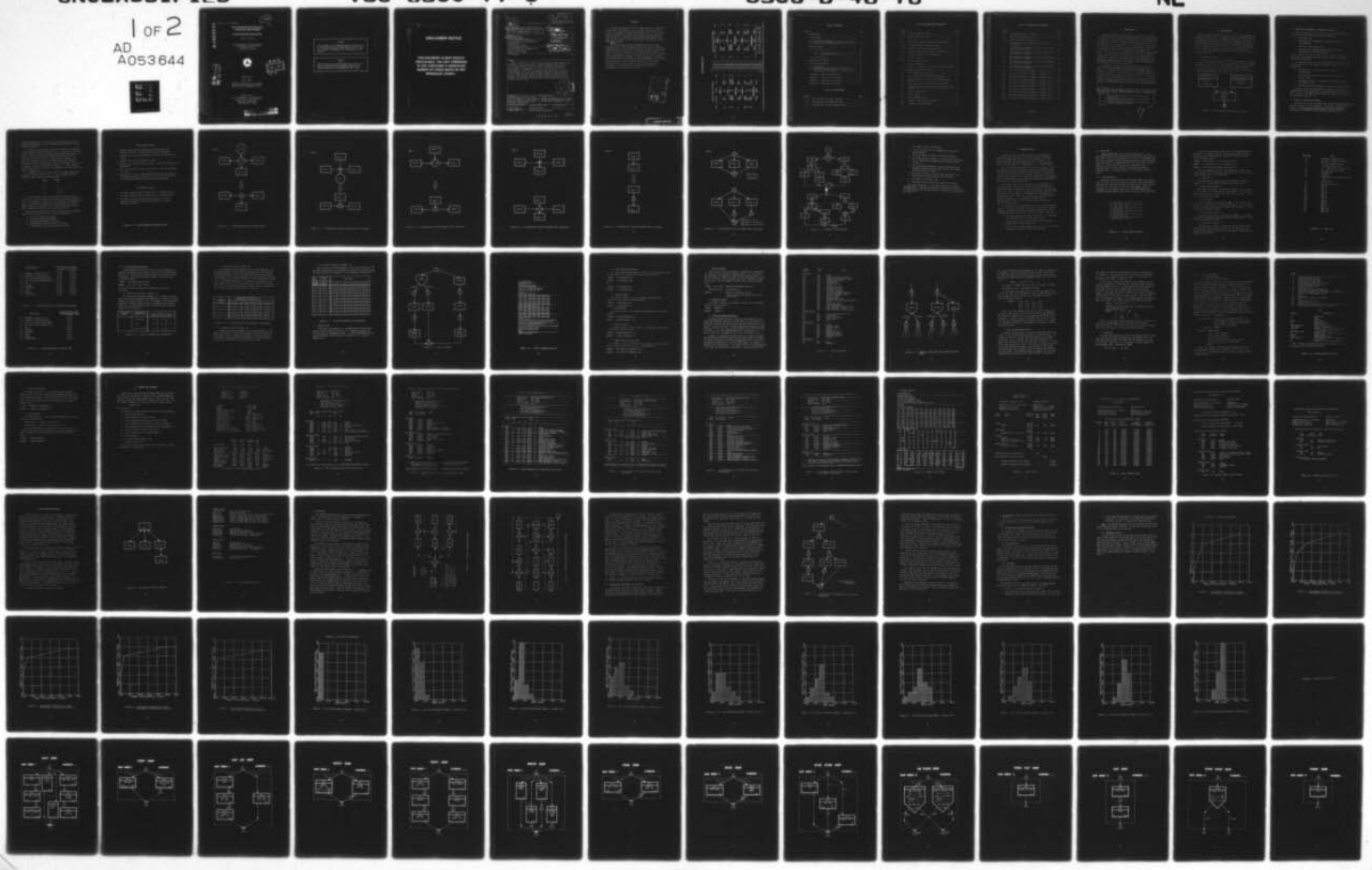
AD-A053 644 TRANSPORTATION SYSTEMS CENTER CAMBRIDGE MASS F/G 13/10
THE CUTTER RESOURCE EFFECTIVENESS EVALUATION (CREE) PROGRAM -- --ETC(U)
MAR 78 D S PRERAU

UNCLASSIFIED TSC-USCG-77-3 USCG-D-48-78 NL

1 OF 2
AD
A053644



AD A053644

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

AD No. DDC FILE COPY

DDC RECEIVED MAY 8 1978 F

MARCH 1978

FINAL REPORT

Document is available to the U.S. public through the Defense Documentation Center, Cameron Station, Alexandria VA 22314

Prepared for

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

COPY AVAILABLE TO DDC DOES NOT PERMIT FULLY LEGIBLE PRODUCTION

NOTICE

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

NOTICE

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

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DDC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

(18) USCG

(12) 174p.

Technical Report Documentation Page

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

(6) 4. Title and Subtitle
THE CUTTER RESOURCE EFFECTIVENESS EVALUATION (CREE) PROGRAM -- A GUIDE FOR USERS AND ANALYSTS.

(11) 5. Report Date
Mar[redacted]78

6. Performing Organization Code

(14) 8. Performing Organization Report No.
DOT-TSC-USCG-77-3

(10) 7. Author(s)
David S. Prerau

9. Performing Organization Name and Address
U.S. Department of Transportation
Transportation Systems Center
Kendall Square
Cambridge MA 02142

10. Work Unit No. (TRAIS)
CG811/R8010

11. Contract or Grant No.

(9) 13. Type of Report and Period Covered
Final Report.
Jun[redacted]75-Jun[redacted]77,

12. Sponsoring Agency Name and Address
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.

DDC
RECEIVED
MAY 8 1978
F

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

407 082 Gul

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.



PRECEDING PAGE BLANK

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
	LENGTH			
in	inches	2.5	centimeters	cm
ft	feet	30	centimeters	cm
yd	yards	0.9	meters	m
mi	miles	1.6	kilometers	km
	AREA			
in^2	square inches	6.5	square centimeters	cm^2
ft^2	square feet	0.09	square meters	m^2
yd^2	square yards	0.8	square meters	m^2
mi^2	square miles	2.6	square kilometers	km^2
	acres	0.4	hectares	ha
	MASS (weight)			
oz	ounces	28	grams	g
lb	pounds	0.45	kilograms	kg
	short tons (2000 lb)	0.9	tonnes	t
	VOLUME			
tsp	teaspoons	5	milliliters	ml
Tbsp	tablespoons	15	milliliters	ml
fl oz	fluid ounces	30	milliliters	ml
c	cups	0.24	liters	l
pt	pints	0.47	liters	l
qt	quarts	0.95	liters	l
gal	gallons	3.8	liters	l
ft^3	cubic feet	0.03	cubic meters	m^3
yd^3	cubic yards	0.76	cubic meters	m^3
	TEMPERATURE (exact)			
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C



Approximate Conversions from Metric Measures

Symbol	When You Know	Multiply by	To Find	Symbol
	LENGTH			
mm	millimeters	0.04	inches	in
cm	centimeters	0.4	inches	in
m	meters	3.3	feet	ft
m	meters	1.1	yards	yd
km	kilometers	0.6	miles	mi
	AREA			
cm^2	square centimeters	0.16	square inches	in^2
m^2	square meters	1.2	square yards	yd^2
km^2	square kilometers	0.4	square miles	mi^2
ha	hectares (10,000 m^2)	2.5	acres	
	MASS (weight)			
g	grams	0.035	ounces	oz
kg	kilograms	2.2	pounds	lb
t	tonnes (1000 kg)	1.1	short tons	
	VOLUME			
ml	milliliters	0.03	fluid ounces	fl oz
l	liters	2.1	pints	pt
l	liters	1.06	quarts	qt
l	liters	0.26	gallons	gal
m^3	cubic meters	35	cubic feet	ft^3
m^3	cubic meters	1.3	cubic yards	yd^3
	TEMPERATURE (exact)			
°C	Celsius temperature	9/5 (then add 32)	Fahrenheit temperature	°F

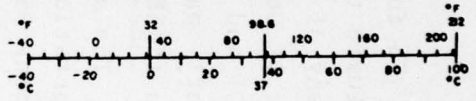


TABLE OF CONTENTS

Section		Page
1.	INTRODUCTION	1
2.	CREE PROGRAM	2
	2.1 CREE Program Overall	2
	2.2 Craft and Environmental Information Required	3
	2.3 Scenario Information Required	3
3.	PROGRAM INPUTS	14
	3.1 Craft Data	15
	3.1.1 Craft Selection	15
	3.1.2 Environment Specification	19
	3.2 Scenario Data	21
4.	RUNNING CREE PROGRAM	33
5.	CREE PROGRAM STRUCTURE	46
	5.1 Methodology	49
	5.2 Methodology Enhancement for Efficiency	52
	5.3 Algorithm for Minimum Fuel Use from Any Node to the End of the Sortie	56
	APPENDIX A - TOWING DISTRIBUTIONS	58
	APPENDIX B - SEA-STATE DISTRIBUTIONS	63
	APPENDIX C - FUNCTIONAL TASK GROUPS	73
	APPENDIX D - CREE PROGRAM LISTING	92

LIST OF ILLUSTRATIONS

Figure		Page
2.1-1.	CREE PROGRAM FROM USER'S VIEWPOINT	2
2.3-1.	NODE-PLACEMENT AND NUMBERING RULES	5
2.3-2.	ILLUSTRATIONS OF NODE-PLACEMENT RULES	6

LIST OF ILLUSTRATIONS (CONTINUED)

Figure		Page
2.3-3.	EXAMPLE OF NODE PLACEMENT	12
3.1-1.	TYPICAL CRAFT.DATA FILE	15
3.1-2.	CRAFT TYPE	17
3.1-3.	ACCEPTABLE RANGES FOR LENGTH AND DISPLACEMENT	18
3.1-4.	ACCEPTABLE RANGES FOR DESIGN SPEED	18
3.1-5.	VISIBILITY PROBABILITY DISTRIBUTIONS	19
3.1-6.	DISTRIBUTIONS FOR DISPLACEMENT OF TOWED CRAFT	20
3.1-7.	SEA-STATE PROBABILITY DISTRIBUTIONS	21
3.2-1.	TYPICAL SCENARIO	22
3.2-2.	TYPICAL SCENARIO.DATA FILE	23
3.2-3.	TASK CODE NUMBERS	26
3.2-4.	EXAMPLE OF THREE-PORT (ONE INPUT/TWO OUTPUT GROUPS)	27
3.2-5.	REQUIRED GROUP-DATA INPUTS	31
4-1.	CRAFT CHARACTERISTICS OUTPUT	34
4-2.	CRAFT PARAMETERS FOR MASTER TASKS OUTPUT	35
4-3.	TASK PROBABILITIES OF SUCCESS FOR MASTER TASKS OUTPUT	36
4-4.	CRAFT PARAMETERS FOR INDIVIDUAL TASKS OUTPUT	37
4-5.	TASK PROBABILITIES OF SUCCESS FOR INDIVIDUAL TASKS OUTPUT	39
4-6.	SCENARIO DATA OUTPUT	41
4-7.	SORTIE OUTPUT	42
4-8.	SORTIE SUMMARY OUTPUT	43
4-9.	SCENARIO OVERALL RESULTS OUTPUT	44
4-10.	SCENARIO EVALUATION OUTPUT	45

LIST OF ILLUSTRATIONS (CONTINUED)

Figure		Page
5-1.	CREE PROGRAM STORAGE STRUCTURE	47
5-2.	CREE PROGRAM COMPUTER FILES	48
5.1-1.	PROPOS ALGORITHM FLOW CHART	50
5.2-1.	ILLUSTRATION FOR METHODOLOGY FOR EFFICIENCY ENHANCEMENT	54
A-1.	TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT -- TOW DISTRIBUTION NUMBER 1	58
A-2.	TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT -- TOW DISTRIBUTION NUMBER 2	59
A-3.	TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT -- TOW DISTRIBUTION NUMBER 3	60
A-4.	TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT -- TOW DISTRIBUTION NUMBER 4	61
A-5.	TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT -- TOW DISTRIBUTION NUMBER 5	62
B-1.	SEA STATE DISTRIBUTION NUMBER 1--AVERAGE SS=0.5	63
B-2.	SEA-STATE DISTRIBUTION NUMBER 2--AVERAGE SS=1.0	64
B-3.	SEA-STATE DISTRIBUTION NUMBER 3--AVERAGE SS=1.5	65
B-4.	SEA-STATE DISTRIBUTION NUMBER 4--AVERAGE SS=2.0	66
B-5.	SEA-STATE DISTRIBUTION NUMBER 5--AVERAGE SS=2.5	67
B-6.	SEA-STATE DISTRIBUTION NUMBER 6--AVERAGE SS=3.0	68
B-7.	SEA-STATE DISTRIBUTION NUMBER 7--AVERAGE SS=3.5	69
B-8.	SEA-STATE DISTRIBUTION NUMBER 8--AVERAGE SS=4.0	70
B-9.	SEA-STATE DISTRIBUTION NUMBER 9--AVERAGE SS=4.5	71
B-10.	SEA-STATE DISTRIBUTION NUMBER 10--AVERAGE SS=5.0	72

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. CREE PROGRAM

The CREE model computer program is written in the FORTRAN IV computer language, and is presently resident in the IBM Model 370/158 computer at the Mitre Corporation, Bedford MA. The user of the CREE program need not know the FORTRAN IV language, but he must know the Time Sharing Option (TSO) of the IBM Model 370 operating system. Specifically, he must know how to create and edit data files on TSO from a computer terminal, since craft and Coast Guard program data must be put in TSO data files before the CREE program can be run.

2.1 CREE PROGRAM OVERALL

From the user's point of view, the CREE program may be thought of as one large program with two sets of inputs. One set of inputs describes the craft and the environmental conditions under which it is to be evaluated. The second describes the Coast Guard program scenario in which the craft will be tested. This structure is shown diagrammatically in Figure 2.1-1.

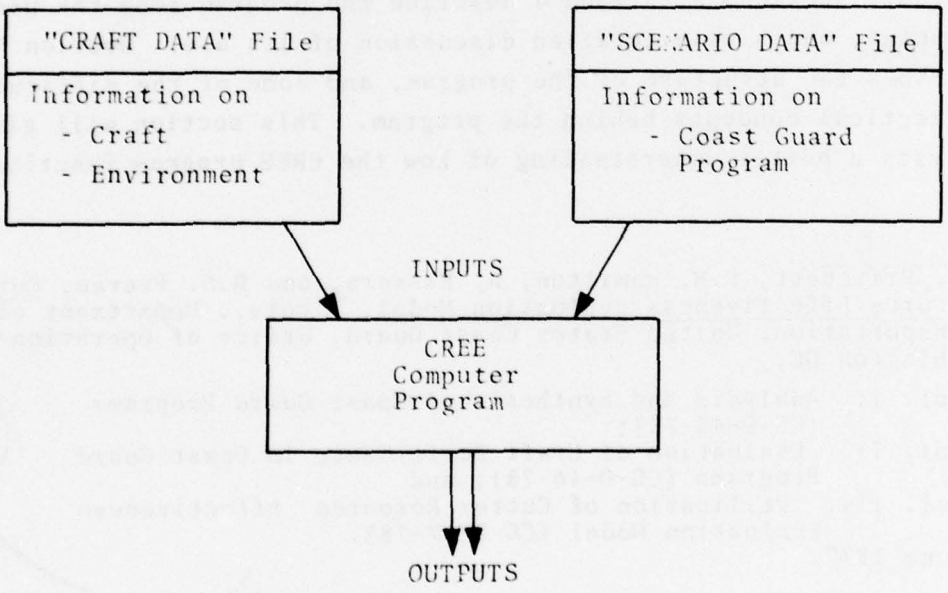


FIGURE 2.1-1. CREE PROGRAM FROM USER'S VIEWPOINT

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

required information for each Group of the flowchart scenario on the Group Data Sheets. (A set of blank Group Data Sheets can be found in Appendix C.)

To prepare the flow-chart scenario for input to the computer, a noded flowchart must be prepared. This is a version of the flow-chart scenario in which only Groups and Probabilities appear (i.e., "Start", "Stop", and decision diamonds do not appear) and in which each junction or "node" is identified by a number. The method for constructing the noded flowchart from the original flow-chart scenario is described by the node-placement rules listed in Figure 2.3-1 and illustrated by Figure 2.3-2. These should be studied carefully. The nodes are numbered by the node-numbering rules in Figure 2.3-1.

An example of the use of these rules on a sample flow-chart scenario is shown in Figure 2.3-3. In this example, the node-placement rules produce the following nodes:

RULE	NODES
1	1
2	2
3	8
4	7, 10
5	3, 4, 5, 6
6	9.

It is efficient to draw the original flow-chart scenario so that there are no unnecessary junction points leading to unnecessary nodes. For example, in Figure 2.3-3 the junction corresponding to Node 7 could have been eliminated by drawing the line from the ESCORT directly to the STOP. This would eliminate the need for Node 7, making the CREE program run slightly faster.

When the noded flowchart has been constructed, the user must supply the CREE program with the following information:

a. The Coast Guard Program Name,
b. A Scenario-identifying Number,
c. The Maximum Allowable Sortie Time,
d. The Range Fraction (the fraction of total fuel capacity that can be used on a sortie),

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 1

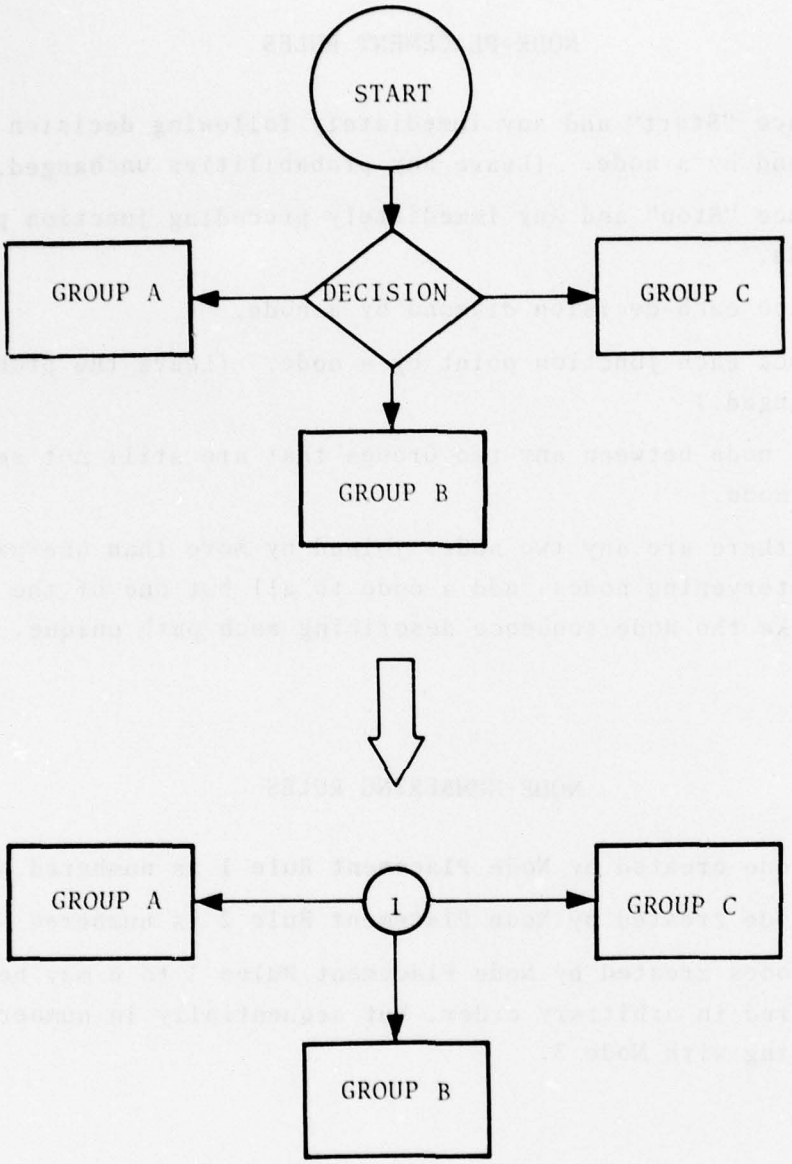


FIGURE 2.3-2. ILLUSTRATIONS OF NODE-PLACEMENT RULES

RULE 2

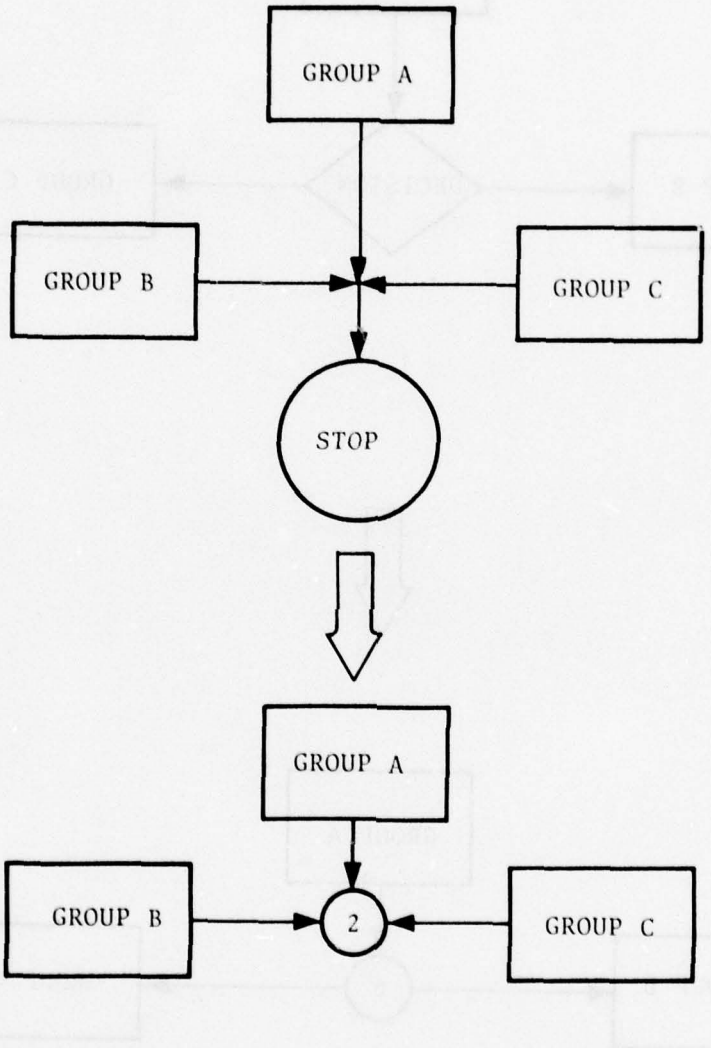


FIGURE 2.3-2. ILLUSTRATIONS OF NODE-PLACEMENT RULES (CONTINUED)

