Lt13
9830 Rate=Gp18tk(Node1,Node2)/100
9835 Taskno=Gp13tk(Node1,Node2)-Rate*100
9840 SUBEXIT
9845 !
9850 ! 90. SAR SEARCH GROUP
9855 ! FAILURE
9860 !
9865 Lt90: MAT Gp90tk=ZER
9870 Gp90tk(3,9)=303
9875 Gp90tk(4,9)=304
9880 Rate=Gp90tk(Node1,Node2)/100
9885 Taskno=Gp90tk(Node1,Node2)-Rate*100
9890 SUBEXIT
9895 !
9900 ! 93. SENSOR SEARCH GROUP
9905 ! FAILURE
9910 !
9915 Lt93: MAT Gp93tk=ZER
9920 Gp93tk(9,1)=206
9925 Rate=Gp93tk(Node1,Node2)/100
9930 Taskno=Gp93tk(Node1,Node2)-Rate*100
9935 SUBEND
9940 !
9945 ! FMINPH
9950 !
9955 ! FINDS THE MINIMUM TIME AND FUEL PATHS FROM ANY OVERALL NODE TO
9960 ! THE END OF THE SCENARIO
9965 !
9970 SUB Fminph(N,SHORT Oucnmx(*),Gpplx(*),Mintim(*),Minfue(*))
9975 OPTION BASE 1
9980 SHORT Gpmt(50,50),Gpmnf(50,50),Flag(50),Flagnw(50)
9985 INTEGER Group,Inst,Flagnw,Again,I,J,K
9990 !
9995 ! INITIALIZATION
10000 !
10005 FOR I=1 TO N
10010 Mintim(I)=99999
10015 Minfue(I)=99999
10020 Flag(I)=0
10025 Flagnw(I)=0
10030 FOR J=1 TO N
10035 Gpmt(I,J)=0
10040 Gpmnf(I,J)=0
10045 NEXT J
10050 NEXT I
10055 Again=0
10060 !
10065 ! FIND THE MINIMUM GROUP TIME PATH AND FUEL PATH FOR ALL LINKS
10070 !
10075 FOR I=1 TO N
10080 FOR J=1 TO N
10085 K=Gpplx(I,J)
10090 Group=K/100
10095 Inst=K-Group*100
10100 IF Group=0 THEN Lm20
10105 CALL Fgpmin(T,F,Group,Inst)
10110 Gpmt(I,J)=T
10115 Gpmnf(I,J)=F
10120 Lm20: NEXT J
10125 NEXT I
10130 !
10135 ! FINDS THE MINIMUM PATH WITH RESPECT TO TIME

```

```

10140 !
10145 Mintim(2)=0
10150 Flag(2)=1
10155 Lm30: FOR J=1 TO N
10160 IF Flag(J)=0 THEN Lm50
10165 FOR I=1 TO N
10170 IF Overnmx(I,J)=0 THEN Lm40
10175 Tpath=Gpmnt(I,J)+Mintim(J)
10180 IF Tpath>Mintim(I) THEN Lm40
10185 Mintim(I)=Tpath
10190 Flagnu(I)=1
10195 Again=1
10200 Lm40: NEXT I
10205 Lm50: NEXT J
10210 FOR I=1 TO N
10215 Flag(I)=Flagnu(I)
10220 Flagnu(I)=0
10225 Lm60: NEXT I
10230 IF Again=0 THEN Lm100
10235 Again=0
10240 GOTO Lm30
10245 !
10250 ! FINDS THE MINIMUM PATH WITH RESPECT TO FUEL
10255 !
10260 Lm100: Minfue(2)=0
10265 Flag(2)=1
10270 Lm110: FOR J=1 TO N
10275 IF Flag(J)=0 THEN Lm130
10280 FOR I=1 TO N
10285 IF Overnmx(I,J)=0 THEN Lm120
10290 Fpath=Gpmnf(I,J)+Minfue(J)
10295 IF Fpath>Minfue(I) THEN Lm120
10300 Minfue(I)=Fpath
10305 Flagnu(I)=1
10310 Again=1
10315 Lm120: NEXT I
10320 Lm130: NEXT J
10325 IF Again=0 THEN Lm200
10330 Again=0
10335 FOR I=1 TO N
10340 Flag(I)=Flagnu(I)
10345 Flagnu(I)=0
10350 NEXT I
10355 GOTO Lm110
10360 Lm200: SUBEXIT
10365 !
10370 ! FGPMIN
10375 !
10380 ! COMPUTES THE SHORTEST PATH THROUGH A GROUP
10385 ! FOR TIME AND FUEL CONSUMPTION
10390 !
10395 SUB Fgpmn(Mintim,Minfue,INTEGER Group,Inst)
10400 Mintim=99999
10405 Minfue=99999
10410 IF Group=90 THEN Lg90
10415 IF Group=93 THEN Lg93
10420 ON Group GOTO Lg1,Lg2,Lg3,Lg4,Lg5,Lg6,Lg7,Lg8,Lg9,Lg10,Lg11,Lg12,Lg1
3,Lg14,Lg15,Lg16,Lg17,Lg18,Lg90,Lg93
10425 !
10430 ! 1. ASSIST GROUP
10435 !
10440 Lg1: CALL Flkdat(Group,Inst,1,2,P12,T12,F12)
10445 CALL Flkdat(Group,Inst,1,3,P13,T13,F13)
10450 CALL Flkdat(Group,Inst,1,5,P15,T15,F15)
10455 CALL Flkdat(Group,Inst,1,6,P16,T16,F16)
10460 CALL Flkdat(Group,Inst,3,4,P34,T34,F34)

```

