77-3

Report No. CG-D-48-78

THE CUTTER RESOURCE EFFECTIVENESS EVALUATION (CREE) PROGRAM

A GUIDE FOR USERS AND ANALYSTS

David S. Prerau

U.S. Department of Transportation
Transportation Systems Center
Kendall Square
Cambridge MA 02142

MARCH 1978

FINAL REPORT

Prepared for

U.S. DEPARTMENT OF TRANSPORTATION
United States Coast Guard
Office of Operations
Washington DC 20590

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
CG-D-48-78		

4. Title and Subtitle	5. Report Date
THE CUTTER RESOURCE EFFECTIVENESS EVALUATION (CREE) PROGRAM -- A GUIDE FOR USERS AND ANALYSTS	March 1978
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
David S. Prerau	DOT-TSC-USCG-77-3
9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
U.S. Department of Transportation Transportation Systems Center Kendall Square Cambridge MA 02142	CG811/R8010
	11. Contract or Grant No.
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Final Report June 1975-June 1977
U.S. Department of Transportation United States Coast Guard Office of Operations Washington DC 20590	14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The Cutter Resource Effectiveness Evaluation (CREE) project has developed a sophisticated, user-oriented computer model which can evaluate the effectiveness of any existing Coast Guard craft, or the effectiveness of any of a number of proposed alternative craft (such as a hydrofoil or an air cushion vehicle), in the performance of a selected set of Coast Guard missions, in a given location under specified environmental conditions. The first part of this report describes the CREE Model computer program from the user's viewpoint, and includes complete details on the use of the program. The second part of the report discusses for analysts the structure of the CREE program and some of the difficult theoretical concepts behind it.

17. Key Words	18. Distribution Statement
Advanced Marine Vehicles, Watercraft Assessment, Mission Analysis, Operations Modeling, Effectiveness Analysis, Computer Model, High-Performance Watercraft	Document is available to the U.S. public through the Defense Documentation Center, Cameron Station, Alexandria VA 22314.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	174	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

PREFACE

To fulfill the need of the U.S. Coast Guard for a method to evaluate the effectiveness of any existing Coast Guard craft or of any of a spectrum of proposed alternative craft in the performance of Coast Guard programs, the Transportation Systems Center (TSC) and the U.S. Coast Guard Research and Development Center jointly participated in the Cutter Resource Effectiveness Evaluation (CREE) project.

The author would like to acknowledge the fine work and cooperation of his fellow members of the CREE project study team: Anthony Passera of TSC, LCDR Fred Hamilton of the Coast Guard R&D Center, and Clark Pritchett of the Coast Guard R&D Center. The author is indebted to Patricia Concannon of TSC for her assistance in the preparation and running of the CREE program; to Stephen Stark of TSC for his assistance in the preparation of this report; and especially to Jeffrey Garlitz formerly of Input Output Computer Services, Inc. and now of TSC for his excellent assistance in the upgrading of the CREE program and the preparation and editing of this report.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (CONTINUED)

1. INTRODUCTION

The Cutter Resource Effectiveness Evaluation (CREE) model is a sophisticated, user-oriented computer model which evaluates the effectiveness of an existing Coast Guard craft, or the effectiveness of any of a number of proposed alternatives (such as an air-cushion vehicle or a hydrofoil), in the performance of a selected set of Coast Guard missions in a given location under specified environmental conditions. Selected craft--even those not actually in existence--can be compared in performance against each other. The CREE model can determine which of several possible craft is the best match for a given Coast Guard operational requirement. Conversely, the model can be used to determine the operational procedures which will optimize the accomplishment of Coast Guard missions with a given craft.

A complete description of the CREE model is found in the three-volume Cutter Resource Effectiveness Evaluation Model report.* The present report describes the CREE model computer program. Sections 2, 3, and 4 describe the program from the user's viewpoint, and give a detailed discussion of its use. Section 5 describes the structure of the program, and some of the difficult theorectical concepts behind the program. This section will give analysts a better understanding of how the CREE program functions.

*C.W. Pritchett, F.M. Hamilton, A. Passera, and D.S. Prerau, Cutter Resource Effectiveness Evaluation Model, 3 Vols., Department of Transportation, United States Coast Guard, Office of Operations, Washington DC,

Vol. I: Analysis and Synthesis of Coast Guard Programs (CG-D-45-78);
Vol. II: Evaluation of Craft Performance in Coast Guard Programs (CG-D-46-78); and
Vol. III: Utilization of Cutter Resource Effectiveness Evaluation Model (CG-D-47-78).
June 1977.

2.2 CRAFT AND ENVIRONMENTAL INFORMATION REQUIRED

The user must select the craft to be evaluated and then supply to the CREE program the following information:

a. The Craft Type,

b. The Craft Size (indicated by Craft Displacement or Craft Length),

c. The Craft Design Speed, and

d. The Craft Fuel Fraction (the fraction of useful payload that is carried as fuel).

For an existing Coast Guard craft, only the Craft Type need be input since size, speed, and fuel-fraction data for such craft are stored in the computer.

The user must select the environment under which the craft is to be evaluated and then supply to the CREE program the following information:

1. The Visibility (as indicated by a Visibility Distribution),

2. A Distribution of the Sizes of Possible Craft which must be Towed,

3. The Depth of Water (as indicated by a Cumulative Depth Distribution), and

4. The Sea State (as indicated by a Sea-State Distribution).

The above craft and environmental information are put into a computer file called "CRAFT.DATA". Detailed descriptions of the data required in the CRAFT.DATA file and the formats for these data are discussed in Section 3.1.

2.3 SCENARIO INFORMATION REQUIRED

The user must design a scenario for the Coast Guard program under which the craft is to be evaluated. This is done (as discussed in Volume I of Cutter Resource Effectiveness Evaluation Model), by constructing a flowchart scenario and then supplying all the

NODE-PLACEMENT RULES

1. Replace "Start" and any immediately following decision diamond by a node. (Leave any probabilities unchanged.)
2. Replace "Stop" and any immediately preceding junction point by a node.
3. Replace each decision diamond by a node.
4. Replace each junction point by a node. (Leave the probabilities unchanged.)
5. Put a node between any two Groups that are still not separated by a node.
6. When there are any two nodes joined by more than one path with no intervening nodes, add a node to all but one of the paths to make the node sequence describing each path unique.

NODE-NUMBERING RULES

1. The node created by Node Placement Rule 1 is numbered Node 1.
2. The node created by Node Placement Rule 2 is numbered Node 2.
3. The nodes created by Node Placement Rules 3 to 6 may be numbered in arbitrary order, but sequentially in number starting with Node 3.

FIGURE 2.3-1. NODE-PLACEMENT AND NUMBERING RULES

RULE 2

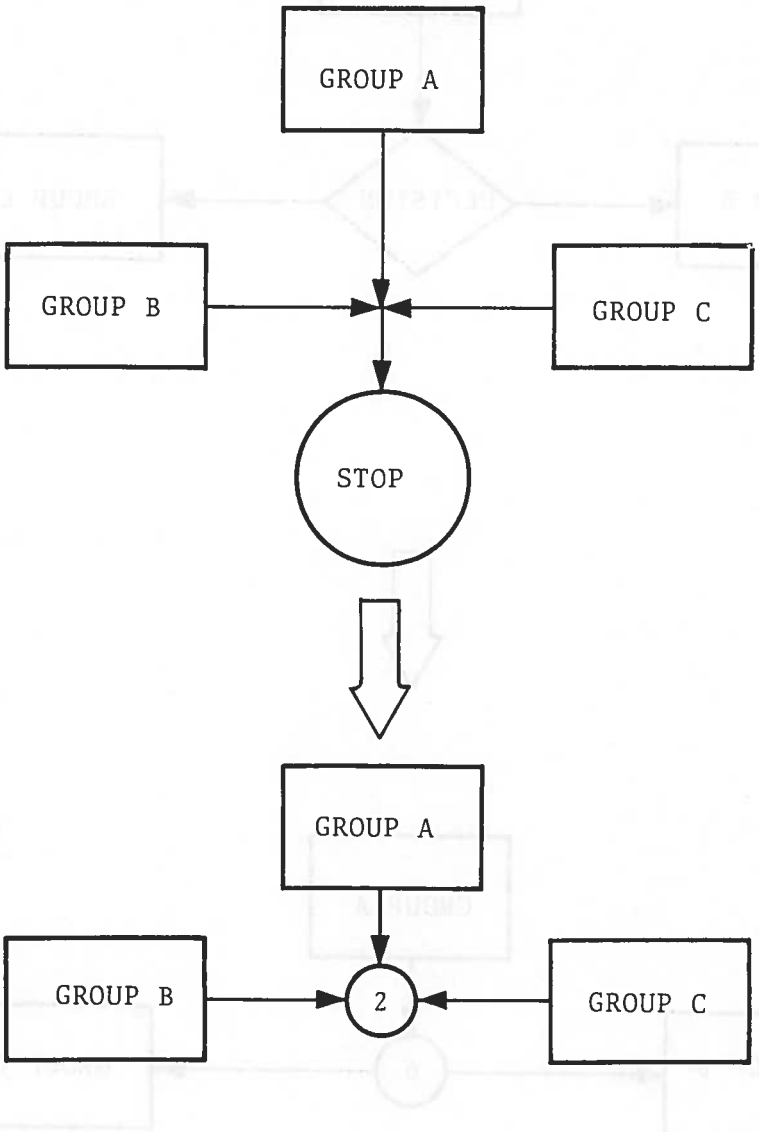


FIGURE 2.3-2. ILLUSTRATIONS OF NODE-PLACEMENT RULES (CONTINUED)

RULE 4

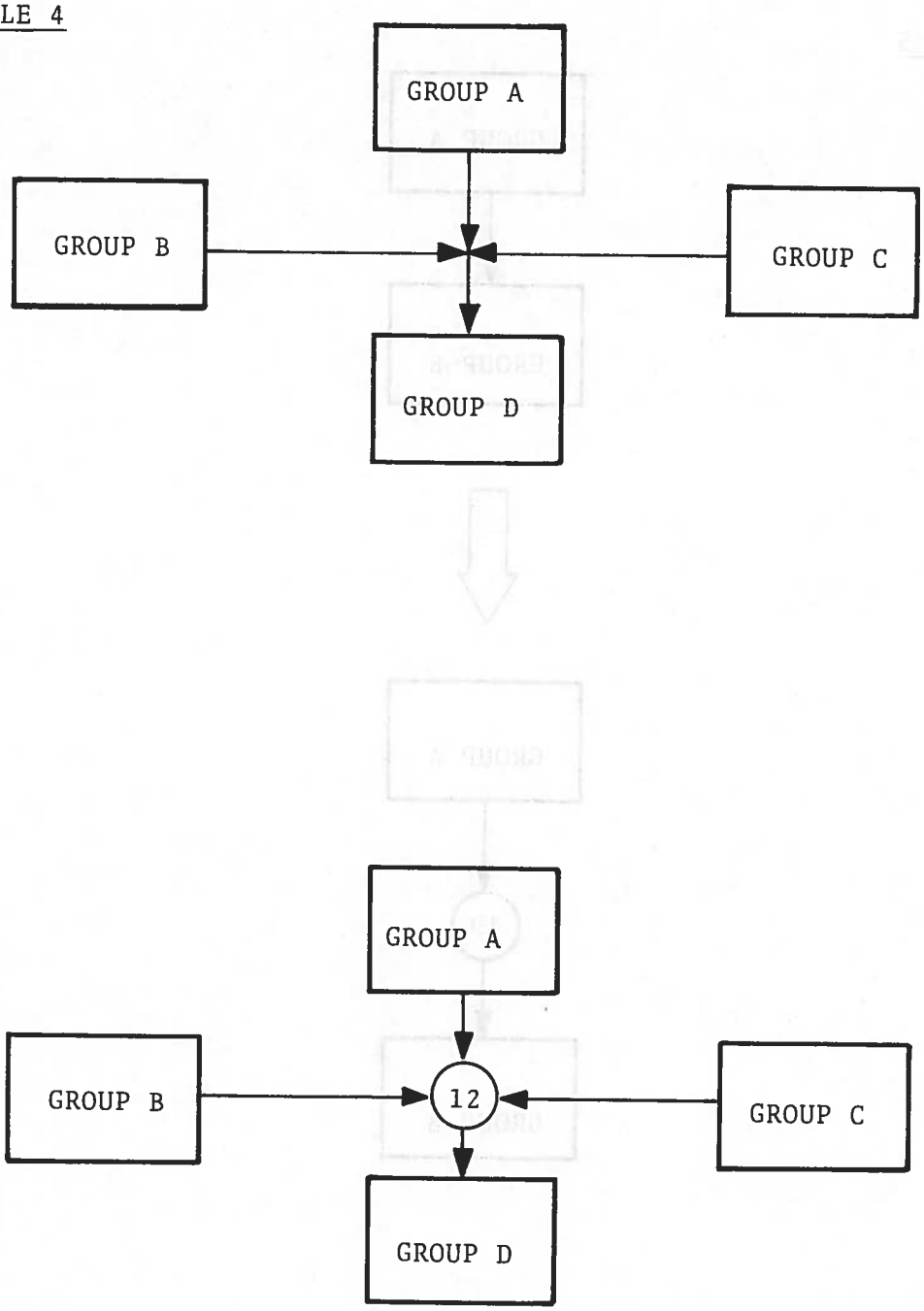
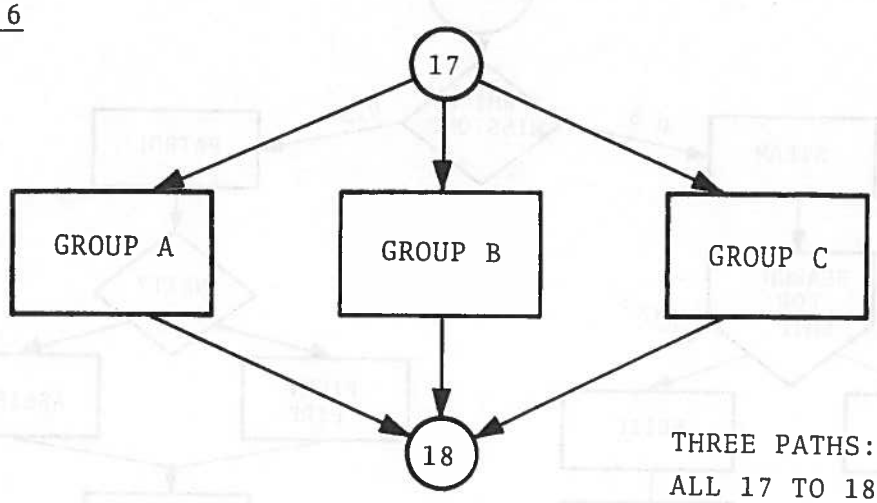


FIGURE 2.3-2. ILLUSTRATIONS OF NODE-PLACEMENT RULES (CONTINUED)

RULE 6



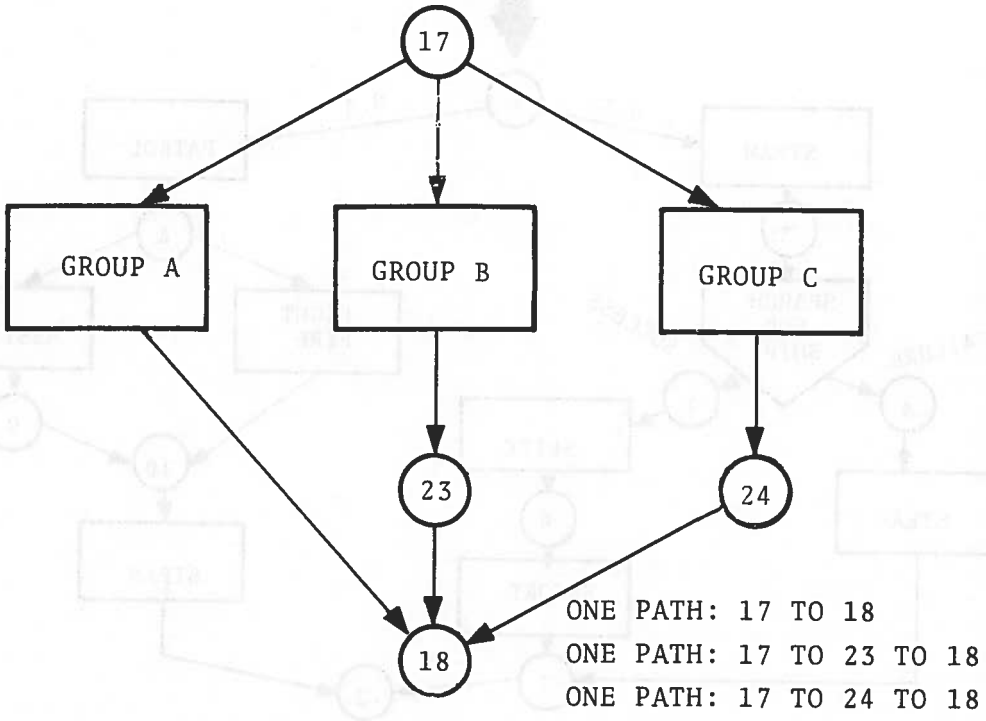


FIGURE 2.3-2. ILLUSTRATIONS OF NODE-PLACEMENT RULES (CONTINUED)

e. The Number of Days of Operation,

f. A List of Tasks that are "Important" in this Coast Guard program,

g. The Total Number of Nodes in the Noded Flowchart,

h. The Group Connection Matrix (a matrix where each entry, C(i,j), denotes the probability of going from Node i to Node j),

i. The Group-Placement Matrix (a matrix where each entry indicates which Functional Task Group, if any, occurs between each pair of nodes),

j. Group Data (the data from the Group Data Sheets, for each instance of each Group in the scenario),

k. The Number of Printouts, and

1. The Output Format (either printing sorties or not).

The above information is put into a computer file called "SCENARIO.DATA". Detailed descriptions of the data required in the SCENARIO.DATA file and the formats for these data are discussed in Section 3.2.

3.1 CRAFT DATA

A sample CRAFT.DATA file is shown in Figure 3.1-1. For each craft-effectiveness evaluation, the CRAFT.DATA file contains two lines of data, one describing the craft to be considered, and the second describing the environmental conditions under which the craft will operate. A CRAFT.DATA file may contain the specifications for several craft-effectiveness evaluations, each represented by two lines of data. The format requirements are given below. (In these formats, "#" represents a numeric input.)

3.1.1 Craft Selection

The craft is described by the following inputs: the craft type, the craft size (as specified by either craft displacement or craft length), the craft design speed and the craft fuel fraction. For craft size, the user only has to input either craft displacement or craft length (not both). The CREE program will provide the other value, for which the user must only input zero.

```
T= 10,D=0000.0,L= 100.0,S=50.0,F=0.50
US= 2,TW= 1,DH= 1,SS= 6
T= 30,D= 200.0,L=0000.0,S=60.0,F=0.50
US= 2,TW= 1,DH= 1,SS= 6
T= 40,D=0000.0,L= 100.0,S=40.0,F=0.50
US= 2,TW= 1,DH= 1,SS= 6
T= 60,D=0000.0,L= 250.0,S=20.0,F=0.50
US= 2,TW= 1,DH= 1,SS= 6
T=110,D=0000.0,L=0000.0,S=00.0,F=0.00
US= 2,TW= 1,DH= 1,SS= 6
T=111,D=0000.0,L=0000.0,S=00.0,F=0.00
US= 3,TW= 1,DH= 1,SS= 6
```

FIGURE 3.1-1. TYPICAL CRAFT.DATA FILE

TYPE CODE	CRAFT
10	Hydrofoil, Submerged Foil
11	Hydrofoil, Surface Piercing
20	ACV (Air Cushion Vehicle) - Low Pressure/Length Ratio
21	ACV - High Pressure/Length Ratio
30	SES (Surface-Effect Ship)
40	Planing Craft
50	Catamaran
60	SWATH (Small Waterplane Area Twin Hull)
70	Hybrid Vessel
80	Conventional 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 3.1-2. CRAFT TYPE

3.1.2 Environment Specification

The environment in which the craft is to be evaluated (i.e., in which the scenario takes place) is described by the following inputs: the visibility distribution number, the towing distribution number, the depth distribution number, and the sea-state distribution number.

FORMAT: "VS=##,TW=##,DH=##,SS=##"

EXAMPLE: "VS= 3,TW= 4,DH= 1,SS=10"

The four distributions that must be specified will now be discussed.

a. Visibility-Distribution Number, VS

The Visibility-Distribution Number, VS, is right-justified. Three visibility distributions are presently available in the CREE program, as shown in Figure 3.1-5. For example, Visibility Distribution No. 2, called "Good", implies 70 percent chance of Good Visibility, 20 percent chance of Fair Visibility and 10 percent chance of Poor Visibility.

DISTRIBUTION NUMBER	DISTRIBUTION DESCRIPTION	V I S I B I L I T Y		
		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 3.1-5. VISIBILITY PROBABILITY DISTRIBUTIONS

d. Sea-State Distribution Number, SS

The Sea-State Distribution Number, SS, is right-justified. Ten sea-state distributions are presently available in the CREE program, as shown graphically in Appendix B and as summarized in Figure 3.1-7.

SEA-STATE DISTRI-BUTION NUMBER	AVERAGE OF SEA-STATE DISTRI-BUTION	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	0.05	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 3.1-7. SEA-STATE PROBABILITY DISTRIBUTIONS

3.2 SCENARIO DATA

The format requirements for a SCENARIO.DATA file are given below. In the format descriptions, "+" represents "blank", and "@" represents an alphanumeric input. A sample SCENARIO.DATA file, which corresponds to the scenario of Figure 3.2-1, is shown in Figure 3.2-2.

```
CG PROGRAM=SAR7
SCENARIO NO.= 1
MAXIMUM TIME=  12.0
RANGE FRACTION=0.96
NO. DAYS OF OPERATION= 100
NUMBER OF IMPORTANT TASKS= 6
411 404 305 304 208 204
NODES= 7
CONNECTION MATRIX=
0.00 0.00 0.70 0.70 0.60 0.30 0.60
0.00 0.00 0.00 0.00 0.00 0.00 0.00
0.00 0.60 0.00 0.60 1.00 0.00 0.00
0.00 1.00 0.00 0.00 0.00 0.00 0.00
0.00 1.00 0.00 0.00 0.00 0.60 0.00
0.00 0.00 0.00 0.00 0.00 0.00 1.00
0.00 1.00 0.00 0.00 0.00 0.00 0.00
GROUP PLACEMENT MATRIX=
0000 0000 1001 9001 0000 1501 0006
0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0006 0101 0000 0000
0000 1502 0000 0000 0000 0000 0000
0000 0901 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0102
0000 0902 0000 0000 0000 0000 0000
&GROUP DATA=
1,1,,5,,2,0,,3,,1,,5,,1,1,,2,,1,,5,,1,6*0,
1,2,,3,,7,0,0,,1,,3,,1,,5,0,0,0,0,6*0,
9,1,,6,,1,,3,30,30,10,30,11*0,
9,2,,7,0,,3,20,0,0,20,11*0,
10,1,0,,1,,0,,0,,0,,0,,0,,,1,4,3,,75,8,,6*0,
15,1,0,1,0,0,30,0,12*0,
15,2,0,1,0,0,30,0,12*0
&END
NUMBER OF PRINTOUTS= 1
OUTPUT FORMAT=1
```

FIGURE 3.2-2. TYPICAL SCENARIO.DATA FILE

f. Important Tasks

The tasks which the user chooses to designate as "important" will appear in the Scenario Evaluation output. The first line has the number of "important" tasks chosen, right-justified. The next line contains the Code Numbers for these "important" tasks, 10 to a line. The Code Number for each task is shown in Figure 3.2-3.

FORMAT: First Line: "+Number+of+Important+Tasks=@@"

Following Lines: "+@@@+@@@+@@@+..."

EXAMPLE: "Number of Important Tasks=13

401 411 305 409 302 202 203 102 402 406
204 413 423"

g. Number of Nodes

The Number of Nodes is the number of nodes in the Flow-Chart Scenario, right-justified.

FORMAT: "+NODES=@@"

EXAMPLE: " NODES= 4"

h. The Group-Connection Matrix

The Group-Connection Matrix describes the structure of the Coast Guard program flowchart by indicating which node points are connected to which other node points, and the probabilities of going from a given node point to any of the nodes to which it is connected. Each entry, C(i,j), is the probability of going from Node i to Node j (i.e., the Link Probability for the link i to j). If Nodes i and j are not connected, the entry in the Matrix is 0.

If Node i is connected to Node j and also to Node k through a Three-Port (One Input/Two Output) Search Group, Group 10 or Group 13, then the probability entered for each node pair is the total probability of going from i to the Search Group. For example, in Figure 3.2-4, C(3,5)=C(3,6) = 1.0, since from Node 3 the probability is 1.0 that the Search will be performed. Also, C(8,10)=C(8,12) =

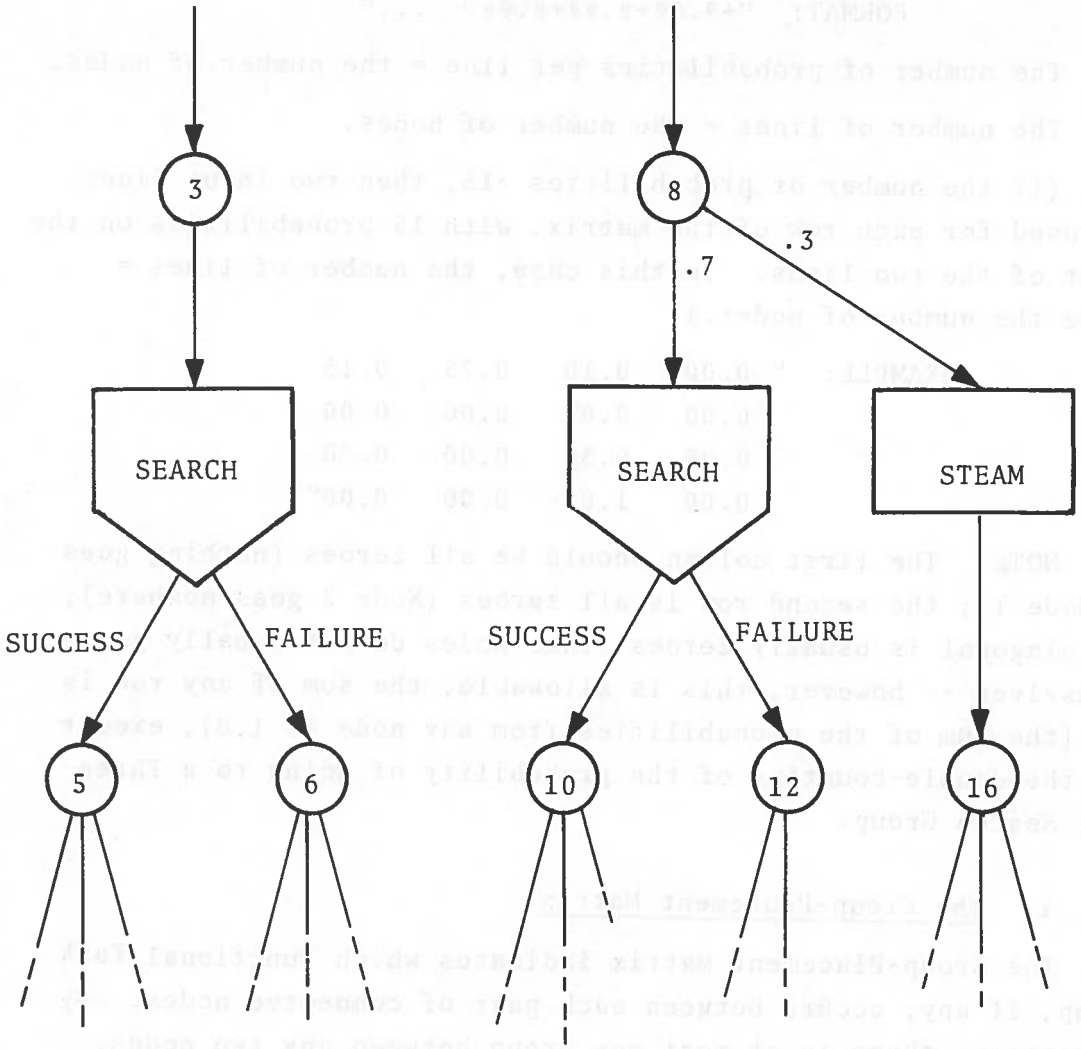


FIGURE 3.2-4. EXAMPLE OF THREE-PORT (ONE INPUT/TWO OUTPUT GROUPS)

each time with different probabilities, times, distances, etc.) For example, a first instance of Group 19 would be represented by "1901", and a second instance by "1902". A zero entry can be indicated by "0000" or simply by "0".

For the case of Three-Port Search Groups, Groups 10 and 13, a special convention is used only for the Group-Placement Matrix: the Group Number (10 or 13) is used between the input node and the output "Success" node. Between the input node and the "Failure" node, the same instance number is used, but the Group Number is increased by 80 (i.e., to 90 or 93). This use indicates to CREE which is the "Success" node and which is the "Failure" node. For example, in Figure 3.2-4 if both Searches are Group 13, then G(3,5)=1301, G(3,6)=9301, G(8,10)=1302, and G(8,12)=9302.

FORMAT: "+@@@@+@@@@+@@@@ ...."

The number of entires per line = the same as for the Group Connection Matrix.

```
EXAMPLE: " 0    1701    1501    0
           0    0       0       0
           0    0701    0       1801
           0    0702    0       0   "
```

NOTE: In the Group Placement Matrix, there will always be a zero entry corresponding to each zero entry in the Group-Connection Matrix. There may also be some zero entries corresponding to non-zero entries in the Group-Connection Matrix.

j. Group Data

For each Group appearing in the Group-Placement Matrix, there will be a line of Group Data indicating the probabilities, the task times and distances, and other data pertinent to the tasks in the Group. Each line is in free format, with the numbers in the following order, separated by commas:

$g,i,e_1,e_2,e_3,\ldots,e_n$, k*0,

Group	
1	P1,P2,P3,P4,T(1),T(2),T(3),T(4),T(5),T(6),T(7),T(8)
2	P1,P2,D(1),SPEED(1),D(2)
3	P1,P2,T(1),T(2),T(3),T(4)
4	P1,P2,T(1),D(1),N(1),T(2)
5	P1,P2,T(1),T(2),T(3),T(4),T(5),T(6)
6	P1,P2,P3,P4,T(1),T(2),T(3),T(4)
7	P1,P2,D(1),SPEED(1),D(2)
8	P1,P2,T(1),T(2)
9	P1,P2,P3,D(1),D(2),SPEED(2),D(3)
10	P1,P2,S.W.(1),AREA(1),# SEARCHES(1), COV.FAC.(1), MAX SEARCH TIME(1), S.W.(2), AREA(2), # SEARCHES(2), COV.FAC.(2), MAX SEARCH TIME(2)
11	D(1)
12	T(1),D(2)
13	S.W.(1),E(1), TRGT SP(1),T MAX(1)
14	T(1)
15	P1,P2,P3,D(1),D(2),D(3)
16	P1,P2,T(1),T(2)
17	P1,P2,T(1),D(2),AREA(2),WT(2),T(3),D(4),AREA(4),WT(4)
18	P1,P2,P3,P4 T(1),T(2),T(3),T(4),T(5),T(6)

Key

Symbol	Definition
P	Probability
T	Time
D	Distance
SPEED	Craft Speed
N	Number of Ships
S.W.	Sweep Width
AREA (Group 10)	Initial SAR Search Area
# SEARCHES	Number of Searches
COV.FAC.	Coverage Factor (Fraction of Search Area Covered in One Search)
MAX SEARCH TIME	Maximum Search Time
E	Initial Error in Target-Ship Position
TRGT SP	Target Speed
T MAX	Maximum Search Time
AREA (Group 17)	Deck Area Required by Transported Equipment
WT	Weight of Transported Equipment

NOTE: Numbers in parentheses denote the task number within the Group to which the data item refers.

FIGURE 3.2-5. REQUIRED GROUP-DATA INPUTS

4. RUNNING CREE PROGRAM

When the user has constructed a CRAFT.DATA file describing the craft and environmental conditions and a SCENARIO.DATA file describing the Coast Guard program, he is ready to run the CREE program. Running the CREE program requires one short step. The TSO "SUBMIT" Command is utilized, and the user simply enters:

SUBMIT CREE

on his terminal.

The output for a CREE model run consists of the following:

a. Craft Characteristics,

b. Craft Parameters for Master Tasks,

c. Task Probabilities of Success for Master Tasks,

d. Craft Parameters for Individual Tasks (two pages),

e. Task Probabilities for Individual Tasks (two pages),

f. Scenario Data (as input by the user),

g. Sortie Outputs (only if Output Format=1),

h. Sortie Summary,

i. Scenario Overall Results, and

j. Scenario Evaluation.

Examples of each of these output pages are shown in Figures 4-1 through 4-10, respectively.