RULE 3

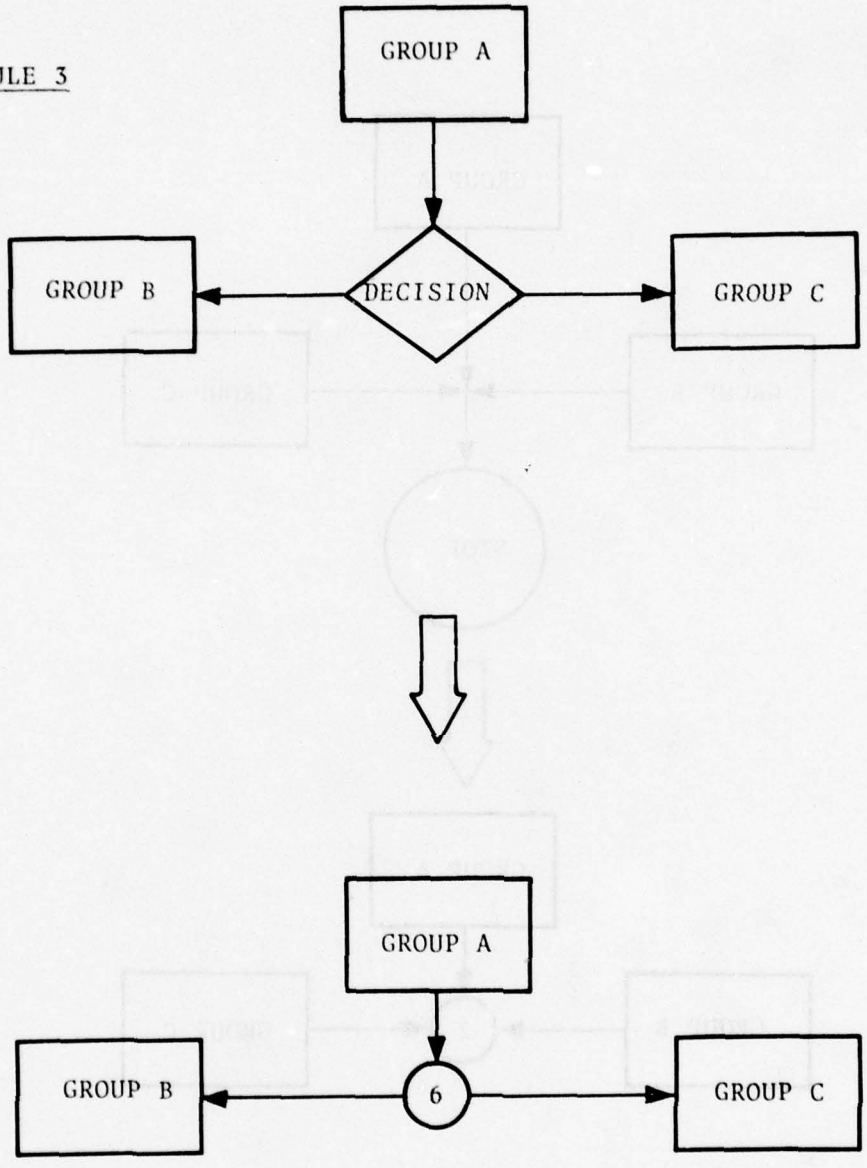
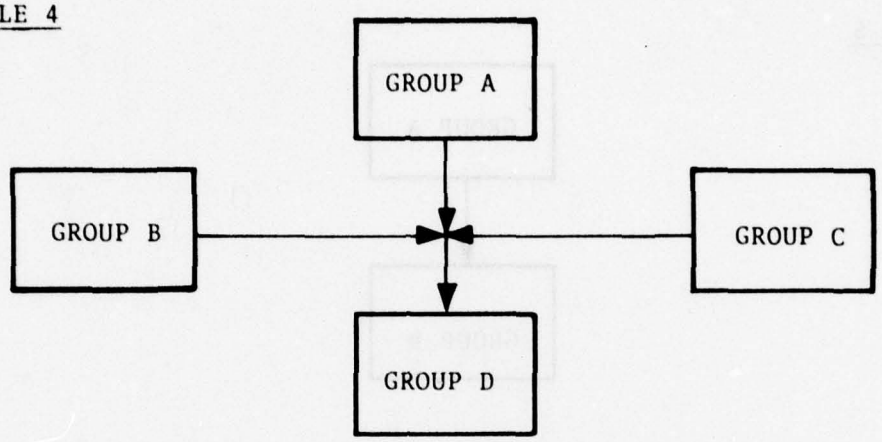


FIGURE 2.3-2. ILLUSTRATIONS OF NODE-PLACEMENT RULES (CONTINUED)

RULE 4



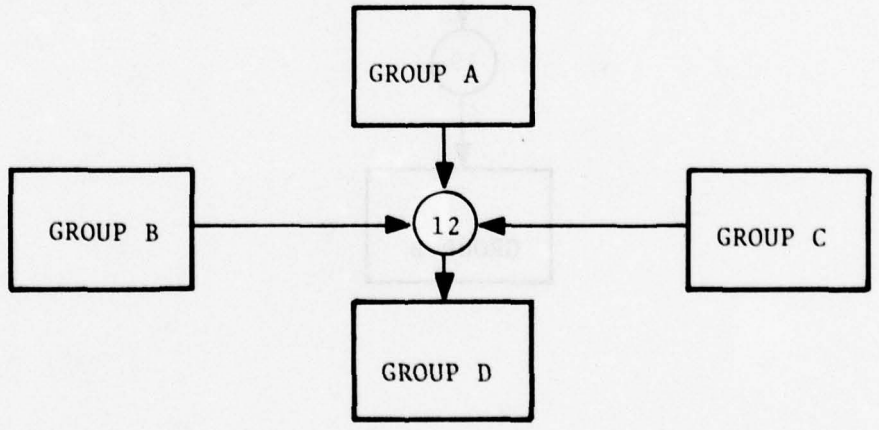


FIGURE 2.3-2. ILLUSTRATIONS OF NODE-PLACEMENT RULES (CONTINUED)

RULE 5

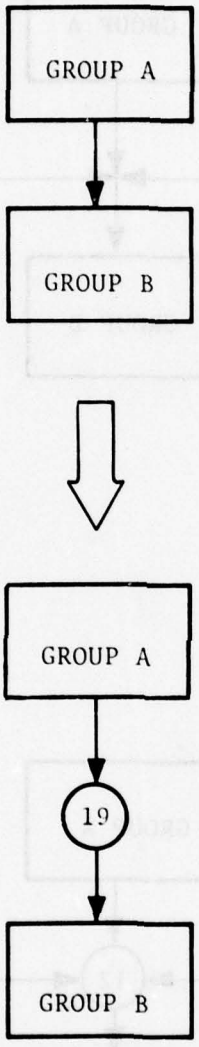
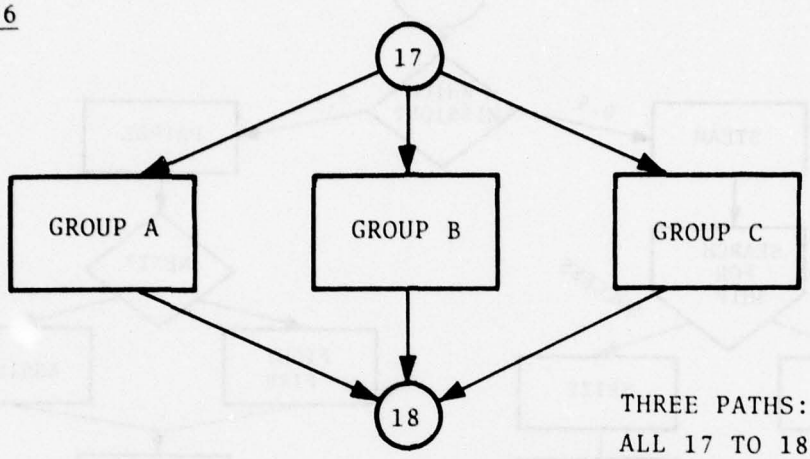
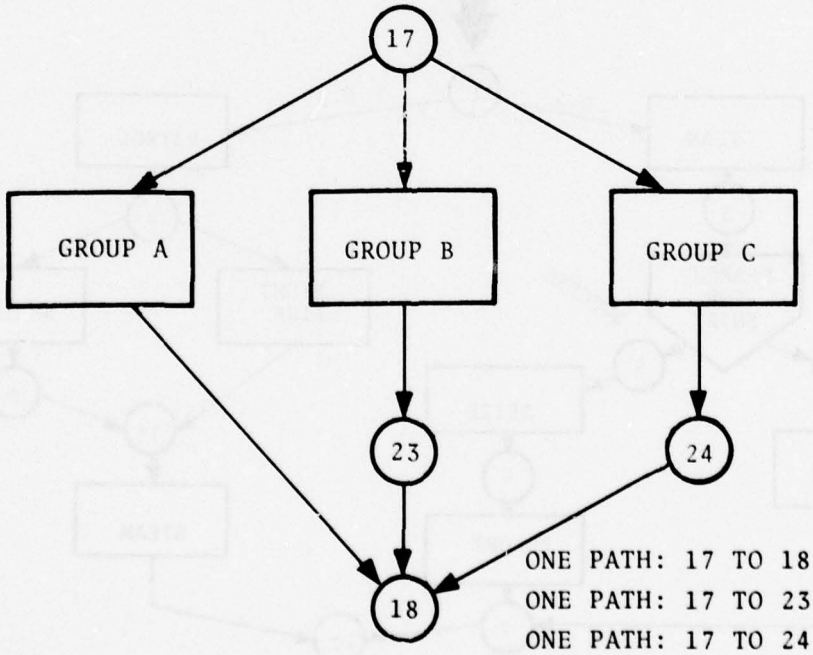


FIGURE 2.3-2. ILLUSTRATIONS OF NODE-PLACEMENT RULES (CONTINUED)

RULE 6



THREE PATHS:
ALL 17 TO 18
(AMBIGUOUS)



ONE PATH: 17 TO 18
ONE PATH: 17 TO 23 TO 18
ONE PATH: 17 TO 24 TO 18

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

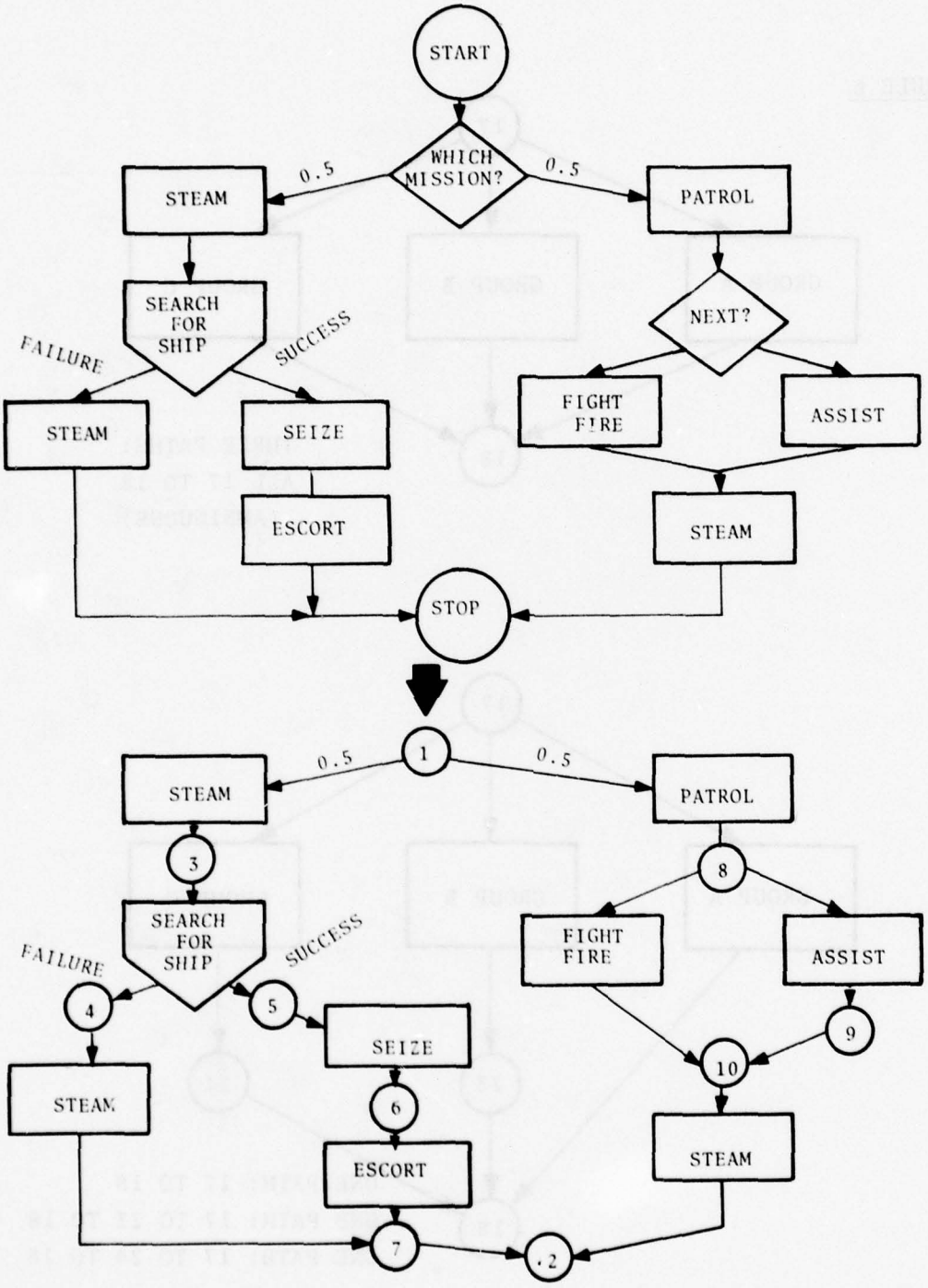


FIGURE 2.3-3. EXAMPLE OF NODE PLACEMENT

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

l. 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. PROGRAM INPUTS

The CREE program must have available to it descriptive information about the chosen Craft and Scenario. Therefore, prior to running CREE, two files must be established: the CRAFT.DATA file, and the SCENARIO.DATA file. These two files must be created using TSO's NONUM mode (i.e., no line numbers).

The files may be established in either of two ways. First, the appropriate craft and scenario data may be input into these files by the user each time he runs the program. Alternatively, the user may establish in advance several Craft data files and several Scenario data files. Using TSO's RENAME or COPY features, he could then name the appropriate files CRAFT.DATA and SCENARIO.DATA just for a single program run.

For example, the user may choose the first approach for craft data and directly input the data to the CRAFT.DATA file before each run of the program, since the data for one craft take up only two lines. However, he may create several scenario files in advance, calling them, say, SCENSAR1.DATA, SCENSAR2.DATA, SCENELT1.DATA, etc. Then, if he wants to run the CREE program with SAR Scenario No. 2, he RENAME's SCENSAR2.DATA to SCENARIO.DATA. After the run, he must rename SCENARIO.DATA back to SCENSAR2.DATA.

The CRAFT.DATA and SCENARIO.DATA files must follow required formats. These are discussed in the next two sections. The user should be cautioned that data not properly allocated into the allowable positions will produce incorrect results. Note that:

a. Most data numbers are "right-justified"; i.e., they fill the right-most part of the allocated space,

b. Decimal points in data numbers must coincide with the decimal point in the format, and

c. In both cases, either blanks or zeroes may be used to fill blank spaces for proper positioning of data.

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,IH= 1,SS= 6
T= 30,D= 200.0,L=0000.0,S=60.0,F=0.50
US= 2,TW= 1,IH= 1,SS= 6
T= 40,D=0000.0,L= 100.0,S=40.0,F=0.50
US= 2,TW= 1,IH= 1,SS= 6
T= 50,D=0000.0,L= 250.0,S=20.0,F=0.50
US= 2,TW= 1,IH= 1,SS= 6
T=110,D=0000.0,L=0000.0,S=00.0,F=0.00
US= 2,TW= 1,IH= 1,SS= 6
T=111,D=0000.0,L=0000.0,S=00.0,F=0.00
US= 3,TW= 1,IH= 1,SS= 6
```

FIGURE 3.1-1. TYPICAL CRAFT.DATA FILE

For existing Coast Guard craft, the craft type alone must be specified. The other values for existing Coast Guard craft will be provided by the CREE program, and the user must only input zeroes for these values.

FORMAT: "T=###,D=####.#,L=####.#,S=##.#,F=#.##"

EXAMPLE: "T= 40,D=0000.0,L= 70.0,S=50.0,F=0.75"

EXAMPLE: "T=108,D=0000.0,L=0000.0,S=00.0,F=0.00"

The first example shows the specification of an HPWC. The second example shows an existing Coast Guard craft. The five data values required will now be discussed:

a. Craft Type, T

The craft type is specified by the TYPE CODE, as shown in Figure 3.1-2. The TYPE CODE is entered right-justified.

b. Craft Displacement, D

The craft displacement is specified in tons. It is input as zero if craft size is indicated by craft length, L; it is input as zero for existing Coast Guard craft (Types 101 and above). Figure 3.1-3 shows acceptable ranges.

c. Craft Length, L

The craft length is specified in feet. It is input as zero if craft size is indicated by craft displacement, D; it is input as zero for existing Coast Guard craft (Types 101 and above). Figure 3.1-3 shows acceptable ranges.

d. Craft Design Speed, S

The craft design speed is specified in knots. It is input as zero for existing Coast Guard craft (Types 101 and above). Figure 3.1-4 shows acceptable ranges.

e. Craft Fuel Fraction, F

The craft fuel fraction is defined as the fraction of total useful payload to be carried as fuel. It is input as zero for existing Coast Guard craft (Types 101 and above). The acceptable range is 0.20 to 0.80.

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

	TYPE AND CRAFT	ACCEPTABLE INPUT RANGE LENGTH OR DISPLACEMENT (feet)	(tons)
10	HYDROFOIL, Submerged Foil	75-150	62-250
11	HYDROFOIL, Surface Piercing	70-150	25-200
20	ACV, Low Pressure/Length Ratio	65-135	15-200
21	ACV, High Pressure/Length Ratio	50-100	15-150
30	SES	100-150	90-250
40	PLANING	85-150	40-275
50	CATAMARAN	40-135	10-140
60	SWATH	100-300	500-3500
70	HYBRID	40-135	10-140
80	CONVENTIONAL	50-400	30-3500

FIGURE 3.1-3. ACCEPTABLE RANGES FOR LENGTH AND DISPLACEMENT

	CRAFT TYPE	MAXIMUM DESIGN SPEED ACCEPTABLE INPUT RANGE (knots)
10	HYDROFOIL, Submerged Foil	40-50
11	HYDROFOIL, Surface Piercing	30-40
20	ACV, Low Pressure/Length Ratio	50-70
21	ACV, High Pressure/Length Ratio	40-60
30	SES	30-50
40	PLANING	35-45
50	CATAMARAN	30-40
60	SWATH	15-25
70	HYBRID	30-40
80	CONVENTIONAL	15-30

FIGURE 3.1-4. ACCEPTABLE RANGES FOR DESIGN SPEED

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

b. Towing-Distribution Number, TW

The Towing-Distribution Number, TW, is right-justified. The towing distribution is the cumulative probability distribution of the displacements of craft in the operating region that may have to be towed. Five towing distributions are presently available in the CREE program, as shown in Appendix A and as summarized in Figure 3.1-6. Towing Distribution No. 1 indicates an operating region where none of the craft to be towed are less than 0.5 ton, 20 percent of the craft to be towed are less than 1.0 ton, 40 percent of the craft to be towed are less than 2.5 tons, etc.

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

FIGURE 3.1-6. DISTRIBUTIONS FOR DISPLACEMENT OF TOWED CRAFT

c. Depth-Distribution Number, DH

The Depth-Distribution Number, DH, is right-justified. The depth distribution is the cumulative probability distribution of water depth. Only one depth distribution is presently available in the CREE program, Depth Distribution No. 1, which corresponds to deep water throughout the entire operating region.

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.

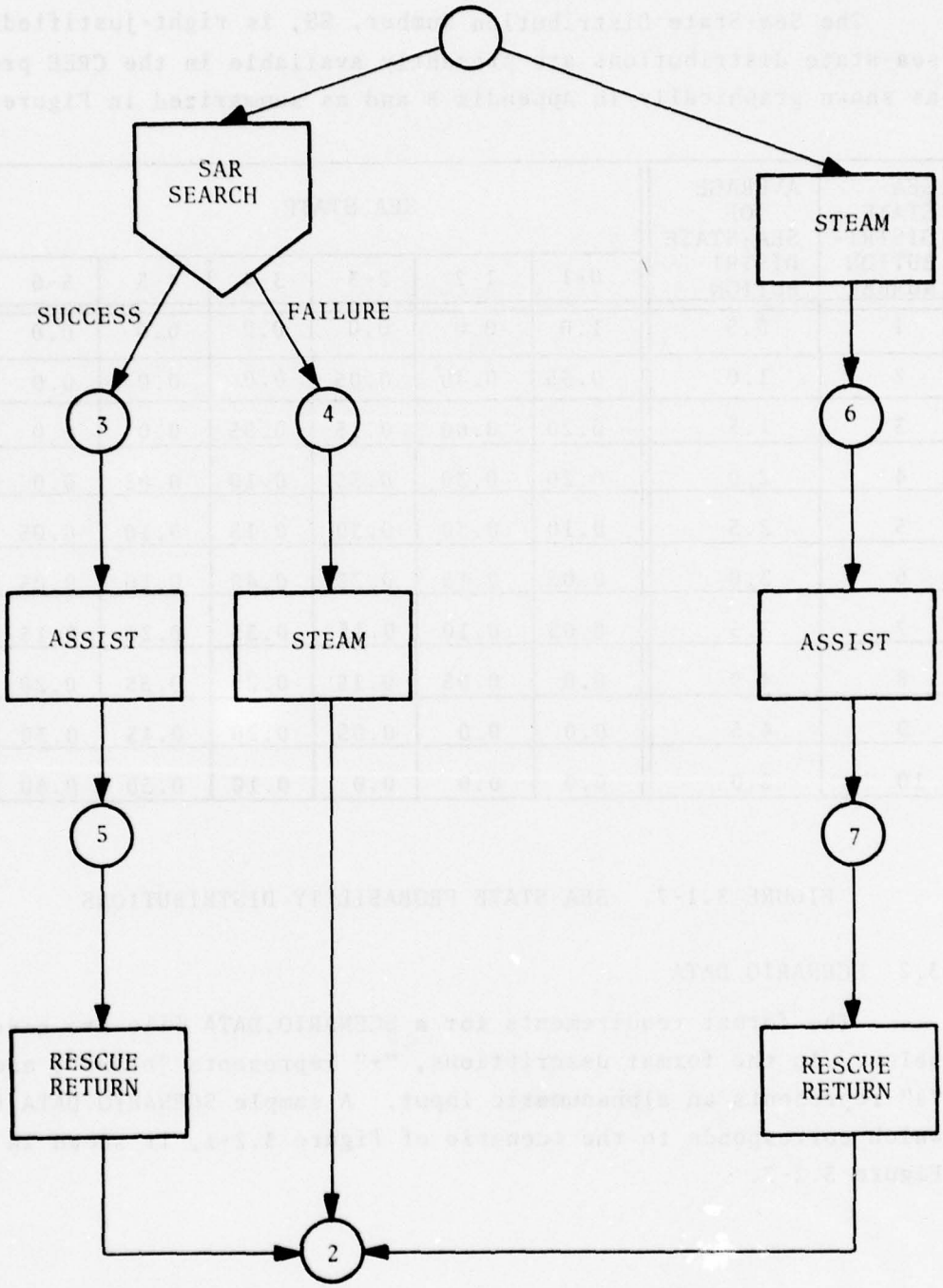


FIGURE 3.2-1. TYPICAL SCENARIO