```

10465      CALL Flkdat(Group,Inst,4,2,P42,T42,F42)
10470      CALL Flkdat(Group,Inst,6,7,P67,T67,F67)
10475      CALL Flkdat(Group,Inst,7,2,P72,T72,F72)
10480      T1342=T13+T34+T42
10485      T1672=T16+T67+T72
10490      IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
10495      IF (P13>0) AND (T1342<Mintim) THEN Mintim=T1342
10500      IF (P15>0) AND (T15<Mintim) THEN Mintim=T15
10505      IF (P16>0) AND (T1672<Mintim) THEN Mintim=T1672
10510      F1342=F13+F34+F42
10515      F1672=F16+F67+F72
10520      IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
10525      IF (P13>0) AND (F1342<Minfue) THEN Minfue=F1342
10530      IF (P15>0) AND (F15<Minfue) THEN Minfue=F15
10535      IF (P16>0) AND (F1672<Minfue) THEN Minfue=F1672
10540      SUBEXIT
10545      !
10550      ! 2. ESCORT GROUP
10555      !
10560      Lg2: CALL Flkdat(Group,Inst,1,2,P12,T12,F12)
10565      CALL Flkdat(Group,Inst,1,3,P13,T13,F13)
10570      IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
10575      IF (P13>0) AND (T13<Mintim) THEN Mintim=T13
10580      IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
10585      IF (P13>0) AND (F13<Minfue) THEN Minfue=F13
10590      SUBEXIT
10595      !
10600      ! 3. FIGHT FIRE GROUP
10605      !
10610      Lg3: CALL Flkdat(Group,Inst,1,2,P12,T12,F12)
10615      CALL Flkdat(Group,Inst,1,3,P13,T13,F13)
10620      CALL Flkdat(Group,Inst,3,4,P34,T34,F34)
10625      CALL Flkdat(Group,Inst,4,2,P42,T42,F42)
10630      T1342=T13+T34+T42
10635      IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
10640      IF (P13>0) AND (T1342<Mintim) THEN Mintim=T1342
10645      F1342=F13+F34+F42
10650      IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
10655      IF (P13>0) AND (F1342<Minfue) THEN Minfue=F1342
10660      SUBEXIT
10665      !
10670      ! 4. IDENTIFY GROUP
10675      !
10680      Lg4: CALL Flkdat(Group,Inst,1,2,P12,T12,F12)
10685      CALL Flkdat(Group,Inst,1,3,P13,T13,F13)
10690      IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
10695      IF (P13>0) AND (T13<Mintim) THEN Mintim=T13
10700      IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
10705      IF (P13>0) AND (F13<Minfue) THEN Minfue=F13
10710      SUBEXIT
10715      !
10720      ! 5. INSPECT GROUP
10725      !
10730      Lg5: CALL Flkdat(Group,Inst,1,3,P13,T13,F13)
10735      CALL Flkdat(Group,Inst,1,5,P15,T15,F15)
10740      CALL Flkdat(Group,Inst,3,4,P34,T34,F34)
10745      CALL Flkdat(Group,Inst,4,2,P42,T42,F42)
10750      CALL Flkdat(Group,Inst,5,6,P56,T56,F56)
10755      CALL Flkdat(Group,Inst,6,2,P62,T62,F62)
10760      T1342=T13+T34+T42
10765      T1562=T15+T56+T62
10770      IF (P13>0) AND (T1342<Mintim) THEN Mintim=T1342
10775      IF (P15>0) AND (T1562<Mintim) THEN Mintim=T1562
10780      F1342=F13+F34+F42
10785      F1562=F15+F56+F62
10790      IF (P13>0) AND (F1342<Minfue) THEN Minfue=F1342

```