C R A F T P A R A M E T E R S

CRAFT TYPE CATAMARAN
DISPLACEMENT 94 TONS
LENGTH 95 FEET
DESIGN SPEED 40 KNOTS
FUEL FRACTION 0.50

VISIBILITY DISTRIBUTION NO. 1
TOW DISTRIBUTION NO. 4
DEPTH DISTRIBUTION NO. 1
SEA STATE DISTRIBUTION NO. 4
(AVERAGE SEA STATE=2.0)

TASK CODE	CARGO CPCTY	DRAFT	MANEUV	SEA STATE	TOW	
	CC	DF	MN	LS	TW	
ON SCENE:						
ASST	--	1.00	0.94	1.00	--	ASSIST
BORD	--	1.00	0.94	1.00	--	BOARD
MNAC	--	1.00	0.94	1.00	--	MONITOR ACTIVITIES
RTRV	--	1.00	0.94	1.00	--	RETRIEVE
WAIT	--	--	--	1.00	--	WAIT
WEQD	--	1.00	--	1.00	--	WORK EQUIPMENT @ DRIFT
WEQP	--	1.00	0.94	1.00	--	WORK EQUIPMENT @ POSITION
REDUCED SPEED:						
SDIU	--	1.00	--	1.00	--	SEARCH FOR DISTRESSED UNIT
SESC	--	--	--	1.00	--	SLOW ESCORT
SPAT	--	1.00	--	1.00	--	SLOW PATROL
SPEO	--	1.00	--	1.00	--	SEARCH FOR PEOPLE
TOWS	--	--	1.00	1.00	0.98	TOWS
CRUISE SPEED:						
ESCT	--	--	--	1.00	--	ESCORT
IDNT	--	--	1.00	1.00	--	IDENTIFY
PATL	--	--	--	1.00	--	PATROL
STGT	--	1.00	--	1.00	--	SEARCH FOR TARGET
TRPT	****	--	--	1.00	--	TRANSPORT
TRST	--	--	--	1.00	--	TRANSIT
FLANK SPEED:						
RSPD	--	--	--	1.00	--	RESPOND

**** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AND WEIGHT OF CARGO)

FIGURE 4-2. CRAFT PARAMETERS FOR MASTER TASKS OUTPUT

CRAFT PARAMETERS

CRAFT TYPE HYDROFOIL-SUBMERGED FOIL
DISPLACEMENT 132 TONS
LENGTH 100 FEET
DESIGN SPEED 50 KNOTS
FUEL FRACTION 0.50

VISIBILITY DISTRIBUTION NO. 2
TOW DISTRIBUTION NO. 1
DEPTH DISTRIBUTION NO. 1
SEA STATE DISTRIBUTION NO. 6
(AVERAGE SEA STATE=3.0)

TASK CODE	CARGO CPCTY	DRAFT MANEUV		SEA STATE	TOW	
	CC	DF	MN	LS	Tw	
ON SCENE:						
BRD	--	1.00	0.93	0.93	--	BOARD
FFF	--	1.00	0.93	0.88	--	FIGHT FIRE FROM CG VESSEL
FFO	--	--	--	0.95	--	FIGHT FIRE ON ANOTHER VESSEL
GAS	--	1.00	0.93	0.95	--	GENERAL ASSISTANCE
INS	--	--	--	0.95	--	INSPECTION
LEQ	--	1.00	0.93	0.88	--	LOAD EQUIPMENT
LOI	--	--	--	0.95	--	LOITER
LSB	--	1.00	0.93	0.88	--	LAUNCH SMALL BOAT
MAC	--	1.00	0.93	0.95	--	MONITOR ACTIVITIES
MOS	--	1.00	0.93	0.95	--	MONITOR OIL SPILL
OBA	--	--	--	0.95	--	ON BOARD ASSISTANCE
OSC	--	--	--	0.95	--	ON SCENE COMMANDER(GENERAL)
RBP	--	1.00	0.93	0.93	--	RETRIEVE BOARDING PARTY
ROB	--	1.00	0.93	0.88	--	RETRIEVE OBJECTS
RPE	--	1.00	0.93	0.88	--	RESCUE PEOPLE
RSB	--	1.00	0.93	0.88	--	RETRIEVE SMALL BOAT
SSI	--	1.00	0.93	0.95	--	STAKEOUT SPECIAL INTEREST VESSEL
SZE	--	--	--	0.95	--	SEIZE
TWS	--	1.00	0.93	0.88	--	TAKE WATER SAMPLE
ULQ	--	1.00	0.93	0.88	--	UNLOAD EQUIPMENT
WOB	--	--	--	0.95	--	WORK EQUIPMENT FROM SMALL BOAT
WOD	--	1.00	--	0.88	--	WORK EQUIPMENT @ DRIFT
WOF	--	1.00	0.93	0.88	--	WORK EQUIPMENT @ FIXED POSITION

FIGURE 4-4. CRAFT PARAMETERS FOR INDIVIDUAL TASKS OUTPUT

TASK PROBABILITIES OF SUCCESS

CRAFT TYPE HYDROFOIL-SUBMERGED FOIL
DISPLACEMENT 132 TONS
LENGTH 100 FEET
DESIGN SPEED 50 KNOTS
FUEL FRACTION 0.50

VISIBILITY DISTRIBUTION NO. 2
TOW DISTRIBUTION NO. 1
DEPTH DISTRIBUTION NO. 1
SEA STATE DISTRIBUTION NO. 6
(AVERAGE SEA STATE=3.0)

TASK CODE	TASK PROB. OF SUCCESS	TASK
ON SCENE:		
BRD	0.864	BOARD
FFF	0.824	FIGHT FIRE FROM CG VESSEL
FFO	0.950	FIGHT FIRE ON ANOTHER VESSEL
GAS	0.887	GENERAL ASSISTANCE
INS	0.950	INSPECTION
LEQ	0.824	LOAD EQUIPMENT
LOI	0.950	LOITER
LSB	0.824	LAUNCH SMALL BOAT
MAC	0.887	MONITOR ACTIVITIES
MOS	0.887	MONITOR OIL SPILL
OBA	0.950	ON BOARD ASSISTANCE
OSC	0.950	ON SCENE COMMANDER(GENERAL)
RBP	0.864	RETRIEVE BOARDING PARTY
ROB	0.824	RETRIEVE OBJECTS
RPE	0.824	RESCUE PEOPLE
RSB	0.824	RETRIEVE SMALL BOAT
SSI	0.887	STAKEOUT SPECIAL INTEREST VESSEL
SZE	0.950	SEIZE
TWS	0.824	TAKE WATER SAMPLE
ULQ	0.824	UNLOAD EQUIPMENT
WQB	0.950	WORK EQUIPMENT FROM SMALL BOAT
WQD	0.883	WORK EQUIPMENT @ DRIFT
WQF	0.824	WORK EQUIPMENT @ FIXED POSITION

FIGURE 4-5. TASK PROBABILITIES OF SUCCESS FOR INDIVIDUAL TASKS OUTPUT

```
** SCENARIO DATA **
CG PROGRAM= ELT
SCENARIO NO.=10
MAXIMUM TIME= 144.0
RANGE FRACTION=0.90
NO. DAYS OF OPERATION=  27
NUMBER OF IMPORTANT TASKS=12
202 203 408 405 416 401 413 101 102 209
418 201
NODES=15
CONNECTION MATRIX=
0.0  0.0  0.80 0.0  0.0  0.0  0.0  0.0  0.0  0.20 0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  1.00 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0  1.00 0.0  0.0  0.0  0.0  0.0  0.0  0.0
0.0  0.0  1.00 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0
0.0  0.10 0.0  0.0  0.0  0.0  0.90 0.0  0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.00 0.0  0.0  0.0  0.0  0.0
0.0  0.70 0.0  0.0  0.0  0.0  0.0  0.0  0.30 0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.00 0.0
0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.00 0.0
0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.00 0.0  0.0  0.0  0.0  0.0
0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  1.00
0.0  0.40 0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.0  0.60 0.0  0.0
GROUP PLACEMENT MATRIX=
   0    0 1501    0    0    0    0    0    0 1502    0    0    0
   0    0    0    0    0    0    0    0    0    0    0    0    0
   0    0    0  401    0    0    0    0    0    0    0    0    0
   0    0    0    0    0  501    0    0    0    0    0    0    0
   0    0    0    0    0    0    0    0    0    0    0    0    0
   0 1201    0    0    0    0    0    0    0    0    0    0    0
   0    0    0    0    0    0    0    0    0    0    0    0    0
   0 1504    0    0    0    0    0    0 1503    0    0    0    0
   0    0    0    0    0    0    0    0    0    0    0  402    0
   0    0    0    0    0    0    0    0    0    0    0  402    0
   0    0    0    0    0    0    0    0    0    0    0    0    0
   0    0    0    0    0    0    0    0    0    0    0    0  502
   0 1202    0    0    0    0    0    0    0    0    0    0    0
&GROUP DATA=
 5  1   0.50    0.50    0.50    1.00    2.00    2.00    1.00    0.50    0.0 ... 0.0
 4  1   0.90    0.10    0.50    1.00  100.00   15.00    0.0     0.0     0.0 ... 0.0
15  1   0.80    0.10    0.10  200.00  200.00  200.00    0.0     0.0     0.0 ... 0.0
15  4   1.00    0.0     0.0   100.00    0.0     0.0     0.0     0.0     0.0 ... 0.0
15  3   0.0     1.00    0.0     0.0   100.00    0.0     0.0     0.0     0.0 ... 0.0
12  2   1.00  100.00    0.0     0.0     0.0     0.0     0.0     0.0     0.0 ... 0.0
 5  2   0.0     1.00    0.0     0.0     0.0     0.50    3.00    0.50    0.0 ... 0.0
 4  2   0.50    0.50    0.25    0.25    0.50    6.00    0.0     0.0     0.0 ... 0.0
15  2   0.0     0.0     1.00    0.0     0.0   100.00    0.0     0.0     0.0 ... 0.0
12  1   1.00  200.00    0.0     0.0     0.0     0.0     0.0     0.0     0.0 ... 0.0
&END
NUMBER OF PRINTOUTS= 1
OUTPUT FORMAT=1
```

10 * 0.0

FIGURE 4-6. SCENARIO DATA OUTPUT

********** SORTIE SUMMARY **********

SAR SCENARIO 1

OPERATIONAL REQUIREMENTS:

MAXIMUM DURATION 12.0 HOURS
RANGE FRACTION 0.50
VISIBILITY VERY GOOD
AVERAGE SEA STATE 2.0

SELECTED CRAFT:

WPB 95
DISPLACEMENT 100 TONS
DESIGN SPEED 20 KNOTS
FUEL FRACTION 0.27

FRACTION OF SCENARIO COMPLETED 0.6687

SORTIE NO.	SORTIE TIME (HRS)	SORTIE FUEL (GALS)	FREQUENCY OF OCCURRENCE	SORTIE PROBABILITY OF SUCCESS	SORTIE SUCCESSFUL OCCURRENCE
1	8.2	438	0.0584	0.4000	0.0234
2	8.2	436	0.0097	0.7520	0.0073
3	8.6	638	0.0292	0.5854	0.0171
4	7.5	535	0.0243	0.5854	0.0143
5	7.9	436	0.1461	0.4000	0.0584
6	7.9	434	0.0243	0.7520	0.0183
7	8.3	636	0.0730	0.5854	0.0428
8	7.2	532	0.0609	0.5854	0.0356
9	7.9	436	0.0877	0.4000	0.0351
10	7.9	434	0.0146	0.7520	0.0110
11	8.3	636	0.0438	0.5854	0.0257
12	7.2	532	0.0365	0.5854	0.0214
13	8.4	458	0.0071	0.4000	0.0028
14	8.7	591	0.0030	0.5854	0.0018
15	8.7	591	0.0151	0.5854	0.0089
16	8.4	458	0.0030	0.4000	0.0012
17	8.7	591	0.0013	0.5854	0.0008
18	8.7	591	0.0065	0.5854	0.0038
19	9.0	727	0.0047	0.3114	0.0015
20	9.3	860	0.0020	0.5854	0.0012
21	9.3	860	0.0101	0.5854	0.0059
22	9.0	727	0.0020	0.3114	0.0006
23	9.3	860	0.0009	0.5854	0.0005
24	9.3	860	0.0043	0.5854	0.0025

FIGURE 4-8. SORTIE SUMMARY OUTPUT

```
************* SCENARIO EVALUATION *************

                SAR SCENARIO  1

OPERATIONAL REQUIREMENTS:              SELECTED CRAFT:

MAXIMUM DURATION   12.0 HOURS          CATAMARAN
RANGE FRACTION 0.90                    DISPLACEMENT   94 TONS
VISIBILITY VERY GOOD                   DESIGN SPEED 40 KNOTS
AVERAGE SEA STATE 2.0                  FUEL FRACTION 0.50

   IMPORTANT TASKS COMPLETED IN  180 DAYS OF OPERATION

          TASK      TIMES      TASK
          CODE    COMPLETED    NAME

   ON SCENE:
          GAS          34      GENERAL ASSISTANCE
          OBA         113      ON BOARD ASSISTANCE

   REDUCED SPEED:
          SPT          27      SLOW PATROL
          TOW          68      TOW

   CRUISE SPEED:
          PAT          20      PATROL
          TPE          34      TRANSPORT PEOPLE

   FLANK SPEED:
          NO IMPORTANT TASKS SPECIFIED
```

FIGURE 4-10. SCENARIO EVALUATION OUTPUT

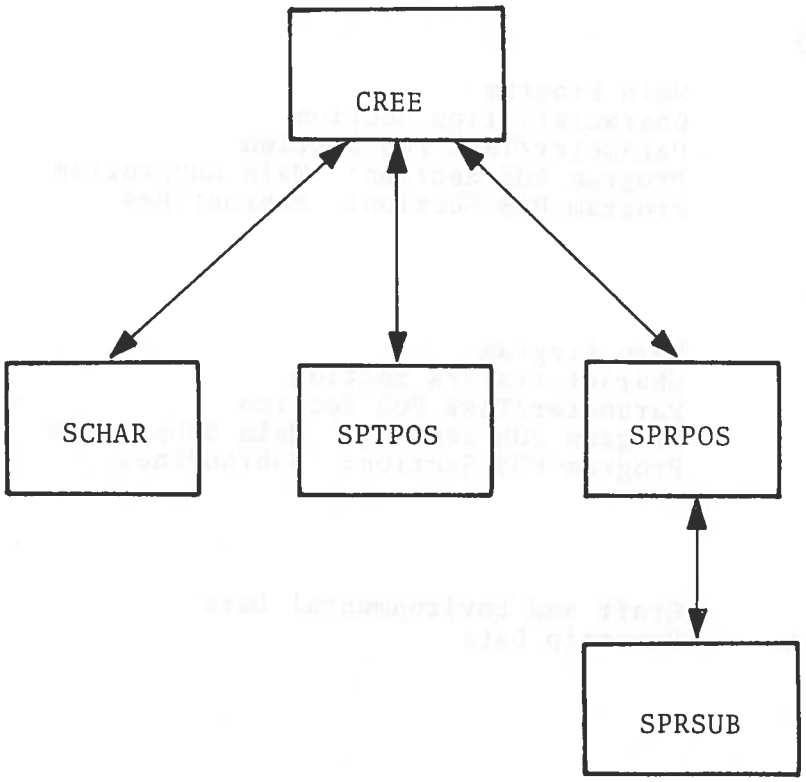


FIGURE 5-1. CREE PROGRAM STORAGE STRUCTURE

5.1 METHODOLOGY

The PROPOS program finds all possible paths through the flowchart. The procedure it uses for doing this is complex. A description of the basic concepts follows.

Consider first the simplified "one-level" case when it is desired only to find all possible paths through the overall Flow-Chart Scenario without looking internally at the Functional Task Groups that occur between overall scenario nodes. Thus, we neglect temporarily the fact that there are several possible paths between any pair of connected overall nodes which have a Functional Task Group between them. The method the PROPOS program uses to solve this still difficult problem of finding all paths through a single flowchart will now be discussed.

The program starts at Node 1, and places the number of this node (i.e., "1") on a last-in/first-out memory structure called a "push-down stack". Then it finds the lowest numbered node, call it Node i, that is connected to Node 1. This second node (i.e., "i") is then added to the push-down stack. Then, starting from Node i, the program looks for the lowest numbered node, call it Node j, which is connected to Node i. This third node (i.e., "j") is also added to the push-down stack. If Node 2 is eventually reached, indicating a complete path through the flowchart, the program prints the path that has been found, as indicated by the numbers of the nodes stored on the push-down stack.

When the program can proceed no further (i.e., a complete path has been found, or a node has been reached which is connected to no other), the program backtracks. It removes the top (i.e., the last) entry on the push-down stack. It then checks if there is another node with a higher number than the node just removed from the stack which is connected to the node now at the top of the stack. If so, this node is added to the push-down stack. If no such node exists, the program removes the new top entry of the push-down stack and tries again. Nodes are added to and removed from the push-down stack until all possible paths through the flowchart have been found.

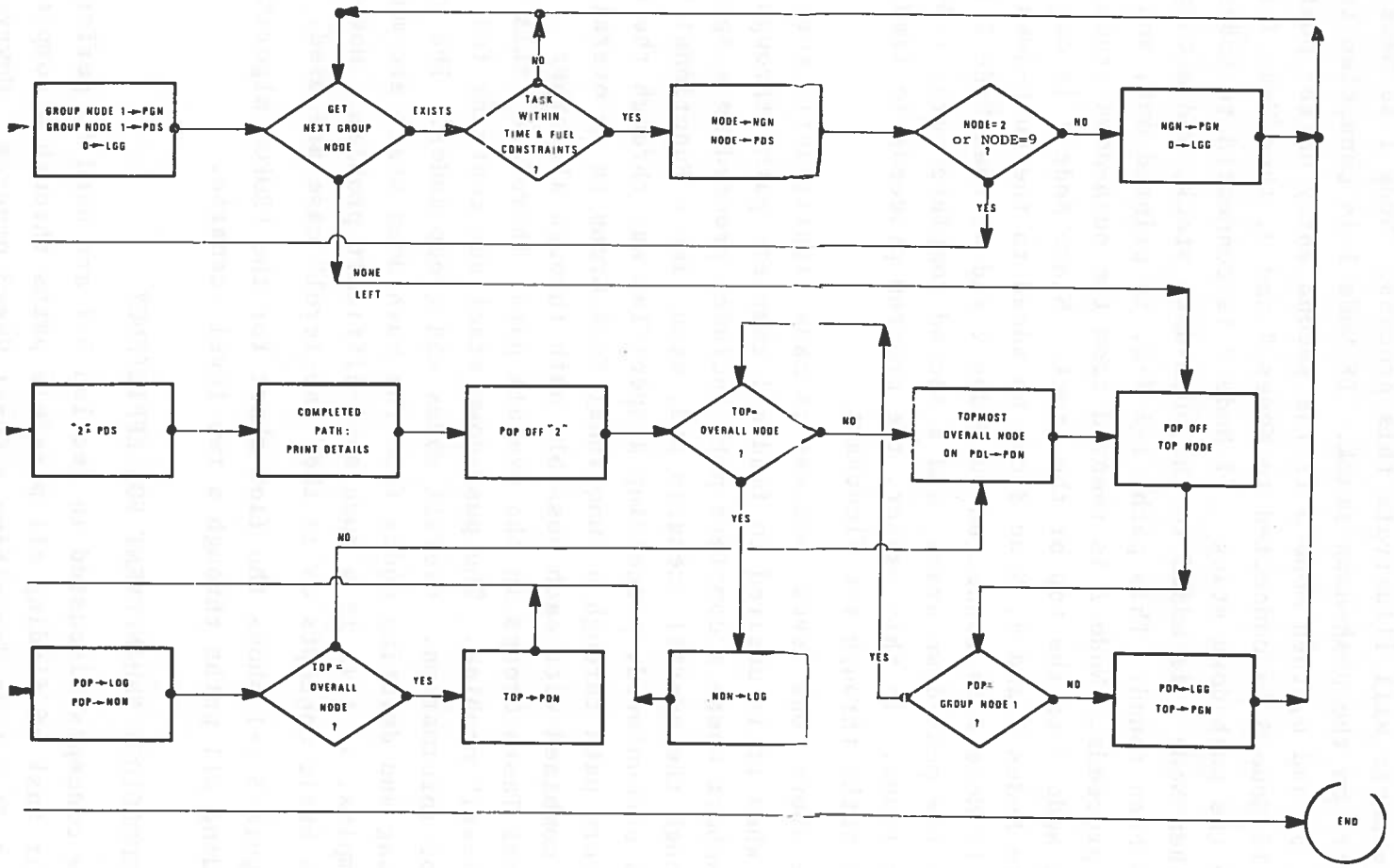


FIGURE 5.1-1. PROPOS ALGORITHM FLOW CHART (CONTINUED)

Many of the paths found will not be realizable (even though they are complete paths) because either they exceed the sortie-time allotment or they cause the craft's fuel use to exceed its fuel supply.

Each craft for which the CREE model is run has associated with it a time restriction and a fuel restriction. The time restriction (maximum time which a craft has to complete a path through the overall scenario) is input by the user. The fuel restriction (the maximum fuel that can be used by the craft) is a function of the craft's fuel capacity and the fuel fraction input by the user.

Any sequence of tasks (i.e., a path) which will take too much time, or use too much fuel, cannot actually be accomplished by the craft. When such a path is found by the methodology, it is rejected. However, computer time is still expended in finding the complete path. A great saving in computer time can be made if a way is found to stop as soon as possible the tracing out of a path that will lead to exceeding the time or fuel restriction. Two such techniques have been found, and have proved to save significant computer time, often cutting computer time by a factor of five or more.

Figure 5.2-1 illustrates the basic concept of the two techniques. (The fuel restriction is illustrated in this figure, but the entire discussion applies equally well to the time restriction.) Path 1-3-4-5-2 will yield a total fuel use of 1700 gallons, exceeding the fuel restriction of 1000 gallons. Thus, path 1-3-4-5-2 should be rejected. If this path can be terminated before it is completely traced out, computer time will be saved.

The basis of the first technique is that when a partial path has been found, the computer will try to continue the path only to a node that will not put the total fuel use over the fuel restriction. In the case of path 1-3-4-5-2, when the partial path 1-3-4 has been found (using a total of 600 gallons), the computer will stop tracing the path. It will not be able to find any way to

continue the path that will keep the total fuel under 1000 gallons (the only possible path addition, Node 5, will add 900 to the 600 gallons already used). Thus, fruitless path 1-3-4-5-2 is terminated after 1-3-4, saving two steps.

Using the second technique, the minimum fuel from a node to the final node (Node 2) is found. Then when each node is reached in finding a path, if the minimum fuel from this node to the end plus the fuel used to get to this node is greater than the total fuel restriction, all paths continuing through the node are rejected. In Figure 5.2-1, it can be seen that the minimum fuel to go from Node 3 to the final node (Node 2) is 600 gallons. However when Node 3 has been reached from Node 1, 500 gallons have already been used. Therefore, the computer rejects all paths beginning with 1-3. Thus, paths 1-3-4-5-2, 1-3-6-7-2, and 1-3-6-2 are all rejected at the same time.

This process is even more powerful when the boxes in the flowchart are not simple tasks, as in the example in Figure 5.2-1, but are full Functional Tasks Groups; i.e., little flowcharts in themselves. If the groups in each box in Figure 5.2-1 all had three possible internal paths, then by stopping at Node 3, there would be 45 possible paths that would be rejected when Node 3 was reached, without the computer having had to trace out any of them.

To accomplish the above, the minimum fuel used from each node to Node 2 must be found. This is not easy since there may be many possible paths from each node to Node 2, and they will vary for each Coast Guard program flowchart. If the computer had to trace out each of these paths, it would use more time than would be saved by the entire technique.

Therefore, the technique can only be useful if a method can be conceived which will find the minimum fuel from each node to Node 2 in a very efficient manner. Such a method was found, and is outlined in the next section.

c. if the fuel used from Node A to Node B plus the fuel used in Node B's n-1 step path is less than the minimum fuel from Node A to Node 2 that has been found thus far,

then: the minimum fuel path (thus far found) for Node A is an n-step path which goes to Node B in 1 step, and then follows Node B's n-1 step path the rest of the way.

(4) Stopping Criteria

If there exists an n+1 step minimum fuel path from Node A to Node 2, and if Node B is the second node on this path, then the n step path from Node B to Node 2 must be the n step minimum fuel path from Node B to Node 2. Thus, if there is no n step minimum path from any node to Node 2, there cannot be any minimum path which requires more than n nodes. Therefore, if no n step minimum fuel path exists, the iteration can stop. The minima found for each node will be the minimum fuel consumed to get from that node to Node 2.

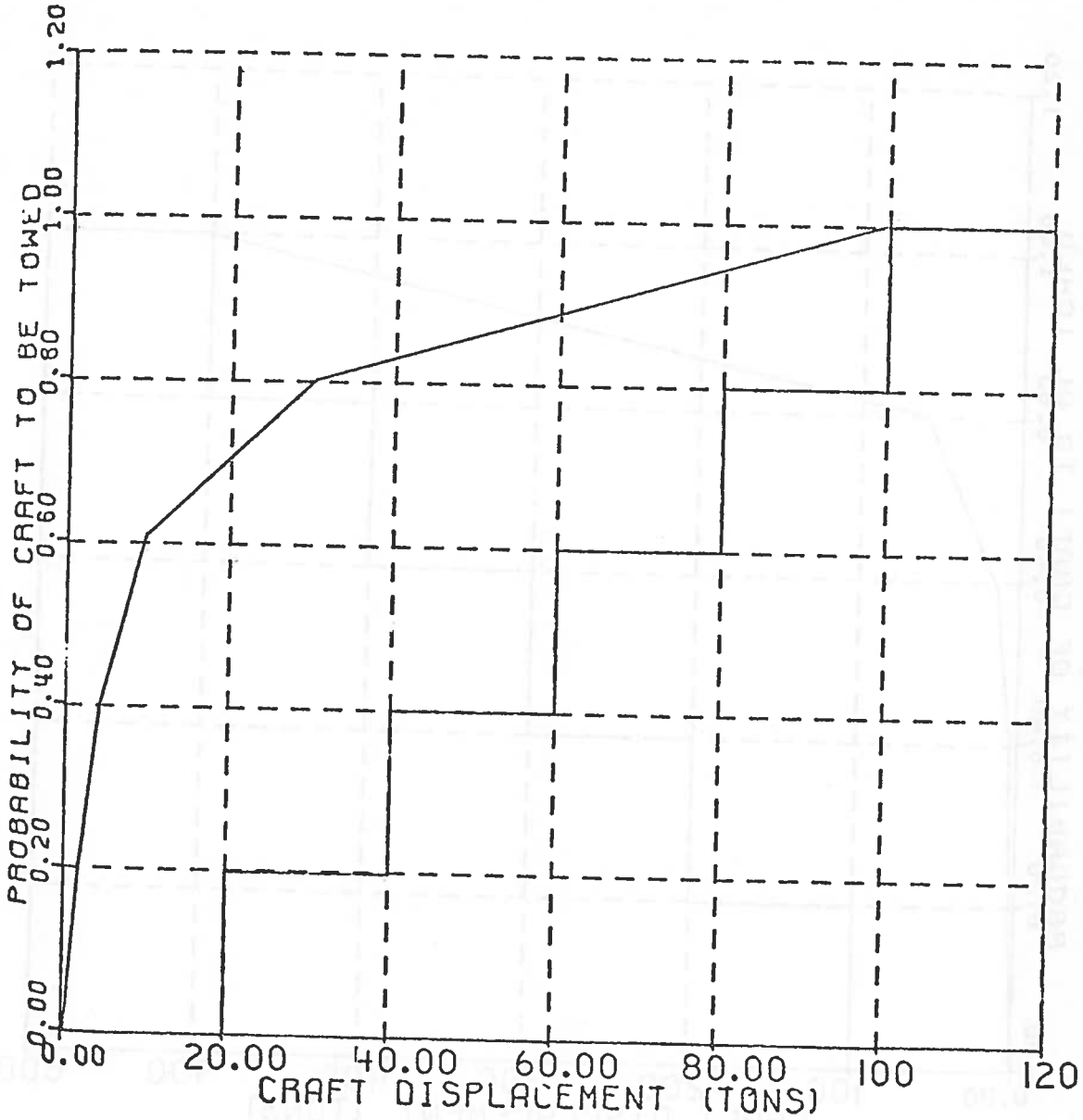


FIGURE A-2. TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT--TOW DISTRIBUTION NUMBER 2

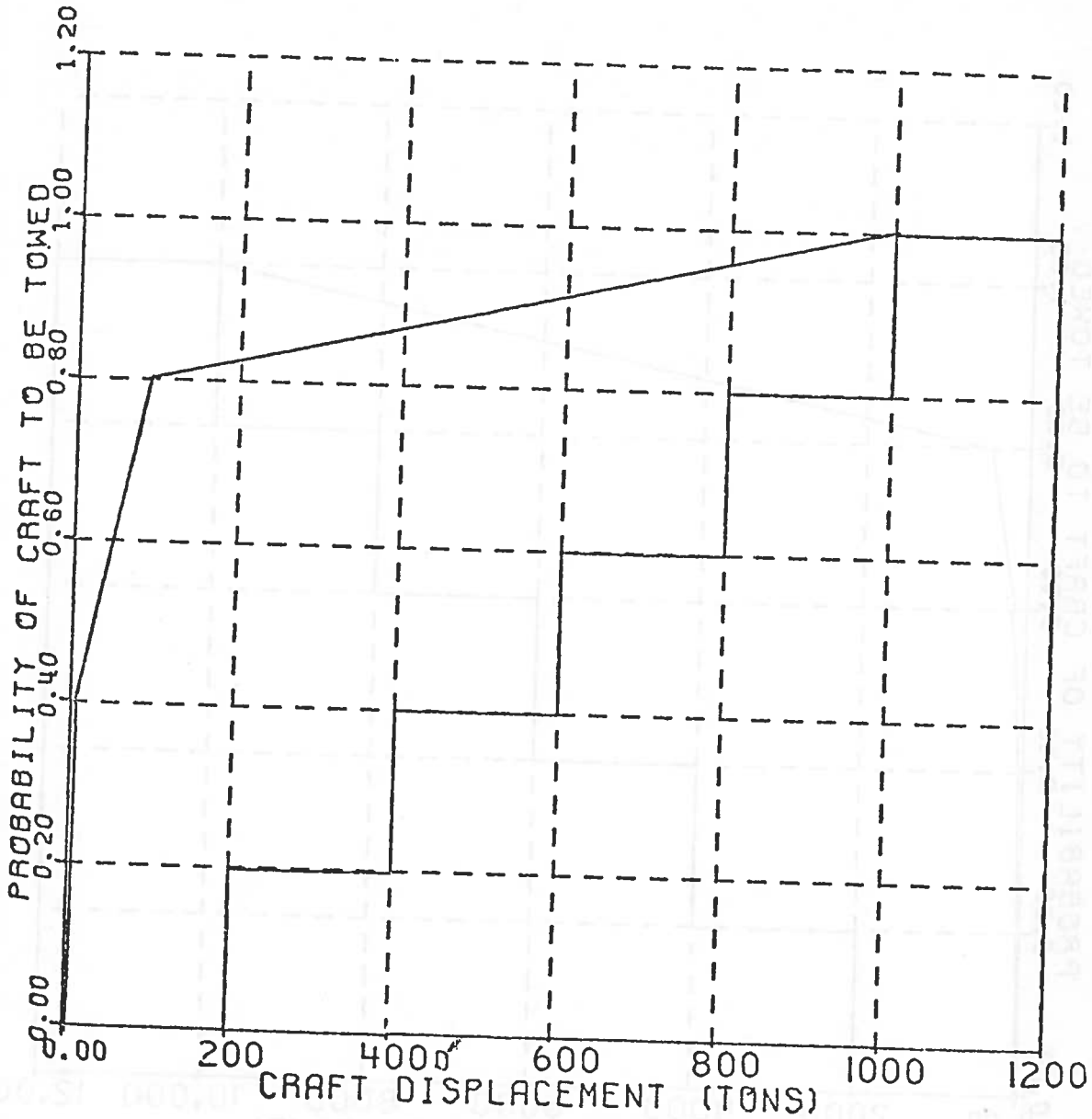


FIGURE A-4. TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT--TOW DISTRIBUTION NUMBER 4