```
LG 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 0.00 1.00 0.60 0.60
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 0.60
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 0000 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,,6,,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

a. Coast Guard Program Name

The Coast Guard Program Name is a three- or four-character code for the name of the Coast Guard program, right-justified.

FORMAT: "+CG+PROGRAM=+@@@"

EXAMPLE: " CG+PROGRAM= SAR"

or

FORMAT: "+CG+PROGRAM=@@@@"

EXAMPLE: " CG PROGRAM=TEST"

b. Scenario Number

The Scenario Number is a user-assigned two-digit number identifying the scenario, right-justified.

FORMAT: "+SCENARIO+NO.=@@"

EXAMPLE: " SCENARIO NO.= 7"

c. Maximum Time

The Maximum Time is the maximum time allowable for any sortie, in hours.

FORMAT: "+MAXIMUM+TIME=@@@@.@"

EXAMPLE: " MAXIMUM TIME= 18.0"

d. Range Fraction

The Range Fraction is the fraction of total fuel capacity that can be used in any one sortie.

FORMAT: "+RANGE+FRACTION=@.@@"

EXAMPLE: " RANGE FRACTION=0.90"

e. Number of Days of Operation

The Number of Days of Operation is the number of days the craft will be operated under the scenario, right-justified.

FORMAT: "+NO.+DAYS+OF+OPERATION=@@@@"

EXAMPLE: " NO. DAYS OF OPERATION= 180"

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) =

Code Number	Task Code	Task
ON SCENE:		
401	BRD	BOARD
402	FFF	FIGHT FIRE FROM CG VESSEL
403	FFO	FIGHT FIRE ON ANOTHER VESSEL
404	GAS	GENERAL ASSISTANCE
405	INS	INSPECTION
406	LEQ	LOAD EQUIPMENT
407	LOI	LOITER
408	LSB	LAUNCH SMALL BOAT
409	MAC	MONITOR ACTIVITIES
410	MOS	MONITOR OIL SPILL
411	OBA	ON-BOARD ASSISTANCE
412	OSC	ON-SCENE COMMANDER (GENERAL)
413	RBP	RETRIEVE BOARDING PARTY
414	ROB	RETRIEVE OBJECTS
415	RPE	RESCUE PEOPLE
416	RSB	RETRIEVE SMALL BOAT
417	SSI	STAKEOUT SPECIAL INTEREST VESSEL
418	SZE	SEIZE
419	TWS	TAKE WATER SAMPLE
420	ULQ	UNLOAD EQUIPMENT
421	WQB	WORK EQUIPMENT FROM SMALL BOAT
422	WQD	WORK EQUIPMENT @ DRIFT
423	WQF	WORK EQUIPMENT @ FIXED POSITION
REDUCED SPEED:		
301	SDU	SEARCH FOR DISTRESSED UNIT
302	SES	SLOW ESCORT
303	SPE	SEARCH FOR PEOPLE
304	SPT	SLOW PATROL
305	TOW	TOW
CRUISE SPEED:		
201	ESC	ESCORT
202	IDC	IDENTIFY CRAFT
203	IDF	IDENTIFY FLEET
204	PAT	PATROL
205	SFL	SEARCH FOR FLEET
206	SSH	SEARCH FOR SHIP
207	TEQ	TRANSPORT EQUIPMENT
208	TPE	TRANSPORT PEOPLE
209	TRA	TRANSIT
FLANK SPEED:		
101	DSH	DASH
102	INT	INTERDICT

FIGURE 3.2-3. TASK CODE NUMBERS

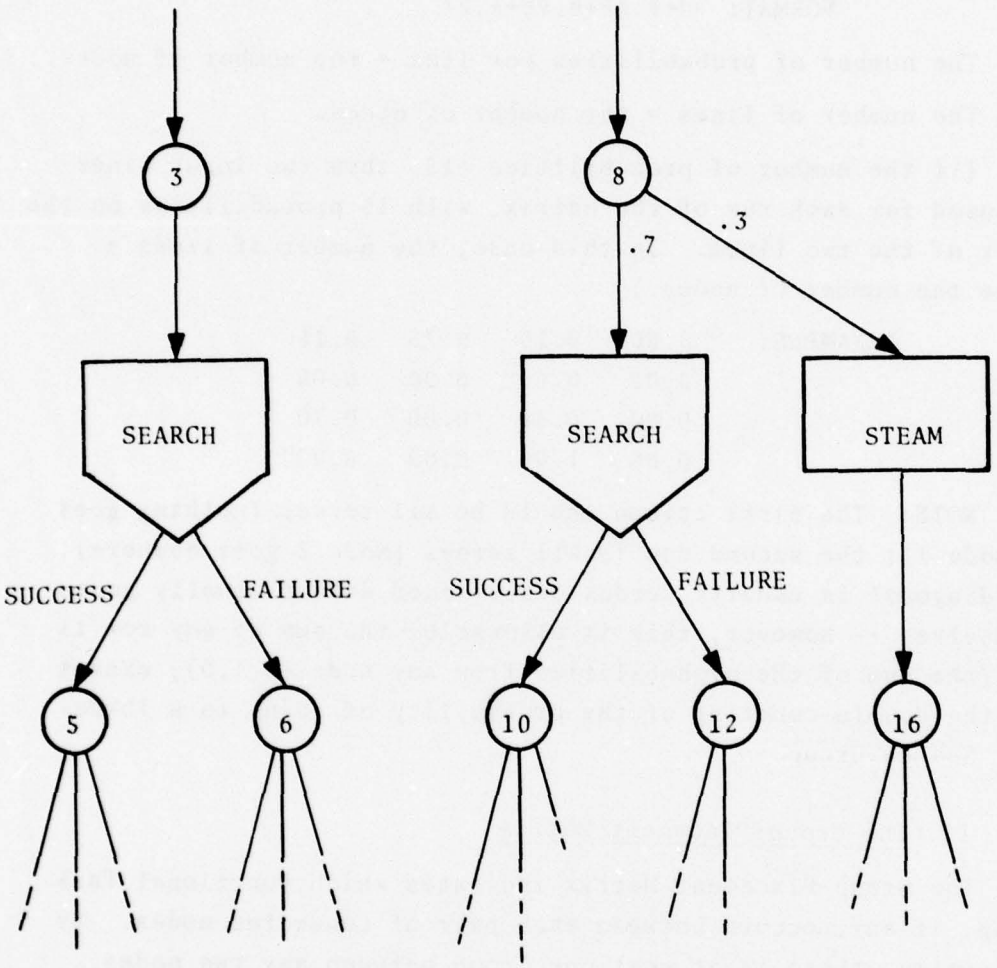


FIGURE 3.2-4. EXAMPLE OF THREE-PORT (ONE INPUT/TWO OUTPUT GROUPS)

0.7, since from Node 8 the probability is 0.7 that the Search will be performed. (Search success and search failure probabilities are found internally by the CREE program, and are used to modify the original Connection Matrix probability inputs for C(i,j) and C(i.k).)

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

The number of probabilities per line = the number of nodes.

The number of lines = the number of nodes.

(If the number of probabilities >15, then two input lines are used for each row of the matrix, with 15 probabilities on the first of the two lines. In this case, the number of lines = twice the number of nodes.)

EXAMPLE: " 0.00 0.10 0.75 0.15
0.00 0.00 0.00 0.00
0.00 0.30 0.00 0.70
0.00 1.00 0.00 0.00"

NOTE: The first column should be all zeroes (nothing goes to Node 1); the second row is all zeroes (Node 2 goes nowhere); the diagonal is usually zeroes since nodes do not usually go to themselves -- however, this is allowable; the sum of any row is 1.0 (the sum of the probabilities from any node is 1.0), except for the double-counting of the probability of going to a Three-Port Search Group.

i. The Group-Placement Matrix

The Group-Placement Matrix indicates which Functional Task Group, if any, occurs between each pair of connected nodes. By convention, there is at most one Group between any two nodes. Each entry, G(i,j), indicates the four-digit identification number for the Group in the link from Node i to Node j. If there is no Group, the entry is zero. The first two digits of the four-digit code indicate the number of the Functional Task Group. The second two digits indicate a user-assigned instance number for this Group. (The instance number is needed since the same Functional Task Group may occur more than once in a scenario,

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,

where: g = Group Number,
i = Instance Number,
e_j = The jth Group-Data Entry.

Each line of Group Data Entries (the e's) must be in the required order for its corresponding group, as shown in Figure 3.2-5. On each line there must be 20 numbers. If the g,i, and e's do not consist of 20 numbers, zeroes must be added to complete the line. A shorthand notation for k zeroes is k*0 (e.g., 4*0 = 0,0, 0,0). For each line, k=20-n-2, where n is the number of required Group-Data Entries. Each line of numbers must end with a comma except the last line of numbers. The entire set of numbers must be preceded by a line that has " &GROUP DATA="on it and followed by a line that has " &END". The order the groups appear in Group Data is arbitrary, but it is convenient to order them numerically. (There should not be any Group-Data entries for artificial Groups 90 and 93, as the data needed will be already included under the entries for the corresponding Groups 10 and 13.)

FORMAT: First Line: "&GROUP+DATA="

Following lines: One line for each Group instance, in free format, 20 numbers per line, ending with a comma. (No comma at the end of the last of these lines).

Last line: "+&END"

EXAMPLE: " &GROUP DATA=
7,1,.5,.5,8,5,4,13*0,
7,2,.6,.4 20,4,10,13*0,
15,1,.1,.9,0,3,3,0,12*0,
17,1,.8,.2,.4,3,60,1.4,.6,3,60,1.4,8*0,
18,1,.5,.2,.1,.2,.1,4,.1,2,1,2,8*0
&END "

NOTE: For example, the third line of numbers gives the following data for Group 15, Instance 1: P1 = 0.1, P2 = 0.9, P3 = 0.0, D1 = 3 nautical miles, D2 = 3 nautical miles, and D3 = 0 nautical miles.

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

k. Number of Printouts

The Number of Printouts is a two-digit right-justified number, greater than zero, indicating the number of times each sortie should be printed. Each sortie will be printed this many times before the following sortie is printed. If it is desired that no sorties be printed out, set this value to 1, and use Output Format = 2.

FORMAT: "+NUMBER+OF+PRINTOUTS=@@"

EXAMPLE: " NUMBER OF PRINTOUTS= 2"

l. Output Format

The Output Format is a one-digit number indicating if sorties should be printed or not:

Output Format = 1 means "print sorties."

Output Format = 2 means "do not print sorties."

Under Output Format 2, all pages up to and including the "Scenario Data" page will be printed as well as the "Sortie Summary", "Scenario Overall Results", and "Scenario Evaluation" pages.

FORMAT: "+OUTPUT+FORMAT=@"

EXAMPLE: " OUTPUT FORMAT=2"

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.

CRAFT CHARACTERISTICS

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

LENGTH	95.0	FEET
BEAM	38.0	FEET
DRAFT	4.8	FEET
LENGTH/BEAM RATIO	2.50	
DRAFT/LENGTH RATIO	0.05	
DISPLACEMENT	93.7	TONS
SURVIVABILITY	5	SEA STATE
TOWS VESSELS UP TO	917.	TONS
USEABLE DECK AREA	1444.	SQUARE FEET
CARGO CAPACITY	14.6	TONS
FUEL CAPACITY	14.6	TONS
USEFUL PAYLOAD	29.3	TONS
INSTALLED POWER	9467.	HORSEPOWER
POWER TO WEIGHT	101.0	HP/TON
TRANSPORT EFFICIENCY	2.53	HP/TON-KNOT
RANGE AT CRUISE SPEED	373.	NAUTICAL MILES
ENDURANCE AT CRUISE SPEED	10.7	HOURS

ENGINE TYPE	FLANK SPEED (DE)	CRUISE SPEED (DE)	REDUCED SPEED (DE)	ON SCENE (DE)	
CALM WATER SPEED	40.0	35.0	12.0	5.0	KNOTS
SFC (WEIGHT)	0.35	0.35	0.35	0.35	LBS/HP-HR
SFC (VOLUME)	0.05	0.05	0.05	0.05	GAL/HP-HR
HP UTILIZED	9467.1	8790.8	2624.6	1065.0	HP
FUEL CONSUMPTION	495.5	460.1	137.4	55.7	GAL/HR
FUEL CONSUMPTION	12.4	13.1	11.4	11.1	GAL/NAUT MI
ENDURANCE (FUEL)	9.9	10.7	35.7	88.0	HOURS
RANGE	396.0	373.2	428.5	440.0	NAUTICAL MI
TURNING RADIUS	430.1	376.3	129.0	53.8	YARDS
CRAFT MOTION	0.1	0.1	0.1	0.1	G
AVG FUEL RATE	463.7	431.8	137.4	55.7	GAL/HR
AVG SPEED	33.0	28.7	12.0	5.0	KNOTS
TOW SPEED	-	-	9.8	-	KNOTS

FIGURE 4-1. CRAFT CHARACTERISTICS OUTPUT

```
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  CARGO DRAFT MANEUV  SEA   TOW
  CODE  CPCTY               STATE

        CC     DF     MN     SS     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

```
T A S K   P R O B A B I L I T I E S   O F   S U C C E S S

     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    TASK PROB.      TASK
     CODE    OF SUCCESS

ON SCENE:
     ASST       0.940       ASSIST
     BORD       0.940       BOARD
     MNAC       0.940       MONITOR ACTIVITIES
     RTRV       0.940       RETRIEVE
     WAIT       1.000       WAIT
     WEQD       1.000       WORK EQUIPMENT @ DRIFT
     WEQP       0.940       WORK EQUIPMENT @ POSITION

REDUCED SPEED:
     SDIU       1.000*      SEARCH FOR DISTRESSED UNIT
     SESC       1.000       SLOW ESCORT
     SPAT       1.000       SLOW PATROL
     SPEO       1.000*      SEARCH FOR PEOPLE
     TOWS       0.962       TOWS

CRUISE SPEED:
     ESCT       1.000       ESCORT
     IDNT       1.000       IDENTIFY
     PATL       1.000       PATROL
     STGT       1.000*      SEARCH FOR TARGET
     TRPT       *****       TRANSPORT
     TRST       1.000       TRANSIT

FLANK SPEED:
     RSPD       0.999       RESPOND

  *   THIS IS THE P.O.S. OF THE ASSET TO SEARCH. CRAFT'S SUCCESS
      IN FINDING THE OBJECT OF THE SEARCH IS DEPENDENT UPON
      SCENARIO (E.G., SEARCH AREA)

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

FIGURE 4-3. TASK PROBABILITIES OF SUCCESS 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	0.68	--	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

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 CC	DF	MANEUV MN	SEA STATE LS	TOW TW	
REDUCED SPEED:							
SDU	--	1.00	--	0.95	--		SEARCH FOR DISTRESSED UNIT
SLS	--	--	--	0.95	--		SLOW ESCORT
SPE	--	1.00	--	0.95	--		SEARCH FOR PEOPLE
SPT	--	1.00	--	0.95	--		SLOW PATROL
TOW	--	--	1.00	0.95	1.00		TOW
CRUISE SPEED:							
ESC	--	--	--	0.95	--		ESCORT
IDC	--	--	1.00	0.95	--		IDENTIFY CRAFT
IDF	--	--	1.00	0.95	--		IDENTIFY
PAT	--	--	--	0.95	--		PATROL
SFL	--	--	--	0.95	--		SEARCH FOR FLEET
SSH	--	1.00	--	0.95	--		SEARCH FOR SHIP
TEQ	****	--	--	0.95	--		TRANSPORT
TPE	--	--	--	0.95	--		TRANSPORT PEOPLE
TRA	--	--	--	0.95	--		TRANSIT
FLANK SPEED:							
DSH	--	--	--	0.95	--		DASH
INT	--	--	--	0.95	--		INTERDICT

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

FIGURE 4-4. CRAFT PARAMETERS FOR INDIVIDUAL TASKS OUTPUT (CONTINUED)

```
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    TASK PROB.     TASK
     CODE    OF SUCCESS

ON SCENE:
     BRD       0.864       BOARD
     FFF       0.824       FIGHT FIRE FROM CG VESSEL
     FFO       0.950       FIGHT FIRE ON ANOTHER VESSEL
     GAS       0.867       GENERAL ASSISTANCE
     INS       0.950       INSPECTION
     LEQ       0.824       LOAD EQUIPMENT
     LOI       0.950       LOITER
     LSB       0.824       LAUNCH SMALL BOAT
     MAC       0.867       MONITOR ACTIVITIES
     MOS       0.867       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       0.867       STAKEOUT SPECIAL INTEREST VESSEL
     SZE       0.950       SEIZE
     TWS       0.824       TAKE WATER SAMPLE
     ULQ       0.824       UNLOAD EQUIPMENT
     WOB       0.950       WORK EQUIPMENT FROM SMALL BOAT
     WOD       0.863       WORK EQUIPMENT @ DRIFT
     WOF       0.824       WORK EQUIPMENT @ FIXED POSITION
```