```

10795      IF (P15>0) AND (F1562<Minfue) THEN Minfue=F1562
10800      SUBEXIT
10805      !
10810      ! 6. MONITOR GROUP
10815      !
10820      Lg6: CALL Fikdat(Group,Inst,1,2,P12,T12,F12)
10825      CALL Fikdat(Group,Inst,1,3,P13,T13,F13)
10830      CALL Fikdat(Group,Inst,1,4,P14,T14,F14)
10835      CALL Fikdat(Group,Inst,1,5,P15,T15,F15)
10840      IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
10845      IF (P13>0) AND (T13<Mintim) THEN Mintim=T13
10850      IF (P14>0) AND (T14<Mintim) THEN Mintim=T14
10855      IF (P15>0) AND (T15<Mintim) THEN Mintim=T15
10860      IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
10865      IF (P13>0) AND (F13<Minfue) THEN Minfue=F13
10870      IF (P14>0) AND (F14<Minfue) THEN Minfue=F14
10875      IF (P15>0) AND (F15<Minfue) THEN Minfue=F15
10880      SUBEXIT
10885      !
10890      ! 7. PATROL GROUP
10895      !
10900      Lg7: CALL Fikdat(Group,Inst,1,2,P12,T12,F12)
10905      CALL Fikdat(Group,Inst,1,3,P13,T13,F13)
10910      IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
10915      IF (P13>0) AND (T13<Mintim) THEN Mintim=T13
10920      IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
10925      IF (P13>0) AND (F13<Minfue) THEN Minfue=F13
10930      SUBEXIT
10935      !
10940      ! 8. RESCUE GROUP
10945      !
10950      Lg8: CALL Fikdat(Group,Inst,1,2,P12,T12,F12)
10955      CALL Fikdat(Group,Inst,1,3,P13,T13,F13)
10960      IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
10965      IF (P13>0) AND (T13<Mintim) THEN Mintim=T13
10970      IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
10975      IF (P13>0) AND (F13<Minfue) THEN Minfue=F13
10980      SUBEXIT
10985      !
10990      ! 9. RESCUE RETURN GROUP
10995      !
11000      Lg9: CALL Fikdat(Group,Inst,1,2,P12,T12,F12)
11005      CALL Fikdat(Group,Inst,1,3,P13,T13,F13)
11010      CALL Fikdat(Group,Inst,1,4,P14,T14,F14)
11015      IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
11020      IF (P13>0) AND (T13<Mintim) THEN Mintim=T13
11025      IF (P14>0) AND (T14<Mintim) THEN Mintim=T14
11030      IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
11035      IF (P13>0) AND (F13<Minfue) THEN Minfue=F13
11040      IF (P14>0) AND (F14<Minfue) THEN Minfue=F14
11045      SUBEXIT
11050      !
11055      ! 10. SAR SEARCH GROUP
11060      ! SUCCESS
11065      !
11070      Lg10: CALL Fikdat(Group,Inst,3,2,P32,T32,F32)
11075      CALL Fikdat(Group,Inst,4,2,P42,T42,F42)
11080      IF (P32>0) AND (T32<Mintim) THEN Mintim=T32
11085      IF (P42>0) AND (T42<Mintim) THEN Mintim=T42
11090      IF (P32>0) AND (F32<Minfue) THEN Minfue=F32
11095      IF (P42>0) AND (F42<Minfue) THEN Minfue=F42
11100      SUBEXIT
11105      !
11110      ! 11. SEARCH FLEET GROUP
11115      !
11120      Lg11: CALL Fikdat(Group,Inst,1,2,P12,T12,F12)

```