APPENDIX B: SEA-STATE DISTRIBUTIONS

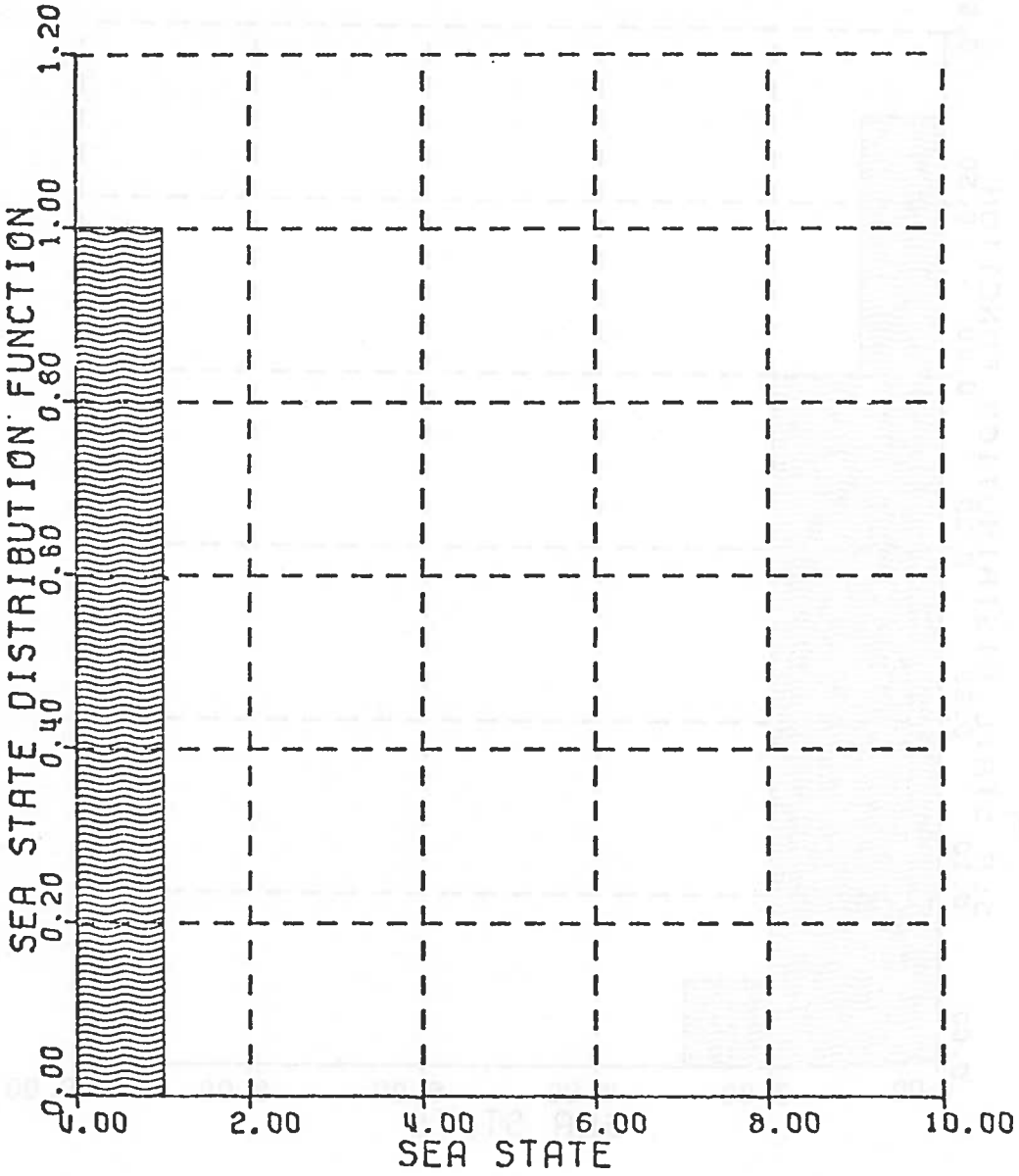


FIGURE B-1. SEA STATE DISTRIBUTION NUMBER 1--AVERAGE SS=0.5

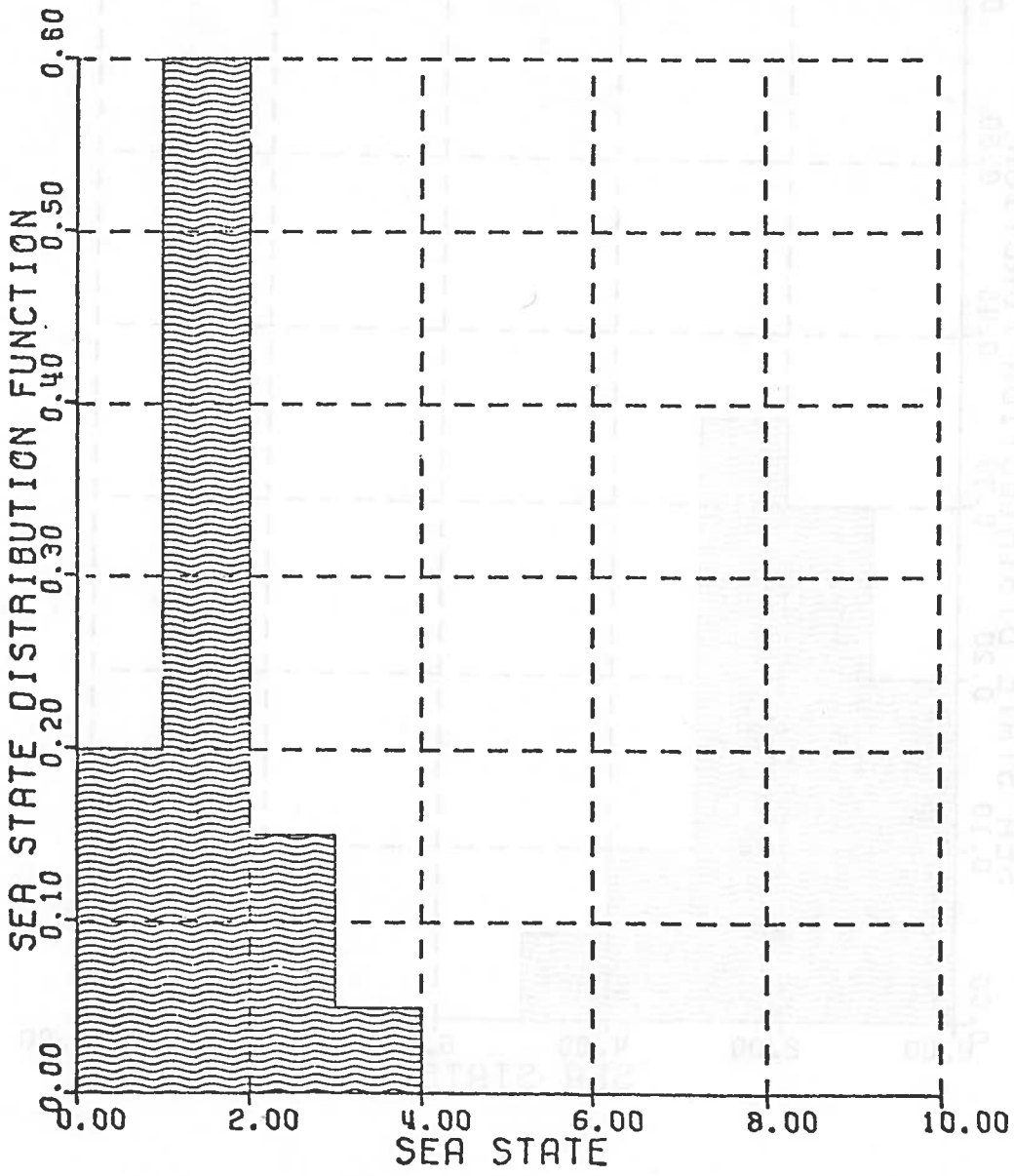


FIGURE B-3. SEA-STATE DISTRIBUTION NUMBER 3--AVERAGE SS=1.5

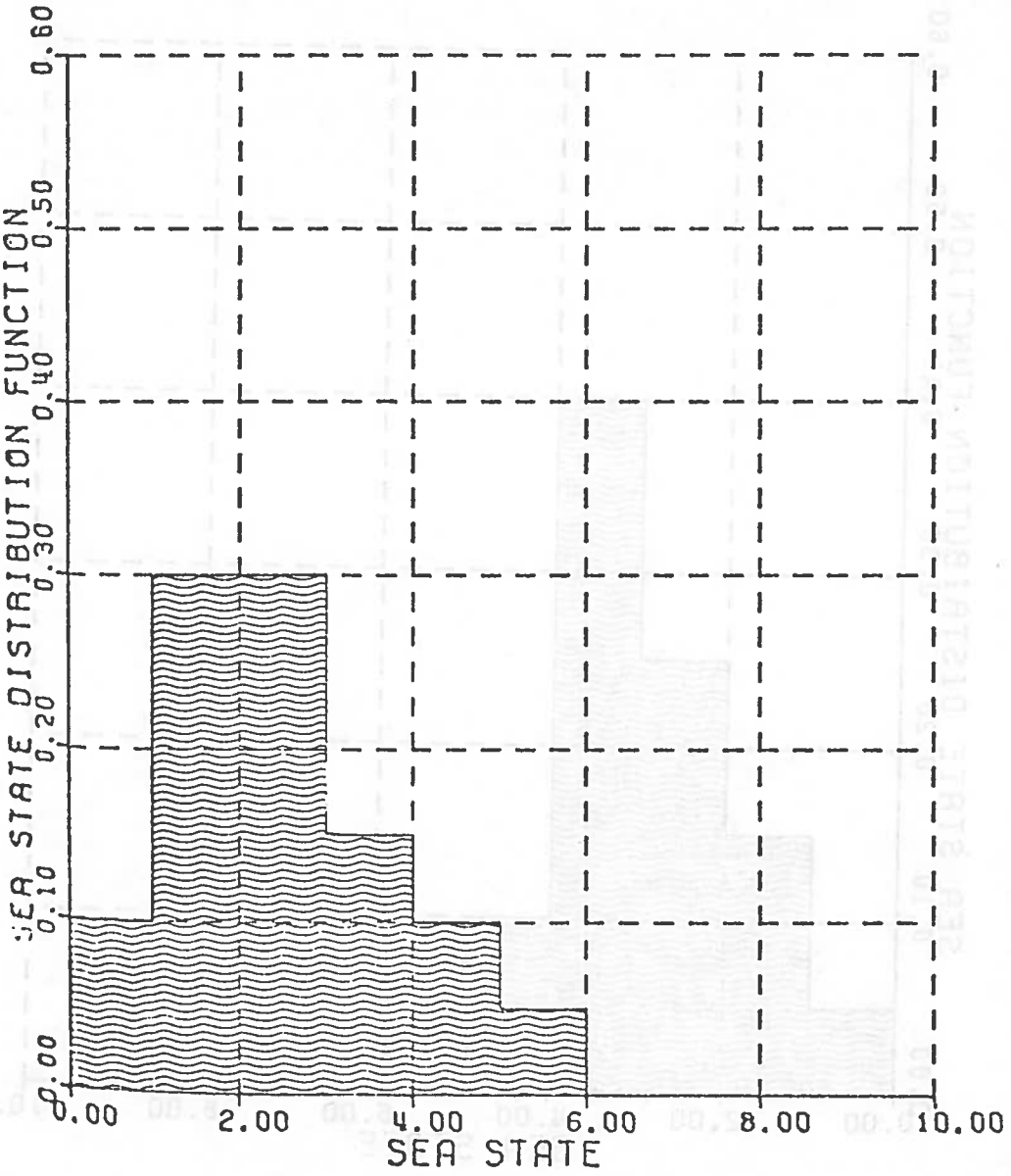


FIGURE B-5. SEA-STATE DISTRIBUTION NUMBER 5--AVERAGE SS=2.5

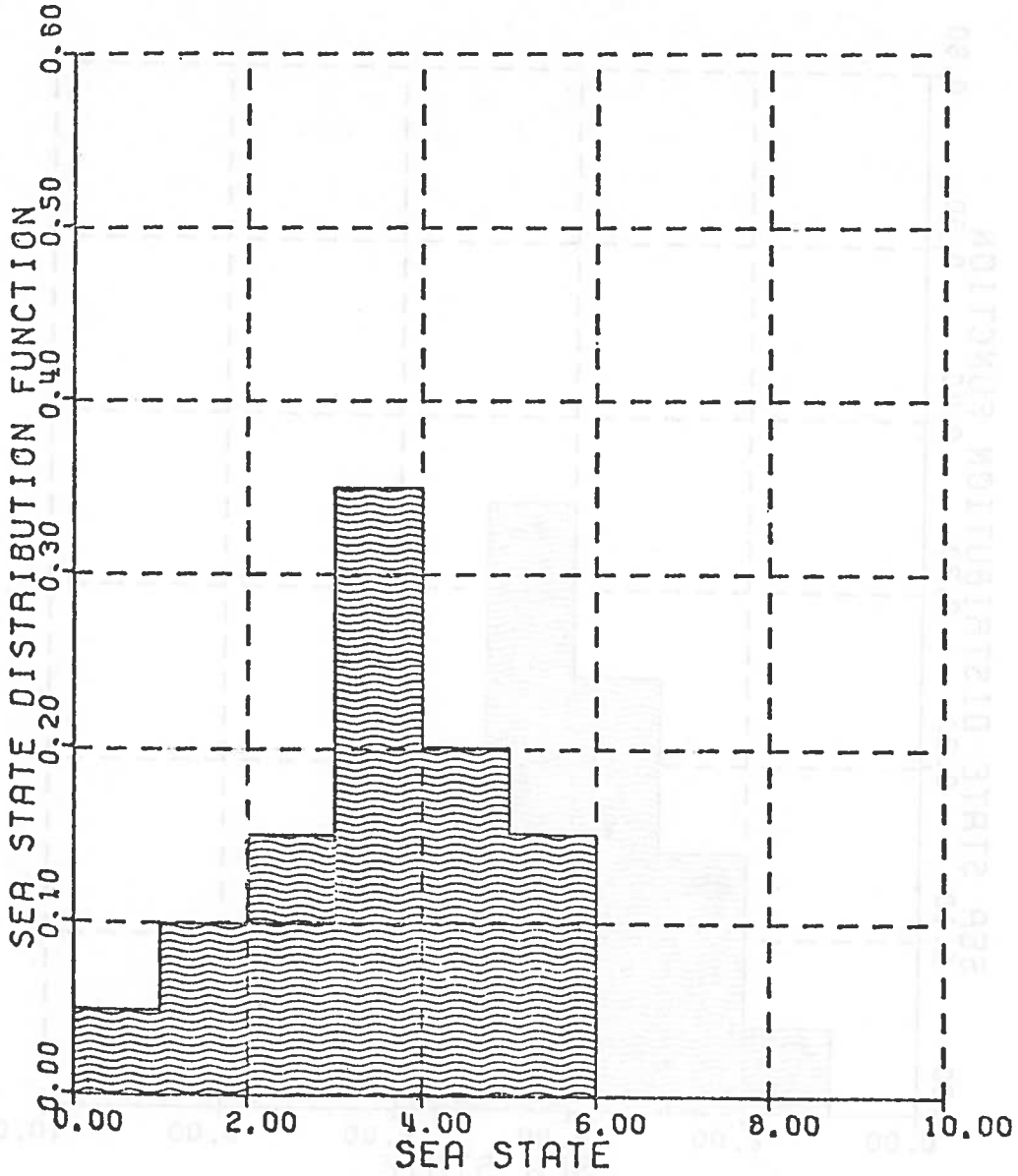


FIGURE B-7. SEA-STATE DISTRIBUTION NUMBER 7--AVERAGE SS=3.5

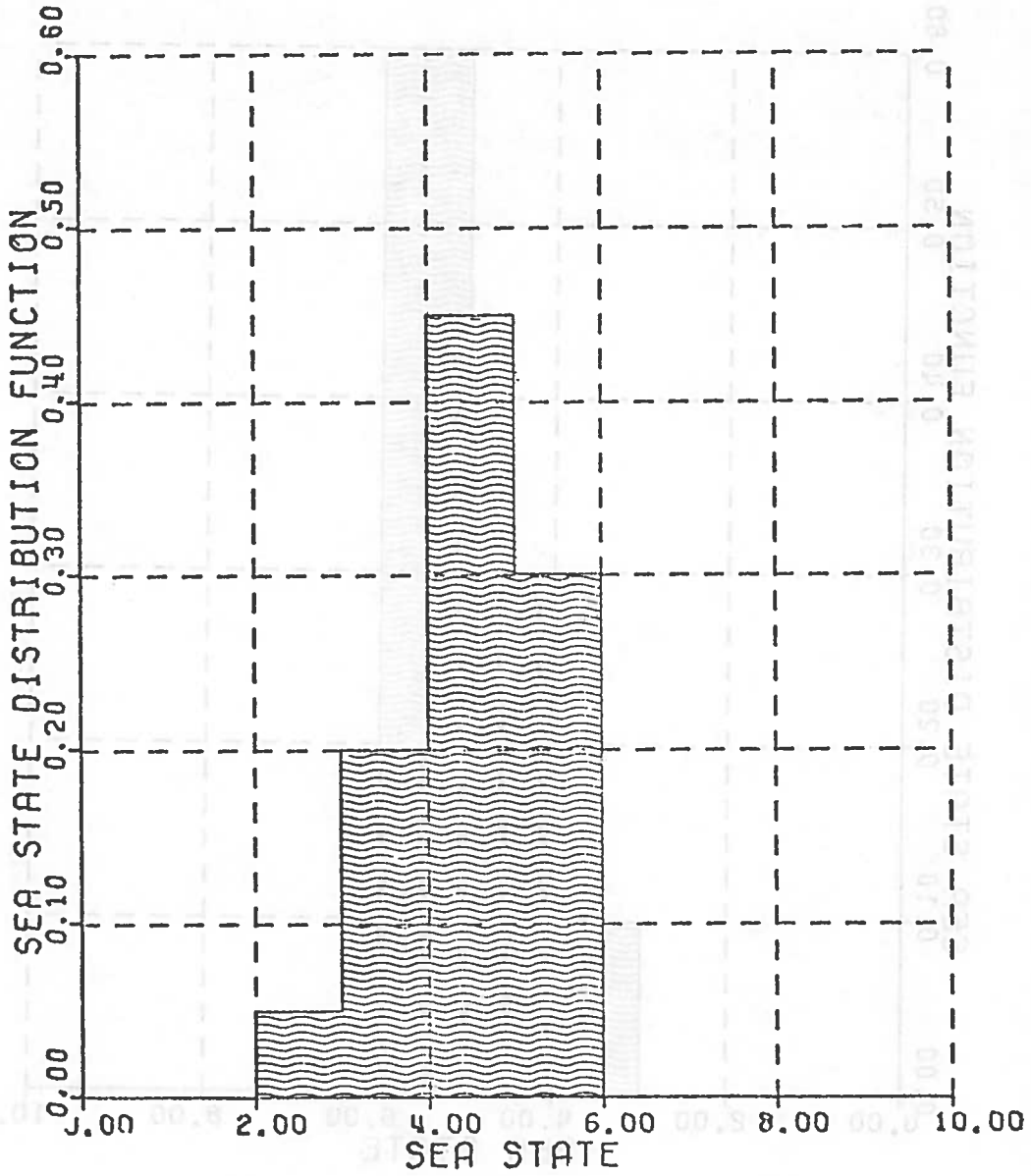
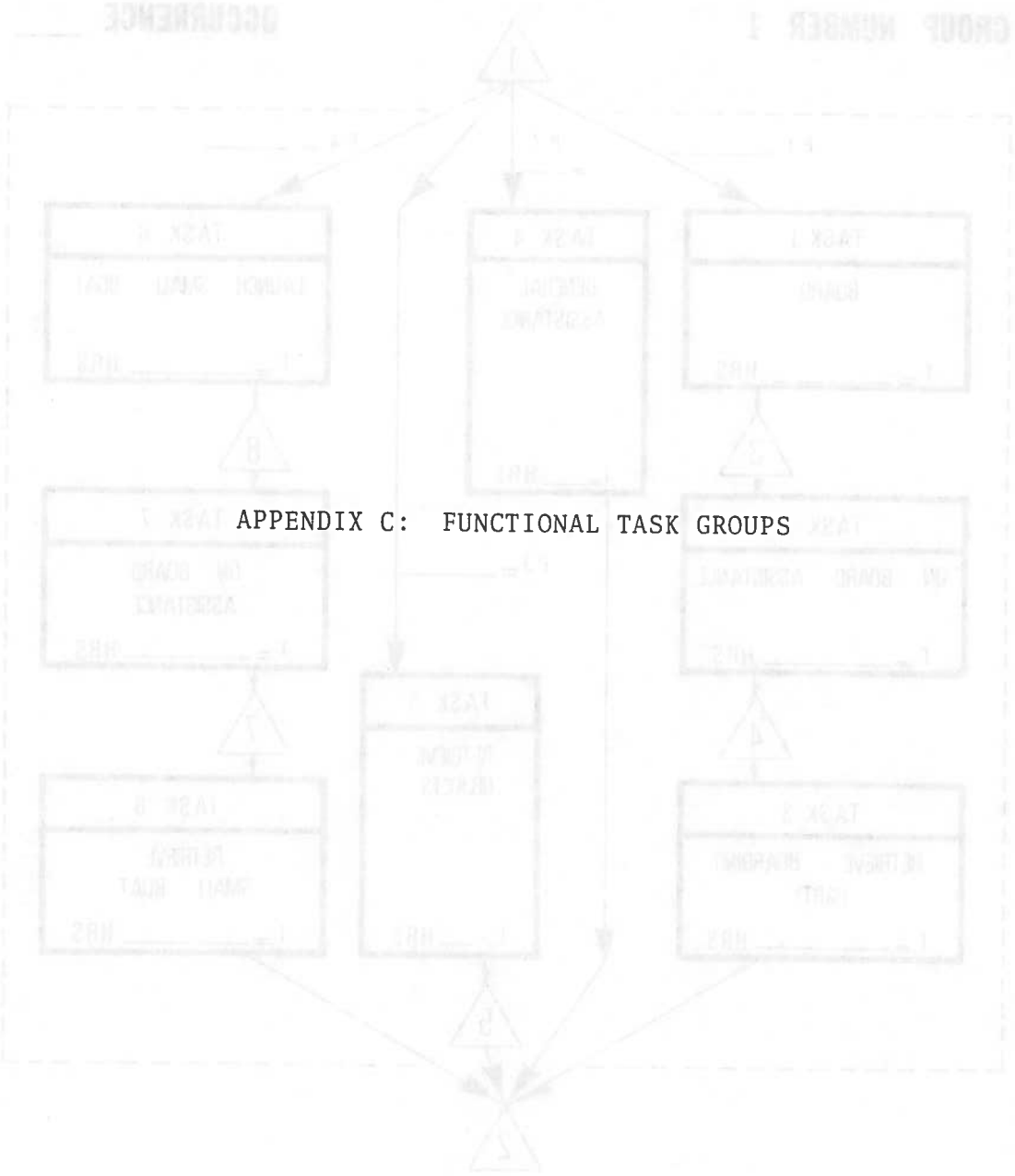
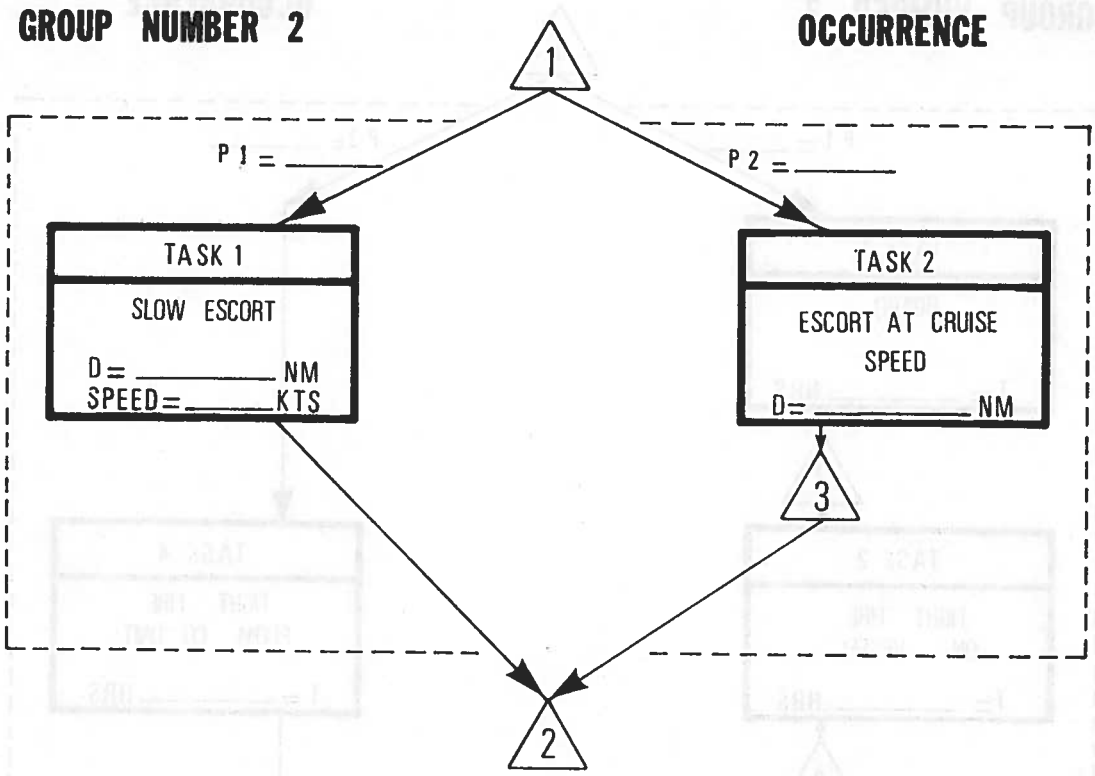