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

```
                    T A S K   P R O B A B I L I T I E S   O F   S U C C E S S

          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    TASK PROB.      TASK
        CODE    OF SUCCESS

    REDUCED SPEED:
        SDU       0.950*      SEARCH FOR DISTRESSED UNIT
        SLS       0.950       SLOW ESCORT
        SPL       0.950*      SEARCH FOR PEOPLE
        SPT       0.950       SLOW PATROL
        TOW       0.950       TOW

    CRUISE SPEED:
        ESC       0.950       ESCORT
        IDC       0.950       IDENTIFY CRAFT
        IDF       0.950       IDENTIFY FLEET
        PAT       0.950       PATROL
        SFL       0.950       SEARCH FOR FLEET
        SSH       0.950*      SEARCH FOR SHIP
        TEQ       *****       TRANSPORT EQUIPMENT
        TPE       0.950       TRANSPORT PEOPLE
        TRA       0.950       TRANSIT

    FLANK SPEED:
        DSH       0.950       DASH
        INT       0.950       INTERDICT

      * THIS IS THE P.O.S. OF THE ABILITY TO SEARCH. CRAFT'S SUCCES
        IN FINDING THE OBJECT OF THE SEARCH IS DEPENDENT UPON
        SCENARIO (E.G. SEARCH AREA)

***** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AND WEIGHT OF CARG
```

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

```
** 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
202 203 408 405 416 401 415 101 102 209
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                                                                      10 * 0.0
NUMBER OF PRINTOUTS= 1
OUTPUT FORMAT=1
```

FIGURE 4-6. SCENARIO DATA OUTPUT

SAR SCENARIO 1
SORTIE NUMBER 7

OPERATIONAL REQUIREMENTS:

MAXIMUM DURATION 12.0 HOURS
RANGE FRACTION 0.90
VISIBILITY VERY GOOD
AVERAGE SEA STATE 2.0

SELECTED CRAFT:

CATAMARAN
DISPLACEMENT 94 TONS
DESIGN SPEED 40 KNOTS
FUEL FRACTION 0.50

GROUP NAME	TASK NAME	LOCATION CODE	TASK TIME (HRS)	TASK FUEL (GALS)	TASK POS
		1			
STEAM		150101			
	*DASH	150103	0.9	421	1.00
		150102			
		3			
SAR SEARCH		100101			
		100104			
	*SEARCH DSTR UNIT: FOUND	100102	2.3	317	1.00
		4			
ASSIST		10101			
	*BOARD	10103	0.1	5	0.94
	*ON BOARD ASSISTANCE	10104	0.5	27	1.00
	*RETRIEVE BOARDING PARTY	10102	0.1	5	0.94
		5			
RESCUE RETURN		90101			
	*TRANSPORT PEOPLE	90104	1.0	451	1.00
		90102			
		2			

TIME TO COMPLETE SORTIE (HRS)	5.0		
FUEL CONSUMED IN SORTIE (GALS)		1229	
SORTIE PROBABILITY OF SUCCESS			0.9393
SORTIE FREQUENCY OF OCCURRENCE			0.0731

FIGURE 4-7. SORTIE 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	8.5	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 OVERALL RESULTS **********

SAR SCENARIO 1

OPERATIONAL REQUIREMENTS:

MAXIMUM DURATION 12.0 HOURS
RANGE FRACTION 0.90
VISIBILITY VERY GOOD
AVERAGE SEA STATE 2.0

SELECTED CRAFT:

CATAMARAN
DISPLACEMENT 94 TONS
DESIGN SPEED 40 KNOTS
FUEL FRACTION 0.50

PERCENT OF SCENARIO COMPLETED 98.1

PROBABILITY OF SUCCESSFULLY COMPLETING SCENARIO 0.92

SPECIFICATIONS OF THE AVERAGE SORTIE:

TIME TO COMPLETE AVERAGE SORTIE 7.2 HRS

FUEL CONSUMED IN AVERAGE SORTIE 1439.1 GALS

TASK COMPOSITION IN AVERAGE SORTIE:

TASK CODE	TIMES COMPLETED	TASK NAME
ON SCENE:		
BRD	0.40	BOARD
GAS	0.19	GENERAL ASSISTANCE
LSB	0.23	LAUNCH SMALL BOAT
OBA	0.63	ON BOARD ASSISTANCE
RBP	0.40	RETRIEVE BOARDING PARTY
RSB	0.23	RETRIEVE SMALL BOAT
REDUCED SPEED:		
SDU	0.76	SEARCH FOR DISTRESSED UNIT: FOUND
SDU	0.10	SEARCH FOR DISTRESSED UNIT: FAILED
SES	0.06	SLOW ESCORT
SPT	0.15	SLOW PATROL
TOW	0.38	TOW
CRUISE SPEED:		
PAT	0.11	PATROL
TPL	0.19	TRANSPORT PEOPLE
TRA	0.29	TRANSIT
FLANK SPEED:		
DSH	1.13	DASH

FIGURE 4-9. SCENARIO OVERALL RESULTS 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  160 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

5. CREE PROGRAM STRUCTURE

The CREE program is stored in the computer as a main program, CREE, and three sections of subprograms: the Characteristics Section, SCHAR, which computes the craft characteristics; the Parameter/Task Probability of Success Section, SPTPOS, which computes parameters and task probabilities of success; and the Program Probability of Success Section, SPRPOS, which finds the sorties and computes the evaluation outputs. Each subprogram section consists of a main subprogram and several subroutines. The main program and the three subprograms are each stored in a separate computer file. Because it is so large, the Program Probability of Success Section is stored in two files: SPRPOS having the main subprogram, and SPRSUB having the subroutines that go with it. Thus the internal structure of the program storage is as shown in Figure 5-1.

The computer files associated with the CREE program are listed in Figure 5-2. There are five object files in which the CREE Program is stored. A control file, CREE.CNTL, contains the Job Control Language (JCL) commands needed to run the CREE Program, obtaining needed inputs from the two input-data files CRAFT.DATA and SCENARIO.DATA.

The above files allow use of the CREE model. However, to make modifications to the CREE program possible, the five Fortran files corresponding to the object files are provided, along with seven control files. If a change is made to one of the Fortran files (e.g., SCHAR.FORT), the corresponding control file (e.g., CMSCHAR.CNTL) can be SUBMITed. This file contains the JCL to recompile the FORTRAN file and then run the entire CREE program with the modification. If more than one program section is changed, the CMALL.CNTL file can be used to recompile all five Fortran files and run the entire program. A LISTCREE.CNTL file contains the JCL to print a listing of the entire CREE Program.

A listing of the CREE program is in Appendix D.

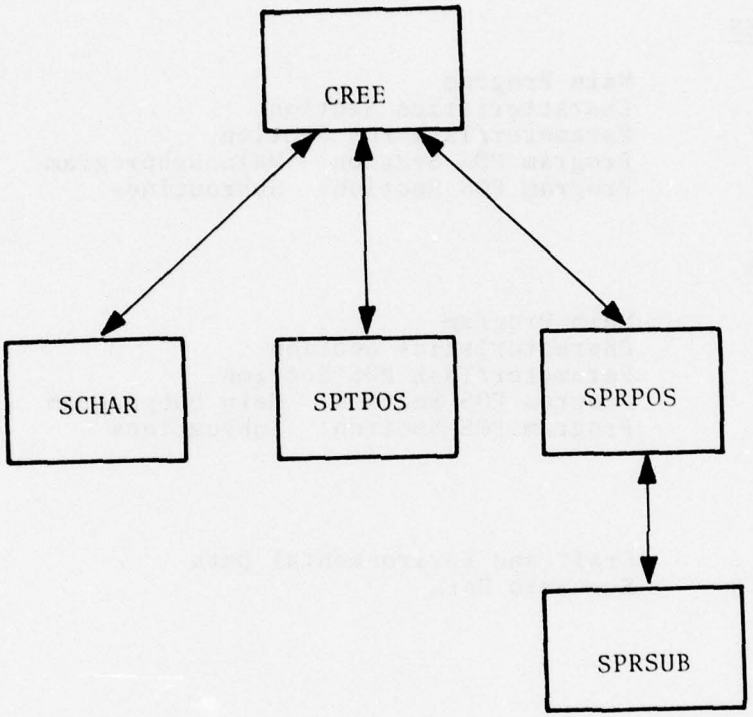


FIGURE 5-1. CREE PROGRAM STORAGE STRUCTURE

CONTROL FILES

LISTCREE.CNTL	Lists all Fortran Files.
CREE.CNTL	Runs CREE Program.
CMALL.CNTL	Compiles all Fortran files and runs CREE Program.
CMCREE.CNTL	Compiles CREE.FORT and runs CREE Program.
CMSCHAR.CNTL	Compiles SCHAR.FORT and runs CREE Program.
CMSPTPOS.CNTL	Compiles SPTPOS.FORT and runs CREE Program.
CMSPRPOS.CNTL	Compiles SPRPOS.FORT and runs CREE Program.
CMSPRSUB.CNTL	Compiles SPRSUB.FORT and runs CREE Program.

FORTRAN FILES

CREE.FORT	Main Program
SCHAR.FORT	Characteristics Section
SPTPOS.FORT	Parameter/Task POS Section
SPRPOS.FORT	Program POS Section: Main Subprogram
SPRSUB.FORT	Program POS Section: Subroutines

OBJECT FILES

CREE.OBJ	Main Program
SCHAR.OBJ	Characteristics Section
SPTPOS.OBJ	Parameter/Task POS Section
SPRPOS.OBJ	Program POS Section: Main Subprogram
SPRSUB.OBJ	Program POS Section: Subroutines

DATA FILES

CRAFT.DATA	Craft and Environmental Data
SCENARIO.DATA	Scenario Data

FIGURE 5-2. CREE PROGRAM COMPUTER FILES

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.

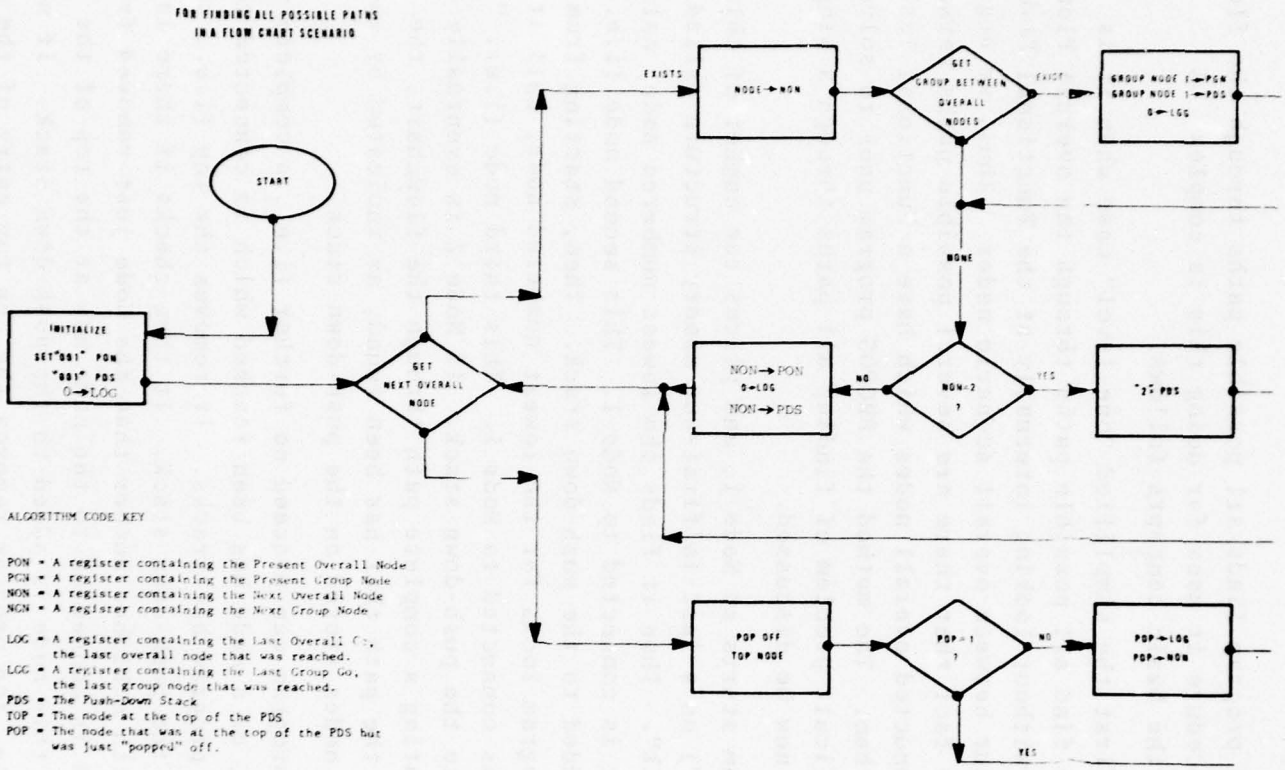


FIGURE 5.1-1. PROPOS ALGORITHM FLOW CHART

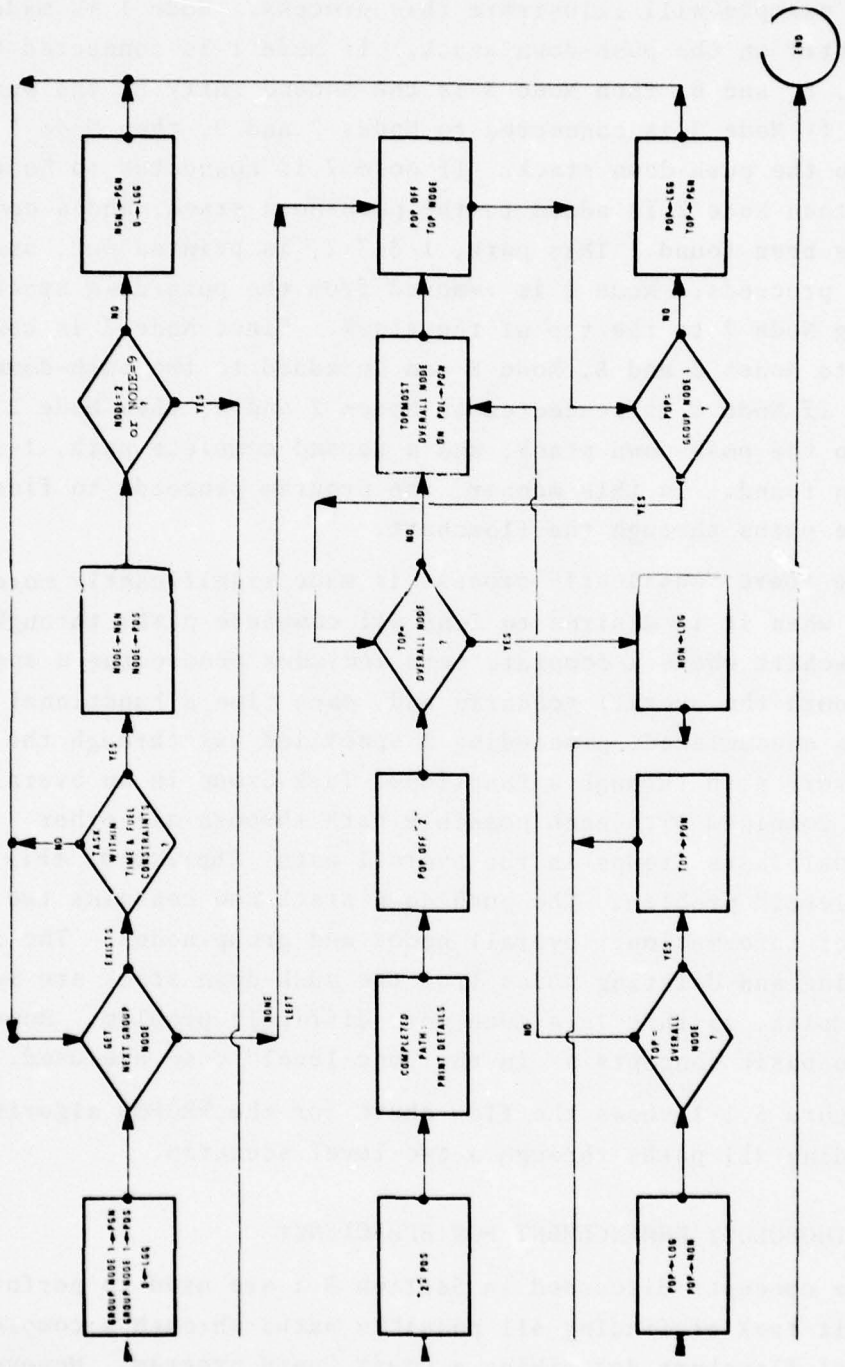


FIGURE 5.1-1. PROPOS ALGORITHM FLOW CHART (CONTINUED)

An example will illustrate this process. Node 1 is made the first entry on the push-down stack. If Node 1 is connected to Nodes 3, 5, and 6, then Node 3 is the second entry on the push-down stack. If Node 3 is connected to Nodes 7 and 9, then Node 7 is added to the push-down stack. If Node 7 is connected to Nodes 2 and 8, then Node 2 is added to the push-down stack, and a complete path has been found. This path, 1-3-7-2, is printed out, and the program proceeds. Node 2 is removed from the push-down stack, bringing Node 7 to the top of the stack. Since Node 7 is connected to Nodes 2 and 8, Node 8 can be added to the push-down stack. If Node 8 is connected to Nodes 2 and 4, then Node 2 is added to the push-down stack, and a second complete path, 1-3-7-8-2, has been found. In this manner, the program proceeds to find all possible paths through the flowchart.

The above "one-level" process is made significantly more complex when it is desired to find all complete paths through the flowchart where a complete path includes proceeding a specified way through the overall scenario and, each time a Functional Task Group is encountered, proceeding a specified way through the Group. Thus, every path through a Functional Task Group in an overall path must be combined with each possible path through all other Functional Tasks Groups in the overall path. Therefore, this is now a "two-level" problem. The push-down stack now contains two levels of information: overall nodes and group nodes. The rules for adding and deleting nodes from the push-down stack are much more complex, as this is a much more difficult problem. However, the same basic concepts as in the "one-level" case are used.

Figure 5.1-1 shows the flow chart for the PROPOS algorithm for finding all paths through a two-level scenario.

5.2 METHODOLOGY ENHANCEMENT FOR EFFICIENCY

The concepts discussed in Section 5.1 are used to perform the difficult task of finding all possible paths through a complex two-level flowchart describing a Coast Guard program. However, although this methodology solves the problem, it is inefficient.

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

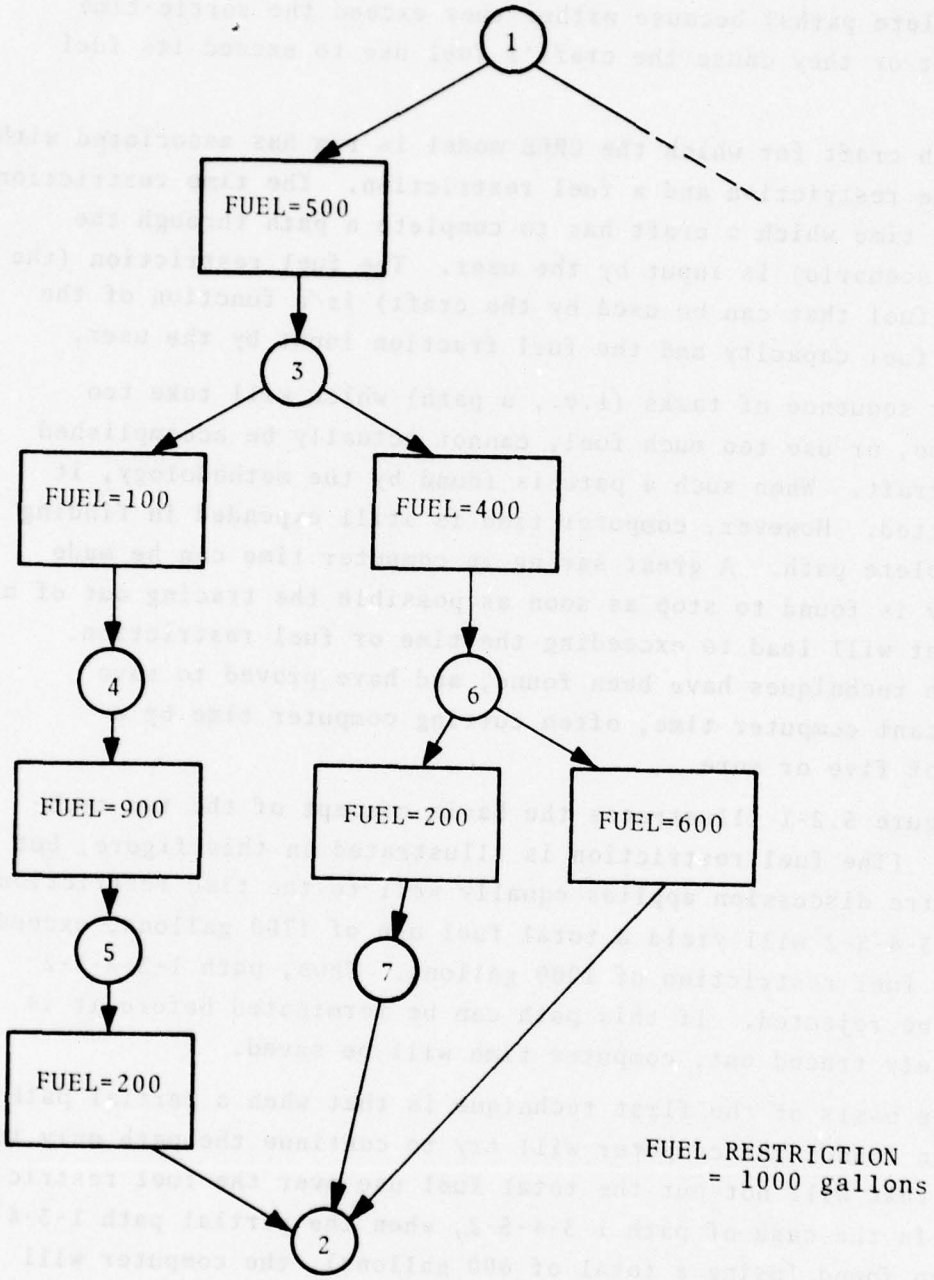


FIGURE 5.2-1. ILLUSTRATION FOR METHODOLOGY FOR EFFICIENCY ENHANCEMENT

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.