```

11125         IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
11130         IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
11135         SUBEXIT
11140         !
11145         ! 12. SEIZE GROUP
11150         !
11155         Lg12: CALL Flkdat(Group,Inst,1,3,P13,T13,F13)
11160         CALL Flkdat(Group,Inst,3,2,P32,T32,F32)
11165         T132=T13+T32
11170         P132=P13+P32
11175         IF (P132>0) AND (T132<Mintim) THEN Mintim=T132
11180         F132=F13+F32
11185         P132=P13+P32
11190         IF (P132>0) AND (F132<Minfue) THEN Minfue=F132
11195         SUBEXIT
11200         !
11205         ! 13. SENSOR SEARCH GROUP
11210         ! SUCCESS
11215         !
11220         Lg13: CALL Flkdat(Group,Inst,1,2,P12,T12,F12)
11225         IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
11230         IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
11235         SUBEXIT
11240         !
11245         ! 14. STANDBY GROUP
11250         !
11255         Lg14: CALL Flkdat(Group,Inst,1,2,P12,T12,F12)
11260         IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
11265         IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
11270         SUBEXIT
11275         !
11280         ! 15. STEAM GROUP
11285         !
11290         Lg15: CALL Flkdat(Group,Inst,1,2,P12,T12,F12)
11295         CALL Flkdat(Group,Inst,1,3,P13,T13,F13)
11300         CALL Flkdat(Group,Inst,1,4,P14,T14,F14)
11305         IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
11310         IF (P13>0) AND (T13<Mintim) THEN Mintim=T13
11315         IF (P14>0) AND (T14<Mintim) THEN Mintim=T14
11320         IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
11325         IF (P13>0) AND (F13<Minfue) THEN Minfue=F13
11330         IF (P14>0) AND (F14<Minfue) THEN Minfue=F14
11335         SUBEXIT
11340         !
11345         ! 16. TRANSFER EQUIPMENT GROUP
11350         !
11355         Lg16: CALL Flkdat(Group,Inst,1,2,P12,T12,F12)
11360         CALL Flkdat(Group,Inst,1,3,P13,T13,F13)
11365         IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
11370         IF (P13>0) AND (T13<Mintim) THEN Mintim=T13
11375         IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
11380         IF (P13>0) AND (F13<Minfue) THEN Minfue=F13
11385         SUBEXIT
11390         !
11395         ! 17. TRANSPORT EQUIPMENT GROUP
11400         !
11405         Lg17: CALL Flkdat(Group,Inst,1,2,P12,T12,F12)
11410         CALL Flkdat(Group,Inst,1,3,P13,T13,F13)
11415         CALL Flkdat(Group,Inst,3,4,P34,T34,F34)
11420         CALL Flkdat(Group,Inst,4,2,P42,T42,F42)
11425         T1342=T13+T34+T42
11430         IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
11435         IF (P13>0) AND (T1342<Mintim) THEN Mintim=T1342
11440         F1342=F13+F34+F42
11445         IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
11450         IF (P13>0) AND (F1342<Minfue) THEN Minfue=F1342

```

```

11455      SUBEXIT
11460      !
11465      ! 18. WORK EQUIPMENT GROUP
11470      !
11475 Lg18: CALL F1kdat(Group,Inst,1,2,P12,T12,F12)
11480      CALL F1kdat(Group,Inst,1,3,P13,T13,F13)
11485      CALL F1kdat(Group,Inst,1,5,P15,T15,F15)
11490      CALL F1kdat(Group,Inst,1,6,P16,T16,F16)
11495      CALL F1kdat(Group,Inst,3,4,P34,T34,F34)
11500      CALL F1kdat(Group,Inst,4,2,P42,T42,F42)
11505      T1342=T13+T34+T42
11510      IF (P12>0) AND (T12<Mintim) THEN Mintim=T12
11515      IF (P13>0) AND (T1342<Mintim) THEN Mintim=T1342
11520      IF (P15>0) AND (T15<Mintim) THEN Mintim=T15
11525      IF (P16>0) AND (T16<Mintim) THEN Mintim=T16
11530      F1342=F13+F34+F42
11535      IF (P12>0) AND (F12<Minfue) THEN Minfue=F12
11540      IF (P13>0) AND (F1342<Minfue) THEN Minfue=F1342
11545      IF (P15>0) AND (F15<Minfue) THEN Minfue=F15
11550      IF (P16>0) AND (F16<Minfue) THEN Minfue=F16
11555      SUBEXIT
11560      !
11565      ! 90. SAR SEARCH GROUP
11570      ! FAILURE
11575      !
11580 Lg90: CALL F1kdat(Group,Inst,3,9,P39,T39,F39)
11585      CALL F1kdat(Group,Inst,4,9,P49,T49,F49)
11590      IF (P39>0) AND (T39<Mintim) THEN Mintim=T39
11595      IF (P49>0) AND (T49<Mintim) THEN Mintim=T49
11600      IF (P39>0) AND (F39<Minfue) THEN Minfue=F39
11605      IF (P49>0) AND (F49<Minfue) THEN Minfue=F49
11610      SUBEXIT
11615      !
11620      ! 93. SENSOR SEARCH GROUP
11625      ! FAILURE
11630      !
11635 Lg93: CALL F1kdat(Group,Inst,1,9,P19,T19,F19)
11640      IF (P19>0) AND (T19<Mintim) THEN Mintim=T19
11645      IF (P19>0) AND (F19<Minfue) THEN Minfue=F19
11650      SUBEND

```

APPENDIX E