FIGURE B-9. SEA-STATE DISTRIBUTION NUMBER 9--AVERAGE SS-4.5



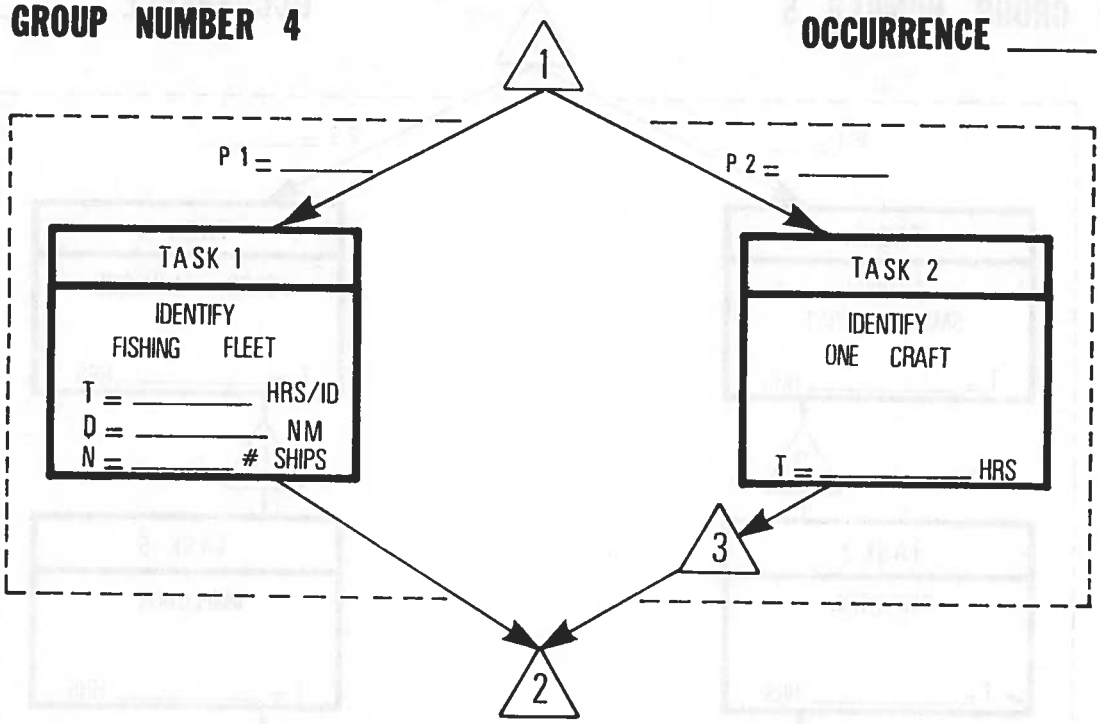
APPENDIX C: FUNCTIONAL TASK GROUPS

ESCORT GROUP



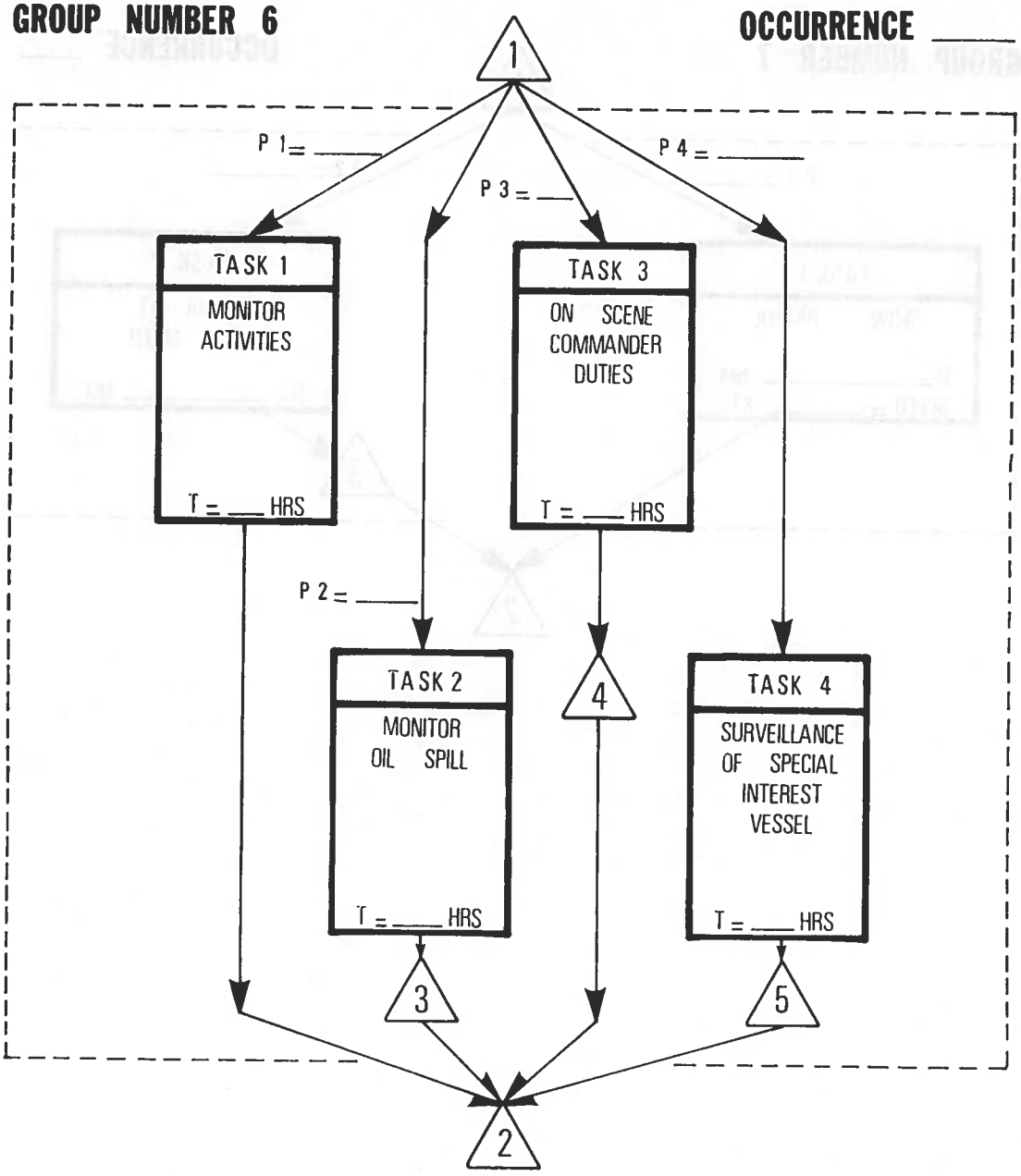
IDENTIFY GROUP

GROUP NUMBER 4 **OCCURRENCE ____**



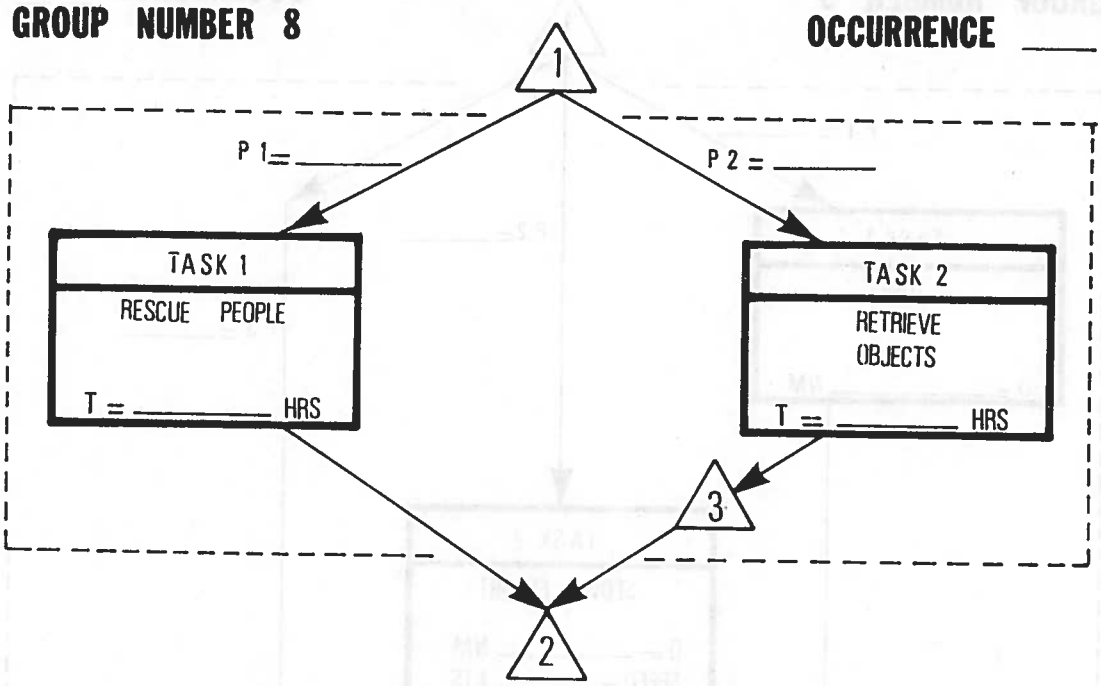
MONITOR GROUP

GROUP NUMBER 6 **OCCURRENCE _____**



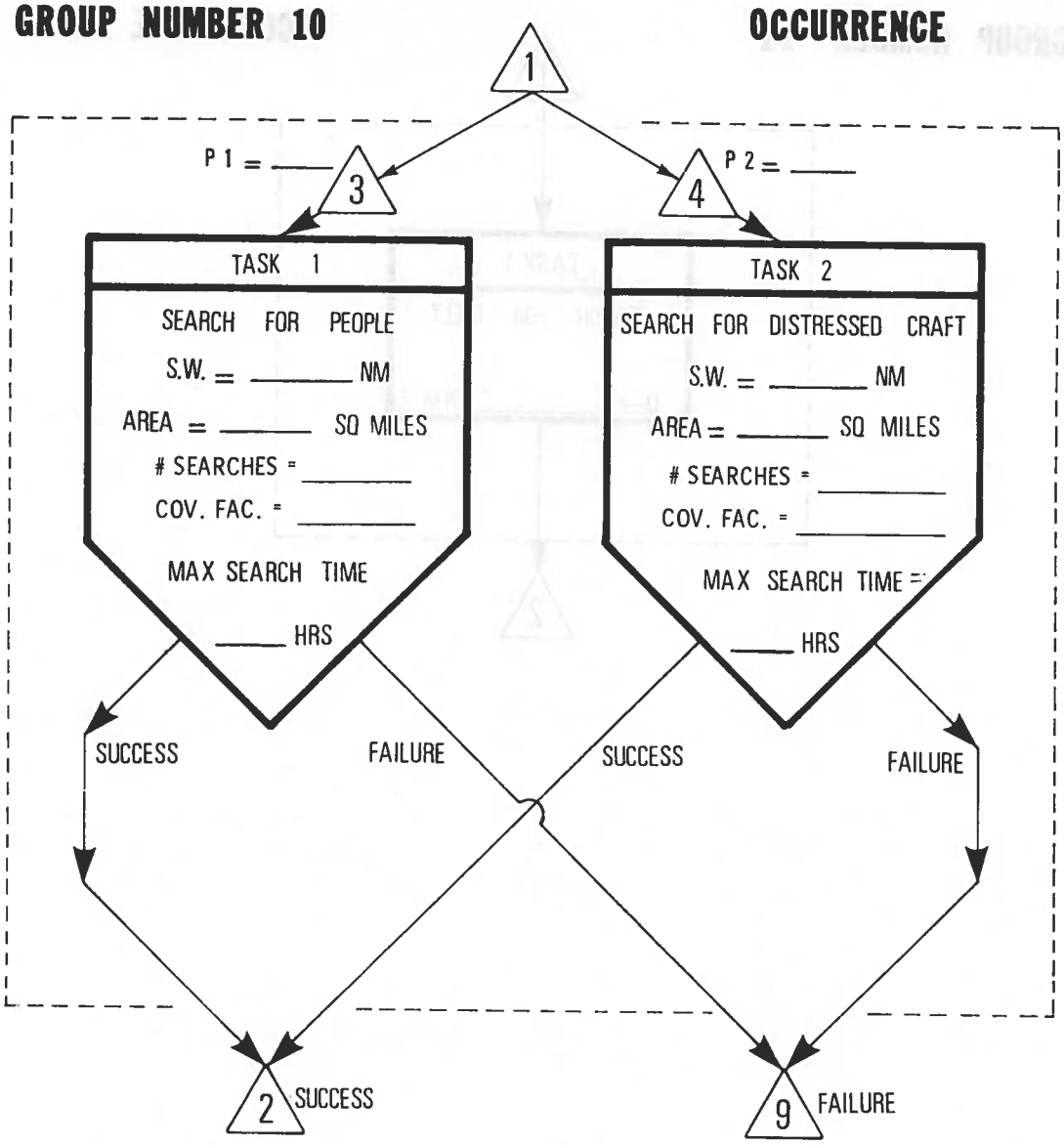
RESCUE GROUP

GROUP NUMBER 8 **OCCURRENCE ____**



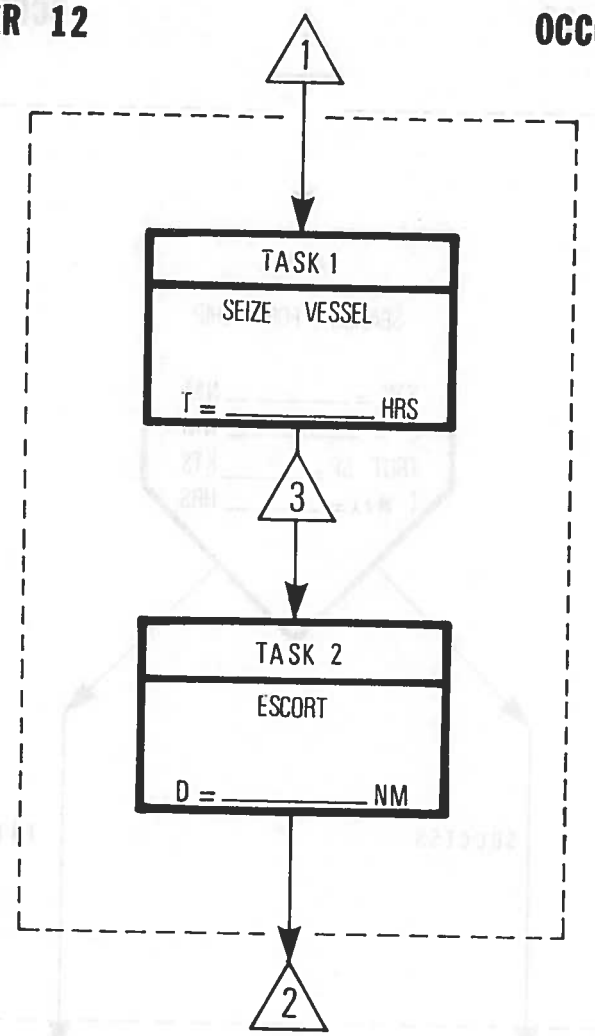
SAR SEARCH GROUP

GROUP NUMBER 10 **OCCURRENCE**



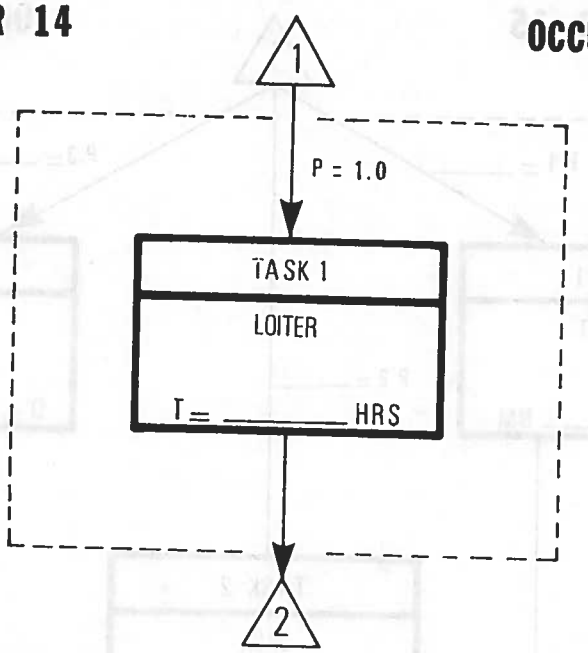
SEIZE GROUP

GROUP NUMBER 12 **OCCURRENCE _____**



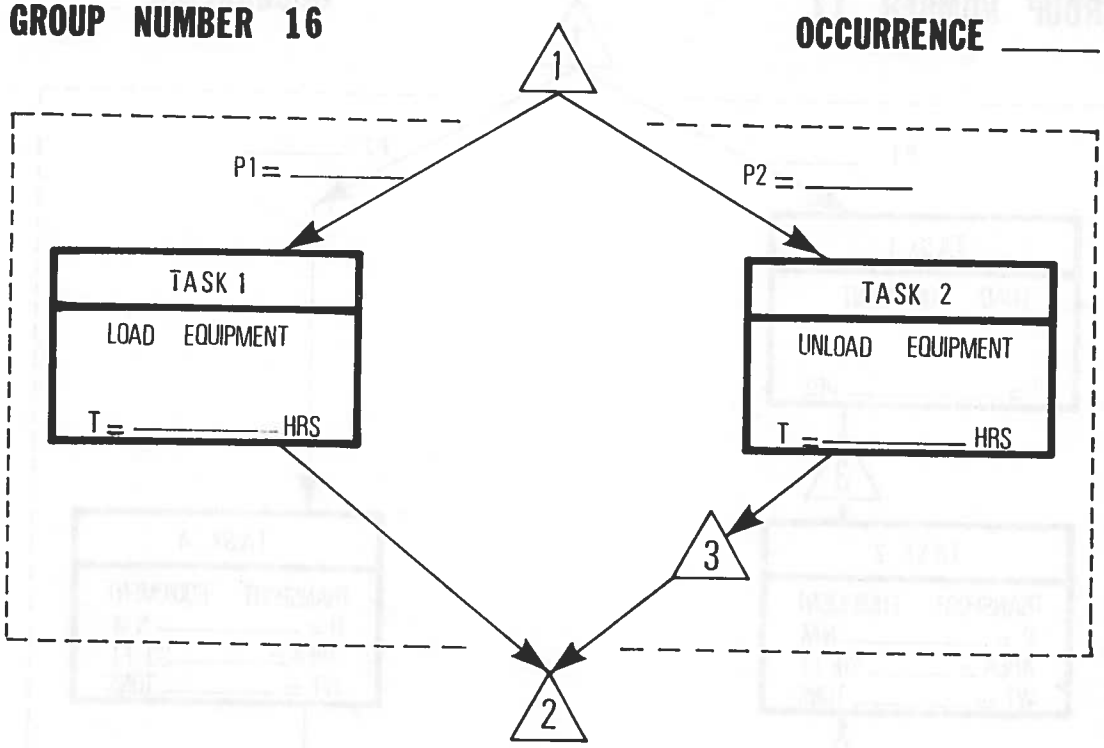
STANDBY GROUP

GROUP NUMBER 14 **OCCURRENCE _____**



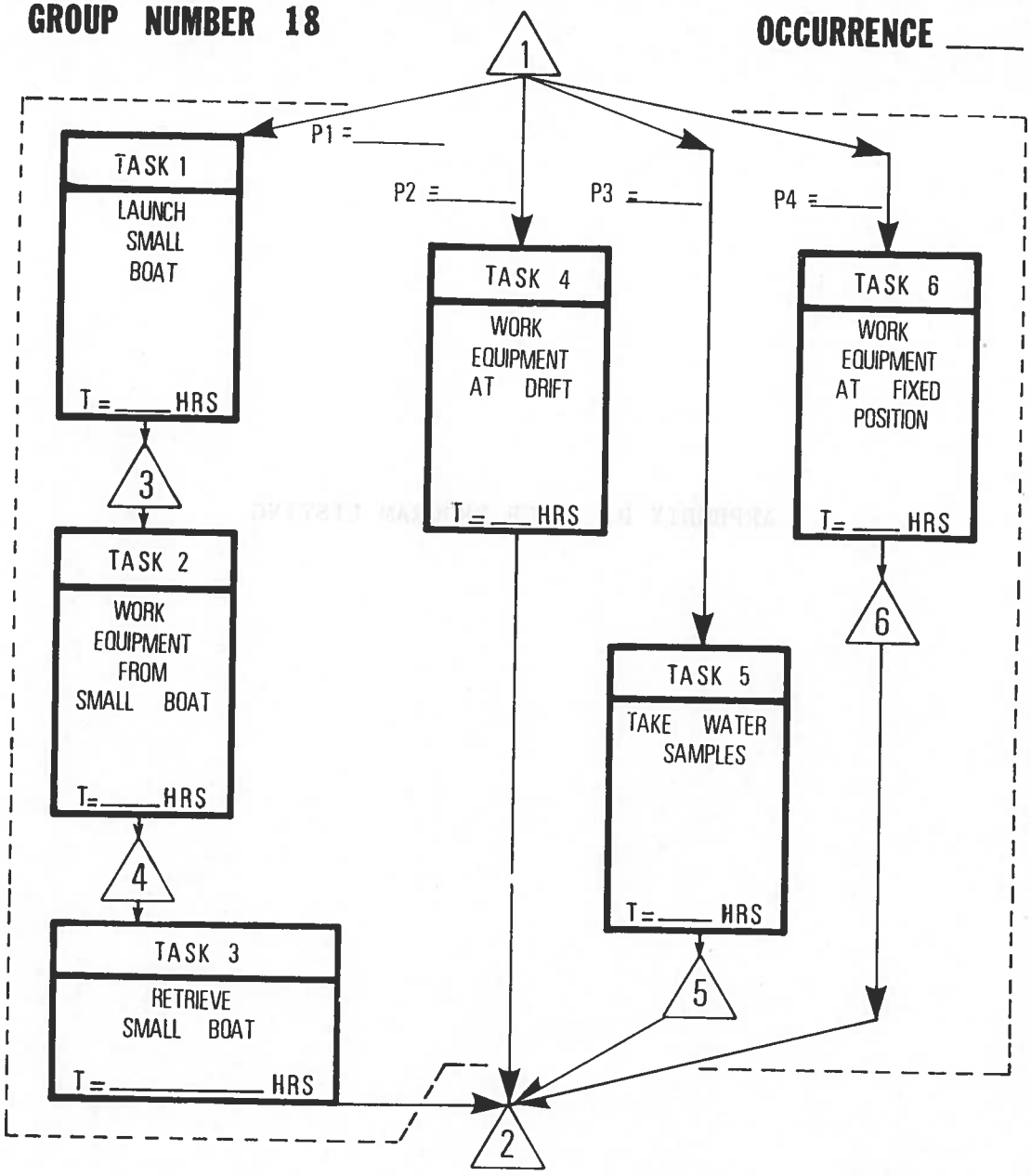
TRANSFER EQUIPMENT GROUP

GROUP NUMBER 16 **OCCURRENCE _____**



WORK EQUIPMENT GROUP

GROUP NUMBER 18 **OCCURRENCE ____**



```
C                                                                       00000001
C                                                                       00000002
C CRFE PROGRAM                                                          00000003
C                                                                       00000004
C                                                                       00000020
C CUTTER RESOURCE EFFECTIVENESS EVALUATION  (CRFE)                      00000021
C                                                                       00000022
C                                                                       00000023
C MAIN PROGRAM - READS CRAFT INPUT DATA AND CALLS SUBROUTINES           00000024
C                TO COMPUTE CRAFT CHARACTERISTICS,CRAFT                 00000025
C                PARAMETERS,TASK PROBABILITIES OF SUCCESS               00000026
C                AND SCENARIOS                                          00000027
C                                                                       00000050
C INPUTS ARE: 1. CRAFT TYPE                                             00000060
C             2. CRAFT DISPLACEMENT (IN TONS)                           00000070
C                  OR CRAFT LENGTH (IN FEET)                            00000080
C             3. DESIGN SPEED (IN KNOTS)                                00000090
C             4. FUEL FRACTION = OF TOTAL PAYLOAD                       00000100
C             5. VISIBILITY,TOW,DEPTH,AND SEA STATE DISTRIBUTION        00000110
C                  NUMBERS                                              00000120
C                                                                       00000130
C CRAFT AND ENGINE ARE IDENTIFIED BY CODES, AS FOLLOWS:                 00000140
C       CRAFT CODES:                                                    00000150
C         10. HYDROFOIL-SUBMERGED FOIL                                  00000160
C         11. HYDROFOIL-SURFACE PIERCING                                00000170
C         20. AIR CUSHION VEHICLE - LOW P/L                             00000180
C         21. AIR CUSHION VEHICLE - HIGH P/L                            00000190
C         30. SURFACE EFFECT SHIP                                       00000200
C         40. PLANING CRAFT                                             00000210
C         50. CATAMARAN                                                 00000220
C         60. SWATH                                                     00000230
C         70. HYBRID VESSEL                                             00000240
C         80. CONVENTIONAL CRAFT                                        00000250
C        101. MRB                                                       00000260
C        102. PWB 32                                                    00000270
C        103. UTB 41                                                    00000280
C        104. MLB 44                                                    00000290
C        105. MLB 52                                                    00000300
C        106. ANB 55                                                    00000310
C        107. ANB 63                                                    00000320
C        108. WPB 82                                                    00000330
C        109. WPB 95                                                    00000340
C        110. WMEC 210                                                  00000350
C        111. WMEC 270                                                  00000360
C        112. WHEC 378                                                  00000370
C                                                                       00000380
C       ENGINE CODES:                                                   00000390
C         1. GAS TURBINE                                                00000400
C         2. DIESEL                                                     00000410
C                                                                       00000420
C                                                                       00000430
C                                                                       00000440
C                                                                       00000450
      IMPLICIT REAL(A-Z)                                                00000460
      INTEGER I,J,K,INFILE,CASNUM,IDISP,IDSPD                           00000470
      INTEGER SS1,SS2                                                   00000480
      INTEGER TYPE,TYPE1,ENG                                            00000490
      INTEGER RATE                                                      00000500
      INTEGER VISDTB,TOWDTB,DPHDTB,SSPDTB                               00000510
C                                                                       00000520
      DATA INFILE/14/                                                   00000530
      DIMENSION SSPRBD(8)                                               00000540
                                                                        00000570
```

```
      COMMON/CSSPRB/SSPROB                                                      00001550
      DATA SSPROB/1.,7*0.,.55,.40,.05,5*0.,.20,.60,.15,.05,4*0.,               00001560
     1 .20,.30,.35,.10,.05,3*0.,.10,.30,.30,.15,.10,.05,2*0.,                  00001570
     2 .05,.15,.25,.40,.10,.05,2*0.,.05,.10,.15,.35,.20,.15,2*0.,              00001580
     3 0.,.05,.15,.25,.35,.20,2*0.,2*0.,.05,.20,.45,.30,2*0.,                  00001590
     4 3*0.,.10,.30,.60,2*0./                                                  00001600
      END                                                                       00001610
C                                                                             00000010
C                                                                             00000020
C                                                                             00000030
C                                                                             00000040
C                                                                             00000050
C                                                                             00000060
C*************************** S C H A R ****************************          00000070
C                                                                             00000080
C SCHAR SUBROUTINE                                                            00000090
C                                                                             00000100
C COMPUTES CRAFT CHARACTERISTICS                                              00000110
C                                                                             00000120
C                                                                             00000130
C                                                                             00000140
      SUBROUTINE SCHAR(TYPE,DISP,LENG,DSPEED,FUFRAC)                          00000150
      IMPLICIT REAL(A-Z)                                                      00000160
      INTEGER I,J,K                                                           00000170
C                                                                             00000180
      INTEGER TYPE,ENG                                                        00000190
      INTEGER OUTFIL,RATE,ISURVV,MPRNTD                                       00000200
      INTEGER TYPLST,$ENG,ENG,IDISP,ILENG,IDSPD,SS1,SS2,SSPDTB                00000210
      INTEGER LENGN,TYPLST                                                    00000220
      INTEGER TYPNUM                                                          00000230
      INTEGER ASURVI,AENG,CGTYPE                                              00000240
      DIMENSION CWSPD(4),SFCENG(4),SFCCF(4),TOTSFC(4),SFCGAL(4)               00000250
      DIMENSION HPUTIL(4),FUELRT(4),ENDUR(4),RANGE(4)                         00000260
      DIMENSION FUELR2(4),ENG(4),SSPROB(8,10),SSPRBD(8)                       00000270
      DIMENSION TNRAD(4),MOTION(4)                                            00000280
      COMMON/CHAR/LTOB,BEAM,DTOL,DRAF,SSPRBD,                                 00000290
     1 DECK,USELD,FUELCP,CARGCP,TOWDSP,                                       00000300
     2 SURVIV,HPINST,HPPTON,HPTNKT,CWSPD,ENG,SFCENG,SFCCF,TOTSFC,SFCGAL,      00000310
     3 HPUTIL,FUELRT,FUELR2,ENDUR,RANGE,MOTION,TNRAD,SSPDTB                   00000320
      COMMON/CSSPRB/SSPROB                                                    00000330
      DIMENSION TYPLST(9)                                                     00000340
      DIMENSION CRFNM(8,10)                                                   00000350
      DATA CRFNM/                                                             00000360
     1 'HYDR','OFOI','L-SU','BMER','GED ','FOIL','    ','    ','    ',      00000370
     2 'HYDR','OFOI','L-SU','RFAC','E PI','ERCI','NG  ','    ','    ',      00000380
     3 'AIR ','CUSH','ION ','VEHI','CLE-',' LOW',' P/L','    ','    ',      00000390
     4 'AIR ','CUSH','ION ','VEHI','CLE-','HIGH',' P/L','    ','    ',      00000400
     5 'SURF','ACE ','EFFE','CT S','HIP ','    ','    ','    ','    ',      00000410
     6 'PLAN','ING ','CRAF','T   ','    ','    ','    ','    ','    ',      00000420
     7 'CATA','MARA','N   ','    ','    ','    ','    ','    ','    ',      00000430
     8 'SWAT','H   ','    ','    ','    ','    ','    ','    ','    ',      00000440
     9 'HYBR','ID V','ESSE','L   ','    ','    ','    ','    ','    ',      00000450
     1 'CONV','ENTI','ONAL',' CRA','FT  ','    ','    ','    ','    '/      00000460
      DIMENSION CGCRNM(2,12)                                                  00000470
      DATA CGCRNM/'MRB ','    ','PWB ','32  ','UTB ','41  ',                 00000480
     1'MLB ','44  ','MLB ','52  ','ANB ','55  ','ANB ','63  ',               00000490
     1'WPB ','82  ','WPB ','95  ','WMEC','210 ',                             00000500
     1'WMEC','270 ','WHEC','378 '/                                           00000510
      DIMENSION ENGNAM(2)                                                     00000520
      DATA ENGNAM/'(GT)','(DE)'/                                              00000530
      DIMENSION ENGNM(3,2)                                                    00000540
      DATA ENGNM/                                                             00000550
                                                                              00000560
```

```
2001  CONTINUE                                                          00001200
      HPTNKT=HPPTON/CWSPD(1)                                            00001210
      IF(TYPE.LT.100)GOTO 1008                                          00001220
C                                                                       00001230
C GET CHARACTERISTICS OF EXISTING COAST GUARD CRAFT                     00001240
C                                                                       00001250
      DIMENSION ALENG(12),ADISP(12),ADSPEE(12),AFUERA(12),ALTOB(12)     00001260
      DIMENSION ABEAM(12),ADTOL(12),ADRAF(12)                           00001270
      DIMENSION ADECK(12),AUSELD(12),AFUELC(12),ACARGC(12)              00001280
      DIMENSION ATOWDS(12),ASURVI(12),AHPINS(12),AHPPTO(12)             00001290
      DIMENSION AHPTNK(12),ACWSPD(4,12),AENG(4,12),AFUERT(4,12)         00001300
      DIMENSION AFUER2(4,12),AENDUR(4,12),ARANGE(4,12)                  00001310
      DIMENSION AHPUTI(4,12)                                            00001320
      DATA ALENG/26.,32.,41.,44.,52.,55.,63.,82.,95.,                   00001330
     1210.,270.,378./                                                   00001340
      DATA ADISP/4.,8.5,15.,18.5,35.,34.,42.,67.,100.,1000.,            00001350
     11780.,3000./                                                      00001360
      DATA ADSPEE/25.,25.,26.,14.,11.,22.,15.,23.5,20.,17.,             00001370
     119.5,28./                                                         00001380
      DATA AFUERA/.375,.429,.556,.333,.432,.6,.304,.25,.273,            00001390
     1.697,.913,.829/                                                   00001400
      DATA ALTOB/3.25,2.66,3.03,3.38,3.58,3.33,3.39,4.55,               00001410
     14.75,6.17,7.10,9./                                                00001420
      DATA ABEAM/8.,12.,13.5,13.,14.5,16.5,18.5,18.,20.,34.,            00001430
     138.,42./                                                          00001440
      DATA ADTOL/.077,.156,.098,.068,.115,.091,.071,.073,.063,          00001450
     1.048,.052,.056/                                                   00001460
      DATA ADRAF/2.,5.,4.,3.,6.,5.,4.5,6.,6.,10.,14.,21./               00001470
      DATA ADECK/5.,100.,200.,50.,100.,250.,375.,200.,400.,             00001480
     11500.,2500.,2500./                                                00001490
      DATA AUSELD/.8,1.75,4.5,3.,8.8,5.,11.5,4.,5.5,33.,                00001500
     1345.,889./                                                        00001510
      DATA AFUELC/.3, .54,1.25,1.,3.14,3.28,3.28,5.67,8.96,23.,         00001520
     1315.,839./                                                        00001530
      DATA ACARGC/.5,1.,2.,2.,5.,2.,8.,3.,4.,10.,30.,50./               00001540
      DATA ATOWDS/20.,100.,150.,200.,400.,340.,420.,1000.,              00001550
     12000.,10000.,20000.,30000./                                       00001560
      DATA ASURVI/4,3,4,5,6,4,5,6,6,7,8,8/                              00001570
      DATA AHPINS/300.,390.,640.,400.,400.,1090.,800.,                  00001580
     11600.,2324.,5000.,7000.,36000./                                   00001590
      DATA AHPPTO/75.,45.8,42.6,21.6,11.4,32.,19.,23.8,                 00001600
     123.2,5.,3.93,12./                                                 00001610
      DATA AHPTNK/3.,1.83,1.64,1.54,1.03,1.45,1.27,1.01,                00001620
     II.15,.29,.20,.42/                                                 00001630
      DATA ACWSPD/25.,17.5,12.,5.,25.,18.,12.,5.,26.,18.,12.,5.,        00001640
     114.,12.,12.,5.,11.,11.,11.,5.,22.,18.,12.,5.,15.,12.,12.,5.,      00001650
     123.5,17.,12.,5.,20.,16.,12.,5.,16.,14.,12.,5.,19.5,               00001660
     115.,12.,5.,28.,16.,12.,5./                                        00001670
      DATA AENG/2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,               00001680
     12,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,                 00001690
     11,2,2,2/                                                          00001700
      DATA AHPUTI/300.,120.,50.,30.,390.,160.,80.,40.,640.,             00001710
     1224.,138.,64.,400.,220.,220.,40.,400.,400.,400.,                  00001720
     180.,1090.,545.,245.,109.,800.,400.,400.,100.,                     00001730
     11600.,640.,320.,160.,2324.,1662.,581.,232.,5000.,                 00001740
     13000.,2000.,500.,7000.,3000.,1750.,700.,36000.,                   00001750
     17000.,2800.,700./                                                 00001760
      DATA AFUERT/25.,10.,6.,3.,42.5,28.5,12.,2.,72.8,                  00001770
     140.,20.4,5.,30.8,25.,25.,10.,23.4,24.4,23.4,10.,77.,              00001780
     150.8,24.,5.,56.,37.,37.,5.,96.1,54.4,30.,7.70,                    00001790
     1130.,88.7,36.,7.5,120.,100.,80.,47.,380.,153.,100.,               00001800
     162.9,3000.,400.,250.,150./                                        00001810
      DATA AFUER2/1.,.57,.50,.60,1.70,1.58,1.,.4,2.8,2.2,               00001820
```

```
3000  FORMAT('1'/15X,'C R A F T  C H A R A C T E R I S T I C S'                00002980
     1,3(/),18X,'CRAFT TYPE',9X,8A4)                                          00002990
      IF(TYPE.GE.100)WRITE(OUTFIL,3025)(CGCRNM(I,CGTYPE),I=1,2)               00003000
3025  FORMAT('1'/15X,'C R A F T  C H A R A C T E R I S T I C S'                00003010
     1 ,3(/),18X,'CRAFT TYPE',9X,'COAST GUARD ',2A4)                          00003020
      WRITE(OUTFIL,3026)IDISP,ILENG                                           00003030
3026  FORMAT(18X,'DISPLACEMENT',4X,I6,1X,'TONS'/18X,'LENGTH',                 00003040
     110X,I6,1X,'FEET')                                                       00003050
      WRITE(OUTFIL,3022)IDSPD,FUFRAC                                          00003060
3022  FORMAT(18X,'DESIGN SPEED',4X,I6,' KNOTS'/                               00003070
     1    18X,'FUEL FRACTION',3X,F7.2)                                       00003080
      WRITE(OUTFIL,3001)LENG,BEAM,DRAF                                        00003090
3001  FORMAT(4(/),13X,'LENGTH',22X,F8.1,2X,'FEET'/13X,'BEAM',                 00003100
     1 24X,F8.1,2X,'FEET'/13X,'DRAFT',23X,F8.1,2X,'FFET')                     00003110
      WRITE(OUTFIL,3002)LTOB,DTOL,DISP                                        00003120
 3002 FORMAT(13X,'LENGTH/BEAM RATIO',12X,F8.2                                 00003130
     1/13X,'DRAFT/LENGTH RATIO',11X,F8.2                                      00003140
     4 /13X,'DISPLACEMENT',16X,F8.1,2X,'TONS')                                00003150
      WRITE(OUTFIL,3003)ISURVV,TOWDSP,DECK,CARGCP,FUFLCP,USFLD                00003160
3003  FORMAT (13X,'SURVIVABILITY',14X,I7,4X,'SEA STATE'                       00003170
     1 /13X,'TOWS VESSELS UP TO',10X,F7.0,3X,'TONS'                           00003180
     2 /13X,'USEABLE DECK AREA',11X,F7.0,3X,'SQUARE FEET'                     00003190
     4 /13X,'CARGO CAPACITY',14X,F8.1,2X,'TONS'                               00003200
     5 /13X,'FUEL CAPACITY',15X,F8.1,2X,'TONS'                                00003210
     3 /13X,'USEFUL PAYLOAD',14X,F8.1,2X,'TONS')                              00003220
      WRITE (OUTFIL,3004)HPTNST,HPPTON,HPTNKT,RANGE(2),ENDUR(2)               00003230
3004  FORMAT(13X,'INSTALLED POWER',13X,F7.0,3X,'HORSEPOWER'                   00003240
     1 /13X,'POWER TO WEIGHT',13X,F8.1,2X,'HP/TON'                            00003250
     2 /13X,'TRANSPORT EFFICIENCY',9X,F8.2,1X,'HP/TON-KNOT'                   00003260
     4 /13X,'RANGE AT CRUISE SPEED',7X,F7.0,3X,'NAUTICAL MILES'               00003270
     5 /13X,'ENDURANCE AT CRUISE SPEED',3X,F8.1,2X,'HOURS')                   00003280
      WRITE(OUTFIL,3010)                                                      00003290
3010  FORMAT(5(/),27X,'  FLANK  ',1X,'CRUISE ',1X,'REDUCED',                  00003300
     1 1X,'  ON  '/29X,'SPEED',3X,'SPEED',3X,' SPEED ',1X,' SCENE  '/)        00003310
      WRITE(OUTFIL,3023)(ENGNAM(ENG(RATE)),RATE=1,4)                          00003320
3023  FORMAT(10X,'ENGINE TYPE',5X,4(4X,A4)/)                                  00003330
      WRITE(OUTFIL,3011)CWSPD                                                 00003340
 3011 FORMAT(10X,'CALM WATER SPEED',4F8.1,4X,'KNOTS')                         00003350
      WRITE(OUTFIL,3012)TOTSFC,SFCGAL                                         00003360
3012  FORMAT(10X,'SFC (WEIGHT) ',4X,4F8.2,3X,'LBS/HP-HR'                      00003370
     1 /10X,'SFC (VOLUME) ',4X,4F8.2,3X,'GAL/HP-HR')                          00003380
      WRITE(OUTFIL,3013)HPUTIL,FUELRT                                         00003390
3013  FORMAT(10X,'HP UTILIZED',5X,4F8.1,4X,'HP'                               00003400
     1 /10X,'FUEL CONSUMPTION',4F8.1,4X,'GAL/HR')                             00003410
      WRITE(OUTFIL,3014)FUELR2,ENDUR                                          00003420
3014  FORMAT(10X,'FUEL CONSUMPTION',4F8.1,4X,'GAL/NAUT MI'                    00003430
     1 /10X,'ENDURANCE (FUEL)',4F8.1,4X,'HOURS')                              00003440
      WRITE(OUTFIL,3015)RANGE                                                 00003450
3015  FORMAT(10X,'RANGE',11X,4F8.1,4X,'NAUTICAL MI')                          00003460
      WRITE(OUTFIL,3031) TNRAD,MOTION                                         00003470
 3031 FORMAT(10X,'TURNING RADIUS',2X,4F8.1,4X,'YARDS'                         00003480
     1/10X,'CRAFT MOTION',4X,4F8.1,4X,'G')                                    00003490
C                                                                            00003500
2021  CONTINUE                                                               00003510
C                                                                            00003520
      RETURN                                                                 00003530
      END                                                                    00003540
C                                                                            00003550
C                                                                            00003560
C $LENG                                                                      00003570
C                                                                            00003580
C LENGTH(IN FEET)                                                            00003590
C                                                                            00003600
```

```
      FUNCTION $DTOL(TYPE,LENG)                                          00004240
      REAL LENG                                                          00004250
      INTEGER TYPE                                                       00004260
      IF(TYPE.EQ.10)$DTOL=.20                                            00004270
      IF(TYPE.EQ.11)$DTOL=.15                                            00004280
      IF(TYPE.EQ.20.OR.TYPE.EQ.21)$DTOL=0.01                             00004290
      IF(TYPE.EQ.30)$DTOL=.05                                            00004300
      IF(TYPE.EQ.40)$DTOL=.06                                            00004310
      IF(TYPE.EQ.50)$DTOL=.05                                            00004320
      IF(TYPE.EQ.60)$DTOL=.10                                            00004330
      IF(TYPE.EQ.70)$DTOL=.06                                            00004340
      IF(TYPE.EQ.80)$DTOL=.06                                            00004350
      RETURN                                                             00004360
      END                                                                00004370
C                                                                        00004380
C                                                                        00004390
C $DECK                                                                  00004400
C                                                                        00004410
C USEABLE DECK AREA IN SQUARE FEET                                       00004420
C                                                                        00004430
      FUNCTION $DECK(TYPE,LENG,BEAM)                                     00004440
      REAL LENG                                                          00004450
      INTEGER TYPE                                                       00004460
      IF(TYPE.EQ.20.OR.TYPE.EQ.21)DA=.50                                 00004470
      IF(TYPE.EQ.10.OR.TYPE.EQ.11.OR.TYPE.EQ.40)DA=.25                   00004480
      IF(TYPE.EQ.30)DA=.75                                               00004490
      IF(TYPE.EQ.50)DA=.40                                               00004500
      IF(TYPE.EQ.60)DA=.55                                               00004510
      IF(TYPE.EQ.70)DA=.30                                               00004520
      IF(TYPE.EQ.80)DA=.25                                               00004530
      $DECK=(LENG*BEAM)*DA                                               00004540
      RETURN                                                             00004550
      END                                                                00004560
C                                                                        00004570
C                                                                        00004580
C $USELD                                                                 00004590
C                                                                        00004600
C TOTAL USEFUL USFLOAD (TONS)                                            00004610
C                                                                        00004620
      FUNCTION $USELD(TYPE,DISP)                                         00004630
      INTEGER TYPE                                                       00004640
      IF(TYPE.EQ.10.)$USELD=$$(DISP,20.,8.,400.,122.)                    00004650
      IF(TYPE.EQ.11.)$USELD=$$(DISP,50.,12.,335.,98.)                    00004660
      IF(TYPE.EQ.20.)$USELD=$$(DISP,15.,6.,200.,68.)                     00004670
      IF(TYPE.EQ.21.)$USELD=$$(DISP,20.,8.,200.,88.)                     00004680
      IF(TYPE.EQ.30.)$USELD=$$(DISP,90.,35.,180.,70.)                    00004690
      IF(TYPE.EQ.40.)$USELD=.525*DISP-7.5                                00004700
      IF(TYPE.EQ.50.)$USELD=$$(DISP,20.,8.,200.,60.)                     00004710
      IF(TYPE.EQ.60.)$USELD=$$(DISP,700.,200.,4250.,1250.)               00004720
      IF(TYPE.EQ.70.)$USELD=$$(DISP,40.,12.,400.,100.)                   00004730
      IF(TYPE.EQ.80.)$USELD=$$LGLG(DISP,3.5,1.,3000.,1000.)              00004740
      RETURN                                                             00004750
      END                                                                00004760
C                                                                        00004770
C                                                                        00004780
C $HPBIN                                                                 00004790
C                                                                        00004800
C INSTALLED BASE HORSEPOWER                                              00004810
C   (FOR A CRAFT WITH DESIGN SPEED=BASE SPEED)                           00004820
C                                                                        00004830
      FUNCTION $HPBIN(TYPE,DISP)                                         00004840
      INTEGER TYPE                                                       00004850
      IF(TYPE.EQ.10)$HPBIN=$$(DISP,55.,2750.,300.,20000.)                00004860
```

```
      IF(TYPE.EQ.11)$BSSPD=40.                                          00005500
      IF(TYPE.EQ.20)$BSSPD=50.                                          00005510
      IF(TYPE.EQ.21)$BSSPD=50.                                          00005520
      IF(TYPE.EQ.30)$BSSPD=60.                                          00005530
      IF(TYPE.EQ.40)$BSSPD=45.                                          00005540
      IF(TYPE.EQ.50)$BSSPD=30.                                          00005550
      IF(TYPE.EQ.60)$BSSPD=20.                                          00005560
      IF(TYPE.EQ.70)$BSSPD=40.                                          00005570
      IF(TYPE.EQ.80)$BSSPD=25.                                          00005580
        RETURN                                                          00005590
        END                                                             00005600
C                                                                       00005610
C                                                                       00005620
C                                                                       00005630
C                                                                       00005640
C $CWSPD                                                                00005650
C                                                                       00005660
C CALM WATER SPEED AT GOOD VISIBILITY                                   00005670
C                                                                       00005680
      FUNCTION $CWSPD(TYPE,RATE,DSPEED)                                 00005690
      INTEGER TYPE,RATE                                                 00005700
      IF(RATE.EQ.1)$CWSPD=DSPEED                                        00005710
      IF(RATE.EQ.2)GOTO 2                                               00005720
      IF(RATE.EQ.3)$CWSPD=12.                                           00005730
      IF(RATE.EQ.4)$CWSPD=5.                                            00005740
      RETURN                                                            00005750
2     IF(TYPE.EQ.10)$CWSPD=.85*DSPEED                                   00005760
      IF(TYPE.EQ.11)$CWSPD=.9*DSPEED                                    00005770
      IF(TYPE.EQ.30.OR.TYPE.EQ.40.OR.TYPE.EQ.50                         00005780
     1     .OR.TYPE.EQ.70)$CWSPD=.875*DSPEED                            00005790
      IF(TYPE.EQ.20.OR.TYPE.EQ.21)$CWSPD=.85*DSPEED                     00005800
      IF(TYPE.EQ.60)$CWSPD=.60*DSPEED                                   00005810
      IF(TYPE.EQ.80)$CWSPD=.5*DSPEED                                    00005820
      RETURN                                                            00005830
      END                                                               00005840
C                                                                       00005850
C                                                                       00005860
C                                                                       00005870
C                                                                       00005880
C $TOWDS                                                                00005890
C                                                                       00005900
C TOW DISPLACEMENT CAPABILITY IN TONS                                   00005910
C                                                                       00005920
      FUNCTION $TOWDS(TYPE,DISP)                                        00005930
      INTEGER TYPE                                                      00005940
      F=10                                                              00005950
      IF(TYPE.EQ.20.OR.TYPE.EQ.21)F=2                                   00005960
      IF(TYPE.EQ.60)F=5                                                 00005970
      $TOWDS=F*DISP*(DISP/100.)**.3333                                  00005980
      RETURN                                                            00005990
      END                                                               00006000
C                                                                       00006010
C                                                                       00006020
C $SFCEN                                                                00006030
C                                                                       00006040
C SPECIFIC FUEL CONSUMPTION (LBS PER HORSEPOWER HOUR PER ENGINE)        00006050
C                                                                       00006060
      FUNCTION $SFCEN(ENG,HPINST)                                       00006070
      INTEGER ENG                                                       00006080
      HPINS2=HPINST/2.                                                  00006090
      IF(ENG.EQ.2)$SFCEN=.35                                            00006100
      IF(ENG.EQ.1)$SFCEN=$$3(HPINS2,400...7,4000...48,16000...40)       00006110
      RETURN                                                            00006120
```

```
      IF(PCTDSP.GT.60)HPPCT=2.*PCTDSP-100.                              00006750
      GOTO 99                                                           00006760
99    $HPFCT=HPPCT/100.                                                 00006770
      RETURN                                                            00006780
C                                                                       00006790
      END                                                               00006800
C                                                                       00006810
C                                                                       00006820
C $TNRAD                                                                00006830
C                                                                       00006840
      FUNCTION $TNRAD(TYPE,CWSPD)                                       00006850
C                                                                       00006860
      INTEGER TYPE,RATE                                                 00006870
      IF(TYPE.EQ.10) OMEGA = 8.                                         00006880
      IF(TYPE.EQ.20.OR.TYPE.EQ.21) OMEGA = 2.                           00006890
      IF(TYPE.EQ.30) OMEGA = 1.5                                        00006900
      IF(TYPE.EQ.40) OMEGA = 4.                                         00006910
      IF(TYPE.EQ.11.OR.TYPE.GE.50) OMEGA = 3.                           00006920
      $TNRAD = (1.689*CWSPD)/((3.14159265/180.)*OMEGA)                  00006930
      RETURN                                                            00006940
      END                                                               00006950
C                                                                       00006960
C                                                                       00006970
C                                                                       00006980
C $$                                                                    00006990
C                                                                       00007000
C FINDS Y VALUE ON A STRAIGHT LINE, GIVEN X VALUE AND TWO POINTS        00007010
C ON THE LINE (ASSUMING LINE EXTENDS INFINITELY)                        00007020
C                                                                       00007030
      FUNCTION $$(X,X1,Y1,X2,Y2)                                        00007040
C                                                                       00007050
      IF(ABS(X2-X1).LT..0001)GOTO 1                                     00007060
      SLOPE=(Y2-Y1)/(X2-X1)                                             00007070
      IF(ABS(Y2-Y1).LT..0001) SLOPE = 0.                                00007080
      B= Y1 - SLOPE*X1                                                  00007090
      $$= SLOPE*X + B                                                   00007100
      RETURN                                                            00007110
C                                                                       00007120
1     $$=(Y1+Y2)/2.                                                     00007130
      RETURN                                                            00007140
      END                                                               00007150
C                                                                       00007160
C                                                                       00007170
C $$3                                                                   00007180
C                                                                       00007190
C FINDS Y VALUE ON BROKEN LINE OF 3 POINTS, GIVEN X VALUE               00007200
C   AND THE 3 POINTS                                                    00007210
C (ASSUMING ENDS OF LINE EXTEND INFINITELY)                             00007220
C                                                                       00007230
      FUNCTION $$3(X,X1,Y1,X2,Y2,X3,Y3)                                 00007240
C                                                                       00007250
      IF(X.LE.X2)$$3=$$(X,X1,Y1,X2,Y2)                                  00007260
      IF(X.GT.X2)$$3=$$(X,X2,Y2,X3,Y3)                                  00007270
      RETURN                                                            00007280
      END                                                               00007290
C                                                                       00007300
C                                                                       00007310
C $$4                                                                   00007320
C                                                                       00007330
C FINDS Y VALUE ON BROKEN LINE OF 4 POINTS, GIVEN X VALUE               00007340
C   AND THE 4 POINTS                                                    00007350
C (ASSUMING ENDS OF LINE EXTEND INFINITELY)                             00007360
C                                                                       00007370
```

```
C      CURVE IS A STRAIGHT LINE                                          00007950
C                                                                        00007960
C     FLAG = 0 MEANS Y VS X                                              00007970
C     FLAG = 1 MEANS X VS Y (NEGATIVE SLOPE)                             00007980
C     FLAG = 2 MEANS X VS Y (POSITIVE SLOPE)                             00007990
C                                                                        00008000
      FUNCTION $$$(XORY,FLAG,X1,Y1,X2,Y2)                               00008010
      INTEGER FLAG                                                       00008020
      IF(FLAG.EQ.0) $$$=$$(XORY,X1,Y1,X2,Y2)                            00008030
      IF(FLAG.EQ.1) $$$=$$(XORY,Y2,X2,Y1,X1)                            00008040
      IF(FLAG.EQ.2) $$$=$$(XORY,Y1,X1,Y2,X2)                            00008050
      RETURN                                                             00008060
      END                                                                00008070