5.3 ALGORITHM FOR MINIMUM FUEL USE FROM ANY NODE TO THE END OF THE SORTIE

The algorithm for finding the minimum fuel that can be used to get from any node to the end of the sortie (Node 2) is as follows:

(1) Minimum Fuel through Each Group

The minimum fuel through each Functional Task Group is found by calculating and comparing the fuel use in each of the possible paths through that Group. This can be done relatively easily, since the Groups are pre-set and thus the possible paths through each Group are known in advance.

(2) Initialization

To initialize the algorithm, the fuel consumed in getting to Node 2 (the end of the sortie) from each node in zero node-to-node steps is found. This value must be zero for Node 2 itself, and infinity for all other nodes (i.e., it takes no fuel to get from Node 2 to Node 2 in zero steps, but it takes infinite fuel--that is, it is impossible--to get to Node 2 from any other node in zero node-to-node steps).

(3) Iteration

This iteration describes a general method of finding the minimum fuel consumed to go from each node to Node 2 in n node-to-node steps, based upon knowledge of the minimum fuel consumed to go from each node to Node 2 in n-1 node-to-node steps. The iteration for n=1 is based upon the values for n=0 found in the Initialization. The iteration continues for n=2,3,4, etc., until a stopping criterion, discussed below, is met.

The minimum fuel from each node to Node 2 in n node-to-node steps (n=1,2,3,4, etc.) is found as follows: For any n,

a. if Node A is connected directly to Node B,

b. if the minimum fuel path from Node B to Node 2 that has been found thus far in the iteration takes exactly n-1 steps, and

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.

APPENDIX A: TOWING DISTRIBUTIONS

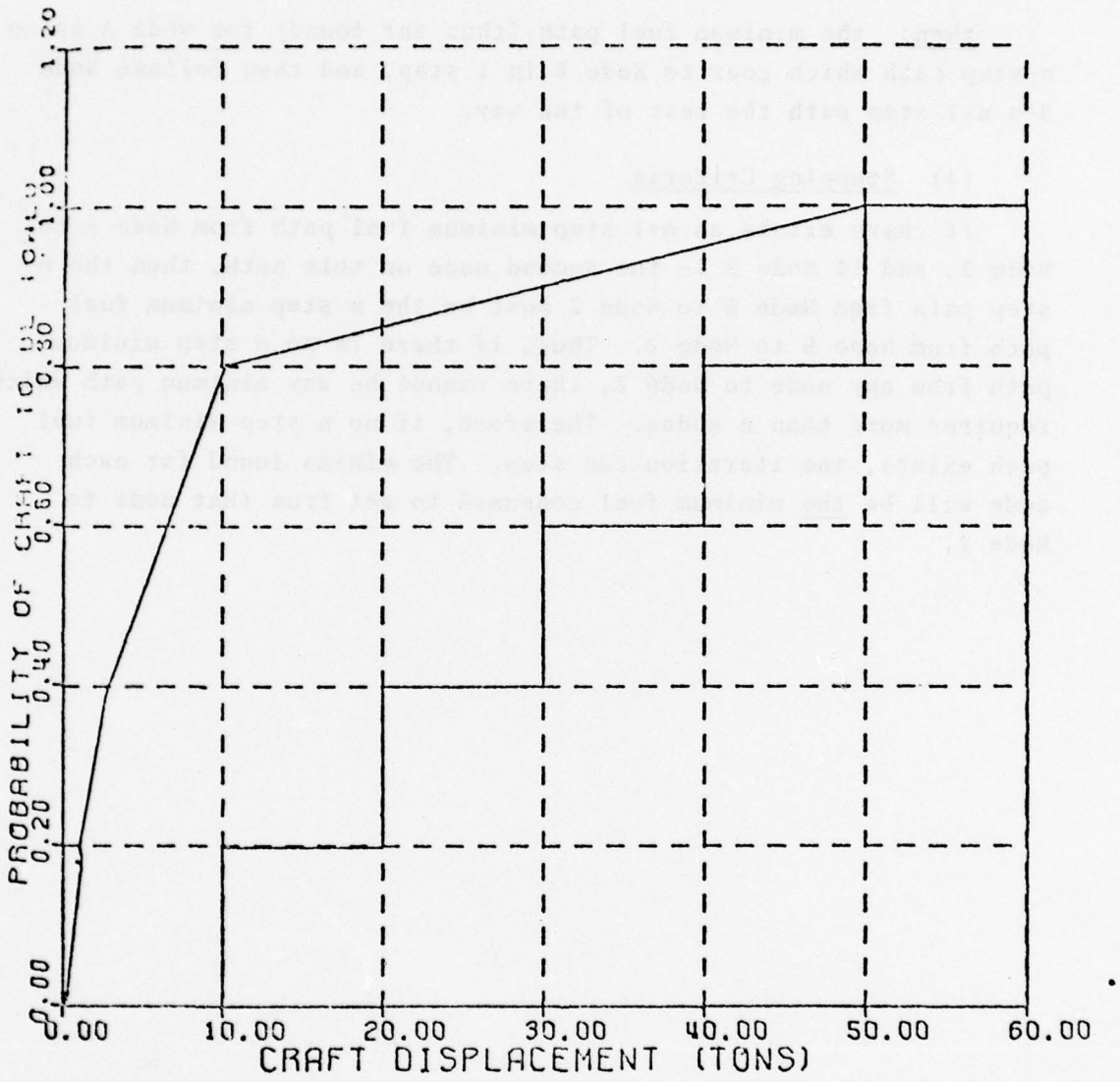


FIGURE A-1. TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT--TOW DISTRIBUTION NUMBER 1

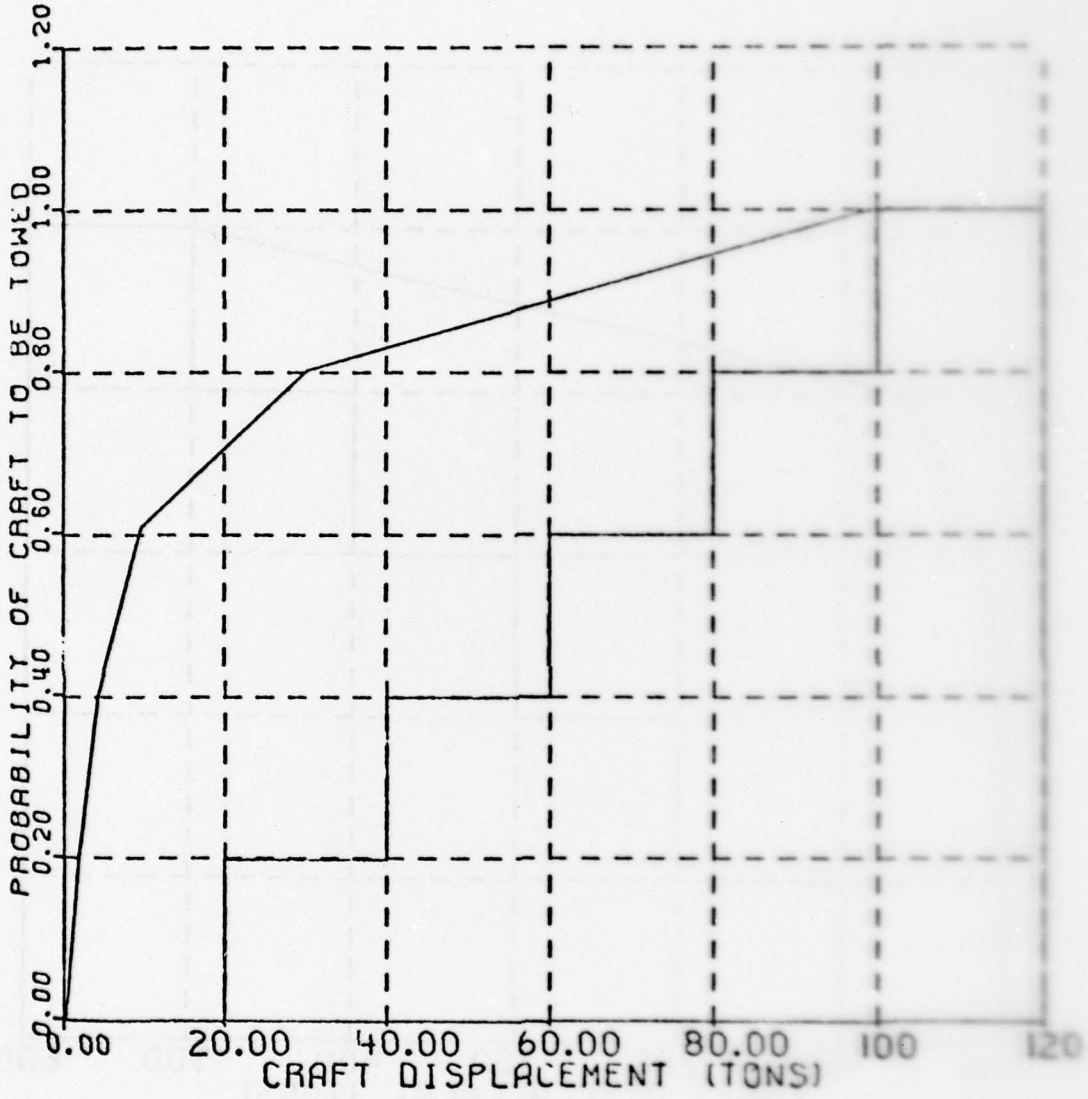


FIGURE A-2. TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT--TOW DISTRIBUTION NUMBER 2

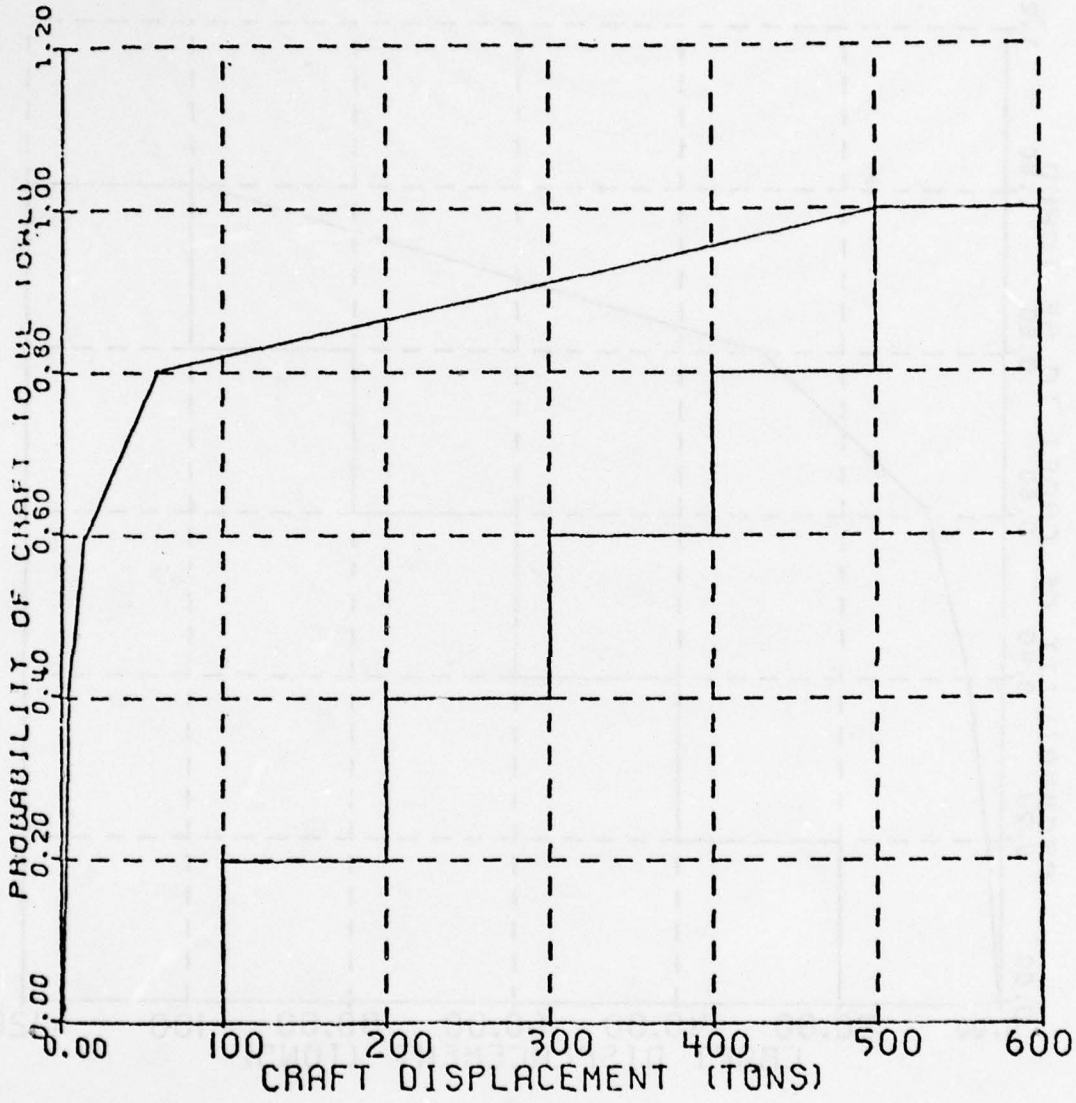


FIGURE A-3. TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT--TOW DISTRIBUTION NUMBER 3

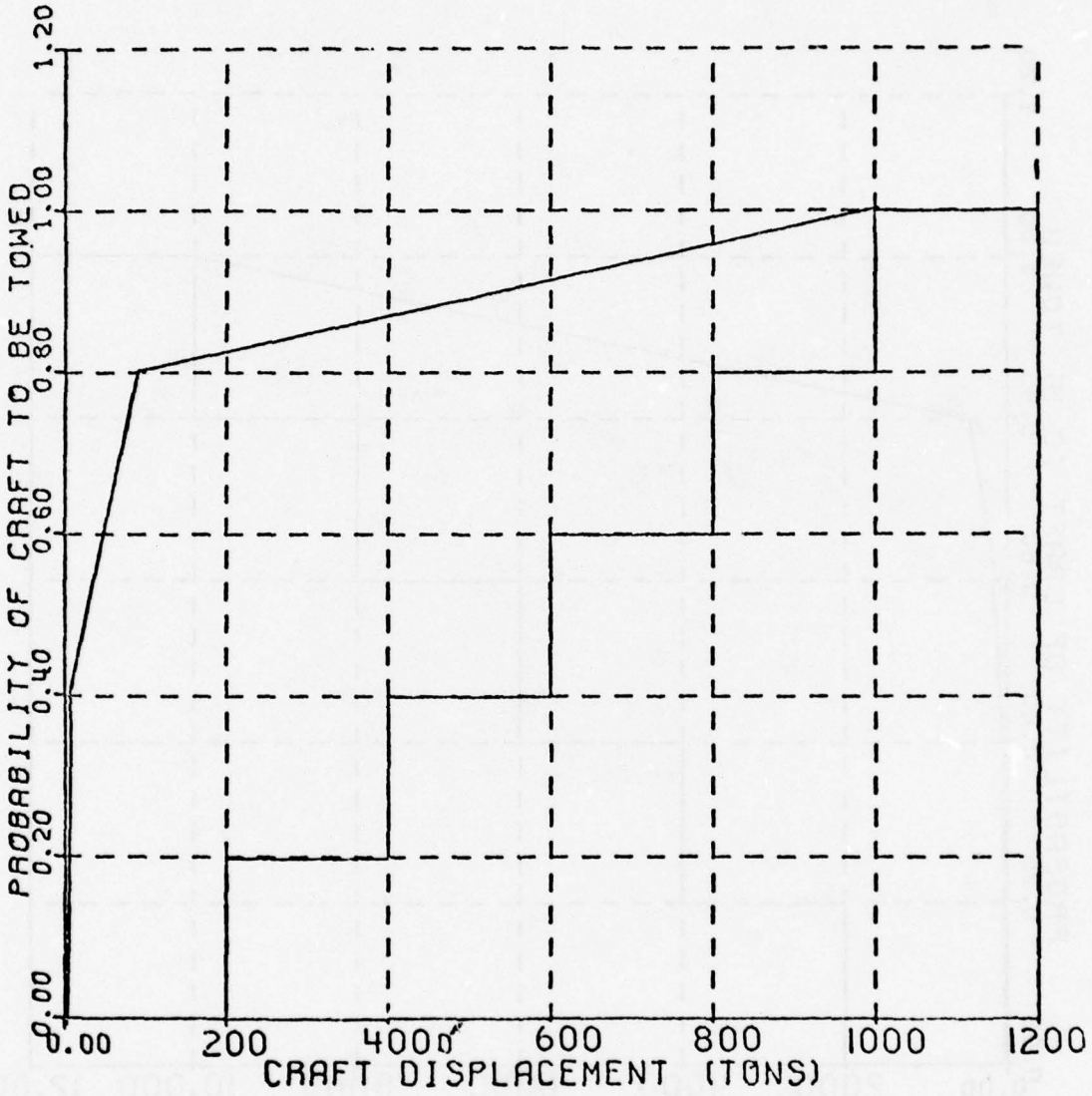


FIGURE A-4. TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT--TOW DISTRIBUTION NUMBER 4