```

5      !
10     ! PROGRAM TO CREATE A SCENARIO DATA FILE
15     ! FOR READING IN THE CREE PROGRAM SEGMENT
20     ! SPRPOS.
25     !
30     OPTION BASE 1
35     DIM A(15),A1(15),B(20,20),C(20,20),D(20,20),Ax(20)
36     INTEGER Temp
40     PRINT PAGE
45     INPUT "TYPE NAME OF SCENARIO FILE ",S$
50     CREATE S$&"T14",1,25000
55     ASSIGN #1 TO S$&"T14"
60     PRINT PAGE
65     INPUT "INPUT NAME OF SCENARIO",P$
70     PRINT #1;P$
75     PRINT PAGE
80     INPUT "INPUT SCENARIO NUMBER ",Scenno
85     PRINT #1;Scenno
90     PRINT PAGE
95     INPUT "INPUT MAXIMUM TIME ALLOWABLE FOR SORTIE (HRS) ",Mxtime
100    PRINT #1;Mxtime
105    PRINT PAGE
110    INPUT "INPUT RANGE FRACTION OF TOTAL FUEL CPACITY",Rangfr
115    PRINT #1;Rangfr
120    PRINT PAGE
125    INPUT "INPUT NUMBER OF DAYS OF OPERATION ",Ndays
130    PRINT #1;Ndays
135    PRINT PAGE
140    INPUT "INPUT NUMBER OF IMPORTANT TASKS ",Nmimtk
145    PRINT #1;Nmimtk
150    FOR I=1 TO Nmimtk
155    PRINT "INPUT ";I;" TASK NUMBER"
160    INPUT Ax(I)
165    Temp=Ax(I)/100-.5
170    A(I)=Temp
175    A1(I)=Ax(I)-A(I)*100
176    PRINT "A ";A(I);" A1(I) ";A1(I)
180    NEXT I
185    FOR I=1 TO 6
190    PRINT #1;A(I),A1(I)
195    NEXT I
200    PRINT PAGE
205    INPUT "INPUT NUMBER OF NODES IN FLOW CHART",Nnodov
210    PRINT #1;Nnodov
215    PRINT PAGE
220    PRINT "YOU ARE ABOUT TO ENTER THE GROUP-CONNECTION MATRIX "
225    FOR I=1 TO Nnodov
230    FOR J=1 TO Nnodov
235    PRINT "INPUT ENTRY ";I;" ";J
240    INPUT B(I,J)
245    NEXT J
250    PRINT "NEXT ROW"
255    NEXT I
260    FOR I=1 TO Nnodov
265    FOR J=1 TO Nnodov
270    PRINT #1;B(I,J)
275    NEXT J
280    NEXT I
285    Iright=20
290    Idown=Nnodov
295    PRINT PAGE
300    PRINT "YOU ARE ABOUT TO ENTER THE GROUP PLACEMENT MATRIX"
305    FOR I=1 TO Nnodov
310    FOR J=1 TO Nnodov
315    PRINT "ENTRY ";I;" ";J
320    INPUT C(I,J)

```

```

325     NEXT J
330     NEXT I
335     FOR I=1 TO 7
340     FOR J=1 TO 7
345     PRINT #1;C(I,J)
350     NEXT J
355     NEXT I
360     PRINT #1;Iright,Idown
365     PRINT PAGE
370     PRINT "NOW ENTER THE GROUP DATA"
375     FOR I=1 TO Nnodov
380     FOR J=1 TO 20
385     PRINT "ENTRY ";I;" ";J
390     INPUT D(I,J)
395     NEXT J
400     NEXT I
405     FOR I=1 TO Nnodov
410     FOR J=1 TO 20
415     PRINT #1;D(I,J)
420     NEXT J
425     NEXT I
430     PRINT PAGE
435     INPUT "INPUT NUMBER OF PRINT OUTS DESIRED",Nptout
440     PRINT #1;Nptout
445     PRINT PAGE
450     PRINT "THAT COMPLETES THE SCENARIO INPUT"
455     Flag=1
460     PRINT #1;Flag
465     ASSIGN * TO #1
470     END

```

U211776