C                                                                        00008080
C                                                                        00008090
C                                                                        00008100
C $$$3                                                                   00008110
C                                                                        00008120
C READ CURVE Y VS X OR X VS Y DEPENDING UPON FLAG                        00008130
C     CURVE IS A BROKEN LINE OF 3 POINTS                                 00008140
C                                                                        00008150
C     FLAG = 0 MEANS Y VS X                                              00008160
C     FLAG = 1 MEANS X VS Y (NEGATIVE SLOPE)                             00008170
C     FLAG = 2 MEANS X VS Y (POSITIVE SLOPE)                             00008180
C                                                                        00008190
      FUNCTION $$$3(XORY,FLAG,X1,Y1,X2,Y2,X3,Y3)                        00008200
      INTEGER FLAG                                                       00008210
      IF(FLAG.EQ.0) $$$3 = $$3(XORY,X1,Y1,X2,Y2,X3,Y3)                  00008220
      IF(FLAG.EQ.1) $$$3 = $$3(XORY,Y3,X3,Y2,X2,Y1,X1)                  00008230
      IF(FLAG.EQ.2) $$$3 = $$3(XORY,Y1,X1,Y2,X2,Y3,X3)                  00008240
      RETURN                                                             00008250
      END                                                                00008260
C                                                                        00008270
C                                                                        00008280
C                                                                        00008290
C $$$4                                                                   00008300
C                                                                        00008310
C  READ CURVE Y VS X OR X VS Y DEPENDING UPON FLAG                       00008320
C     CURVE IS A BROKEN LINE OF 4 POINTS                                 00008330
C                                                                        00008340
C    FLAG = 0  MEANS Y VS X                                              00008350
C    FLAG = 1  MEANS X VS Y (NEGATIVE SLOPE)                             00008360
C    FLAG = 2  MEANS X VS Y (POSITIVE SLOPE)                             00008370
C                                                                        00008380
      FUNCTION $$$4(XORY,FLAG,X1,Y1,X2,Y2,X3,Y3,X4,Y4)                  00008390
      INTEGER FLAG                                                       00008400
      IF(FLAG.EQ.0) $$$4 = $$4(XORY,X1,Y1,X2,Y2,X3,Y3,X4,Y4)            00008410
      IF(FLAG.EQ.1) $$$4 = $$4(XORY,Y4,X4,Y3,X3,Y2,X2,Y1,X1)            00008420
      IF(FLAG.EQ.2) $$$4 = $$4(XORY,Y1,X1,Y2,X2,Y3,X3,Y4,X4)            00008430
      RETURN                                                             00008440
      END                                                                00008450
C                                                                        00008460
C                                                                        00008470
C                                                                        00008480
C $$$5                                                                   00008490
C                                                                        00008500
C  READ CURVE Y VS X OR X VS Y DEPENDING UPON FLAG                       00008510
C    CURVE IS A BROKEN LINE OF 5 POINTS                                  00008520
C                                                                        00008530
C    FLAG = 0 MEANS Y VS X                                               00008540
C    FLAG = 1 MEANS X VS Y (NEGATIVE SLOPE)                              00008550
C    FLAG = 2 MEANS X VS Y (POSITIVE SLOPE)                              00008551
      FUNCTION $$$5(XORY,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5)                  00008560
```

```
     1SPEO,SPAT,TOWS,ESCT,IDNT,PATL,STGT,TRPT,TRST,RSPD                  00000440
      COMMON/CSSPRB/SSPROB                                              00000450
      DIMENSION GO(4)                                                   00000460
      DIMENSION MNACC(4),MNRKG(4),MNTUR(4)                              00000470
      DIMENSION SU(4),LLS(4)                                            00000480
C                                                                       00000490
      DATA MO/.7,.6,.8,.5,.9,.5,.5,1.0,99.,99.,1.0,1.0,                 00000500
     199.,1.0,99.,1.0,99.,99.,99./                                      00000510
C                                                                       00000520
      DIMENSION GOMIN(4),VISFUE(3)                                      00000530
      DIMENSION XX(4)                                                   00000540
      DATA XX/9999.,9999.,9999.,9999./                                  00000550
      DIMENSION CRFNM(8,10)                                             00000560
      DATA CRFNM/                                                       00000570
     1 'HYDR','OFOI','L-SU','BMER','GED ','FOIL','    ','    ',        00000580
     2 'HYDR','OFOI','L-SU','RFAC','E PT','FRCI','NG  ','    ',        00000590
     3 'AIR ','CUSH','ION ','VEHI','CLE-','LOW ',' P/L','    ',        00000600
     4 'AIR ','CUSH','ION ','VEHI','CLE-','HIGH',' P/L','    ',        00000610
     5 'SURF','ACE ','EFFE','CT S','HIP ','  ','    ','    ',         00000620
     6 'PLAN','ING ','CRAF','T   ','  ','  ','  ','  ',              00000630
     7 'CATA','MARA','N   ','    ','  ','  ','  ','  ',              00000640
     8 'SWAT','H   ','    ','    ','  ','  ','  ','  ',              00000650
     9 'HYBR','ID V','ESSL','L   ','  ','  ','  ','  ',              00000660
     I 'CONV','ENTI','ONAL',' CRA','FT  ','  ','  ','  '/             00000670
      DIMENSION CGCRNM(2,12)                                            00000680
      DATA CGCRNM/'MRB ','  ','FWB','32 ','UTB ','41 ',                00000690
     1'MLB ','44 ','MLB ','52 ','AWB ','55 ','ANB ','63 ',             00000700
     1'WPB ','82 ','WPB ','95 ','WMEC','210 ',                         00000710
     1'WMEC','270 ','WMEC','378 '/                                      00000720
      DIMENSION ENGNAM(2)                                               00000730
      DATA ENGNAM/'(GT)','(PE)'/                                        00000740
C                                                                       00000750
      DIMENSION SSPROB(8,10)                                            00000760
C                                                                       00000770
      DIMENSION AVESS(10)                                               00000780
      DATA AVESS/0.5,1.0,1.5,2.0,2.5,3.0,3.5,4.0,4.5,5.0/               00000790
C                                                                       00000800
      DIMENSION TOWDIS(6,5)                                             00000810
      DATA TOWDIS/.5,1.,2.5,7.,10.,50., .7,2.,4.,10.,30.,100.,          00000820
     1  1.,4.,7.,20.,60.,500., 2.,6.,20.,50.,80.,1000.,                 00000830
     2  10.,20.,50.,100.,300.,10000./                                   00000840
C                                                                       00000850
      DIMENSION VISDIS(3,5),VMXVIS(5)                                   00000860
      DATA VISDIS/.9,.1,.0,.7,.2,.1,.5,.3,.2/                           00000870
      DATA VMXVIS/99999.,20.,10./                                       00000880
C                                                                       00000890
      INTEGER TOWDTB,DPHDTB,VISDTB,SSPDTB                               00000900
      INTEGER VISTYP                                                    00000910
C                                                                       00000920
      DIMENSION SSPRBD(8)                                               00000930
C                                                                       00000940
      DIMENSION CGFR20(12),CGFR10(12)                                   00000941
      DATA CGFR20/15.,32.5,48.2,9999.,9999.,63.9,9999.,73.6,130.,9999., 00000942
     19999.,1266.7/                                                     00000943
      DATA CGFR10/5.1,9.1,16.0,20.7,21.2,18.6,27.8,23.6,27.9,           00000944
     170.6,89.4,221.4/                                                  00000945
C                                                                       00000950
      IF(TYPE.EQ.10)TYPNUM=1                                            00000960
      IF(TYPE.EQ.11)TYPNUM=2                                            00000970
      IF(TYPE.EQ.20)TYPNUM=3                                            00000980
      IF(TYPE.EQ.21)TYPNUM=4                                            00000990
      IF(TYPE.EQ.30)TYPNUM=5                                            00001000
      IF(TYPE.EQ.40)TYPNUM=6                                            00001010
```

```
     1RATE,DSPEED,FUELRT,VISFUE,VAVG,AVFURT)                              00001570
      SPEED(RATE) = VAVG                                                  00001580
      MFULRT(RATE) = AVFURT                                               00001590
  711 CONTINUE                                                            00001600
C                                                                         00001610
C                                                                         00001620
C GO: GO FRACTION  (USED IN LIMITING SEA STATE PARAMETER)                 00001630
C                                                                         00001640
      IFILE = 6                                                           00001650
      DATA GOMIN/15.,8.,5.,0./                                            00001660
      DO 4701 RATE=1,4                                                    00001670
      IF(DSPEED.LT.GOMIN(RATE))GO TO 4702                                 00001680
      PCDSPD=(GOMIN(RATE)/DSPEED)*100.                                    00001690
      SSMX=$SSPOS(TYPE,DISP,RATE,DSPEED,PCDSPD)                           00001700
      GO(RATE)=$CPBSS(SSPROB,SSPDTB,SSMX)                                 00001710
      GO TO 4701                                                          00001720
 4702 GO(RATE)=0.                                                         00001730
 4701 CONTINUE                                                            00001740
C                                                                         00001750
C                                                                         00001760
C Tw: TOW FRACTION PARAMETER                                              00001770
C                                                                         00001780
      CALL PTWD(TOWDIS,TOWDTB,TOWDSP,PTOWD,AVTWDS)                        00001790
      TW(I2) = PTOWD                                                      00001800
      FCTDSP = AVTWDS/DISP                                                00001810
      TOWSPD = $$3(FCTDSP,0.,.5,.2,10.,10.,0.)                            00001820
C                                                                         00001830
C SK: SEAKINDLINESS PARAMETER  (USED IN LIMITING SEA STATE PARAMETER)     00001840
C                                                                         00001850
      DO 9020 JTPOS=1,19                                                  00001860
      IF(MO(JTPOS).EQ.99.) GO TO 9020                                     00001870
      MTN = MO(JTPOS)                                                     00001880
C                                                                         00001890
      IF(JTPOS.LE.7) RATE = 4                                             00001900
      IF(JTPOS.GE.8.AND.JTPOS.LE.12) RATE = 3                             00001910
      IF(JTPOS.GE.13.AND.JTPOS.LE.18) RATE = 2                            00001920
      IF(JTPOS.EQ.19) RATE = 1                                            00001930
C                                                                         00001940
      WVHTBS = $WHVSM(TYPE,RATE,MTN)                                      00001950
      LAMBDA = (100./DISP)**.333                                          00001960
      IF(TYPE.EQ.60) LAMBDA = (1500./DISP)**.333                          00001970
      WVHTCF = WVHTBS/LAMBDA                                              00001980
      ARG = .8*WVHTCF+.4                                                  00001990
      SS = 2.5*ALOG(ARG)                                                  00002000
      SK(JTPOS) = $CPBSS(SSPROB,SSPDTB,SS)                                00002010
 9020 CONTINUE                                                            00002020
C                                                                         00002030
C MN: MANEUVERABILITY PARAMETER                                           00002040
C                                                                         00002050
      MN(1) = $$4(LENG,0.,1.,50.,1.,200.,.6,99999.,.8)                    00002060
      MN(2) = MN(1)                                                       00002070
      MN(3) = MN(1)                                                       00002080
      MN(4) = MN(1)                                                       00002090
      MN(7) = MN(1)                                                       00002100
      MN(12) = $$4(TNRAD(3),0.,1.,500.,1.,1500.,.5,99999.,.5)             00002110
      MN(14) = $$4(TNRAD(2),0.,1.,500.,1.,1500.,.5,99999.,.5)             00002120
C                                                                         00002130
C SU: SURVIVABILITY  (USED IN LIMITING SEA STATE PARAMETER)               00002140
C                                                                         00002150
      SU0=$CPBSS(SSPROB,SSPDTB,SURVIV)                                    00002160
      SU(1)=SU0                                                           00002170
      SU(2)=SU0                                                           00002180
      SU(3)=SU0                                                           00002190
```

```
      IF(TYPE.LT.100)WRITE(IFILE,4901)(CRFNM(I,TYPNUM),I=1,8)                00002890
 4901  FORMAT('1'/10X,12X,'C R A F T   P A R A M E T E R S'/                 00002900
     1   //11X,11X,'CRAFT TYPE',5X,8A4)                                     00002910
      IF(TYPE.GE.100)WRITE(IFILE,4936)(CGCRNM(I,CGTYPE),I=1,2)               00002920
 4936  FORMAT('1'/10X,12X,'C R A F T   P A R A M E T E R S '/                00002930
     1//11X,11X,'CRAFT TYPE',5X,'COAST GUARD ',2A4)                         00002940
      WRITE(IFILE,4937)IDISP,ILENG,IDSPD,FUFRAC                              00002950
 4937  FORMAT(11X,11X,'DISPLACEMENT',I6,2X,'TONS'/                          00002960
     3   11X,11X,'LENGTH',6X,I6,2X,'FEET'/                                  00002970
     4   11X,11X,'DESIGN SPEED',I6,2X,'KNOTS'/                              00002980
     3   11X,11X,'FUEL FRACTION',F6.2/)                                     00002990
      WRITE(IFILE,4945)VISDTB,TOWDTB,DPHDTB,SSPDTB,AVESS(SSPDTB)             00003000
 4945  FORMAT(11X,11X,4X,'VISIBILITY DISTRIBUTION NO.',I2/                  00003010
     2        11X,11X,4X,'TOW DISTRIBUTION NO.',I2/                         00003020
     2        11X,11X,4X,'DEPTH DISTRIBUTION NO.',I2/                       00003030
     3        11X,11X,4X,'SEA STATE DISTRIBUTION NO.',I2/                   00003040
     4        11X,11X,4X,'(AVERAGE SEA STATE=',F3.1,')')                    00003050
      WRITE(IFILE,5001)                                                      00003060
   01  FORMAT(//14X,'TASK',2X,'CARGO',1X,'DRAFT',1X,'MANEUV',2X,            00003070
     1 'SEA',3X,'TOW'/14X,'CODE',2X,'CPCTY',14X,'STATE'//22X,               00003080
     2 'CC',4X,'DF',4X,'MN',4X,'LS',4X,'TW')                                00003090
C                                                                           00003100
      WRITE(IFILE,5002)                                                     00003110
 5002 FORMAT(//10X,'ON SCENE:')                                             00003120
      WRITE(IFILE,5003)DF(1),MN(1),LS(1)                                    00003130
 5003 FORMAT(14X,'ASST',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'ASSIST')       00003140
      WRITE(IFILE,5006) DF(2),MN(2),LS(2)                                   00003150
 5006 FORMAT(14X,'BORD',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'BOARD')        00003160
      WRITE(IFILE,5005)DF(3),MN(3),LS(3)                                    00003170
 5005 FORMAT(14X,'MNAC',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'MONITOR ACT00003180
     1IVITIES')                                                             00003181
      WRITE(IFILE,5007)DF(4),MN(4),LS(4)                                    00003190
 5007 FORMAT(14X,'RTRV',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'RETRIEVE')     00003200
      WRITE(IFILE,5004)LS(5)                                                00003210
 5004 FORMAT(14X,'WAIT',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'WAIT')            00003220
      WRITE(IFILE,5008)DF(6),LS(6)                                          00003230
 5008 FORMAT(14X,'WFOD',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'WORK EQUIPM00003240
     1ENT @ DRIFT')                                                         00003241
      WRITE(IFILE,5009)DF(7),MN(7),LS(7)                                    00003250
 5009 FORMAT(14X,'WFOP',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'WORK EQUIPM00003260
     1ENT @ POSITION')                                                      00003261
C                                                                           00003270
      WRITE(IFILE,5010)                                                     00003280
 5010 FORMAT(//10X,'REDUCED SPEED:')                                        00003290
      WRITE(IFILE,5013)DF(8),LS(8)                                          00003300
 5013 FORMAT(14X,'SDIU',3X,2(1X,'--',1X,2X,F4.2,2X),1X,'--',3X,'SEARCH FO00003310
     1OR DISTRESSED UNIT')                                                  00003311
      WRITE(IFILE,5012)LS(9)                                                00003320
 5012 FORMAT(14X,'SESC',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'SLOW ESCORT')     00003330
      WRITE(IFILE,5015)DF(10),LS(10)                                        00003340
 5015 FORMAT(14X,'SPAT',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'SLOW PATROL00003350
     1')                                                                    00003351
      WRITE(IFILE,5014)DF(11),LS(11)                                        00003360
 5014 FORMAT(14X,'SPEO',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'SEARCH FOR 00003370
     1PEOPLE')                                                              00003371
      WRITE(IFILE,5011)MN(12),LS(12),TW(12)                                 00003380
 5011 FORMAT(14X,'TOWS',3X,2(1X,'--',3X),3(F4.2,2X),'TOWS')                 00003390
C                                                                           00003400
      WRITE(IFILE,5016)                                                     00003410
 5016 FORMAT(//10X,'CRUISE SPEED:')                                         00003420
      WRITE(IFILE,5030)LS(13)                                               00003430
 5030 FORMAT(14X,'ESCT',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'ESCORT')          00003440
      WRITE(IFILE,5017)MN(14),LS(14)                                        00003450
```

```
     1 3X,'OF SUCCESS')                                                 00004110
C                                                                       00004120
      WRITE(IFILE,6002)                                                 00004130
 6002 FORMAT(//10X,'ON SCENE:')                                         00004140
      WRITE(IFILE,6003)TPOS(1)                                          00004150
 6003 FORMAT(14X,'ASST',5X,F5.3,5X,'ASSIST')                            00004160
      WRITE(IFILE,6006) TPOS(2)                                         00004170
 6006 FORMAT(14X,'BORD',5X,F5.3,5X,'BOARD')                             00004180
      WRITE(IFILE,6005)TPOS(3)                                          00004190
 6005 FORMAT(14X,'MNAC',5X,F5.3,5X,'MONITOR ACTIVITIES')                00004200
      WRITE(IFILE,6007)TPOS(4)                                          00004210
 6007 FORMAT(14X,'RTRV',5X,F5.3,5X,'RETRIEVE')                          00004220
      WRITE(IFILE,6004)TPOS(5)                                          00004230
 6004 FORMAT(14X,'WAIT',5X,F5.3,5X,'WAIT')                              00004240
      WRITE(IFILE,6008)TPOS(6)                                          00004250
 6008 FORMAT(14X,'WEQD',5X,F5.3,5X,'WORK EQUIPMENT @ DRIFT')            00004260
      WRITE(IFILE,6009)TPOS(7)                                          00004270
 6009 FORMAT(14X,'WEQP',5X,F5.3,5X,'WORK EQUIPMENT @ POSITION')         00004280
C                                                                       00004290
      WRITE(IFILE,6010)                                                 00004300
 6010 FORMAT(/10X,'REDUCED SPEED:')                                     00004310
      WRITE(IFILE,6013)TPOS(8)                                          00004320
 6013 FORMAT(14X,'SDIU',5X,F5.3,'*',4X,'SEARCH FOR DISTRESSED UNIT')    00004330
      WRITE(IFILE,6012)TPOS(9)                                          00004340
 6012 FORMAT(14X,'SESC',5X,F5.3,5X,'SLOW ESCORT')                       00004350
      WRITE(IFILE,6015)TPOS(10)                                         00004360
 6015 FORMAT(14X,'SPAT',5X,F5.3,5X,'SLOW PATROL')                       00004370
      WRITE(IFILE,6014)TPOS(11)                                         00004380
 6014 FORMAT(14X,'SPEO',5X,F5.3,'*',4X,'SEARCH FOR PEOPLE')             00004390
      WRITE(IFILE,6011)TPOS(12)                                         00004400
 6011 FORMAT(14X,'TOWS',5X,F5.3,5X,'TOWS')                              00004410
C                                                                       00004420
      WRITE(IFILE,6016)                                                 00004430
 6016 FORMAT(//10X,'CRUISE SPEED:')                                     00004440
      WRITE(IFILE,6030)TPOS(13)                                         00004450
 6030 FORMAT(14X,'ESCT',5X,F5.3,5X,'ESCORT')                            00004460
      WRITE(IFILE,6017)TPOS(14)                                         00004470
 6017 FORMAT(14X,'IDNT',5X,F5.3,5X,'IDENTIFY')                          00004480
      WRITE(IFILE,6018)TPOS(15)                                         00004490
 6018 FORMAT(14X,'PATL',5X,F5.3,5X,'PATROL')                            00004500
      WRITE(IFILE,6019)TPOS(16)                                         00004510
 6019 FORMAT(14X,'STGT',5X,F5.3,'*',4X,'SEARCH FOR TARGET')             00004520
      WRITE(IFILE,6021)TPOS(17)                                         00004530
 6021 FORMAT(14X,'TRPT',5X,F5.3,5X,'TRANSPORT')                         00004540
      WRITE(IFILE,6020)TPOS(18)                                         00004550
 6020 FORMAT(14X,'TRST',5X,F5.3,5X,'TRANSIT')                           00004560
C                                                                       00004570
      WRITE(IFILE,6022)                                                 00004580
 6022 FORMAT(/10X,'FLANK SPEED:')                                       00004590
      WRITE(IFILE,6023)TPOS(19)                                         00004600
 6023 FORMAT(14X,'RSPD',5X,F5.3,5X,'RESPOND')                           00004610
C                                                                       00004620
      WRITE(IFILE,6025)                                                 00004630
 6025 FORMAT(//12X,                                                     00004640
     1'*   THIS IS THE P.O.S. OF THE ABILITY TO SEARCH. CRAFT''S SUCCESS00004641
     2'/16X,'IN FINDING THE OBJECT OF THE SEARCH IS DEPENDENT UPON')    00004642
      WRITE(IFILE,6026)                                                 00004670
 6026 FORMAT(16X,' SCENARIO (E.G., SEARCH AREA)')                       00004680
      WRITE(IFILE,6024)                                                 00004690
 6024 FORMAT(/10X,                                                      00004700
     1'***** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AND WEIGHT OF CARG00004710
     20)')                                                              00004711
                                                                        00004740
```

```
     1RDING PARTY')                                                     00005321
      WRITE(IFILE,2017)DF(4),MN(4),LS(4)                                00005330
 2017 FORMAT(14X,'ROB',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'RETRIEVE OBJ00005340
     1ECTS')                                                            00005341
      WRITE(IFILE,2018)DF(4),MN(4),LS(4)                                00005350
 2018 FORMAT(14X,'RPE',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'RESCUE PEOPL00005360
     1E')                                                               00005361
      WRITE(IFILE,2019)DF(7),MN(7),LS(7)                                00005370
 2019 FORMAT(14X,'RSB',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'RETRIEVE SMA00005380
     1LL BOAT')                                                         00005381
      WRITE(IFILE,2020)DF(3),MN(3),LS(3)                                00005390
 2020 FORMAT(14X,'SSI',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,             00005400
     1'STAKEOUT SPECIAL INTEREST VESSEL')                               00005410
      WRITE(IFILE,2021)LS(5)                                            00005420
 2021 FORMAT(14X,'SZE',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'SEIZE')        00005430
      WRITE(IFILE,2022)DF(7),MN(7),LS(7)                                00005440
 2022 FORMAT(14X,'TWS',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'TAKE WATER S00005450
     1AMPLE')                                                           00005451
      WRITE(IFILE,2023)DF(7),MN(7),LS(7)                                00005460
 2023 FORMAT(14X,'ULQ',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'UNLOAD EQUIP00005470
     1MENT')                                                            00005471
      WRITE(IFILE,2026)LS(5)                                            00005480
 2026 FORMAT(14X,'WQB',3X,3(1X,'--',3X),F4.2,3X,'--',3X,                 00005490
     1'WORK EQUIPMENT FROM SMALL BOAT')                                 00005500
      WRITE(IFILE,2024)DF(6),LS(6)                                      00005510
 2024 FORMAT(14X,'WQD',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'WORK EQUIPMF00005520
     1NT @ DRIFT')                                                      00005521
      WRITE(IFILE,2025)DF(7),MN(7),LS(7)                                00005530
 2025 FORMAT(14X,'WQF',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,              00005540
     1'WORK EQUIPMENT @ FIXED POSITION')                                00005550
C                                                                       00005560
C                                                                       00005570
      IF(TYPE.LT.100)WRITE(IFILE,3031)(CRFNM(I,TYPNUM),I=1,8)           00005580
 3031 FORMAT('1'/14X,12X,'C R A F T   P A R A M E T E R S'/             00005640
     1   //11X,15X,'CRAFT TYPE',5X,8A4)                                00005650
      IF(TYPE.GE.100)WRITE(IFILE,3032)(CGCRNM(I,CGTYPE),I=1,2)          00005660
 3032  FORMAT('1'/14X,12X,'C R A F T   P A R A M E T E R S '/           00005670
     1//11X,15X,'CRAFT TYPE',5X,'COAST GUARD ',2A4)                     00005680
      WRITE(IFILE,3033)IDISP,ILENG,IDSPD,FUFRAC                         00005690
 3033 FORMAT(11X,15X,'DISPLACEMENT',I6,2X,'TONS'/                       00005700
     3   11X,15X,'LENGTH',6X,I6,2X,'FEET'/                              00005710
     4   11X,15X,'DESIGN SPEED',I6,2X,'KNOTS'/                          00005720
     3   11X,15X,'FUEL FRACTION',F6.2/)                                 00005730
      WRITE(IFILE,3112)VISDTB,TOWDTB,DPHDTB,SSPDTB,AVESS(SSPDTB)        00005740
 3112 FORMAT(11X,15X,4X,'VISIBILITY DISTRIBUTION NO.',I2/               00005750
     2       11X,15X,4X,'TOW DISTRIBUTION NO.',I2/                      00005760
     2       11X,15X,4X,'DEPTH DISTRIBUTION NO.',I2/                    00005770
     3       11X,15X,4X,'SEA STATE DISTRIBUTION NO.',I2/                00005780
     4       11X,15X,4X,'(AVERAGE SEA STATE=',F3.1,')')                 00005790
      WRITE(IFILE,3001)                                                 00005800
 3001 FORMAT(//14X,'TASK',2X,'CARGO',1X,'DRAFT',1X,'MANEUV',2X,         00005810
     1 'SEA',3X,'TOW'/14X,'CODE',2X,'CPCTY',14X,'STATE'//22X,           00005820
     2 'CC',4X,'DF',4X,'MN',4X,'LS',4X,'TW')                            00005830
C                                                                       00005840
      WRITE(IFILE,3002)                                                 00005850
 3002 FORMAT(//10X,'REDUCED SPEED:')                                    00005860
C                                                                       00005870
      WRITE(IFILE,3004)DF(8),LS(8)                                      00005880
 3004 FORMAT(14X,'SDU',3X,2(1X,'--',1X,2X,F4.2,2X),1X,'--',1X,2X,'SEARCH00005890
     1 FOR DISTRESSED UNIT')                                            00005900
      WRITE(IFILE,3003)LS(9)                                            00005901
 3003 FORMAT(14X,'SES',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'SLOW ESCORT')  00005910
      WRITE(IFILE,3005)DF(11),LS(11)                                    00005920
                                                                        00005930
```

```
C TPOS(14)=ICNT=IDC=IDF                                                  00006500
C TPOS(15)=PATL=PAT                                                      00006510
C TPOS(16)=STGT=SSH                                                      00006520
C TPOS(17)=TRPT=TFQ=****                                                 00006530
C TPOS(18)=TRST=SFL=TPE=TKA                                              00006540
C TPOS(19)=RSPD=INT=DSH                                                  00006550
C                                                                        00006560
      IFILE=6                                                            00006590
      IF(TYPE.LT.100)WRITE(IFILE,7031)(CRFNM(I,TYPNUM),I=1,8)            00006620
 7031 FORMAT('1'/5X,8X,'T A S K  P R O B A B I L I T I F S  O F  ',      00006630
     1'S U C C F S S '                                                   00006640
     1//2X,15X,'CRAFT TYPE',5X,8A4)                                      00006650
      IF(TYPE.GE.100)WRITE(IFILE,7032)(CGCRNM(I,CGTYPE),I=1,2)           00006660
 7032 FORMAT('1'/5X,8X,'T A S K  P R O B A B I L I T I F S  O F  ',      00006670
     1'S U C C F S S '/                                                  00006680
     1//2X,15X,'CRAFT TYPE',5X,'COAST GUARD ',2A4)                       00006690
      WRITE(IFILE,7033)IDISP,ILENG,ILSPD,FUFRAC                          00006700
 7033 FORMAT(2X,15X,'DISPLACEMENT',I6,2X,'TONS'/                         00006710
     3   2X,15X,'LENGTH',6X,I6,' FEET'/                                  00006720
     4   2X,15X,'DESIGN SPEED',I6,2X,'KNOTS'/                            00006730
     5   2X,15X,'FUEL FRACTION',F6.2/)                                   00006740
      WRITE(IFILE,7112)VISDTB,TOWDTB,DPHDTB,SSPDTB,AVESS(SSPDTB)         00006750
 7112 FORMAT(2X,15X,4X,'VISIBILITY DISTRIBUTION NO.',I2/                 00006760
     2       2X,15X,4X,'TOW DISTRIBUTION NO.',I2/                        00006770
     2       2X,15X,4X,'DEPTH DISTRIBUTION NO.',I2/                      00006780
     3       2X,15X,4X,'SEA STATE DISTRIBUTION NO.',I2/                  00006790
     4       2X,15X,4X,'(AVERAGE SEA STATE=',F3.1,')')                   00006800
      WRITE(IFILE,7001)                                                  00006810
 7001 FORMAT(//14X,'TASK',3X,'TASK PROB.',4X,'TASK'/14X,'CODE',          00006820
     1 3X,'OF SUCCESS')                                                  00006830
C                                                                        00006840
      WRITE(IFILE,7002)                                                  00006850
 7002 FORMAT(//10X,'ON SCENE:')                                          00006860
C                                                                        00006870
      WRITE(IFILE,7003) TPOS(2)                                          00006880
 7003 FORMAT(14X,'BRD',6X,F5.3,5X,'BOARD')                               00006890
      WRITE(IFILE,7004)TPOS(7)                                           00006900
 7004 FORMAT(14X,'FFF',6X,F5.3,5X,'FIGHT FIRE FROM CG VESSEL')           00006910
      WRITE(IFILE,7005)TPOS(5)                                           00006920
 7005 FORMAT(14X,'FFO',6X,F5.3,5X,'FIGHT FIRE ON ANOTHER VESSEL')        00006930
      WRITE(IFILE,7006)TPOS(1)                                           00006940
 7006 FORMAT(14X,'GAS',6X,F5.3,5X,'GENERAL ASSISTANCE')                  00006950
      WRITE(IFILE,7007)TPOS(5)                                           00006960
 7007 FORMAT(14X,'INS',6X,F5.3,5X,'INSPECTION')                          00006970
      WRITE(IFILE,7008)TPOS(7)                                           00006980
 7008 FORMAT(14X,'LEQ',6X,F5.3,5X,'LOAD EQUIPMENT')                      00006990
      WRITE(IFILE,7009)TPOS(5)                                           00007000
 7009 FORMAT(14X,'LOI',6X,F5.3,5X,'LOITER')                              00007010
      WRITE(IFILE,7010)TPOS(7)                                           00007020
 7010 FORMAT(14X,'LSB',6X,F5.3,5X,'LAUNCH SMALL BOAT')                   00007030
      WRITE(IFILE,7011)TPOS(3)                                           00007040
 7011 FORMAT(14X,'MAC',6X,F5.3,5X,'MONITOR ACTIVITIES')                  00007050
      WRITE(IFILE,7012)TPOS(3)                                           00007060
 7012 FORMAT(14X,'MOS',6X,F5.3,5X,'MONITOR OIL SPILL')                   00007070
      WRITE(IFILE,7013)TPOS(5)                                           00007080
 7013 FORMAT(14X,'OBA',6X,F5.3,5X,'ON BOARD ASSISTANCE')                 00007090
      WRITE(IFILE,7015)TPOS(5)                                           00007100
 7015 FORMAT(14X,'OSC',6X,F5.3,5X,'ON SCENE COMMANDER(GENERAL)')         00007110
      WRITE(IFILE,7016) TPOS(2)                                          00007120
 7016 FORMAT(14X,'RBP',6X,F5.3,5X,'RETRIEVE BOARDING PARTY')             00007130
      WRITE(IFILE,7017)TPOS(4)                                           00007140
 7017 FORMAT(14X,'ROB',6X,F5.3,5X,'RETRIEVE OBJECTS')                    00007150
      WRITE(IFILE,7018)TPOS(4)                                           00007160
```

```
      WRITE(IFILE,8009)TPOS(14)                                          00007860
 8009 FORMAT(14X,'IDC',6X,F5.3,5X,'IDENTIFY CRAFT')                      00007870
      WRITE(IFILE,8010)TPOS(14)                                          00007880
 8010 FORMAT(14X,'IDF',6X,F5.3,5X,'IDENTIFY FLEET')                      00007890
      WRITE(IFILE,8011)TPOS(15)                                          00007900
 8011 FORMAT(14X,'PAT',6X,F5.3,5X,'PATROL')                              00007910
      WRITE(IFILE,8035)TPOS(18)                                          00007920
 8035 FORMAT(14X,'SFL',6X,F5.3,5X,'SEARCH FOR FLEET')                    00007930
      WRITE(IFILE,8014)TPOS(16)                                          00007940
 8014 FORMAT(14X,'SSH',6X,F5.3,'*',4X,'SEARCH FOR SHIP')                 00007950
      WRITE(IFILE,8015) TPOS(17)                                         00007960
 8015 FORMAT(14X,'TFQ',6X,F5.3,5X,'TRANSPORT FQUIPMENT')                 00007970
      WRITE(IFILE,8016)TPOS(18)                                          00007980
 8016 FORMAT(14X,'TPE',6X,F5.3,5X,'TRANSPORT PEOPLE')                    00007990
      WRITE(IFILE,8017)TPOS(18)                                          00008000
 8017 FORMAT(14X,'TRA',6X,F5.3,5X,'TRANSIT')                             00008010
C                                                                        00008020
      WRITE(IFILE,8018)                                                  00008030
 8018 FORMAT(/10X,'FLANK SPEED:')                                        00008040
C                                                                        00008050
      WRITE(IFILE,8019)TPOS(19)                                          00008060
 8019 FORMAT(14X,'DSH',6X,F5.3,5X,'DASH')                                00008070
      WRITE(IFILE,8020)TPOS(19)                                          00008080
 8020 FORMAT(14X,'INT',6X,F5.3,5X,'INTERDICT')                           00008090
C                                                                        00008100
      WRITE(IFILE,8022)                                                  00008110
 8022 FORMAT(//12X,                                                      00008120
     1'*   THIS IS THE P.O.S. OF THE ABILITY TO SEARCH. CRAFT''S SUCCESS00008121
     2'/16X,'IN FINDING THE OBJECT OF THE SEARCH IS DEPENDENT UPON')     00008122
      WRITE(IFILE,8023)                                                  00008150
 8023 FORMAT(16X,' SCENARIO (E.G. SEARCH AREA)')                         00008160
      WRITE(IFILE,8021)                                                  00008170
 8021 FORMAT(/10X,                                                       00008180
     1'***** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AND WEIGHT OF CARG00008190
     20)')                                                               00008200