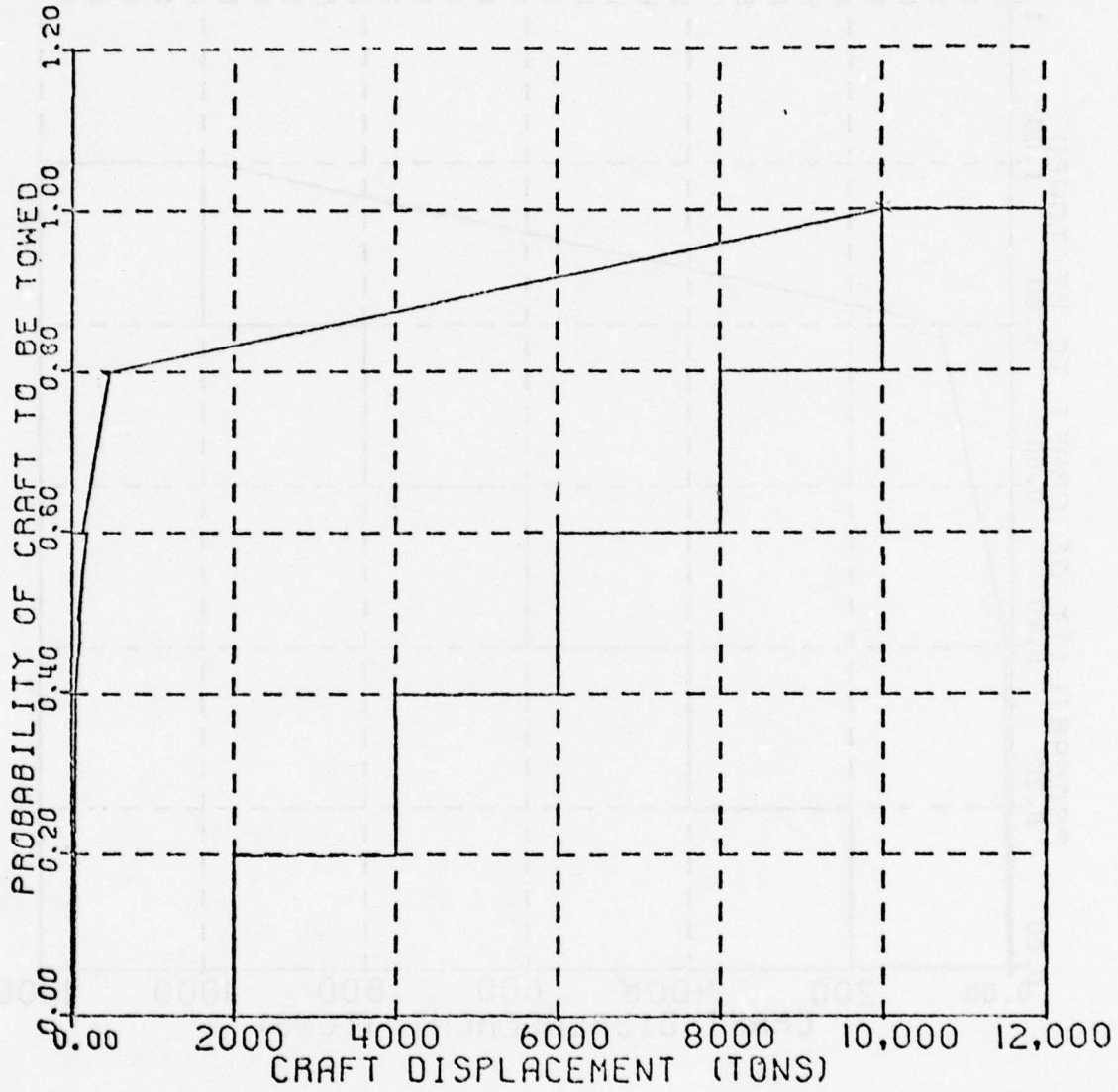


FIGURE A-5. TOW CUMULATIVE PROBABILITY VS CRAFT DISPLACEMENT--TOW DISTRIBUTION NUMBER 5

APPENDIX B: SEA-STATE DISTRIBUTIONS

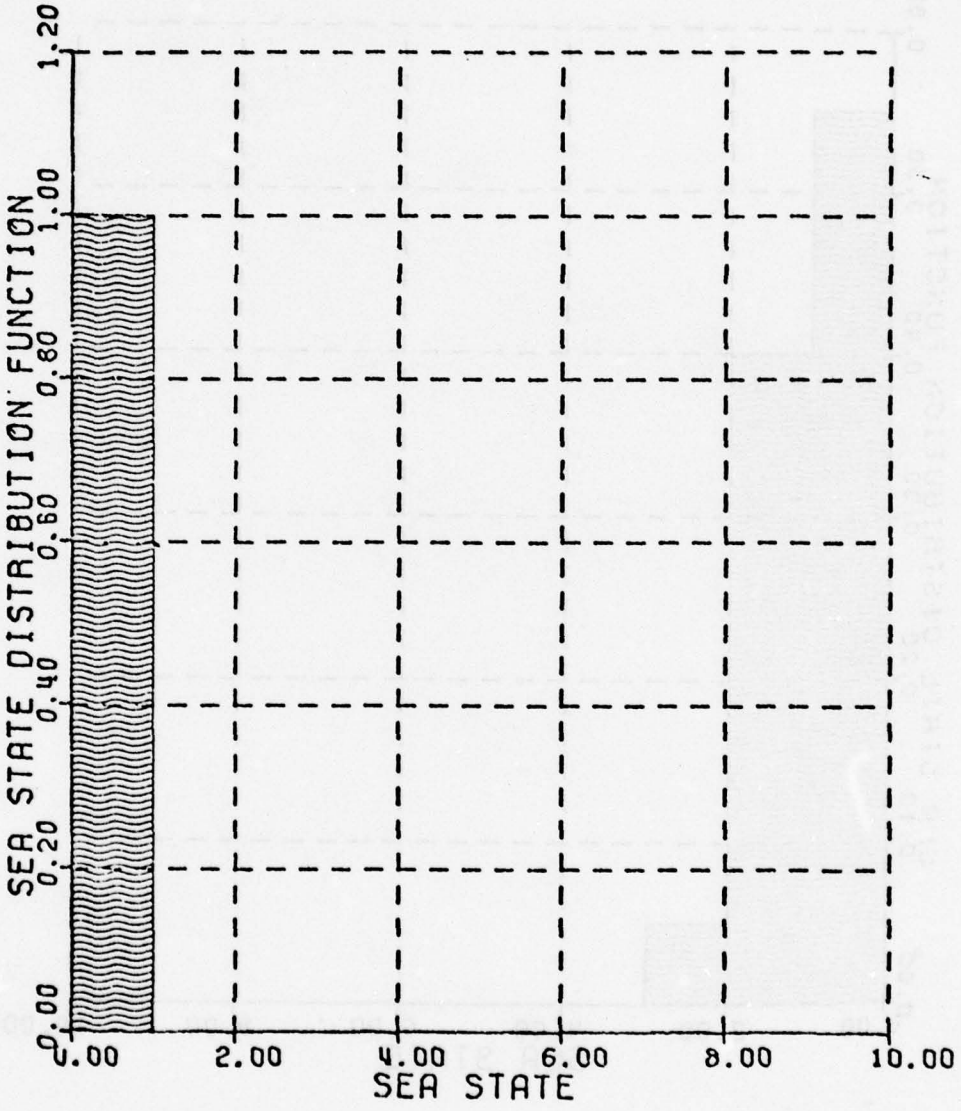


FIGURE B-1. SEA STATE DISTRIBUTION NUMBER 1--AVERAGE SS=0.5

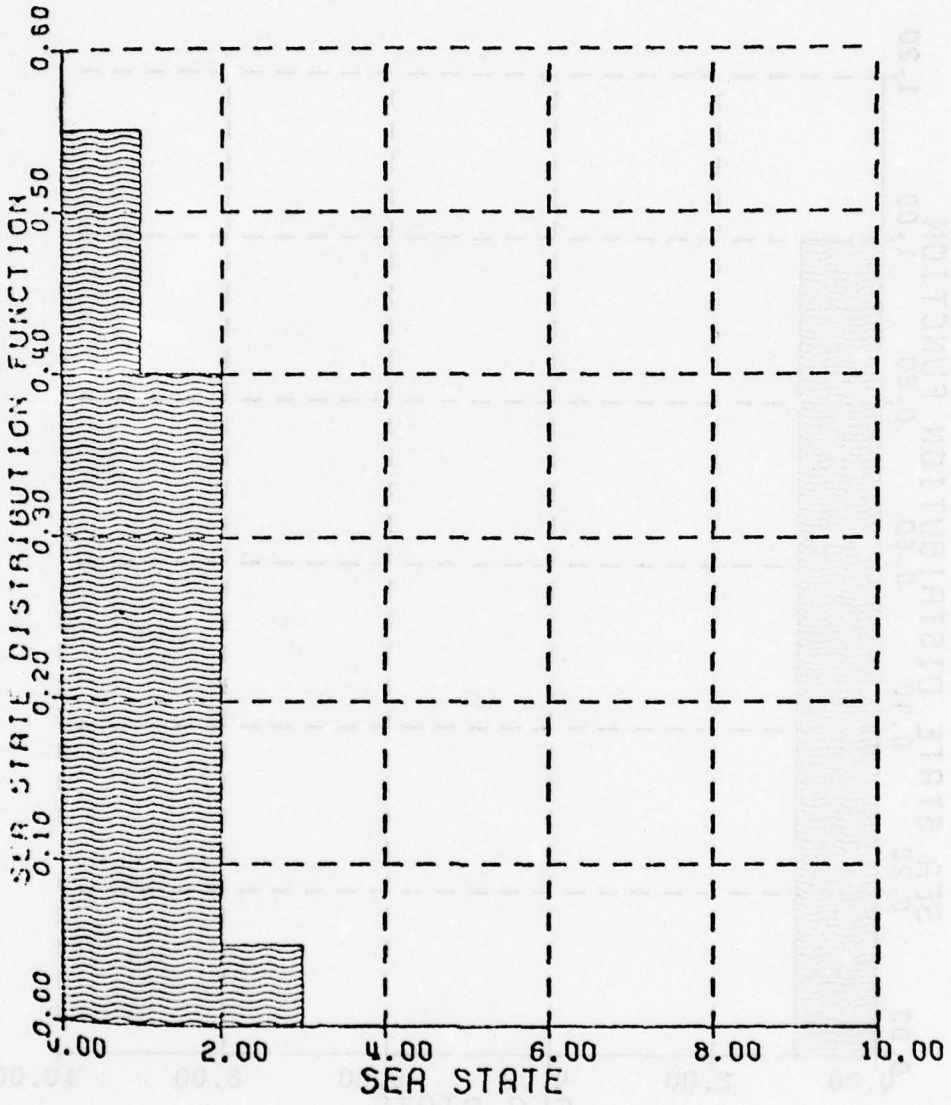


FIGURE B-2. SEA-STATE DISTRIBUTION NUMBER 2--AVERAGE SS=1.0

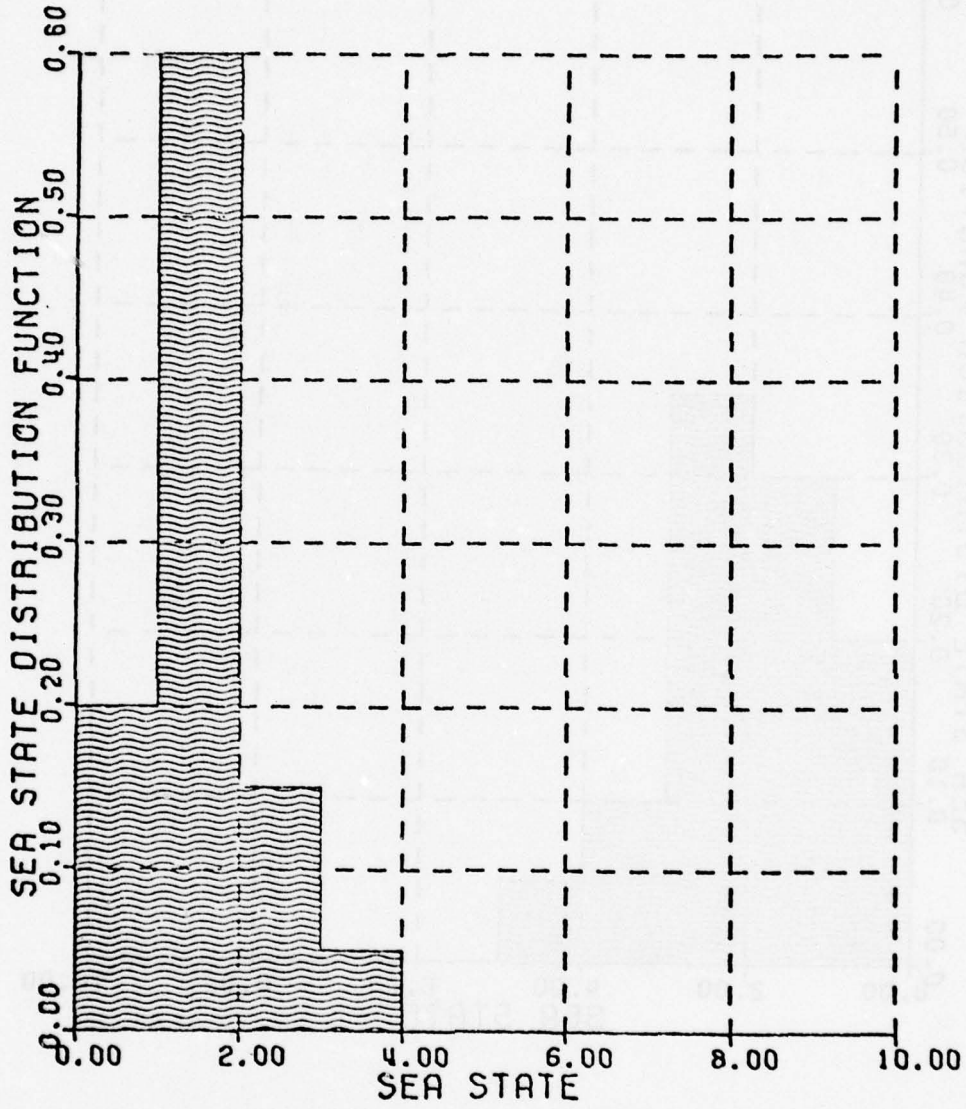


FIGURE B-3. SEA-STATE DISTRIBUTION NUMBER 3--AVERAGE SS=1.5

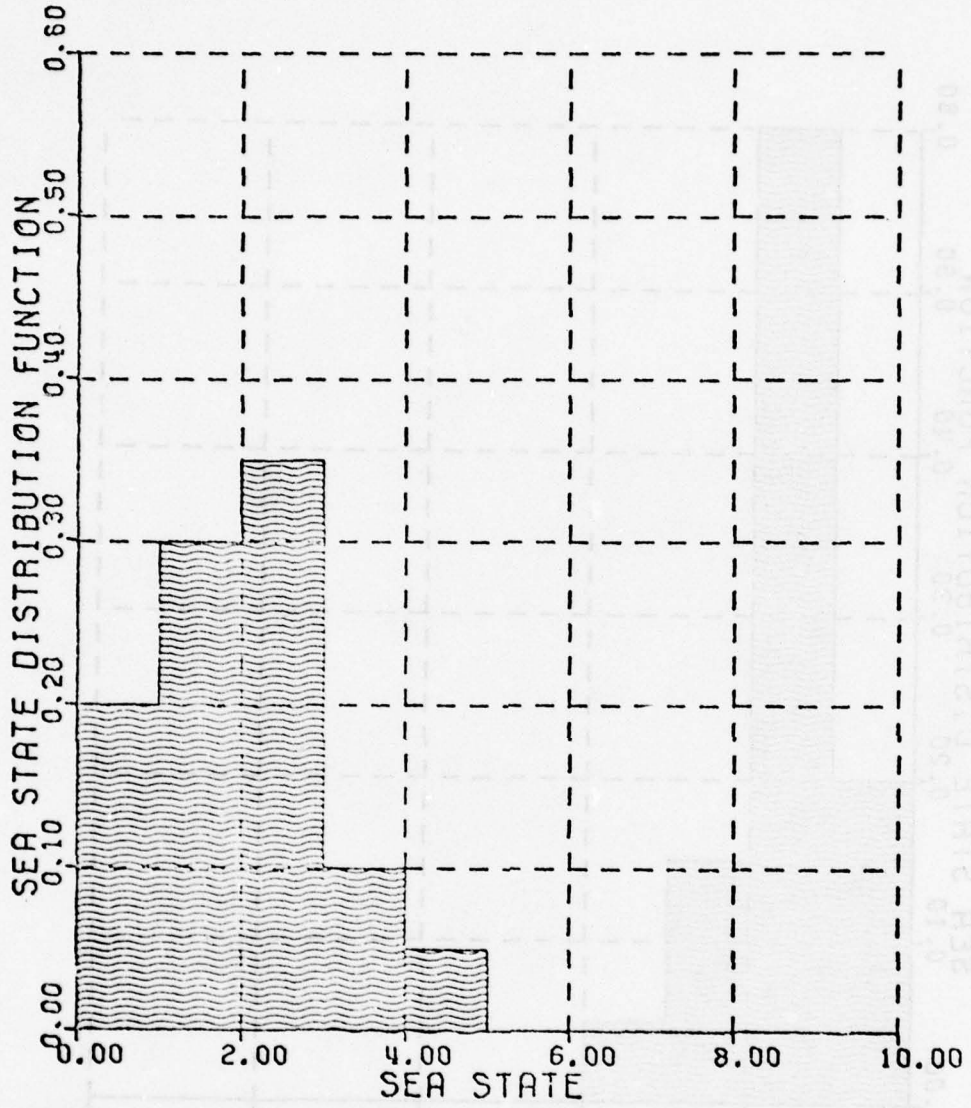


FIGURE B-4. SEA-STATE DISTRIBUTION NUMBER 4--AVERAGE SS=2.0