C                                                                        00008220
      WRITE(6,6040)                                                      00008230
 6040 FORMAT('I')                                                        00008240
      RETURN                                                             00008250
      END                                                                00008260
C                                                                        00008270
C                                                                        00008280
C                                                                        00008290
C VwTAV                                                                  00008300
C                                                                        00008310
C WEIGHTED AVERAGE VELOCITY AND FUEL RATE                                00008320
C                                                                        00008330
      SUBROUTINE VWTAV(SSPRED,VISDIS,VISDTR,VMXVIS,TYPE,DISP,RATE,       00008340
     1DSPEED,FUELRT,VISFUE,VAVG,AVFURT)                                  00008350
      IMPLICIT REAL(A-Z)                                                 00008360
      INTEGER TYPE,RATE,SS1,DELTA,VISTYP,SSO,VISDTR                      00008370
      DIMENSION SSPRED(8),VMXVIS(3),VISDIS(3,3)                          00008380
      DIMENSION FUELRT(4),VISFUE(3)                                      00008390
      VAVG = 0.                                                          00008400
      AVFURT = 0.                                                        00008410
      DO 10 SS1 = 1,8                                                    00008420
      SSO = SS1 - 1                                                      00008430
      DO 100 DELTA = 1,9                                                 00008440
      SS = SSO + DELTA/10                                                00008450
      DO 200 VISTYP = 1,3                                                00008460
      PVINSS = $PDSSS(TYPE,DISP,RATE,DSPEED,SS)                          00008470
      VINSS = DSPEED*PVINSS/100.                                         00008480
                                                                         00008490
```

```
      END                                                                00009130
C                                                                        00009140
C                                                                        00009150
C $MWTAV                                                                 00009160
C                                                                        00009170
C WEIGHTED AVERAGE MOTION OF CRAFT                                       00009180
C                                                                        00009190
      FUNCTION $MWTAV(SSPRBD,TYPE,DISP,RATE)                             00009200
      IMPLICIT REAL(A-Z)                                                 00009210
      INTEGER TYPE,RATE,SS1,SS                                           00009220
      DIMENSION SSPRBD(8)                                                00009230
C                                                                        00009240
      SUM=0.                                                             00009250
C LAMBDA CONVERTS WAVE HEIGHT FROM CRAFT DISPLACEMENT TO                 00009260
C BASE DISPLACEMENT (=100 TONS)                                          00009270
      LAMBDA=(100/DISP)**.333                                            00009280
      IF(TYPE.EQ.60)LAMBDA=(1500/DISP)**.333                             00009290
      DO 100 SS1=1,8                                                     00009300
      SS=SS1-1                                                           00009310
      WVHTCF=.5*(-1.+2.5*EXP(.4*SS))                                     00009320
      WVHTBS=LAMBDA*WVHTCF                                               00009330
      SUM=SUM + SSPRBD(SS1)*$MVSWH(TYPE,RATE,WVHTBS)                     00009340
100   CONTINUE                                                           00009350
      $MWTAV=SUM                                                         00009360
      RETURN                                                             00009370
      END                                                                00009380
C                                                                        00009390
C                                                                        00009400
C $MVSWH AND $WHVSM                                                      00009410
C                                                                        00009420
C  MOTION OF BASE CRAFT VS. WAVE HEIGHT (FOR DISPLACEMENT=100 TONS       00009430
C  EXCEPT TYPE 60 DISPLACEMENT=1500 TONS) AND REVERSE                    00009440
C                                                                        00009450
      FUNCTION $MVSWH(TYPE,RATE,WVHTBS)                                  00009460
      IMPLICIT REAL(A-Z)                                                 00009470
      INTEGER TYPE,RATE,FLAG                                             00009480
C                                                                        00009490
      IN = WVHTBS                                                        00009500
      FLAG = 0                                                           00009510
      GO TO 1                                                            00009520
C                                                                        00009530
      ENTRY $WHVSM(TYPE,RATE,MTN)                                        00009540
      IN = MTN                                                           00009550
      FLAG = 2                                                           00009560
C                                                                        00009570
    1 IF(TYPE.NE.10)GO TO 11                                             00009580
      IF(RATE.EQ.1.OR.RATE.EQ.2) OUT = $$$3(IN,FLAG,0.,0.,15.,.5,        00009590
     118.,1.0)                                                           00009600
      IF(RATE.EQ.3) OUT = $$$(IN,FLAG,0.,0.,8.,1.0)                      00009610
      IF(RATE.EQ.4) OUT = $$$(IN,FLAG,0.,0.,12.,1.0)                     00009620
      GO TO 999                                                          00009630
11    IF(TYPE.NE.11)GO TO 20                                             00009640
      IF(RATE.EQ.1.OR.RATE.EQ.2) OUT = $$$3(IN,FLAG,0.,0.,5.,.25,        00009650
     18.,1.0)                                                            00009660
      IF(RATE.EQ.3) OUT = $$$(IN,FLAG,0.,0.,10.,1.0)                     00009670
      IF(RATE.EQ.4) OUT = $$$(IN,FLAG,0.,0.,14.,1.0)                     00009680
      GO TO 999                                                          00009690
20    IF(TYPE.NE.20.OR.TYPE.NE.21)GO TO 40                               00009700
      IF(RATE.EQ.1) OUT = $$$3(IN,FLAG,0.,0.,4.,.2,5.,1.0)               00009710
      IF(RATE.EQ.2) OUT = $$$3(IN,FLAG,0.,0.,4.,.2,6.,1.0)               00009720
      IF(RATE.EQ.3) OUT = $$$3(IN,FLAG,0.,0.,4.,.2,7.,1.)                00009730
      IF(RATE.EQ.4) OUT = $$$3(IN,FLAG,0.,0.,4.,.2,10.,1.0)              00009740
                                                                         00009750
```

```
      IN = SS                                                           00010560
      FLAG = 0                                                          00010570
      GO TO 1                                                           00010580
C                                                                       00010590
      ENTRY $SSPDS(TYPE,DISP,RATE,DSPEED,PCDSPD)                        00010600
      IN = PCDSPD                                                       00010610
      FLAG = 1                                                          00010620
C                                                                       00010630
    1 IF(TYPE.EQ.10) GO TO 10                                           00010640
      IF(TYPE.EQ.11) GO TO 11                                           00010650
      IF(TYPE.EQ.20.OR.TYPE.EQ.21) GO TO 20                             00010660
      IF(TYPE.EQ.30.OR.TYPE.EQ.70) GO TO 30                             00010670
      IF(TYPE.EQ.80.OR.TYPE.EQ.102.OR.TYPE.EQ.103.OR.                   00010680
     1   TYPE.EQ.106.OR.TYPE.EQ.107.OR.TYPE.EQ.108.OR.                  00010690
     1   TYPE.EQ.109.OR.TYPE.EQ.110.OR.TYPE.EQ.111.OR.                  00010700
     1   TYPE.EQ.112)  GO TO 80                                         00010710
      IF(TYPE.EQ.50) GO TO 50                                           00010720
      IF(TYPE.EQ.60) GO TO 60                                           00010730
      IF(TYPE.EQ.40) GO TO 40                                           00010740
      IF(TYPE.EQ.101.OR.TYPE.EQ.104.OR.TYPE.EQ.105) GO TO 101           00010750
C                                                                       00010760
   10 IF(RATE.EQ.2) GO TO 1002                                          00010770
      IF(DISP.LE.100.)OUT = $$$4(IN,FLAG,0.,100.,5.,91.7,5.,            00010780
     120.,7.,0.)                                                        00010790
      IF(DISP.GT.100..AND.DISP.LE.200.) OUT = $$$4(IN,FLAG,0.,          00010800
     1100.,5.5,90.8,5.5,20.,7.5,0.)                                     00010810
      IF(DISP.GT.200.) OUT = $$$4(IN,FLAG,0.,100.,6.,90.,6.,            00010820
     120.,8.,0.)                                                        00010830
      GO TO 991                                                         00010840
C  FOR RATE = 2 AND TYPE = 10                                           00010850
 1002 IF(DISP.LE.100.) OUT = $$$4(IN,FLAG,0.,85.,5.,76.7,5.,            00010860
     120.,7.,0.)                                                        00010870
      IF(DISP.GT.100..AND.DISP.LE.200.) OUT = $$$4(IN,FLAG,0.,85.,      00010880
     15.5,75.8,5.5,20.,7.5,0.)                                          00010890
      IF(DISP.GT.200.) OUT = $$$4(IN,FLAG,0.,85.,6.,75.,6.,20.,         00010900
     18.,0.)                                                            00010910
      GO TO 999                                                         00010920
C                                                                       00010930
   11 IF(RATE.EQ.2) GO TO 1102                                          00010940
      IF(DISP.LE.100.) OUT = $$$4(IN,FLAG,0.,100.,4.5,62.5,5.18,        00010950
     120.,7.,0.)                                                        00010960
      IF(DISP.GT.100..AND.DISP.LE.200.) OUT = $$$4(IN,FLAG,0.,          00010970
     1100.,5.,58.3,5.66,20.,7.5,0.)                                     00010980
      IF(DISP.GT.200.) OUT = $$$4(IN,FLAG,0.,100.,5.5,54.2,6.3,         00010990
     120.,8.,0.)                                                        00011000
      GO TO 991                                                         00011010
C                                                                       00011020
 1102 IF(DISP.LE.100.) OUT = $$$4(IN,FLAG,0.,90.,4.72,50.5,             00011030
     15.18,20.,7.0,0.)                                                  00011040
      IF(DISP.GT.100..AND.DISP.LE.200.) OUT = $$$4(IN,FLAG,0.,90.,      00011050
     15.25,46.,5.66,20.,7.5,0.)                                         00011060
      IF(DISP.GT.200.) OUT = $$$4(IN,FLAG,0.,90.,5.8,41.5,              00011070
     16.13,20.,8.,0.)                                                   00011080
      GO TO 999                                                         00011090
C                                                                       00011100
   20 IF(RATE.EQ.2) GO TO 1020                                          00011110
      IF(DISP.LE.20.) OUT = $$$(IN,FLAG,0.,100.,3.,0.)                  00011120
      IF(DISP.GT.20..AND.DISP.LE.50.) OUT = $$$4(IN,FLAG,0.,100.,       00011130
     11.5,100.,2.5,80.,4.,0.)                                           00011140
      IF(DISP.GT.50..AND.DISP.LE.100.) OUT = $$$4(IN,FLAG,0.,           00011150
     1100.,2.,100.,3.,80.,4.5,0.)                                       00011160
      IF(DISP.GT.100..AND.DISP.LE.150.) OUT = $$$4(IN,FLAG,0.,          00011170
     1100.,2.5,100.,3.5,80.,5.0,0.)                                     00011180
```

```
      IF(DISP.LE.5.) OUT = $$$3(IN,FLAG,0.,100.,2.,83.,5.,0.)                00011820
      IF(DISP.GT.5..AND.DISP.LE.20.) OUT = $$$3(IN,FLAG,0.,100.,             00011830
     12.67,77.,5.5,0.)                                                       00011840
      IF(DISP.GT.20..AND.DISP.LE.50.) OUT = $$$3(IN,FLAG,0.,100.,            00011850
     13.34,72.,6.,0.)                                                        00011860
      IF(DISP.GT.50..AND.DISP.LE.100.) OUT = $$$3(IN,FLAG,0.,100.,           00011870
     14.,65.,6.5,0.)                                                         00011880
      IF(DISP.GT.100..AND.DISP.LE.200.) OUT = $$$3(IN,FLAG,0.,               00011890
     1100.,4.67,60.,7.,0.)                                                   00011900
      IF(DISP.GT.200.) OUT = $$$3(IN,FLAG,0.,100.,5.34,54.,7.5,0.)           00011910
      GO TO 991                                                              00011920
 1080 IF(DISP.LE.5.) OUT = $$$3(IN,FLAG,0.,60.,4.1,25.6,5.,0.)               00011930
      IF(DISP.GT.5..AND.DISP.LE.20.) OUT = $$$3(IN,FLAG,                     00011940
     10.,60.,4.85,19.5,5.5,0.)                                               00011950
      IF(DISP.GT.20..AND.DISP.LE.50.) OUT = $$$3(IN,FLAG,                    00011960
     10.,60.,5.58,13.0,6.,0.)                                                00011970
      IF(DISP.GT.50..AND.DISP.LE.100.) OUT = $$$3(IN,FLAG,0.,60.,            00011980
     16.3,6.4,6.5,0.)                                                        00011990
      IF(DISP.GT.100..AND.DISP.LE.200.) OUT = $$$(IN,FLAG,0.,60.,            00012000
     17.,0.)                                                                 00012010
      IF(DISP.GT.200.) OUT = $$$3(IN,FLAG,0.,60.,7.,0.,7.5,0.)               00012020
      GO TO 999                                                              00012030
  101 IF(RATE.EQ.2) GO TO 1101                                               00012040
      IF(DISP.LE.10.) OUT = $$$3(IN,FLAG,0.,100.,1.,96.7,5.,0.)              00012050
      IF(DISP.GT.10..AND.DISP.LE.25.) OUT = $$$3(IN,FLAG,0.,100.,            00012060
     12.,93.3,6.,0.)                                                         00012070
      IF(DISP.GT.25.) OUT = $$$3(IN,FLAG,0.,100.,3.,90.,7.,0.)               00012080
      GO TO 991                                                              00012090
 1101 IF(DISP.LE.10.) OUT = $$$3(IN,FLAG,0.,70.,2.45,62.,5.,0.)              00012100
      IF(DISP.GT.10..AND.DISP.LE.25.) OUT = $$$3(IN,FLAG,0.,85.,             00012110
     12.75,76.,6.,0.)                                                        00012120
      IF(DISP.GT.25.) OUT = $$$3(IN,FLAG,0.,100.,3.,90.,6.5,0.)              00012130
      GO TO 999                                                              00012140
C                                                                            00012150
  991 IF(RATE.EQ.1) GO TO 999                                                00012160
      IF(RATE.EQ.3) GO TO 993                                                00012170
      IF(RATE.EQ.4) GO TO 994                                                00012180
  993 PCOSMX = 12./OSPEED*100.                                               00012190
  995 IF(FLAG.EQ.0.AND.OUT.GT.PCOSMX) OUT = PCOSMX                           00012200
      IF(FLAG.EQ.1.AND.IN.GT.PCOSMX) OUT = 0.                                00012210
      GO TO 999                                                              00012220
  994 PCOSMX = 5./OSPEED*100.                                                00012230
      GO TO 995                                                              00012240
C                                                                            00012250
C     IF CALCULATED OUTPUT OF SEA STATE VS PERCENT DESIGN SPEED              00012260
C      IS LESS THAN ZERO ,SET VALUE EQUAL TO ZERO                            00012270
C                                                                            00012280
  999 IF(OUT.LT.0.) OUT = 0.                                                 00012290
      $POSSS = OUT                                                           00012300
      $SSPOS = OUT                                                           00012310
      RETURN                                                                 00012320
      END                                                                    00012330
C PROPOS                                                                     00000010
C FINDS THE PROGRAM PROBABILITY OF SUCCESS                                   00000020
C                                                                            00000030
C FOR A GIVEN FLOW CHART OF GROUPS, FINDS ALL POSSIBLE PATHS                 00000040
C  (I.E., SEQUENCES OF TASKS) THAT DO NOT VIOLATE TIME AND                   00000050
C  FUEL MAXIMA. FINDS THE PATH PROBABILITY OF SUCCESS (PTHPOS) FOR           00000060
C  EACH PATH. COMBINES THE PATH POS'S TO PRODUCE THE PROPOS.                 00000070
C                                                                            00000080
C LIMITS:                                                                    00000090
C   MAXIMUM NUMBER OF GROUPS = 40                                            00000100
C   MAXIMUM NUMBER OF NODES IN FLOWCHART = 50                                00000110
```

```
      EQUIVALENCE (TASKNM(451),TASKN3(1))                                00000670
      EQUIVALENCE (TASKNM(523),TASKN4(1))                                00000680
      DATA TASKNM/                                                       00000690
     1 'DASH','    ','    ','    ','    ','    ',                       00000700
     1 'INTE','RDIC','T   ','    ','    ','    ',                       00000710
     1 138*0.,                                                          00000720
     1 'ESCO','RT  ','    ','    ','    ','    ',                       00000730
     1 'IDEN','TIFY',' CRA','FT  ','    ','    ',                       00000740
     1 'IDEN','TIFY',' FLE','ET  ','    ','    ',                       00000750
     1 'PATR','OL  ','    ','    ','    ','    ',                       00000760
     1 'SEAR','CH F','OR F','LEET','    ','    ',                       00000770
     1 'SEAR','CH F','OR S','HIP ',': FO','UND ',                       00000780
     1 'TRAN','SPOR','T EQ','UIPM','ENT ','    ',                       00000790
     1 'TRAN','SPOR','T PE','OPLE','    ','    ',                       00000800
     1 'TRAN','SIT ','    ','    ','    ','    ',                       00000810
     1 96*0./                                                           00000820
      DATA TASKN2/                                                      00000830
     1 'SEAR','CH D','STR ','UNIT',': FO','UND ',                       00000840
     1 'SLOW',' ESC','ORT ','    ','    ','    ',                       00000850
     1 'SEAR','CH F','OR P','EOPL',': FO','UND ',                       00000860
     1 'SLOW',' PAT','ROL ','    ','    ','    ',                       00000870
     1 'TOW ','    ','    ','    ','    ','    ',                       00000880
     1 120*0./                                                          00000890
      DATA TASKN3/                                                      00000900
     1 'BOAR','D   ','    ','    ','    ','    ',                       00000910
     1 'FIGH','T FI','RE F','ROM ','CG V','ESSL',                       00000920
     1 'FGHT',' FIR','E ON',' OTH','ER V','ESSL',                       00000930
     1 'GENE','RAL ','ASSI','STAN','CE  ','    ',                       00000940
     1 'INSP','ECTI','ON  ','    ','    ','    ',                       00000950
     1 'LOAD',' EQU','IPME','NT  ','    ','    ',                       00000960
     1 'LOIT','ER  ','    ','    ','    ','    ',                       00000970
     1 'LAUN','CH S','MALL',' BOA','T   ','    ',                       00000980
     1 'MONI','TOR ','ACTI','VITI','ES  ','    ',                       00000990
     1 'MONI','TOR ','OIL ','SPIL','LS  ','    ',                       00001000
     1 'ON B','OARD',' ASS','ISTA','NCE ','    ',                       00001010
     1 'ON S','CENE',' COM','MAND','ER  ','    '/                       00001020
      DATA TASKN4/                                                      00001030
     1 'RETR','IEVE',' BOA','RDIN','G PA','RTY ',                       00001040
     1 'RETR','IEVE',' OPS','ECTS','    ','    ',                       00001050
     1 'RESC','UE P','EOPL','E   ','    ','    ',                       00001060
     1 'RETR','IEVE',' SMA','LL B','OAT ','    ',                       00001070
     1 'STAK','EOUT',' SPE','C IN','T VE','SSL ',                       00001080
     1 'SEIZ','E   ','    ','    ','    ','    ',                       00001090
     1 'TAKE',' WAT','ER S','AMPL','E   ','    ',                       00001100
     1 'UNLO','AD E','QUIP','MENT','    ','    ',                       00001110
     1 'WORK',' EQU','IP F','ROM ','SM B','OAT ',                       00001120
     1 'WORK',' EQU','IPME','NT @',' DRI','FT  ',                       00001130
     1 'WORK',' EQU','IP @',' FIX','ED P','OS  ',                       00001140
     1 12*0./                                                           00001150
      DIMENSION GRPNM(4,20),GRPNM2(4,10)                                00001160
      EQUIVALENCE(GRPNM(41),GRPNM2(1))                                  00001170
      DATA GRPNM/                                                       00001180
     1 'ASSI','ST  ','    ','    ',                                     00001190
     1 'ESCO','RT  ','    ','    ',                                     00001200
     1 'FIGH','T FI','RE  ','    ',                                     00001210
     1 'IDEN','TIFY','    ','    ',                                     00001220
     1 'INSP','ECT ','    ','    ',                                     00001230
     1 'MONI','TOR ','    ','    ',                                     00001240
     1 'PATR','OL  ','    ','    ',                                     00001250
     1 'RESC','UE  ','    ','    ',                                     00001260
     1 'RESC','UE R','ETUR','N   ',                                     00001270
     1 'SAR ','SEAR','CH  ','    '/                                     00001280
      DATA GRPNM2/                                                      00001290
```

```
      WRITE(6,802)(MTASK(I),I=1,NMIMTK)                                 00001930
802   FORMAT((10(1X,I3)))                                              00001940
      DO 509 I=1,NMIMTK                                                 00001950
      IF(IMRATE(I).EQ.0)GO TO 510                                       00001960
      IMPTSK(IMTSKN(I),IMRATE(I))=1                                     00001970
509   CONTINUE                                                          00001980
510   CONTINUE                                                          00001990
C                                                                       00002000
C READ NUMBER OF NODES                                                  00002010
C FORMAT ' NODES=NN'                                                    00002020
      READ (13,123)NNODOV                                               00002030
123   FORMAT(7X,I2)                                                     00002040
      WRITE(6,126)NNODOV                                                00002050
126   FORMAT(' NODES=',I2)                                              00002060
C                                                                       00002070
C READ CONNECTION MATRIX                                                00002080
C FORMAT: ' P.PP P.PP P.PP     ...'                                     00002090
C   THE NUMBER OF PROBABILITIES PER LINE = THE NUMBER OF NODES.         00002100
C   IF >15, USE TWO LINES WITH 15 PROBABILITIES ON THE FIRST LINE.      00002110
      READ(13,124)                                                      00002120
124   FORMAT()                                                          00002130
      WRITE(6,120)                                                      00002140
120   FORMAT(' CONNECTION MATRIX=')                                     00002150
      DO 108 I=1,NNODOV                                                 00002160
      READ(13,107)(OVCNMX(I,J),J=1,NNODOV)                              00002170
107   FORMAT(15(1X,F4.2))                                               00002180
      WRITE(6,107)(OVCNMX(I,J),J=1,NNODOV)                              00002190
108   CONTINUE                                                          00002200
C                                                                       00002210
C READ GROUP PLACEMENT MATRIX                                           00002220
C FORMAT: ' GGII GGII GGII     ...'                                     00002230
C   THE NUMBER OF GROUPS PER LINE = THE NUMBER OF NODES.                00002240
C   IF >15, USE TWO LINES WITH 15 GROUPS ON THE FIRST LINE.             00002250
      READ(13,124)                                                      00002260
      WRITE(6,121)                                                      00002270
121   FORMAT(' GROUP PLACEMENT MATRIX=')                                00002280
      DO 118 I=1,NNODOV                                                 00002290
      READ(13,117)(GPPLMX(I,J),J=1,NNODOV)                              00002300
117   FORMAT(15(1X,I4))                                                 00002310
      WRITE(6,117)(GPPLMX(I,J),J=1,NNODOV)                              00002320
118   CONTINUE                                                          00002330
C                                                                       00002340
C READ GROUP DATA                                                       00002350
      READ(13,GROUP)                                                    00002360
      DO 100 I=1,25                                                     00002370
      DO 101 J=1,18                                                     00002380
      GPDAT2(I,J)=DATA(J+2,I)                                           00002390
101   CONTINUE                                                          00002400
100   CONTINUE                                                          00002410
      DO 103I=1,25                                                      00002420
      DO 104J=1,2                                                       00002430
      GPDAT1(I,J)=DATA(J,I)                                             00002440
104   CONTINUE                                                          00002450
103   CONTINUE                                                          00002460
      WRITE(6,122)                                                      00002470
122   FORMAT(' &GROUP DATA=')                                           00002480
      DO 106 I=1,25                                                     00002490
      IF(GPDAT1(I,1).EQ.0)GOTO 105                                      00002500
      WRITE(6,109)(GPDAT1(I,J),J=1,2),(GPDAT2(I,J),J=1,18)              00002510
109   FORMAT(1X,I2,1X,I2,18(F7.2))                                      00002520
106   CONTINUE                                                          00002530
105   WRITE(6,129)                                                      00002540
129   FORMAT(' &END')                                                   00002550
```

```
C                                                                       00003590
C                                                                       00003600
C START AT OVERALL NODE 1                                               00003610
      PROVND=000001                                                     00003620
      LSOVGO=0                                                          00003630
      CALL $PUSH(000001,0.,0.,1.)                                       00003640
C GET NEXT OVERALL NODE                                                 00003650
10    IF(MINTIM(PROVND)+PTHTIM.GT.MXTIME)GO TO 90                       00003660
      IF(MINFUE(PROVND)+PTHFUE.GT.MXGALS)GO TO 90                       00003670
      BEGIN=LSOVGO+1                                                    00003680
      IF (BEGIN.GT.NNODOV)GOTO 90                                       00003690
      DO 22 J=BEGIN,NNODOV                                              00003700
      OVPROB=OVCNMX(PROVND,J)                                           00003710
      IF(OVPROB.GT.0.)GO TO 24                                          00003720
22    CONTINUE                                                          00003730
      OVPROB=1.                                                         00003740
      GOTO 90                                                           00003750
C FOUND AN OVERALL NODE TO GO TO                                        00003760
24    NXOVND=J                                                          00003770
C GET THIS LINK'S GROUP (IF ANY) AND START AT GROUPNODE 1               00003780
20    LNKSGP=GPPLMX(PROVND,NXOVND)                                      00003790
      IF (LNKSGP.EQ.0)GOTO40                                            00003800
      PRGPND=100*LNKSGP + 1                                             00003810
      LSGPGO=0                                                          00003820
      CALL $PUSH(PRGPND,0.,0.,OVPROB)                                   00003830
      PTHPRB=PTHPRB*OVPROB                                              00003840
      OVPROB=1.                                                         00003850
C GET NEXT GROUP NODE IN PRESENT GROUP                                  00003860
60    BEGIN=$NODE(LSGPGO)+1                                             00003870
      GROUP=$GROUP(PRGPND)                                              00003880
      INST=$INST(PRGPND)                                                00003890
      NODE=$NODE(PRGPND)                                                00003900
      IF(GROUP.GE.90)GO TO 890                                          00003910
      NUMNDS=NNODE(GROUP)                                               00003920
      GO TO 891                                                         00003930
890   NUMNDS=9                                                          00003940
891   IF(BEGIN.GT.NUMNDS)GOTO 80                                        00003950
      DO 61 J=BEGIN,NUMNDS                                              00003960
      CALL $LKDAT(GROUP,INST,NODE,J,LKPROB,LKTIME,LKFUEL)               00003970
C     CHECK FOR TIME AND FUEL                                           00003980
      IF( LKPROB.GT.0. .AND. (PTHTIM+LKTIME).LE.MXTIME                  00003990
     1          .AND. (PTHFUE+LKFUEL).LE.MXGALS )  GOTO 62             00004000
61    CONTINUE                                                          00004010
      GOTO 80                                                           00004020
C FOUND A GROUP NODE TO GO TO IN PRESENT GROUP                          00004030
62    NODE=J                                                            00004040
      NXGPND=$PACK(GROUP,INST,NODE)                                     00004050
      CALL $PUSH(NXGPND,LKTIME,LKFUEL,LKPROB)                           00004060
      PTHTIM=PTHTIM+LKTIME                                              00004070
      PTHFUE=PTHFUE+LKFUEL                                              00004080
      PTHPRB=PTHPRB*LKPROB                                              00004090
C TEST IF END OF PATH IN GROUP                                          00004100
      IF(NODE.EQ.2 .OR. NODE.EQ.9)GO TO 40                              00004110
      PRGPND=NXGPND                                                     00004120
      LSGPGO=0                                                          00004130
      GOTO 60                                                           00004140
C NO GROUP FOR THIS OVERALL LINK -OR- FINISHED THIS LINK                00004150
40    IF(NXOVND.EQ.000002)GOTO 42                                       00004160
      LSOVGO=0                                                          00004170
      PROVND=NXOVND                                                     00004180
      CALL $PUSH(NXOVND,0.,0.,OVPROB)                                   00004190
      PTHPRB=PTHPRB*OVPROB                                              00004200
      GOTO 10                                                           00004210
```

```
 202   FORMAT(//10X,' GROUP',5X,'TASK',18X,'LOCATION',2X,'TASK',4X,'TASK'00004850
      1  ,4X,'TASK'/1X,10X,'NAME',6X,'NAME',20X,'CODE',4X,'TIME',       00004860
      2   4X,'FUEL',5X,'POS'/                                           00004870
      3   1X,52X,'(HRS)',2X,'(GALS)'//)                                 00004880
       POSPRD=1.                                                        00004890
       CCMIN=9999.                                                      00004900
       DFMIN=9999.                                                      00004910
       MNMIN=9999.                                                      00004920
       LSMIN=9999.                                                      00004930
       TWMIN=9999.                                                      00004940
C ZERO OUT TASK COUNTER AFTER A CRAFT HAS                               00004950
C  COMPLETED A SORTIE                                                   00004960
       DO 421 I=1,25                                                    00004970
       DO 422 J=1,4                                                     00004980
       COUNT(I,J)=0                                                     00004990
 422   CONTINUE                                                         00005000
 421   CONTINUE                                                         00005010
       NTASK=0                                                          00005020
       DO 47 I=1,PTR                                                    00005030
       LOCATN=PSHLST(I)                                                 00005040
       INST=$INST(LOCATN)                                               00005050
       GROUP=$GROUP(LOCATN)                                             00005060
       IF(GROUP.EQ.0)GOTO 204                                           00005070
       NODE=$NODE(LOCATN)                                               00005080
       IF(NODE.EQ.1)GOTO205                                             00005090
       NODE1=NODE2                                                      00005100
       NODE2=NODE                                                       00005110
       CALL $TASK(GROUP,NODE1,NODE2,TASKNO,RATE)                        00005120
       TASKN1=TASKNO                                                    00005130
       IF(GROUP.GE.90)TASKN1=TASKN1+19                                  00005140
       RATE1=RATE                                                       00005150
       IF(GROUP.GE.90)RATE1=1                                           00005160
C NOTE: COUNTER FOR THE 3 SEARCH FAILURES ARE STORED                    00005170
C  IN RATE 1 AND TASKNOS 20,22 AND 25                                   00005180
       IF(TASKNO.EQ.0)GOTO204                                           00005190
C TASK IS PERFORMED INCREMENT TASK COUNTER                              00005200
       COUNT(TASKN1,RATE1)=COUNT(TASKN1,RATE1)+1                        00005210
 88    MASTSK=MASTER(TASKNO,RATE)                                       00005220
C FIND CC FOR MASTER TASK 17                                            00005230
C FIND MINIMUM VALUE OF EACH PARAMETER: CC,DF,MN,LS,TW                  00005240
       CC0=CC(MASTSK)                                                   00005250
       IF(MASTSK.EQ.17)CALL $CC17(GROUP,INST,NODE1,NODE2,DECK,CARGCP,CC0)00005260
       IF(CC0.LE.CCMIN)CCMIN=CC0                                        00005270
       IF(DF(MASTSK).LE.DFMIN)DFMIN=DF(MASTSK)                          00005280
       IF(MN(MASTSK).LE.MNMIN)MNMIN=MN(MASTSK)                          00005290
       IF(LS(MASTSK).LE.LSMIN)LSMIN=LS(MASTSK)                          00005300
       IF(TW(MASTSK).LE.TWMIN)TWMIN=TW(MASTSK)                          00005310
       TPOS=TPOSMX(MASTSK)                                              00005320
       IF(MASTSK.EQ.17)TPOS=CC0*DF(17)*MN(17)*LS(17)*TW(17)             00005330
       IFULST=FUELST(I)                                                 00005340
       IF(GROUP.GE.90)GO TO 896                                         00005350
       IF(FLAG.EQ.1)WRITE(6,48)(TASKNM(J,TASKNO,RATE),J=1,6),           00005360
      1  LOCATN,TIMLST(I),IFULST,TPOS                                   00005370
 48    FORMAT(1X,16X,'*',6A4,2X,I6,2X,F5.1,2X,I6,3X,F5.2)               00005380
       GO TO 825                                                        00005390
C TASK FAILURE                                                          00005400
 896   IF(FLAG.EQ.1)WRITE(6,893)(TASKNM(J,TASKNO,RATE),J=1,4),          00005410
      1  LOCATN,TIMLST(I),IFULST,TPOS                                   00005420
 893   FORMAT(1X,16X,'*',4A4,': FAILED',2X,I6,2X,F5.1,2X,I6,3X,F5.2)    00005430
 825   NTASK=NTASK+1                                                    00005440
       GOTO 47                                                          00005450
 205   BASGRP=GROUP                                                     00005460
       IF(GROUP.GE.90)BASGRP=GROUP-80                                   00005470
```

```
C CALCULATE TIME TO COMPLETE AVERAGE SORTIE AND                          00006110