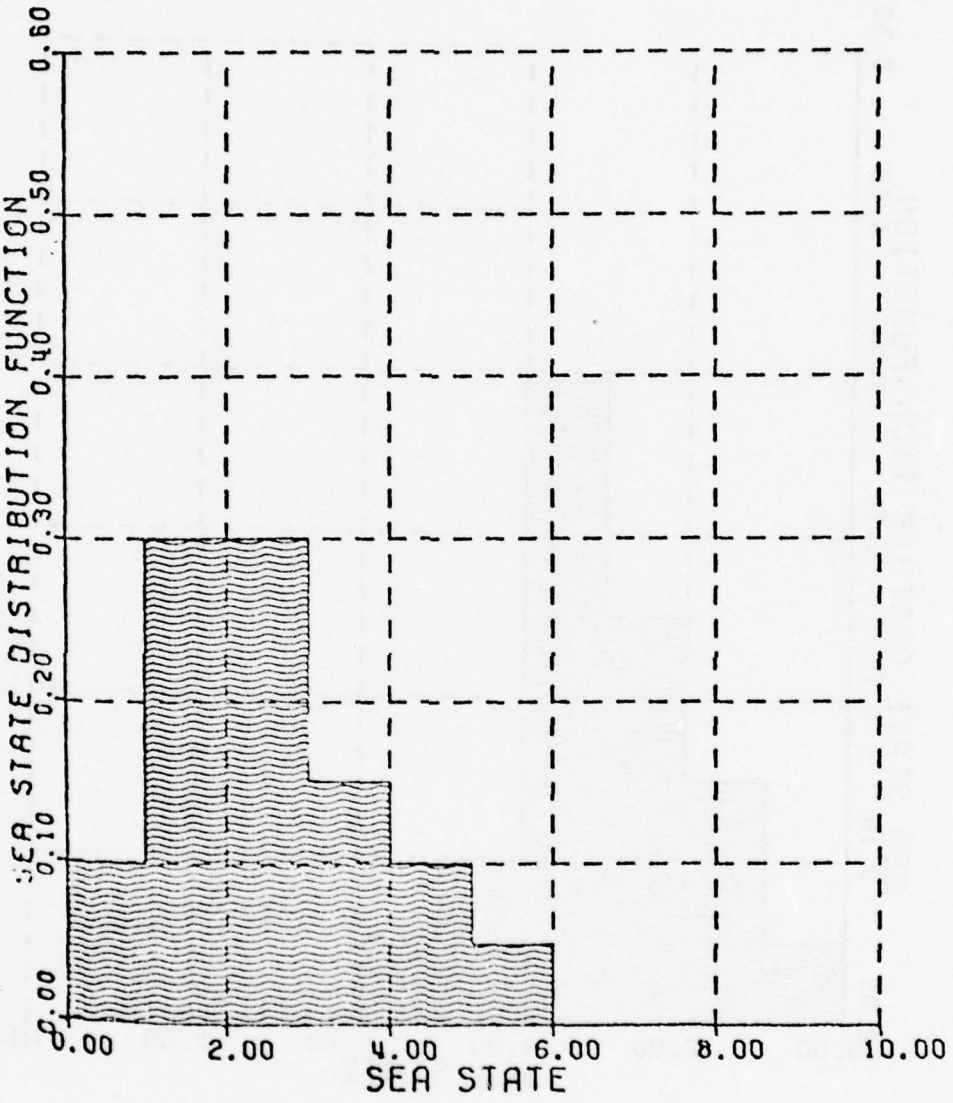


FIGURE B-5. SEA-STATE DISTRIBUTION NUMBER 5--AVERAGE SS=2.5

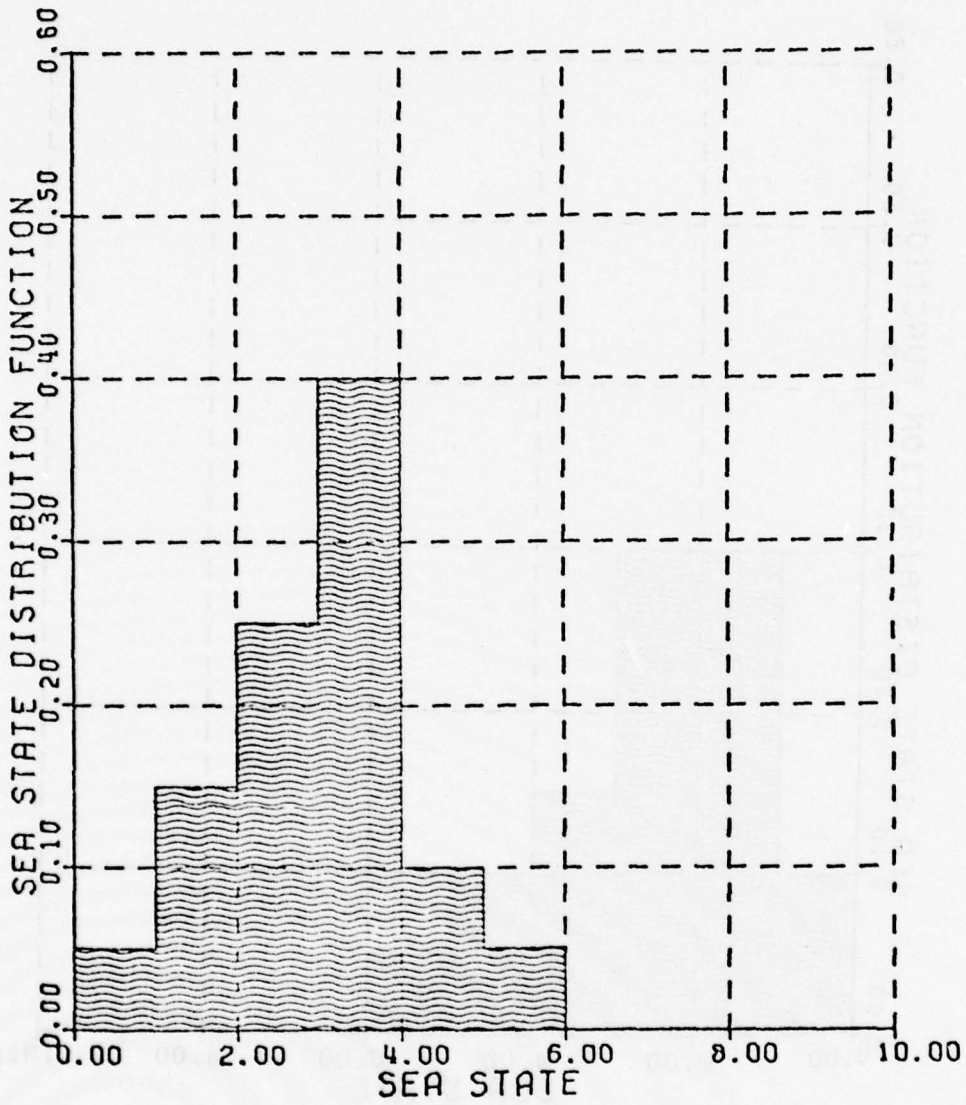


FIGURE 6. SEA-STATE DISTRIBUTION NUMBER 6--AVERAGE SS=3.0

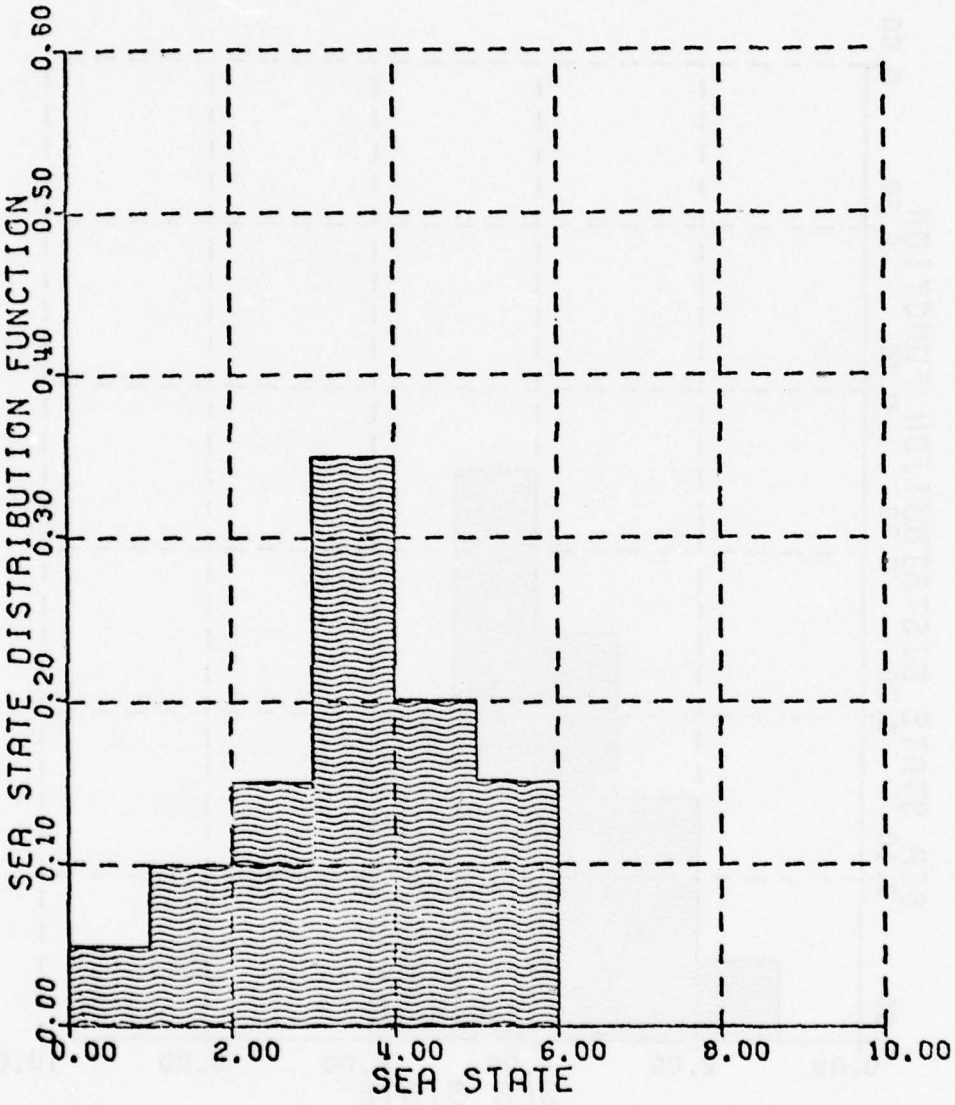


FIGURE B-7. SEA-STATE DISTRIBUTION NUMBER 7--AVERAGE SS=3.5

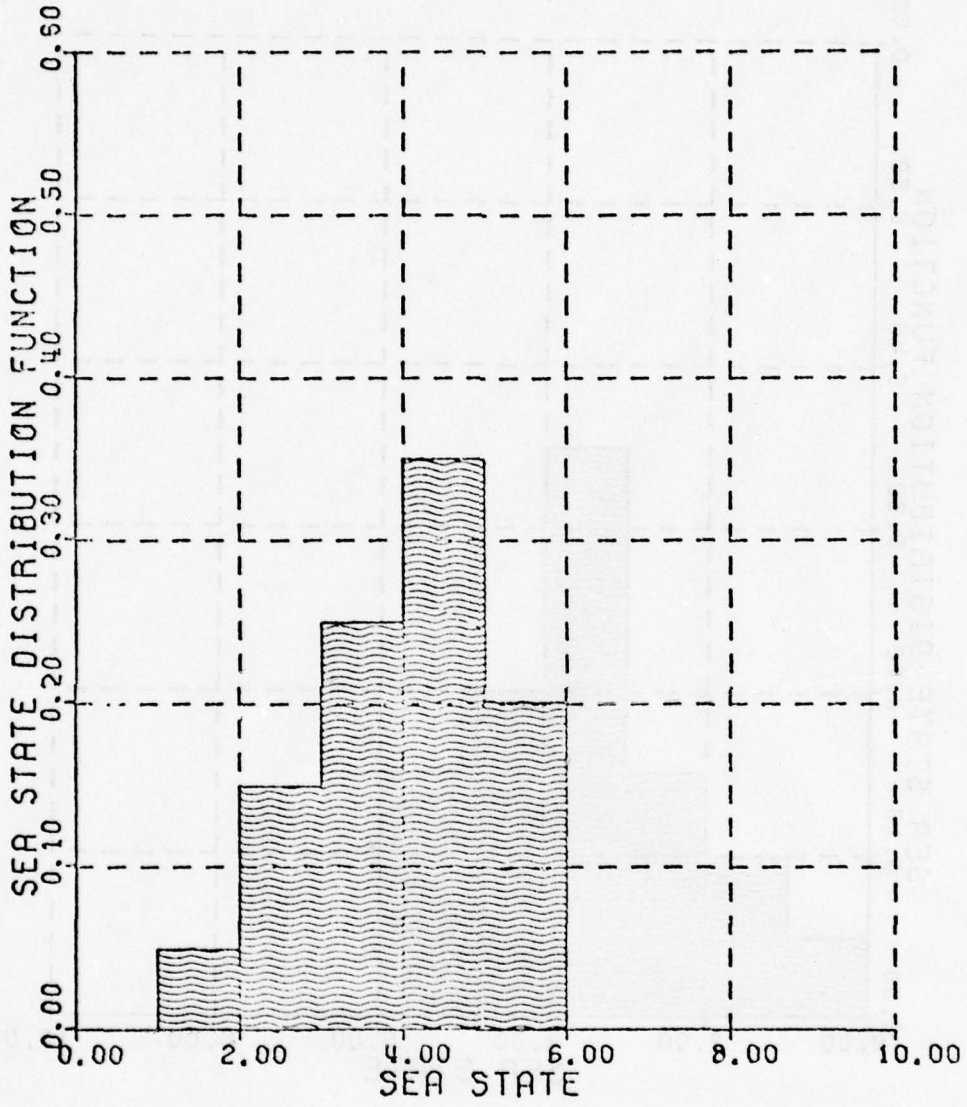


FIGURE B-8. SEA-STATE DISTRIBUTION NUMBER 8--AVERAGE SS=4.0

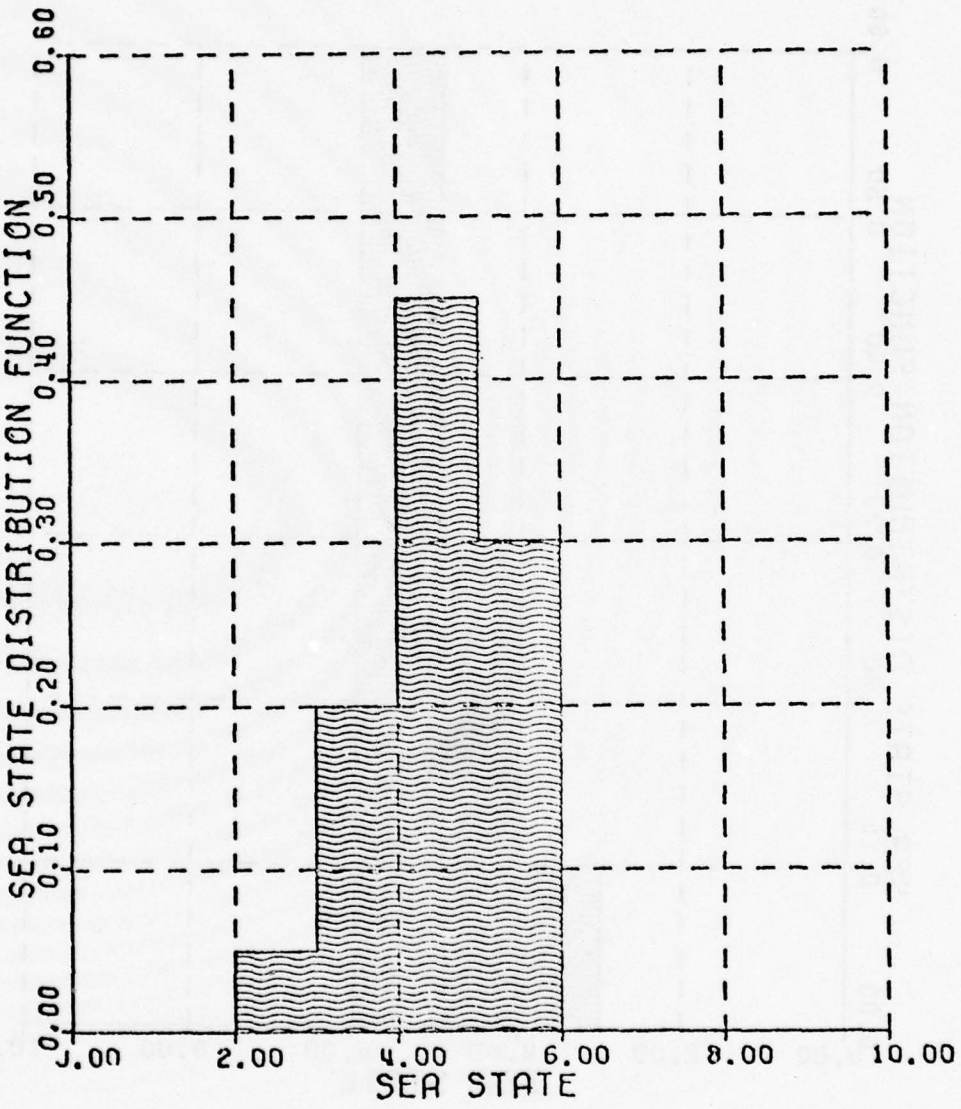


FIGURE B-9. SEA-STATE DISTRIBUTION NUMBER 9--AVERAGE SS-4.5

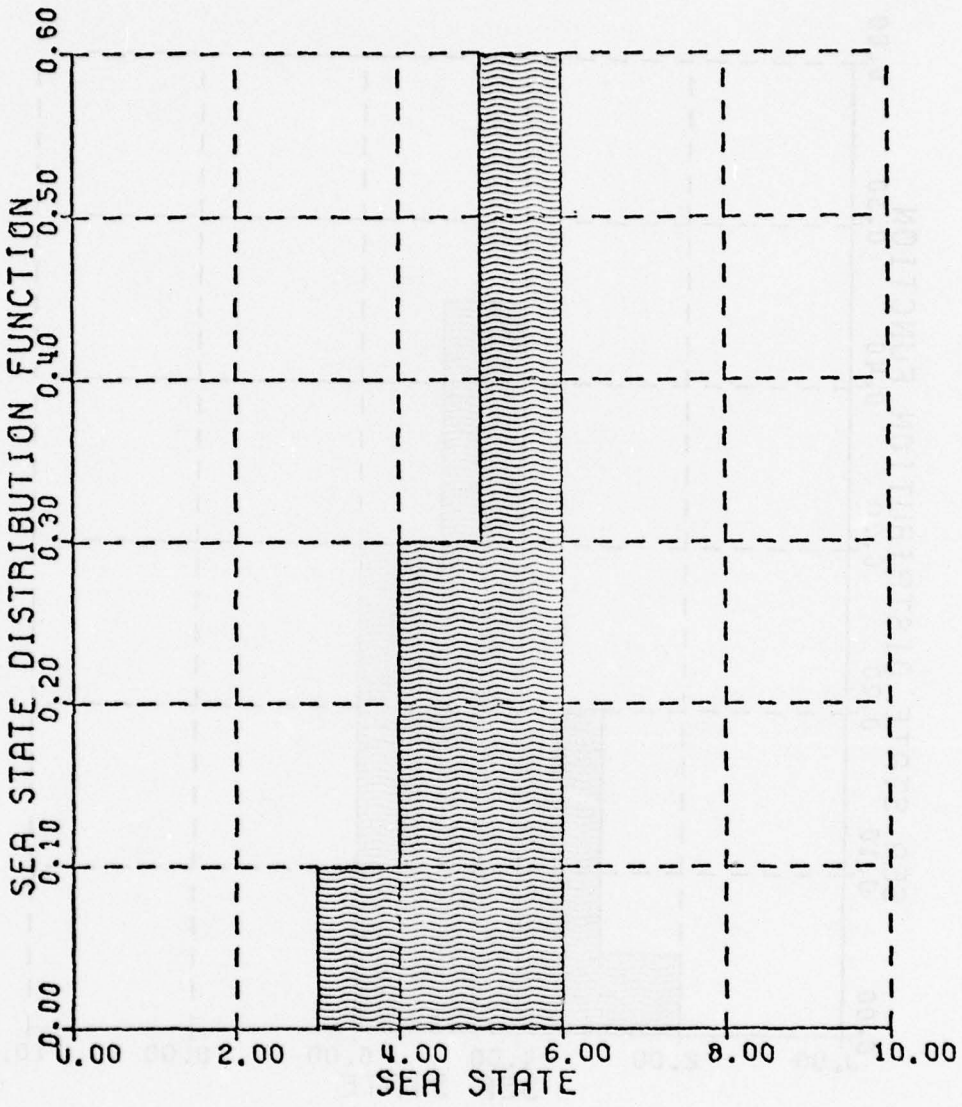
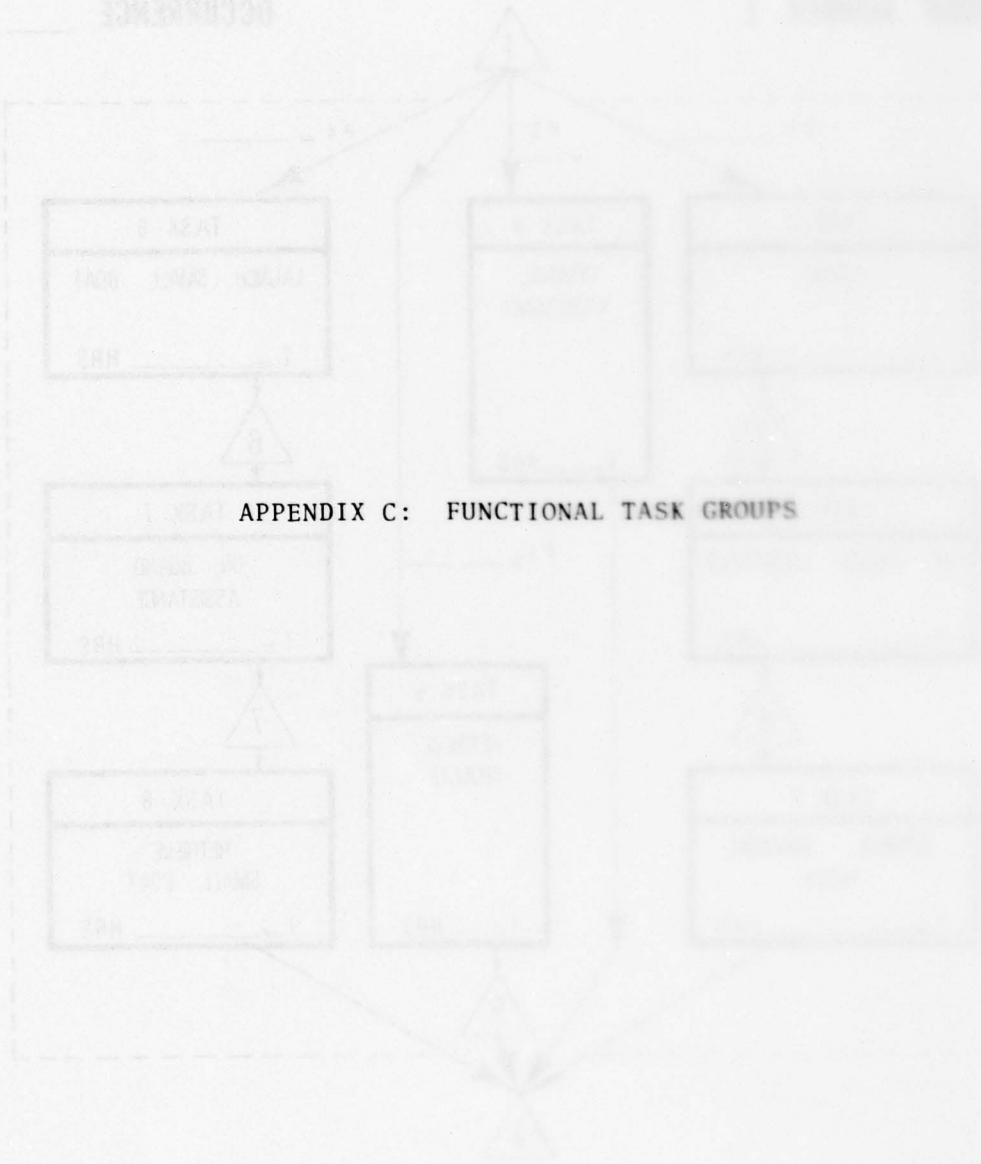