C  FUEL CONSUMED IN AVERAGE SORTIE                                       00006120
      AVETIM=TOTIM/PROPOS                                               00006130
      AVEFUE=TOTFUE/PROPOS                                              00006140
C                                                                       00006150
C PRINT OVERALL RESULTS                                                 00006160
C                                                                       00006170
      WRITE(6,480)PROGRM,SCENNO                                         00006180
480   FORMAT('1'////                                                    00006190
     X 13X,'************* SCENARIO OVERALL RESULTS *************'//    00006200
     1   30X,A4,' SCENARIO ',I2/)                                       00006210
      WRITE(6,211)MXTIME,CFTNAM,RANGER,IDTSP,                           00006220
     1 (VISDS2(VISDTB,I),I=1,3),IDSPD,SSAVG,FUFRAC                      00006230
      PERPRB=TOTPRB*100.                                                00006240
      WRITE(6,405)PERPRB                                                00006250
405   FORMAT(//1X,22X,' PERCENT OF SCENARIO COMPLETED  ',F5.1/)         00006260
      WRITE(6,406)PROPOS                                                00006270
406   FORMAT(1X,                                                        00006280
     1 14X,'PROBABILITY OF SUCCESSFULLY COMPLETING SCENARIO ',          00006290
     2 2X,F4.2//)                                                       00006300
      WRITE(6,407)                                                      00006310
407   FORMAT(1X,14X,'SPECIFICATIONS OF THE AVERAGE SORTIE:'/)           00006320
      WRITE(6,408)AVETIM                                                00006330
408   FORMAT(1X,22X,'TIME TO COMPLETE AVERAGE SORTIE',F8.1,' HRS'/)     00006340
      WRITE(6,409)AVEFUE                                                00006350
409   FORMAT(1X,22X,'FUEL CONSUMED IN AVERAGE SORTIE',F8.1,' GALS'/)    00006360
      WRITE(6,410)                                                      00006370
410   FORMAT(//1X,                                                      00006380
     1 14X,'TASK COMPOSITION IN AVERAGE SORTIE:'/)                       00006390
      WRITE(6,522)                                                      00006400
522   FORMAT(1X,22X,'TASK',6X,'TIMES',5X,'TASK')                        00006410
      WRITE(6,523)                                                      00006420
523   FORMAT(1X,22X,'CODE',4X,'COMPLETED',3X,'NAME')                    00006430
      WRITE(6,7002)                                                     00006440
 7002 FORMAT(/1X,18X,'ON SCENE:')                                       00006450
      IF(TOTCNT(1,4).GT.0.)WRITE(6,7003) TOTCNT(1,4)                    00006460
 7003 FORMAT(1X,22X,'BRD',6X,F5.2,5X,'BOARD')                           00006470
      IF(TOTCNT(2,4).GT.0.)WRITE(6,7004)TOTCNT(2,4)                     00006480
 7004 FORMAT(1X,22X,'FFF',6X,F5.2,5X,'FIGHT FIRE FROM CG VESSEL')       00006490
      IF(TOTCNT(3,4).GT.0.)WRITE(6,7005)TOTCNT(3,4)                     00006500
 7005 FORMAT(1X,22X,'FFO',6X,F5.2,5X,'FIGHT FIRE ON ANOTHER VESSEL')    00006510
      IF(TOTCNT(4,4).GT.0.)WRITE(6,7006)TOTCNT(4,4)                     00006520
 7006 FORMAT(1X,22X,'GAS',6X,F5.2,5X,'GENERAL ASSISTANCE')              00006530
      IF(TOTCNT(5,4).GT.0.)WRITE(6,7007)TOTCNT(5,4)                     00006540
 7007 FORMAT(1X,22X,'INS',6X,F5.2,5X,'INSPECTION')                      00006550
      IF(TOTCNT(6,4).GT.0.)WRITE(6,7008)TOTCNT(6,4)                     00006560
 7008 FORMAT(1X,22X,'LEQ',6X,F5.2,5X,'LOAD EQUIPMENT')                  00006570
      IF(TOTCNT(7,4).GT.0.)WRITE(6,7009)TOTCNT(7,4)                     00006580
 7009 FORMAT(1X,22X,'LOI',6X,F5.2,5X,'LOITER')                          00006590
      IF(TOTCNT(8,4).GT.0.)WRITE(6,7010)TOTCNT(8,4)                     00006600
 7010 FORMAT(1X,22X,'LSB',6X,F5.2,5X,'LAUNCH SMALL BOAT')               00006610
      IF(TOTCNT(9,4).GT.0.)WRITE(6,7011)TOTCNT(9,4)                     00006620
 7011 FORMAT(1X,22X,'MAC',6X,F5.2,5X,'MONITOR ACTIVITIES')              00006630
      IF(TOTCNT(10,4).GT.0.)WRITE(6,7012)TOTCNT(10,4)                   00006640
 7012 FORMAT(1X,22X,'MOS',6X,F5.2,5X,'MONITOR OIL SPILL')               00006650
      IF(TOTCNT(11,4).GT.0.)WRITE(6,7013)TOTCNT(11,4)                   00006660
 7013 FORMAT(1X,22X,'OBA',6X,F5.2,5X,'ON BOARD ASSISTANCE')             00006670
      IF(TOTCNT(12,4).GT.0.)WRITE(6,7015)TOTCNT(12,4)                   00006680
 7015 FORMAT(1X,22X,'OSC',6X,F5.2,5X,'ON SCENE COMMANDER(GENERAL)')     00006690
      IF(TOTCNT(13,4).GT.0.)WRITE(6,7016) TOTCNT(13,4)                  00006700
 7016 FORMAT(1X,22X,'RBP',6X,F5.2,5X,'RETRIEVE BOARDING PARTY')         00006710
      IF(TOTCNT(14,4).GT.0.)WRITE(6,7017)TOTCNT(14,4)                   00006720
 7017 FORMAT(1X,22X,'ROB',6X,F5.2,5X,'RETRIEVE OBJECTS')                00006730
```

```
 3032  FORMAT(1X,22X,'SSH',6X,F5.2,5X,'SEARCH FOR SHIP: FAILED')          00007370
       IF(TOTCNT(7,2).GT.0.)WRITE(6,8015) TOTCNT(7,2)                     00007380
  8015 FORMAT(1X,22X,'TEQ',6X,F5.2,5X,'TRANSPORT EQUIPMENT')              00007390
       IF(TOTCNT(8,2).GT.0.)WRITE(6,8016)TOTCNT(8,2)                      00007400
  8016 FORMAT(1X,22X,'TPE',6X,F5.2,5X,'TRANSPORT PEOPLE')                 00007410
       IF(TOTCNT(9,2).GT.0.)WRITE(6,8017)TOTCNT(9,2)                      00007420
  8017 FORMAT(1X,22X,'TRA',6X,F5.2,5X,'TRANSIT')                          00007430
       DO 496 I=1,9                                                       00007440
       IF(TOTCNT(I,2).GT.0.)GO TO 497                                     00007450
  496  CONTINUE                                                           00007460
       WRITE(6,7027)                                                      00007470
C                                                                         00007480
  497  WRITE(6,8018)                                                      00007490
  8018 FORMAT(/1X,18X,'FLANK SPEED:')                                     00007500
       IF(TOTCNT(1,1).GT.0.)WRITE(6,8019)TOTCNT(1,1)                      00007510
  8019 FORMAT(1X,22X,'DSH',6X,F5.2,5X,'DASH')                             00007520
       IF(TOTCNT(2,1).GT.0.)WRITE(6,8020)TOTCNT(2,1)                      00007530
  8020 FORMAT(1X,22X,'INT',6X,F5.2,5X,'INTERDICT')                        00007540
       DO 501 I=1,2                                                       00007550
       IF(TOTCNT(I,1).GT.0.)GO TO 502                                     00007560
  501  CONTINUE                                                           00007570
       WRITE(6,7027)                                                      00007580
  502  CONTINUE                                                           00007590
C                                                                         00007600
C PRINT EVALUATION                                                        00007610
C                                                                         00007620
       WRITE(6,482)PROGRM,SCENNO                                          00007630
  482  FORMAT('1'////                                                     00007640
      X 16X,'************* SCENARIO EVALUATION **************'//         00007650
      1   30X,A4,' SCENARIO ',I2/)                                        00007660
       WRITE(6,211)MXTIME,CFTNAM,RANGER,IDISP,                            00007670
      1 (VISOS2(VISDTB,I),I=1,3),IDSPD,SSAVG,FUFRAC                       00007680
       DO 503 I=1,25                                                      00007690
       DO 504 J=1,4                                                       00007700
       IM(I,J)=TOTCNT(I,J)*NDAYS+.500001                                  00007710
  504  CONTINUE                                                           00007720
  503  CONTINUE                                                           00007730
       WRITE(6,511)NDAYS                                                  00007740
  511  FORMAT(//1X,                                                       00007750
      X 17X,'IMPORTANT TASKS COMPLETED IN ',                              00007760
      X I4,' DAYS OF OPERATION'//)                                        00007770
       WRITE(6,512)                                                       00007780
  512  FORMAT(1X,22X,'TASK',6X,'TIMES',5X,'TASK')                         00007790
       WRITE(6,513)                                                       00007800
  513  FORMAT(1X,22X,'CODE',4X,'COMPLETED',3X,'NAME')                     00007810
       WRITE(6,2002)                                                      00007820
 2002  FORMAT(/1X,18X,'ON SCENE:')                                        00007830
C IF AN IMPORTANT TASK IS NOT PERFORMED, ITS TASK CODE                    00007840
C AND TASK NAME WILL STILL APPEAR IN THE OUTPUT, WITH                     00007850
C THE NUMBER OF TIMES COMPLETED BEING 0.0. IF IT IS                       00007860
C DESIRED AN IMPORTANT TASK NOT PERFORMED WILL NOT APPEAR                 00007870
C IN THE OUTPUT.                                                          00007880
       IF(IM(1,4).GE.0..AND.IMPTSK(1,4).EQ.1)WRITE(6,2003) IM(1,4)        00007890
 2003  FORMAT(1X,22X,'BRD',6X,I5,5X,'BOARD')                              00007900
       IF(IM(2,4).GE.0..AND.IMPTSK(2,4).EQ.1)WRITE(6,2004)IM(2,4)         00007910
 2004  FORMAT(1X,22X,'FFF',6X,I5,5X,'FIGHT FIRE FROM CG VESSEL')         00007920
       IF(IM(3,4).GE.0..AND.IMPTSK(3,4).EQ.1)WRITE(6,2005)IM(3,4)         00007930
 2005  FORMAT(1X,22X,'FFO',6X,I5,5X,'FIGHT FIRE ON ANOTHER VESSEL')      00007940
       IF(IM(4,4).GE.0..AND.IMPTSK(4,4).EQ.1)WRITE(6,2006)IM(4,4)         00007950
 2006  FORMAT(1X,22X,'GAS',6X,I5,5X,'GENERAL ASSISTANCE')                 00007960
       IF(IM(5,4).GE.0..AND.IMPTSK(5,4).EQ.1)WRITE(6,2007)IM(5,4)         00007970
 2007  FORMAT(1X,22X,'INS',6X,I5,5X,'INSPECTION')                         00007980
       IF(IM(6,4).GE.0..AND.IMPTSK(6,4).EQ.1)WRITE(6,2008)IM(6,4)         00007990
```

```
      WRITE(6,2027)                                                     00008590
C                                                                       00008600
695   WRITE(6,3008)                                                     00008610
 3008 FORMAT(/1X,18X,'CRUISE SPEED:')                                   00008620
      IF(IM(1,2).GE.0..AND.IMPTSK(1,2).EQ.1)WRITE(6,3030)IM(1,2)        00008630
 3030 FORMAT(1X,22X,'ESC',6X,I5,5X,'ESCORT')                            00008640
      IF(IM(2,2).GE.0..AND.IMPTSK(2,2).EQ.1)WRITE(6,3009)IM(2,2)        00008650
 3009 FORMAT(1X,22X,'IDC',6X,I5,5X,'IDENTIFY CRAFT')                    00008660
      IF(IM(3,2).GE.0..AND.IMPTSK(3,2).EQ.1)WRITE(6,3010)IM(3,2)        00008670
 3010 FORMAT(1X,22X,'IDF',6X,I5,5X,'IDENTIFY FLEET')                    00008680
      IF(IM(4,2).GE.0..AND.IMPTSK(4,2).EQ.1)WRITE(6,3011)IM(4,2)        00008690
 3011 FORMAT(1X,22X,'PAT',6X,I5,5X,'PATROL')                            00008700
      IF(IM(5,2).GE.0..AND.IMPTSK(5,2).EQ.1)WRITE(6,3035)IM(5,2)        00008710
 3035 FORMAT(1X,22X,'SFL',6X,I5,5X,'SEARCH FOR FLEET')                  00008720
      IF(IM(6,2).GE.0..AND.IMPTSK(6,2).EQ.1)WRITE(6,3014)IM(6,2),       00008730
     X IM(25,1)                                                         00008731
 3014 FORMAT(1X,22X,'SSH',6X,I5,5X,'SEARCH FOR SHIP: FOUND'/            00008740
     X 1X,22X,'SSH',6X,I5,5X,'SEARCH FOR SHIP: FAILED')                 00008741
      IF(IM(7,2).GE.0..AND.IMPTSK(7,2).EQ.1)WRITE(6,3015) IM(7,2)       00008750
 3015 FORMAT(1X,22X,'TEQ',6X,I5,5X,'TRANSPORT EQUIPMENT')               00008760
      IF(IM(8,2).GE.0..AND.IMPTSK(8,2).EQ.1)WRITE(6,3016)IM(8,2)        00008770
 3016 FORMAT(1X,22X,'TPE',6X,I5,5X,'TRANSPORT PEOPLE')                  00008780
      IF(IM(9,2).GE.0..AND.IMPTSK(9,2).EQ.1)WRITE(6,3017)IM(9,2)        00008790
 3017 FORMAT(1X,22X,'TRA',6X,I5,5X,'TRANSIT')                           00008800
      DO 696 I=1,9                                                      00008810
      IF(IM(I,2).GE.0..AND.IMPTSK(I,2).EQ.1)GO TO 697                   00008820
696   CONTINUE                                                          00008830
      WRITE(6,2027)                                                     00008840
C                                                                       00008850
697   WRITE(6,3018)                                                     00008860
 3018 FORMAT(/1X,18X,'FLANK SPEED:')                                    00008870
      IF(IM(1,1).GE.0..AND.IMPTSK(1,1).EQ.1)WRITE(6,3019)IM(1,1)        00008880
 3019 FORMAT(1X,22X,'DSH',6X,I5,5X,'DASH')                              00008890
      IF(IM(2,1).GE.0..AND.IMPTSK(2,1).EQ.1)WRITE(6,3020)IM(2,1)        00008900
 3020 FORMAT(1X,22X,'INT',6X,I5,5X,'INTERDICT')                         00008910
      DO 701 I=1,2                                                      00008920
      IF(IM(I,1).GE.0..AND.IMPTSK(I,1).EQ.1)GO TO 702                   00008930
701   CONTINUE                                                          00008940
      WRITE(6,2027)                                                     00008950
702   CONTINUE                                                          00008960
481   GOTO 9990                                                         00008970
9994  WRITE(6,9993)                                                     00008980
9993  FORMAT(//                                                         00008990
     1   1X,23X,'NO SORTIES CAN BE COMPLETED')                          00009000
C                                                                       00009010
9990  CONTINUE                                                          00009020
3001  CONTINUE                                                          00009030
C                                                                       00009040
4999  CONTINUE                                                          00009050
      RETURN                                                            00009060
      END                                                               00009070
C                                                                       00000010
C PUSHDOWN LIST SUBROUTINES                                             00000020
C                                                                       00000030
C $PUSH                                                                 00000040
C                                                                       00000050
C PUT A NUMBER ON THE PUSHDOWN LIST                                     00000060
      SUBROUTINE $PUSH(ENTRY,LKTIME,LKFUEL,LKPROB)                      00000070
      IMPLICIT REAL(A-Z)                                                00000080
      INTEGER PSHLST,PTR,I,ENTRY                                        00000090
      COMMON/PSHDWN/PTR,PSHLST(100),TIMLST(100),FUELST(100),PRBLST(100) 00000100
      PTR=PTR+1                                                         00000110
      PSHLST(PTR)=ENTRY                                                 00000120
```

```
C                                                                       00000760
C $INST                                                                 00000770
C                                                                       00000780
C-----------------------------------------------------------------------00000790
C                                                                       00000800
      INTEGER FUNCTION $INST(Z)                                         00000810
      IMPLICIT INTEGER(A-Z)                                             00000820
      $INST=Z/100-(Z/10000)*100                                         00000830
      RETURN                                                            00000840
      END                                                               00000850
C                                                                       00000860
C $NODE                                                                 00000870
C                                                                       00000880
C-----------------------------------------------------------------------00000890
      INTEGER FUNCTION $NODE(Z)                                         00000900
      IMPLICIT INTEGER(A-Z)                                             00000910
      $NODE=Z-(Z/100)*100                                               00000920
      RETURN                                                            00000930
      END                                                               00000940
C                                                                       00000950
C                                                                       00000960
C-----------------------------------------------------------------------00000970
C $HOVND                                                                00000980
C                                                                       00000990
C FINDS HIGHEST (CLOSEST TO TOP) OVERALL NODE ON PUSHDOWN LIST.         00001000
C    TOP OF PUSHDOWN LIST IS ASSUMED TO BE A GROUP NODE.                00001010
C                                                                       00001020
      INTEGER FUNCTION $HOVND(DUMMY)                                    00001030
C                                                                       00001040
      IMPLICIT INTEGER(A-Z)                                             00001050
      COMMON/PSHDWN/PTR,PSHLST(100)                                     00001060
C                                                                       00001070
      DO 10  I=1,100                                                    00001080
      IF(PSHLST(PTR-I).LT.100000)GOTO 20                                00001090
10    CONTINUE                                                          00001100
20    $HOVND=PSHLST(PTR-I)                                              00001110
      RETURN                                                            00001120
      END                                                               00001130
C                                                                       00001140
C-----------------------------------------------------------------------00001150
C                                                                       00001160
C $PRVTM                                                                00001170
C                                                                       00001180
C FINDS TIME OF THE PREVIOUS (CLOSEST TO TOP) TASK ON PUSHDOWN LIST.    00001190
C                                                                       00001200
      FUNCTION $PRVTM(DUMMY)                                            00001210
C                                                                       00001220
      IMPLICIT REAL(A-Z)                                                00001230
      INTEGER I,PTR,PSHLST                                              00001240
      COMMON/PSHDWN/PTR,PSHLST(100),TIMLST(100)                         00001250
C                                                                       00001260
      DO 10  I=1,100                                                    00001270
      IF((PTR-I).LE.0)GO TO 30                                          00001271
      IF(TIMLST(PTR-I).GT.0.)GOTO 20                                    00001280
10    CONTINUE                                                          00001290
20    $PRVTM=TIMLST(PTR-I)                                              00001300
      RETURN                                                            00001310
30    $PRVTM=0.                                                         00001311
      RETURN                                                            00001312
      END                                                               00001320
C                                                                       00001330
C $LKDAT                                                                00001340
C                                                                       00001350
```

```
      LKFUEL=LKTIME*MFULRT(4)                                          00001970
      RETURN                                                           00001980
134   IF(NODE1.NE.3.OR.NODE2.NE.4)GO TO 142                            00001990
      T2=GPDAT2(ROW,6)                                                 00002000
      LKTIME=T2                                                        00002010
      LKFUEL=LKTIME*MFULRT(4)                                          00002020
      RETURN                                                           00002030
142   IF(NODE1.NE.4.OR.NODE2.NE.2)GO TO 167                            00002040
      T3=GPDAT2(ROW,7)                                                 00002050
      LKTIME=T3*$SKTIM(2)*$MNTIM(2)                                    00002060
      LKFUEL=LKTIME*MFULRT(4)                                          00002070
      RETURN                                                           00002080
167   IF(NODE1.NE.6.OR.NODE2.NE.7)GO TO 172                            00002090
      T7=GPDAT2(ROW,11)                                                00002100
      LKTIME=T7                                                        00002110
      LKFUEL=LKTIME*MFULRT(4)                                          00002120
      RETURN                                                           00002130
172   IF(NODE1.NE.7.OR.NODE2.NE.2)GO TO 199                            00002140
      T8=GPDAT2(ROW,12)                                                00002150
      LKTIME=T8*$SKTIM(7)*$MNTIM(7)                                    00002160
      LKFUEL=LKTIME*MFULRT(4)                                          00002170
      RETURN                                                           00002180
199   RETURN                                                           00002190
C                                                                      00002200
C                                                                      00002210
C                                                                      00002220
C 2. ESCORT GROUP                                                      00002230
C                                                                      00002240
2     CONTINUE                                                         00002250
      DATA GP2PB/3*0., 91.,0.,1., 92.,2*0./                            00002260
      GP2PB(1,2)=GPDAT2(ROW,1)                                         00002270
      GP2PB(1,3)=GPDAT2(ROW,2)                                         00002280
      LKPROB=GP2PB(NODE1,NODE2)                                        00002290
      IF(LKPROB.EQ.0.)RETURN                                           00002300
212   IF(NODE1.NE.1.OR.NODE2.NE.2)GO TO 213                            00002310
      D1=GPDAT2(ROW,3)                                                 00002320
      V1=GPDAT2(ROW,4)                                                 00002330
      LKTIME=D1/V1                                                     00002340
      LKFUEL=LKTIME*MFULRT(3)                                          00002350
      RETURN                                                           00002360
213   IF(NODE1.NE.1.OR.NODE2.NE.3)GO TO 299                            00002370
      D2=GPDAT2(ROW,5)                                                 00002380
      LKTIME=D2/SPEED(2)                                               00002390
      LKFUEL=LKTIME*MFULRT(2)                                          00002400
      RETURN                                                           00002410
299   RETURN                                                           00002420
C                                                                      00002430
C                                                                      00002440
C                                                                      00002450
C 3. FIGHT FIRE GROUP                                                  00002460
C                                                                      00002470
3     CONTINUE                                                         00002480
      DATA GP3PB/4*0., 92.,2*0.,1., 91.,3*0., 2*0.,1.,0./              00002490
      GP3PB(1,2)=GPDAT2(ROW,2)                                         00002500
      GP3PB(1,3)=GPDAT2(ROW,1)                                         00002510
      LKPROB=GP3PB(NODE1,NODE2)                                        00002520
      IF(LKPROB.EQ.0.)RETURN                                           00002530
      IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 313                          00002540
      T4=GPDAT2(ROW,6)                                                 00002550
      LKTIME=T4*$SKTIM(7)*$MNTIM(7)                                    00002560
      LKFUEL=LKTIME*MFULRT(4)                                          00002570
      RETURN                                                           00002580
313   IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 334                          00002590
```

```
      T2=GPDAT2(ROW,4)                                                  00003230
      LKTIME=T2                                                         00003240
      LKFUEL=LKTIME*MFULRT(4)                                           00003250
      RETURN                                                            00003260
542   IF(NODE1.NE.4 .OR. NODE2.NE.2)GO TO 556                           00003270
      T3=GPDAT2(ROW,5)                                                  00003280
      LKTIME=T3*$SKTIM(7)*$MNTIM(7)                                     00003290
      LKFUEL=LKTIME*MFULRT(4)                                           00003300
      RETURN                                                            00003310
556   IF(NODE1.NE.5 .OR. NODE2.NE.6) GO TO 562                          00003320
      T5=GPDAT2(ROW,7)                                                  00003330
      LKTIME=T5                                                         00003340
      LKFUEL=LKTIME*MFULRT(4)                                           00003350
      RETURN                                                            00003360
562   IF(NODE1.NE.6 .OR. NODE2.NE.2)GO TO 599                           00003370
      T6=GPDAT2(ROW,8)                                                  00003380
      LKTIME=T6*$SKTIM(2)*$MNTIM(2)                                     00003390
      LKFUEL=LKTIME*MFULRT(4)                                           00003400
      RETURN                                                            00003410
599   RETURN                                                            00003420
C                                                                       00003430
C                                                                       00003440
C                                                                       00003450
C 6. MONITOR GROUP                                                      00003460
C                                                                       00003470
6     CONTINUE                                                          00003480
      DATA GP6PB/5*0., 91.,0.,3*1., 92.,4*0.,                           00003490
     X 93.,4*0., 94.,4*0./                                              00003500
      GP6PB(1,2)=GPDAT2(ROW,1)                                          00003510
      GP6PB(1,3)=GPDAT2(ROW,2)                                          00003520
      GP6PB(1,4)=GPDAT2(ROW,3)                                          00003530
      GP6PB(1,5)=GPDAT2(ROW,4)                                          00003540
      LKPROB=GP6PB(NODE1,NODE2)                                         00003550
      IF(LKPROB.EQ.0.)RETURN                                            00003560
612   IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 613                           00003570
      T1=GPDAT2(ROW,5)                                                  00003580
      LKTIME=T1                                                         00003590
      LKFUEL=LKTIME*MFULRT(4)                                           00003600
      RETURN                                                            00003610
613   IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 614                           00003620
      T2=GPDAT2(ROW,6)                                                  00003630
      LKTIME=T2                                                         00003640
      LKFUEL=LKTIME*MFULRT(4)                                           00003650
      RETURN                                                            00003660
614   IF(NODE1.NE.1 .OR. NODE2.NE.4)GO TO 615                           00003670
      T3=GPDAT2(ROW,7)                                                  00003680
      LKTIME=T3                                                         00003690
      LKFUEL=LKTIME*MFULRT(4)                                           00003700
      RETURN                                                            00003710
615   IF(NODE1.NE.1 .OR. NODE2.NE.5)GO TO 699                           00003720
      T4=GPDAT2(ROW,8)                                                  00003730
      LKTIME=T4                                                         00003740
      LKFUEL=LKTIME*MFULRT(4)                                           00003750
      RETURN                                                            00003760
699   RETURN                                                            00003770
C                                                                       00003780
C                                                                       00003790
C                                                                       00003800
C 7. PATROL GROUP                                                       00003810
C                                                                       00003820
7     CONTINUE                                                          00003830
      DATA GP7PB/5*0., 91.,0.,1., 92.,2*0./                             00003840
      GP7PB(1,2)=GPDAT2(ROW,1)                                          00003850
```

```
      LKFUEL=LKTIME*MFULRT(2)                                          00004490
      RETURN                                                           00004500
999   RETURN                                                           00004510
C                                                                      00004520
C                                                                      00004530
C                                                                      00004540
C 10. SAR SEARCH GROUP                                                 00004550
C SUCCESS                                                              00004560
10    CONTINUE                                                         00004570
      DATA GP10PB/4*0., 2*0.,915.,925., 91.,3*0.,                      00004580
     X  92.,3*0./                                                      00004590
      GP10PB(1,3)=GPDAT2(ROW,1)                                        00004600
      GP10PB(1,4)=GPDAT2(ROW,2)                                        00004610
      SW1=GPDAT2(ROW,3)                                                00004620
      A1=GPDAT2(ROW,4)                                                 00004630
      NSEAR1=GPDAT2(ROW,5)                                             00004640
      CF1=GPDAT2(ROW,6)                                                00004650
      TMAX1=GPDAT2(ROW,7)                                              00004660
      SW2=GPDAT2(ROW,8)                                                00004670
      A2=GPDAT2(ROW,9)                                                 00004680
      NSEAR2=GPDAT2(ROW,10)                                            00004690
      CF2=GPDAT2(ROW,11)                                               00004700
      TMAX2=GPDAT2(ROW,12)                                             00004710
      CALL $SPEDU(SPEED,SW1,A1,NSEAR1,CF1,TMAX1,PS1,PF1,TS1,TF1)       00004720
      CALL $SPEDU(SPEED,SW2,A2,NSEAR2,CF2,TMAX2,PS2,PF2,TS2,TF2)       00004730
      GP10PB(3,2)=PS1                                                  00004740
      GP10PB(4,2)=PS2                                                  00004750
      LKPROB=GP10PB(NODE1,NODE2)                                       00004760
      IF(LKPROB.EQ.0.)RETURN                                           00004770
1032  IF(NODE1.NE.3 .OR. NODE2.NE.2)GO TO 1042                         00004780
      LKTIME=TS1                                                       00004790
      LKFUEL=LKTIME*MFULRT(3)                                          00004800
      RETURN                                                           00004810
1042  IF(NODE1.NE.4 .OR. NODE2.NE.2)GO TO 1099                         00004820
      LKTIME=TS2                                                       00004830
      LKFUEL=LKTIME*MFULRT(3)                                          00004840
1099  RETURN                                                           00004850
C                                                                      00004860
C                                                                      00004870
C 11. SEARCH FLEET GROUP                                               00004880
C                                                                      00004890
11    CONTINUE                                                         00004900
      DATA GP11PB/2*0., 1.,0./                                         00004910
      LKPROB=GP11PB(NODE1,NODE2)                                       00004920
      IF(LKPROB.EQ.0.)RETURN                                           00004930
1112  IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 1199                         00004940
      D1=GPDAT2(ROW,1)                                                 00004950
      LKTIME=D1/SPEED(2)                                               00004960
      LKFUEL=LKTIME*MFULRT(2)                                          00004970
      RETURN                                                           00004980
1199  RETURN                                                           00004990
C                                                                      00005000
C                                                                      00005010
C                                                                      00005020
C 12. SEIZE GROUP                                                      00005030
C                                                                      00005040
12    CONTINUE                                                         00005050
      DATA GP12PB/3*0., 2*0.,1., 1.,2*0./                              00005060
      LKPROB=GP12PB(NODE1,NODE2)                                       00005070
      IF(LKPROB.EQ.0.)RETURN                                           00005080
1213  IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 1232                         00005090
      T1=GPDAT2(ROW,1)                                                 00005100
                                                                       00005110
```

```
      LKTIME=D2/SPEED(1)                                                00005750
      LKFUEL=LKTIME*MFULRT(1)                                           00005760
      RETURN                                                            00005770
1514  IF(NODE1.NE.1 .OR. NODE2.NE.4)GO TO 1599                          00005780
      D3=GPDAT2(ROW,6)                                                  00005790
      LKTIME=D3/SPEED(1)                                                00005800
      LKFUEL=LKTIME*MFULRT(1)                                           00005810
      RETURN                                                            00005820
1599  RETURN                                                            00005830
C                                                                       00005840
C                                                                       00005850
C                                                                       00005860
C 16. TRANSFER EQUIPMENT GROUP                                          00005870
C                                                                       00005880
16    CONTINUE                                                          00005890
      DATA GP16PB/3*0.,.91.,0.,1.,.92.,3*0./                            00005900
      GP16PB(1,2)=GPDAT2(ROW,1)                                         00005910
      GP16PB(1,3)=GPDAT2(ROW,2)                                         00005920
      LKPROB=GP16PB(NODE1,NODE2)                                        00005930
      IF(LKPROB.EQ.0.)RETURN                                            00005940
1612  IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 1613                          00005950
      T1=GPDAT2(ROW,3)                                                  00005960
      LKTIME=T1*$SKTIM(7)*$MNTIM(7)                                     00005970
      LKFUEL=LKTIME*MFULRT(4)                                           00005980
      RETURN                                                            00005990
1613  IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 1699                          00006000
      T2=GPDAT2(ROW,4)                                                  00006010
      LKTIME=T2*$SKTIM(7)*$MNTIM(7)                                     00006020
      LKFUEL=LKTIME*MFULRT(4)                                           00006030
      RETURN                                                            00006040
1699  RETURN                                                            00006050
C                                                                       00006060
C                                                                       00006070
C                                                                       00006080
C 17. TRANSPORT EQUIPMENT GROUP                                         00006090
C                                                                       00006100
17    CONTINUE                                                          00006110
      DATA GP17PB/4*0.,.92.,2*0.,1.,.91.,3*0.,.2*0.,1.,0./             00006120
      GP17PB(1,2)=GPDAT2(ROW,2)                                         00006130
      GP17PB(1,3)=GPDAT2(ROW,1)                                         00006140
      LKPROB=GP17PB(NODE1,NODE2)                                        00006150
      IF(LKPROB.EQ.0.)RETURN                                            00006160
1712  IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 1713                          00006170
      D4=GPDAT2(ROW,8)                                                  00006180
      A4=GPDAT2(ROW,9)                                                  00006190
      W4=GPDAT2(ROW,10)                                                 00006200
      LKTIME=D4/SPEED(2)                                                00006210
      LKFUEL=LKTIME*MFULRT(2)                                           00006220
      RETURN                                                            00006230
1713  IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 1734                          00006240
      T1=GPDAT2(ROW,3)                                                  00006250
      LKTIME=T1*$SKTIM(7)*$MNTIM(7)                                     00006260
      LKFUEL=LKTIME*MFULRT(4)                                           00006270
      RETURN                                                            00006280
1734  IF(NODE1.NE.3 .OR. NODE2.NE.4)GO TO 1742                          00006290
      D2=GPDAT2(ROW,4)                                                  00006300
      A2=GPDAT2(ROW,5)                                                  00006310
      W2=GPDAT2(ROW,6)                                                  00006320
      LKTIME=D2/SPEED(2)                                                00006330
      LKFUEL=LKTIME*MFULRT(2)                                           00006340
      RETURN                                                            00006350
1742  IF(NODE1.NE.4 .OR. NODE2.NE.2)GO TO 1799                          00006360
      T3=GPDAT2(ROW,7)                                                  00006370
```

```
      A2=GPDAT2(ROW,9)                                                  00007010
      NSEAR2=GPDAT2(ROW,10)                                             00007020
      CF2=GPDAT2(ROW,11)                                                00007030
      TMAX2=GPDAT2(ROW,12)                                              00007040
      CALL $SPEDU(SPEED,SW1,A1,NSEAR1,CF1,TMAX1,PS1,PF1,TS1,TF1)        00007050
      CALL $SPEDU(SPEED,SW2,A2,NSEAR2,CF2,TMAX2,PS2,PF2,TS2,TF2)        00007060
      GP90PB(3,9)=PF1                                                   00007070
      GP90PB(4,9)=PF2                                                   00007080
      LKPROB=GP90PB(NODE1,NODE2)                                        00007090
      IF(LKPROB.EQ.0.)RETURN                                            00007100
 9039 IF(NODE1.NE.3 .OR. NODE2.NE.9)GO TO 9049                          00007110
      LKTIME=TF1                                                        00007120
      LKFUEL=LKTIME*MFULRT(3)                                           00007130
      RETURN                                                            00007140
 9049 IF(NODE1.NE.4 .OR. NODE2.NE.9)GO TO 9099                          00007150
      LKTIME=TF2                                                        00007160
      LKFUEL=LKTIME*MFULRT(3)                                           00007170
      RETURN                                                            00007180
 9099 RETURN                                                            00007190
C                                                                       00007200
C 93. SENSOR SEARCH GROUP                                               00007210
C THIS GROUP MUST ALWAYS FOLLOW A STEAM GROUP                           00007220
C FAILURE                                                               00007230
 93   CONTINUE                                                          00007240
      DATA GP93PB/72*0., 97.,8*0./                                      00007250
      SW=GPDAT2(ROW,1)                                                  00007260
      E=GPDAT2(ROW,2)                                                   00007270
      VTAR=GPDAT2(ROW,3)                                                00007280
      TMAX=GPDAT2(ROW,4)                                                00007290
      TBEF=$PRVTM(0)                                                    00007300
      CALL $SSHP(SPEED,TBEF,SW,E,VTAR,TMAX,PS,PF,TS,TF)                 00007310
      GP93PB(1,9)=PF                                                    00007320
      LKPROB=GP93PB(NODE1,NODE2)                                        00007330
      IF(LKPROB.EQ.0.)RETURN                                            00007340
      LKTIME=TF                                                         00007350
      LKFUEL=LKTIME*MFULRT(2)                                           00007360
 9399 RETURN                                                            00007370
C                                                                       00007380
C                                                                       00007390
C                                                                       00007400
      END                                                               00007410
C                                                                       00007420
C $TASK                                                                 00007430
C                                                                       00007440
C RETURNS NUMBER OF TASK (&RATE) BETWEEN TWO NODES IN A GROUP           00007450
      SUBROUTINE $TASK(GROUP,NODE1,NODE2,TASKNO,RATE)                   00007460
      IMPLICIT INTEGER(A-Z)                                             00007470
      DIMENSION GP1TK(7,7),GP2TK(3,3),GP3TK(4,4),GP4TK(3,3)             00007480
      DIMENSION GP7TK(3,3),GP8TK(3,3),GP5TK(6,6),GP6TK(5,5)             00007490
      DIMENSION GP11TK(2,2),GP12TK(3,3),GP9TK(4,4),GP10TK(4,4)          00007500
      DIMENSION GP15TK(4,4),GP16TK(3,3),GP13TK(2,2),GP14TK(2,2)         00007510
      DIMENSION GP17TK(4,4),GP18TK(6,6),GP90TK(9,9),GP93TK(9,9)         00007520
C                                                                       00007530
      IF(GROUP.EQ.90)GO TO 90                                           00007540
      IF(GROUP.EQ.93)GO TO 93                                           00007550
      GOTO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18),GROUP         00007560
C                                                                       00007570
C 1. ASSIST GROUP                                                       00007580
C                                                                       00007590
 1    CONTINUE                                                          00007600
      DATA GP1TK/7*0, 404,2*0,413,0,0,416, 401,6*0,                     00007610
     X  2*0,411,4*0, 414,6*0, 408,6*0, 5*0,411,0/                       00007620
      RATE=GP1TK(NODE1,NODE2)/100                                       00007630
```

```
 9      CONTINUE                                                        00008270
        DATA GP9TK/4*0, 305,0,2*0, 302,3*0, 208,3*0/                    00008280
        RATE=GP9TK(NODE1,NODE2)/100                                     00008290
        TASKNO=GP9TK(NODE1,NODE2) - RATE*100                            00008300
        RETURN                                                          00008310
C                                                                       00008320
C 10. SAR SEARCH GROUP                                                  00008330
C SUCCESS                                                               00008340
 10     CONTINUE                                                        00008350
        DATA GP10TK/4*0, 2*0,303,301, 4*0, 4*0/                         00008360
        RATE=GP10TK(NODE1,NODE2)/100                                    00008370
        TASKNO=GP10TK(NODE1,NODE2) - RATE*100                           00008380
        RETURN                                                          00008390
C                                                                       00008400
C 11. SEARCH FLEET GROUP                                                00008410
C                                                                       00008420
 11     CONTINUE                                                        00008430
        DATA GP11TK/2*0, 205,0/                                         00008440
        RATE=GP11TK(NODE1,NODE2)/100                                    00008450
        TASKNO=GP11TK(NODE1,NODE2) - RATE*100                           00008460
        RETURN                                                          00008470
C                                                                       00008480
C 12. SEIZE GROUP                                                       00008490
C                                                                       00008500
 12     CONTINUE                                                        00008510
        DATA GP12TK/3*0, 2*0,201, 418,2*0/                              00008520
        RATE=GP12TK(NODE1,NODE2)/100                                    00008530
        TASKNO=GP12TK(NODE1,NODE2) - RATE*100                           00008540
        RETURN                                                          00008550
C                                                                       00008560
C 13. SENSOR SEARCH GROUP                                               00008570
C SUCCESS                                                               00008580
 13     CONTINUE                                                        00008590
        DATA GP13TK/2*0, 206,0/                                         00008600
        RATE=GP13TK(NODE1,NODE2)/100                                    00008610
        TASKNO=GP13TK(NODE1,NODE2) - RATE*100                           00008620
        RETURN                                                          00008630
C                                                                       00008640
C 14. STANDBY GROUP                                                     00008650
C                                                                       00008660
 14     CONTINUE                                                        00008670
        DATA GP14TK/2*0, 407,0/                                         00008680
        RATE=GP14TK(NODE1,NODE2)/100                                    00008690
        TASKNO=GP14TK(NODE1,NODE2) - RATE*100                           00008700
        RETURN                                                          00008710
C                                                                       00008720
C 15. STEAM GROUP                                                       00008730
C                                                                       00008740
 15     CONTINUE                                                        00008750
        DATA GP15TK/4*0, 209,0,0,0, 101,3*0, 102,3*0/                   00008760
        RATE=GP15TK(NODE1,NODE2)/100                                    00008770
        TASKNO=GP15TK(NODE1,NODE2) - RATE*100                           00008780
        RETURN                                                          00008790
C                                                                       00008800
C 16. TRANSFER EQUIPMENT GROUP                                          00008810
C                                                                       00008820
 16     CONTINUE                                                        00008830
        DATA GP16TK/3*0, 406,0,0, 420,2*0/                              00008840
        RATE=GP16TK(NODE1,NODE2)/100                                    00008850
        TASKNO=GP16TK(NODE1,NODE2) - RATE*100                           00008860
        RETURN                                                          00008870
C                                                                       00008880
C 17. TRANSPORT EQUIPMENT GROUP                                         00008890
```

```
      DO 20 J=1,N                                                       00009530
      K=GPPLMX(I,J)                                                     00009540
      GROUP=K/100                                                       00009550
      INST=K-GROUP*100                                                  00009560
      IF(GROUP.EQ.0)GO TO 20                                            00009570
      CALL $GPMIN(GROUP,INST,T,F)                                       00009580
      GPMNT(I,J)=T                                                      00009590
      GPMNF(I,J)=F                                                      00009600
20    CONTINUE                                                          00009610
C                                                                       00009620
C FINDS THE MINIMUM PATH WITH RESPECT TO TIME                           00009630
C                                                                       00009640
      MINTIM(2)=0.                                                      00009650
      FLAG(2)=1                                                         00009660
30    DO 50 J=1,N                                                       00009670
      IF(FLAG(J).EQ.0) GO TO 50                                         00009680
      DO 40 I=1,N                                                       00009690
      IF(OVCNMX(I,J).EQ.0)GO TO 40                                      00009700
      TPATH=GPMNT(I,J)+MINTIM(J)                                        00009710
      IF(TPATH.GE.MINTIM(I))GO TO 40                                    00009720
      MINTIM(I)=TPATH                                                   00009730
      FLAGNW(I)=1                                                       00009740
      AGAIN=1                                                           00009750
40    CONTINUE                                                          00009760
50    CONTINUE                                                          00009770
      DO 60 I=1,N                                                       00009780
      FLAG(I)=FLAGNW(I)                                                 00009790
      FLAGNW(I)=0                                                       00009800
60    CONTINUE                                                          00009810
      IF(AGAIN.EQ.0)GO TO 100                                           00009820
      AGAIN=0                                                           00009830
      GO TO 30                                                          00009840
C                                                                       00009850
100   CONTINUE                                                          00009860
C FINDS THE MINIMUM PATH WITH RESPECT TO FUEL                           00009870
C                                                                       00009880
      MINFUE(2)=0.                                                      00009890
      FLAG(2)=1                                                         00009900
110   DO 130 J=1,N                                                      00009910
      IF(FLAG(J).EQ.0) GO TO 130                                        00009920
      DO 120 I=1,N                                                      00009930
      IF(OVCNMX(I,J).EQ.0)GO TO 120                                     00009940
      FPATH=GPMNF(I,J)+MINFUE(J)                                        00009950
      IF(FPATH.GE.MINFUE(I))GO TO 120                                   00009960
      MINFUE(I)=FPATH                                                   00009970
      FLAGNW(I)=1                                                       00009980
      AGAIN=1                                                           00009990
120   CONTINUE                                                          00010000
130   CONTINUE                                                          00010010
      IF(AGAIN.EQ.0)GO TO 200                                           00010020
      AGAIN=0                                                           00010030
      DO 140 I=1,N                                                      00010040
      FLAG(I)=FLAGNW(I)                                                 00010050
      FLAGNW(I)=0                                                       00010060
140   CONTINUE                                                          00010070
      GO TO 110                                                         00010080
200   RETURN                                                            00010090
      END                                                               00010100
C                                                                       00010110
C $GPMIN                                                                00010120
C                                                                       00010130
C COMPUTES THE SHORTEST PATH THROUGH A GROUP                            00010140
C FOR TIME AND FUEL CONSUMPTION                                         00010150
```

```
      RETURN                                                            00010790
C                                                                       00010800
C                                                                       00010810
C 4.  IDENTIFY GROUP                                                    00010820
C                                                                       00010830
4     CONTINUE                                                          00010840
      CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                           00010850
      CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                           00010860
      IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                         00010870
      IF(P13.GT.0..AND.T13.LT.MINTIM)MINTIM=T13                         00010880
      IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                         00010890
      IF(P13.GT.0..AND.F13.LT.MINFUE)MINFUE=F13                         00010900
      RETURN                                                            00010910
C                                                                       00010920
C                                                                       00010930
C 5.  INSPECT GROUP                                                     00010940
C                                                                       00010950
5     CONTINUE                                                          00010960
      CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                           00010970
      CALL $LKDAT(GROUP,INST,1,5,P15,T15,F15)                           00010980
      CALL $LKDAT(GROUP,INST,3,4,P34,T34,F34)                           00010990
      CALL $LKDAT(GROUP,INST,4,2,P42,T42,F42)                           00011000
      CALL $LKDAT(GROUP,INST,5,6,P56,T56,F56)                           00011010
      CALL $LKDAT(GROUP,INST,6,2,P62,T62,F62)                           00011020
      T1342=T13+T34+T42                                                 00011030
      T1562=T15+T56+T62                                                 00011040
      IF(P13.GT.0..AND.T1342.LT.MINTIM)MINTIM=T1342                     00011050
      IF(P15.GT.0..AND.T1562.LT.MINTIM)MINTIM=T1562                     00011060
      F1342=F13+F34+F42                                                 00011070
      F1562=F15+F56+F62                                                 00011080
      IF(P13.GT.0..AND.F1342.LT.MINFUE)MINFUE=F1342                     00011090
      IF(P15.GT.0..AND.F1562.LT.MINFUE)MINFUE=F1562                     00011100
      RETURN                                                            00011110
C                                                                       00011120
C                                                                       00011130
C 6.  MONITOR GROUP                                                     00011140
C                                                                       00011150
6     CONTINUE                                                          00011160
      CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                           00011170
      CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                           00011180
      CALL $LKDAT(GROUP,INST,1,4,P14,T14,F14)                           00011190
      CALL $LKDAT(GROUP,INST,1,5,P15,T15,F15)                           00011200
      IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                         00011210
      IF(P13.GT.0..AND.T13.LT.MINTIM)MINTIM=T13                         00011220
      IF(P14.GT.0..AND.T14.LT.MINTIM)MINTIM=T14                         00011230
      IF(P15.GT.0..AND.T15.LT.MINTIM)MINTIM=T15                         00011240
      IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                         00011250
      IF(P13.GT.0..AND.F13.LT.MINFUE)MINFUE=F13                         00011260
      IF(P14.GT.0..AND.F14.LT.MINFUE)MINFUE=F14                         00011270
      IF(P15.GT.0..AND.F15.LT.MINFUE)MINFUE=F15                         00011280
      RETURN                                                            00011290
C                                                                       00011300
C                                                                       00011310
C 7.  PATROL GROUP                                                      00011320
C                                                                       00011330
7     CONTINUE                                                          00011340
      CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                           00011350
      CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                           00011360
      IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                         00011370
      IF(P13.GT.0..AND.T13.LT.MINTIM)MINTIM=T13                         00011380
      IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                         00011390
      IF(P13.GT.0..AND.F13.LT.MINFUE)MINFUE=F13                         00011400
      RETURN                                                            00011410
```

```
C                                                                       00012050
C 13. SENSOR SEARCH GROUP                                               00012060
C SUCCESS                                                               00012070
13      CONTINUE                                                        00012080
        CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                         00012090
        IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                       00012100
        IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                       00012110
        RETURN                                                          00012120
C                                                                       00012130
C                                                                       00012140
C 14. STANDBY GROUP                                                     00012150
C                                                                       00012160
14      CONTINUE                                                        00012170
        CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                         00012180
        IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                       00012190
        IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                       00012200
        RETURN                                                          00012210
C                                                                       00012220
C                                                                       00012230
C 15. STEAM GROUP                                                       00012240
C                                                                       00012250
15      CONTINUE                                                        00012260
        CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                         00012270
        CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                         00012280
        CALL $LKDAT(GROUP,INST,1,4,P14,T14,F14)                         00012290
        IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                       00012300
        IF(P13.GT.0..AND.T13.LT.MINTIM)MINTIM=T13                       00012310
        IF(P14.GT.0..AND.T14.LT.MINTIM)MINTIM=T14                       00012320
        IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                       00012330
        IF(P13.GT.0..AND.F13.LT.MINFUE)MINFUE=F13                       00012340
        IF(P14.GT.0..AND.F14.LT.MINFUE)MINFUE=F14                       00012350
        RETURN                                                          00012360
C                                                                       00012370
C                                                                       00012380
C 16. TRANSFER EQUIPMENT GROUP                                          00012390
C                                                                       00012400
16      CONTINUE                                                        00012410
        CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                         00012420
        CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                         00012430
        IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                       00012440
        IF(P13.GT.0..AND.T13.LT.MINTIM)MINTIM=T13                       00012450
        IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                       00012460
        IF(P13.GT.0..AND.F13.LT.MINFUE)MINFUE=F13                       00012470
        RETURN                                                          00012480
C                                                                       00012490
C                                                                       00012500
C 17. TRANSPORT EQUIPMENT GROUP                                         00012510
C                                                                       00012520
17      CONTINUE                                                        00012530
        CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                         00012540
        CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                         00012550
        CALL $LKDAT(GROUP,INST,3,4,P34,T34,F34)                         00012560
        CALL $LKDAT(GROUP,INST,4,2,P42,T42,F42)                         00012570
        T1342=T13+T34+T42                                               00012580
        IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                       00012590
        IF(P13.GT.0..AND.T1342.LT.MINTIM)MINTIM=T1342                   00012600
        F1342=F13+F34+F42                                               00012610
        IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                       00012620
        IF(P13.GT.0..AND.F1342.LT.MINFUE)MINFUE=F1342                   00012630
        RETURN                                                          00012640
C                                                                       00012650
C                                                                       00012660
C 18. WORK EQUIPMENT GROUP                                              00012670
```

```
C
C FINDS Y VALUE ON BROKEN LINE OF 3 POINTS,                          00013310
C GIVEN X VALUE AND THE 3 POINTS (ASSUMING ENDS OF                   00013320
C LINE EXTEND INFINETLY)                                             00013330
C                                                                    00013340
      FUNCTION ZZ3(X,X1,Y1,X2,Y2,X3,Y3)                              00013350
C                                                                    00013360
      IF(X.LE.X2)ZZ3=ZZ(X,X1,Y1,X2,Y2)                               00013370
      IF(X.GT.X2)ZZ3=ZZ(X,X2,Y2,X3,Y3)                               00013380
      RETURN                                                         00013390
      END                                                            00013400
C                                                                    00013410
C                                                                    00013420
C ZZ4                                                                00013430
C                                                                    00013440
C FINDS Y VALUE ON BROKEN LINE OF 4 POINTS, GIVEN X VALUE            00013450
C    AND THE 4 POINTS                                                00013460
C (ASSUMING ENDS OF LINE EXTEND INFINITELY)                          00013470
C                                                                    00013480
      FUNCTION ZZ4(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4)                        00013490
C                                                                    00013500
      IF(X.LE.X2)ZZ4=ZZ(X,X1,Y1,X2,Y2)                               00013510
      IF(X.GT.X2 .AND. X.LE.X3)ZZ4=ZZ(X,X2,Y2,X3,Y3)                 00013520
      IF(X.GT.X3)ZZ4=ZZ(X,X3,Y3,X4,Y4)                               00013530
      RETURN                                                         00013540
      END                                                            00013550
C                                                                    00013560
C                                                                    00013570
C ZZ5                                                                00013580
C FINDS Y VALUE ON BROKEN LINE OF 5 POINTS, GIVEN X                  00013590
C VALUE AND THE 5 POINTS                                             00013600
C (ASSUMING ENDS OF LINE EXTEND INFINITELY)                          00013610
C                                                                    00013620
      FUNCTION ZZ5(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5)                  00013630
C                                                                    00013640
C                                                                    00013650
      IF(X.LE.X2)ZZ5=ZZ(X,X1,Y1,X2,Y2)                               00013660
      IF(X.GT.X2.AND.X.LE.X3)ZZ5=ZZ(X,X2,Y2,X3,Y3)                   00013670
      IF(X.GT.X3.AND.X.LE.X4)ZZ5=ZZ(X,X3,Y3,X4,Y4)                   00013680
      IF(X.GT.X4)ZZ5=ZZ(X,X4,Y4,X5,Y5)                               00013690
      RETURN                                                         00013700
      END                                                            00013710
C                                                                    00013720
C $SKTIM                                                             00013730
C                                                                    00013740
      FUNCTION $SKTIM(MASTSK)                                        00013750
      IMPLICIT REAL(A-Z)                                             00013760
      INTEGER MASTSK,ENG,SSPDTB                                      00013770
      DIMENSION SSPRBD(8),CWSPD(4),SFCENG(4),SFCCF(4),TOTSFC(4)      00013780
      DIMENSION SFCGAL(4),HPUTIL(4),FUELRT(4),ENDUR(4),RANGE(4)      00013781
      DIMENSION MOTION(4),TNRAD(4),FUELR2(4),ENG(4)                  00013782
      COMMON/CHAR/LTOB,BEAM,DTOL,DRAF,SSPRBD,DECK,USECD,             00013783
     1 FUELCP,CARGCP,TOWDSP,SURVIV,HPINST,HPPTON,HPTNKT,             00013784
     2 CWSPD,ENG,SFCENG,SFCCF,TOTSFC,SFCGAL,HPUTIL,                  00013785
     3 FUELRT,FUELR2,ENDUR,RANGE,MOTION,TNRAD,SSPDTB                 00013786
C     ASST,BORD,RTRV,WEGD,WEGP                                       00013787
C                                                                    00013800
      IF(MASTSK.EQ.1 .OR. MASTSK.EQ.2 .OR. MASTSK.EQ.4               00013810
     X  .OR. MASTSK.EQ.6 .OR. MASTSK.EQ.7)GO TO 1                    00013820
      $SKTIM=1.                                                      00013830
      RETURN                                                         00013840
1     $SKTIM=$$3(MOTION(4),0.,1., .5,1., 1.,2.)                      00013850
      RETURN                                                         00013860
                                                                     00013870
```

```
      TFULLS=TF-TNEXT                                                   00014500
      TLAST=TMAX-TFULLS                                                 00014510
      SFRAC=TLAST/TNEXT                                                 00014520
      TF=TMAX                                                           00014530
C     FINDS PS AND PF                                                   00014540
33    PODN=$POD(CF,NFULLS)                                              00014550
      PODN1=$POD(CF,NFULLS+1)                                           00014560
      PODL=SFRAC*(PODN1-PODN)                                           00014570
      PS=PODN+PODL                                                      00014580
      PF=1.0-PS                                                         00014590
C     CALCULATE TIME OF SUCCESSFUL SEARCH                               00014600
      SUM=0.                                                            00014610
      TMBEFJ=0.                                                         00014620
      DO 30  J=1,NFULLS                                                 00014630
      PODJ=$POD(CF,J)-$POD(CF,J-1)                                      00014640
      IF(J.LE.1)GO TO 92                                                00014650
      TMBEFJ=TMBEFJ+TIMJ                                                00014660
92    TIMJ=(CF*A*ALPHA(J))/(SPEED(3)*SW)                                00014670
      AVTMJ=TIMJ/2.0                                                    00014680
      TOTTMJ=AVTMJ+TMBEFJ                                               00014690
      SUM=SUM+TOTTMJ*PODJ                                               00014700
30    CONTINUE                                                          00014710
      IF(SFRAC.EQ.0.)GO TO 95                                           00014720
      TOTTML=TFULLS+TLAST/2.                                            00014730
      SUM=SUM+TOTTML*PODL                                               00014740
95    TS=SUM/PS                                                         00014750
      RETURN                                                            00014760
      END                                                               00014770
C                                                                       00014780
C $POD                                                                  00014790
C                                                                       00014800
C CALCULATES PROBABILITY OF DETECTION USING CURVES FROM                 00014810
C SAR MANUAL, FIGURE 8-65.                                              00014820
      FUNCTION $POD(CF,NRSRCH)                                          00014830
      IMPLICIT REAL(A-Z)                                                00014840
      INTEGER NRSRCH                                                    00014850
      IF(NRSRCH.EQ.0)$POD=0.                                            00014860
      IF(NRSRCH.EQ.1)$POD=$$5(CF,0.,0.,.6,.55,1.1,.83,1.5,.95,1.8,.97)  00014870
      IF(NRSRCH.EQ.2)$POD=$$5(CF,0.,0.,.39,.6,.7,.85,1.,.96,1.3,.99)    00014880
      IF(NRSRCH.EQ.3)$POD=$$5(CF,0.,0.,.21,.5,.4,.76,.6,.9,.99,.99)     00014890
      IF(NRSRCH.EQ.4)$POD=$$5(CF,0.,0.,.21,.6,.4,.85,.6,.95,.8,1.)      00014900
      IF(NRSRCH.EQ.5)$POD=$$5(CF,0.,0.,.22,.7,.4,.9,.5,.96,.75,1.)      00014910
      IF($POD.GT.1.0)$POD=1.0                                           00014920
      RETURN                                                            00014930
      END                                                               00014940
C                                                                       00014950
C $SSHP                                                                 00014960
C                                                                       00014970
C CALCULATES PROBABILITY AND TIME FOR SEARCH                            00014971
C  FOR SHIP TASK                                                        00014972
      SUBROUTINE $SSHP(SPEED,TBEF,SW,E,VTAR,TMAX,PS,PF,TS,TF)           00014980
      IMPLICIT REAL(A-Z)                                                00014990
      INTEGER I,J                                                       00015000
      DIMENSION SPEED(4)                                                00015010
      IF(SW.LE.0. .OR. TMAX.LE.0.)GO TO 99                              00015020
      PI=3.14159                                                        00015030
      DELTAT=0.1                                                        00015040
      DELTH=DELTAT/2.0                                                  00015050
      GO TO 90                                                          00015060
99    PS=0.0                                                            00015070
      PF=1.0                                                            00015080
      TS=0.                                                             00015090
      TF=0.                                                             00015100
```