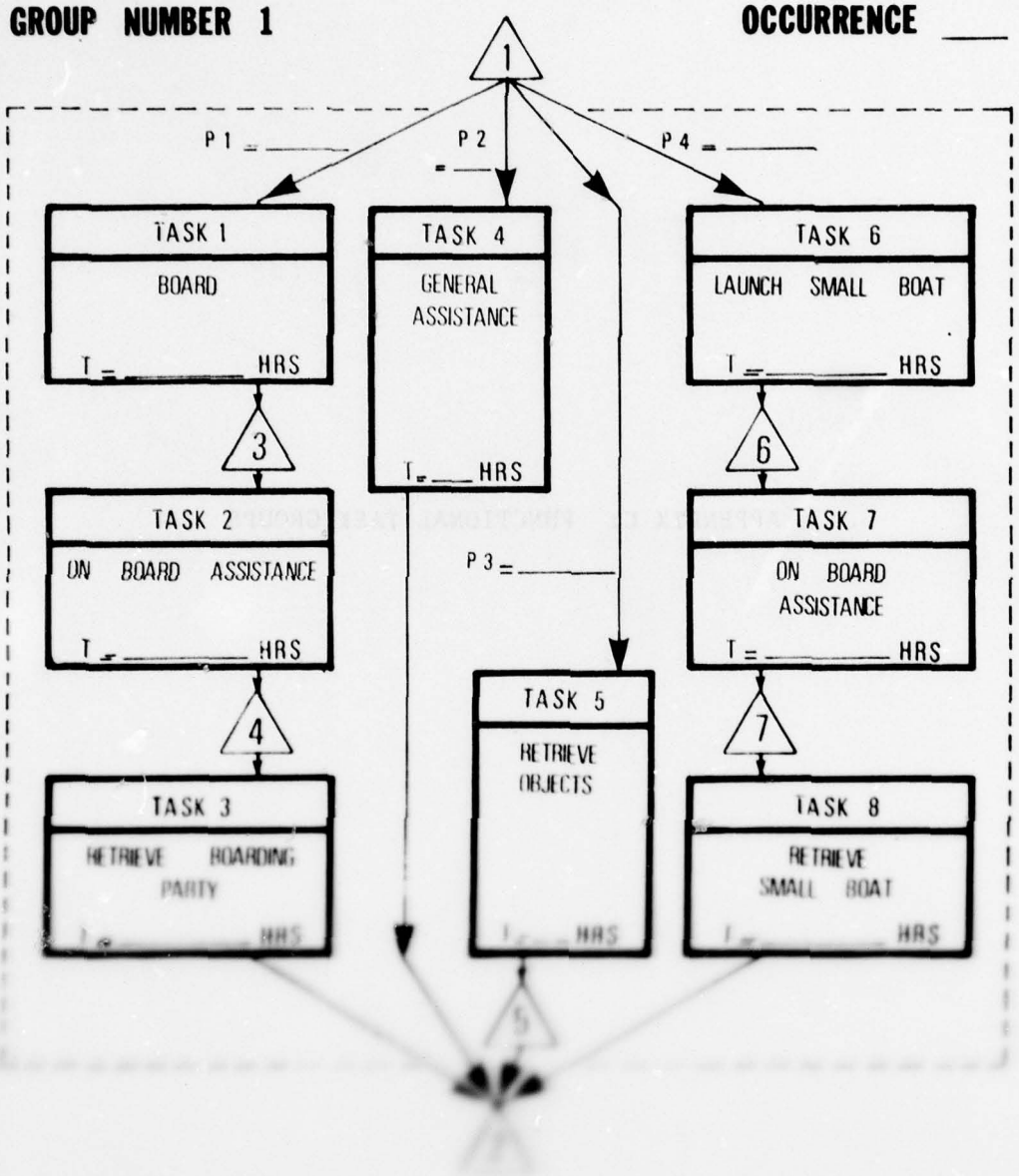
FIGURE B-10. SEA-STATE DISTRIBUTION NUMBER 10--AVERAGE SS=5.0



APPENDIX C: FUNCTIONAL TASK GROUPS

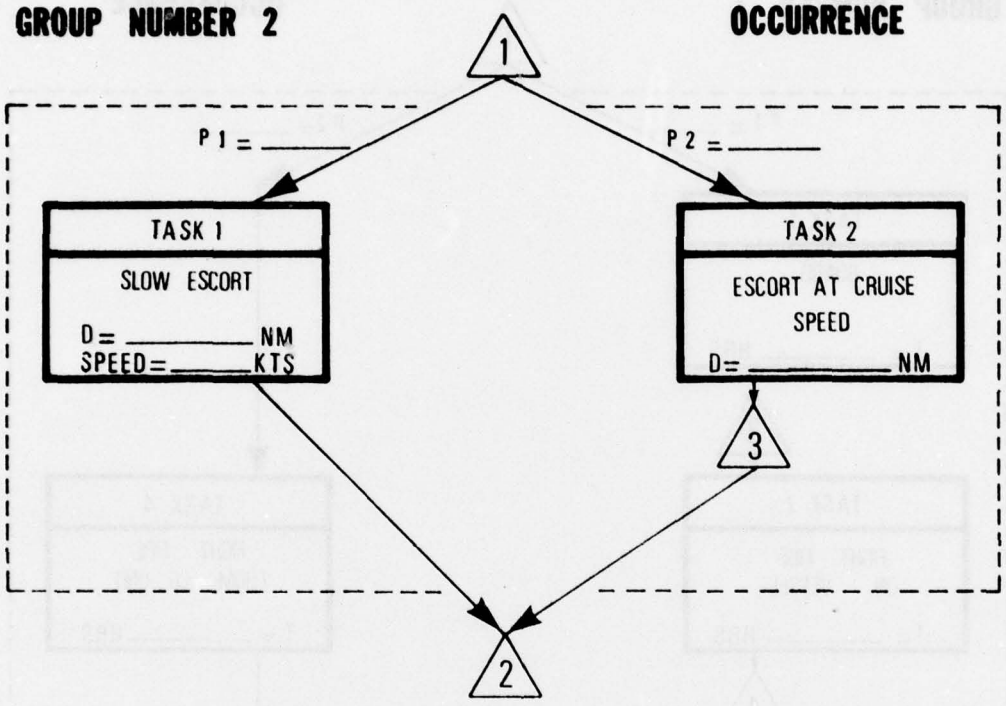
ASSIST GROUP

GROUP NUMBER 1 **OCCURRENCE ____**



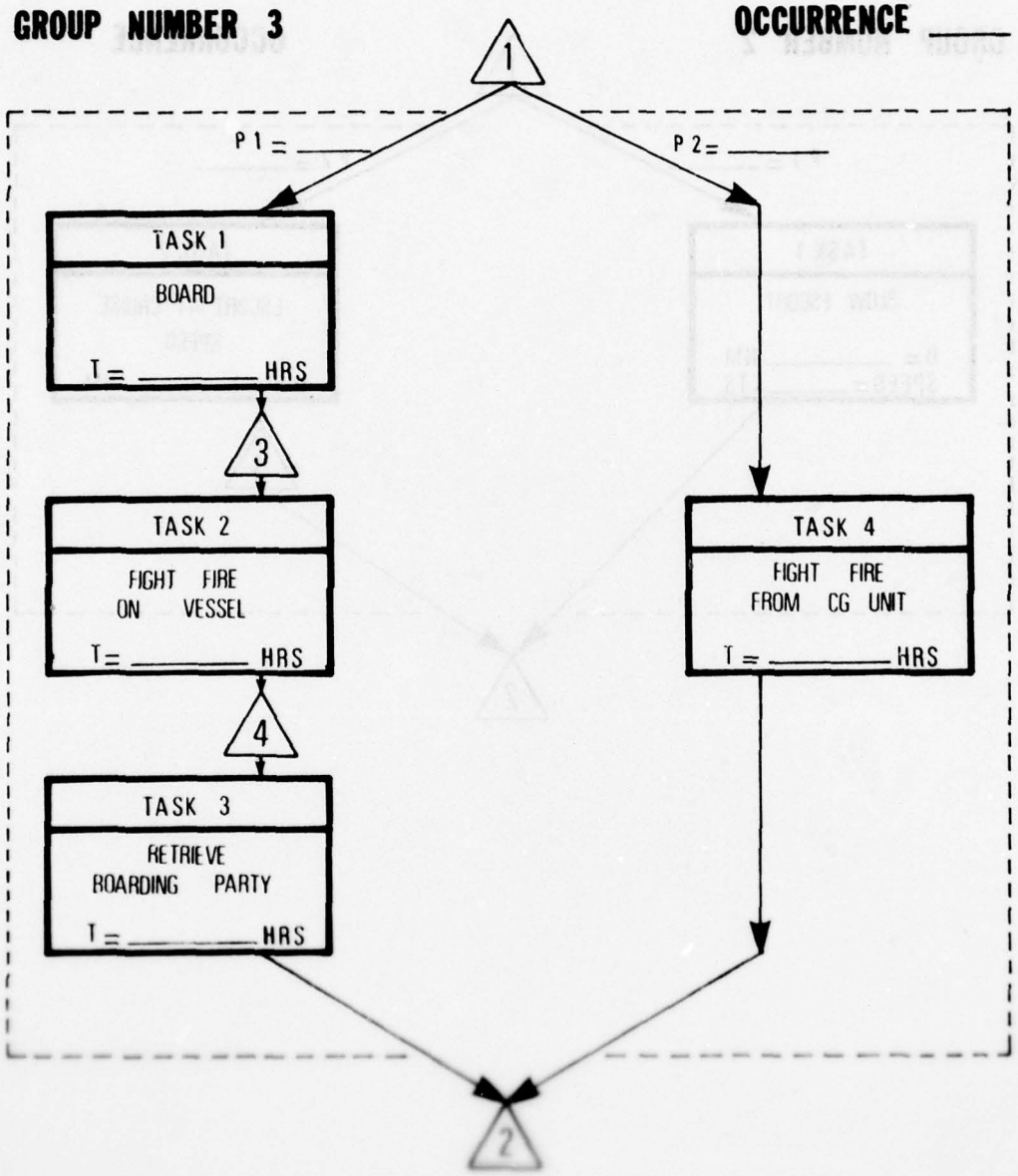
ESCORT GROUP

GROUP NUMBER 2 **OCCURRENCE**



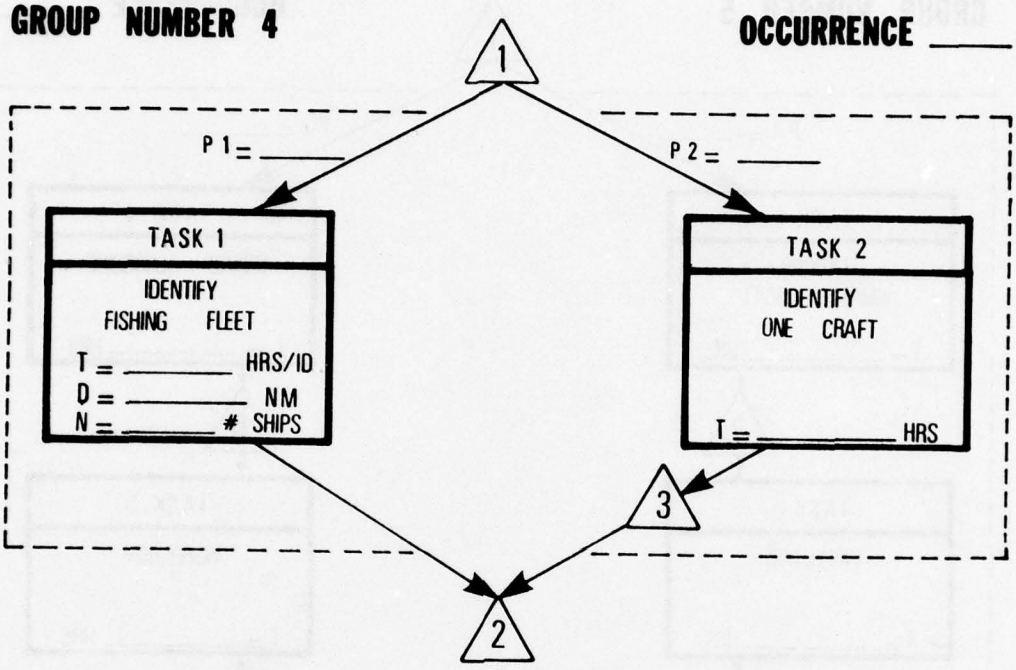
FIGHT FIRE GROUP

GROUP NUMBER 3 **OCCURRENCE ____**



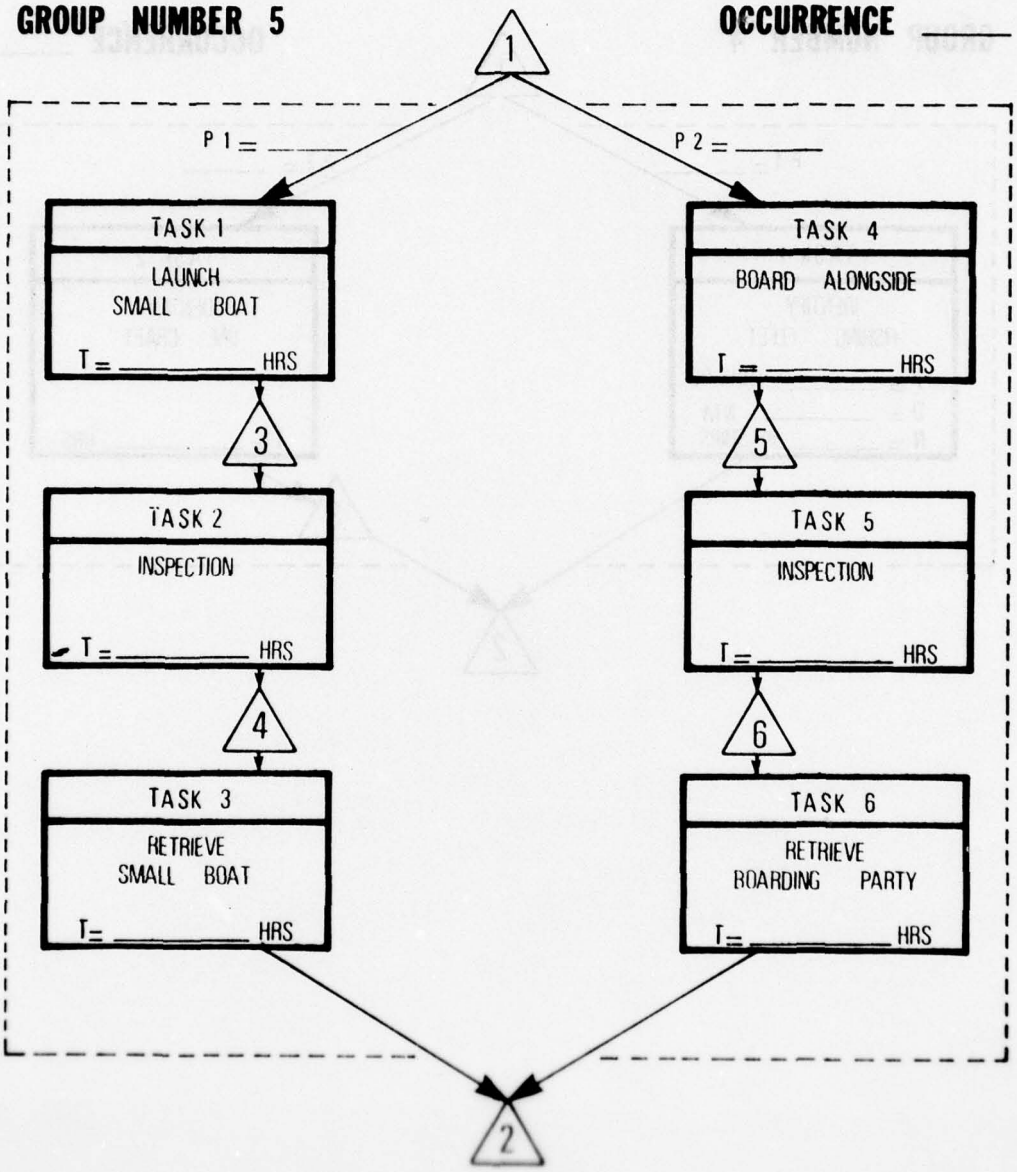
IDENTIFY GROUP

GROUP NUMBER 4 **OCCURRENCE ____**



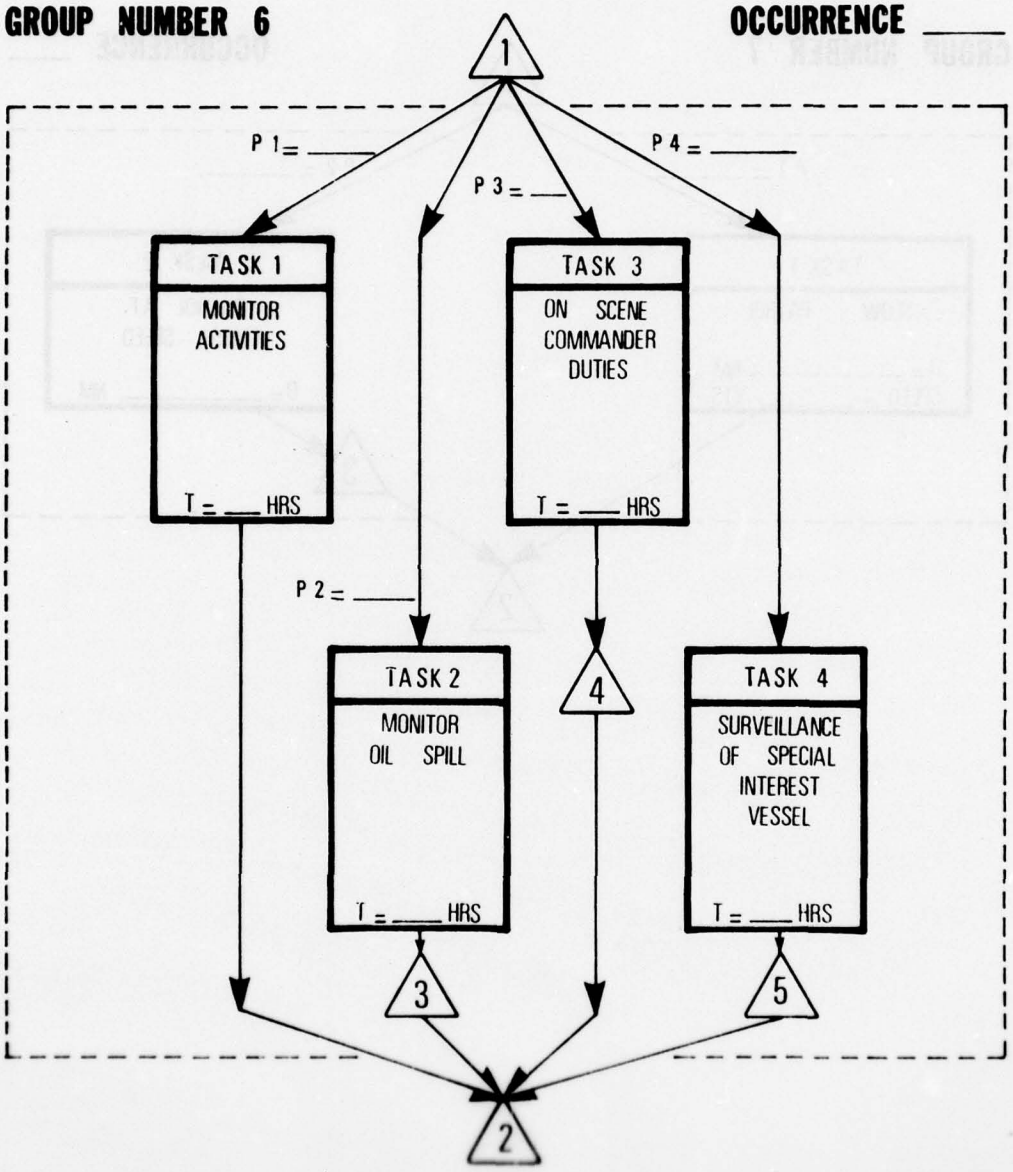
INSPECT GROUP

GROUP NUMBER 5 **OCCURRENCE ____**



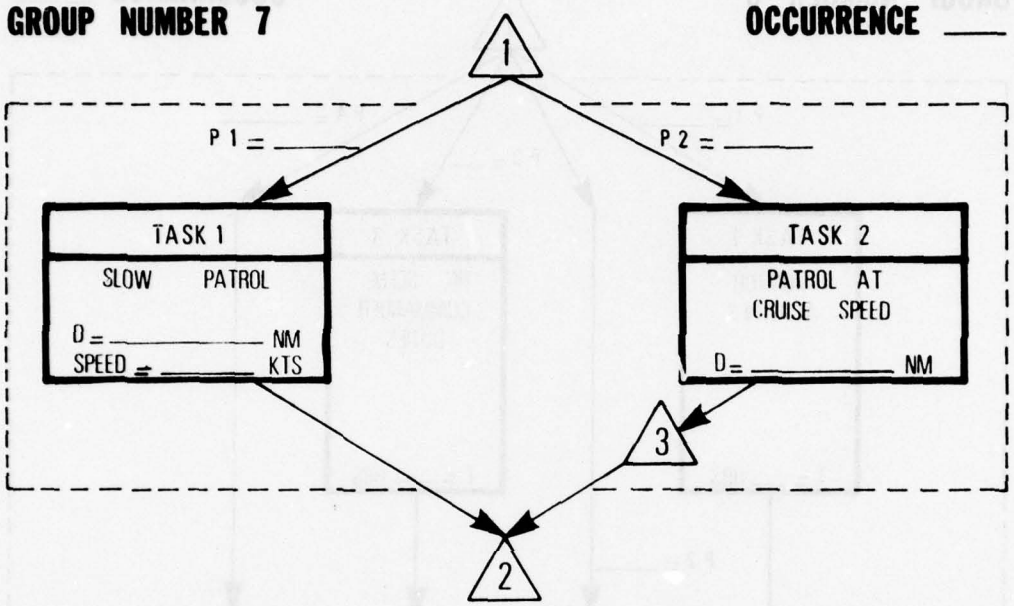
MONITOR GROUP

GROUP NUMBER 6 **OCCURRENCE ____**



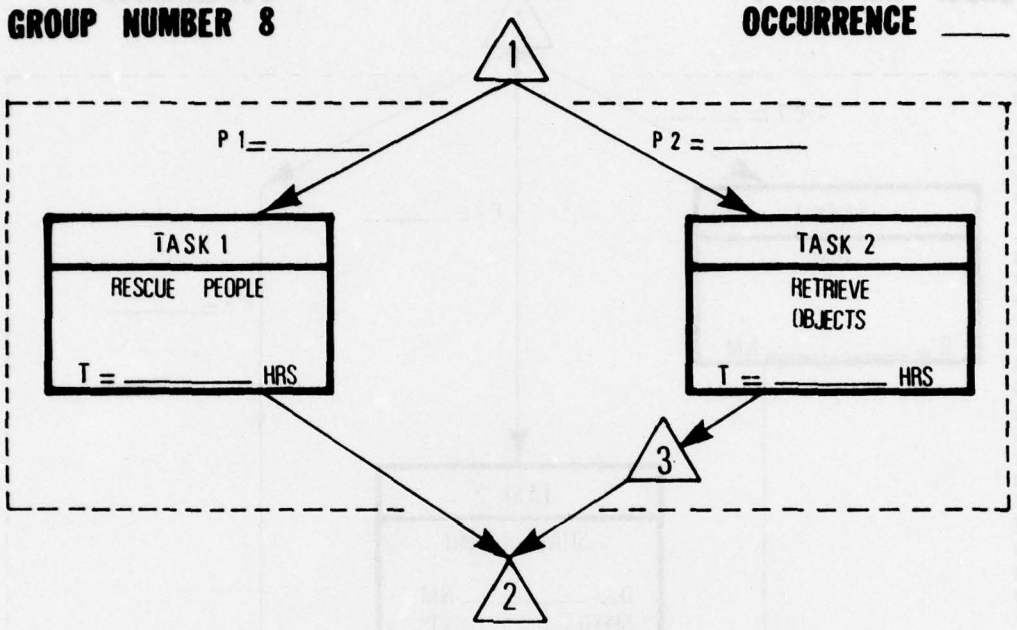
PATROL GROUP

GROUP NUMBER 7 **OCCURRENCE ___**



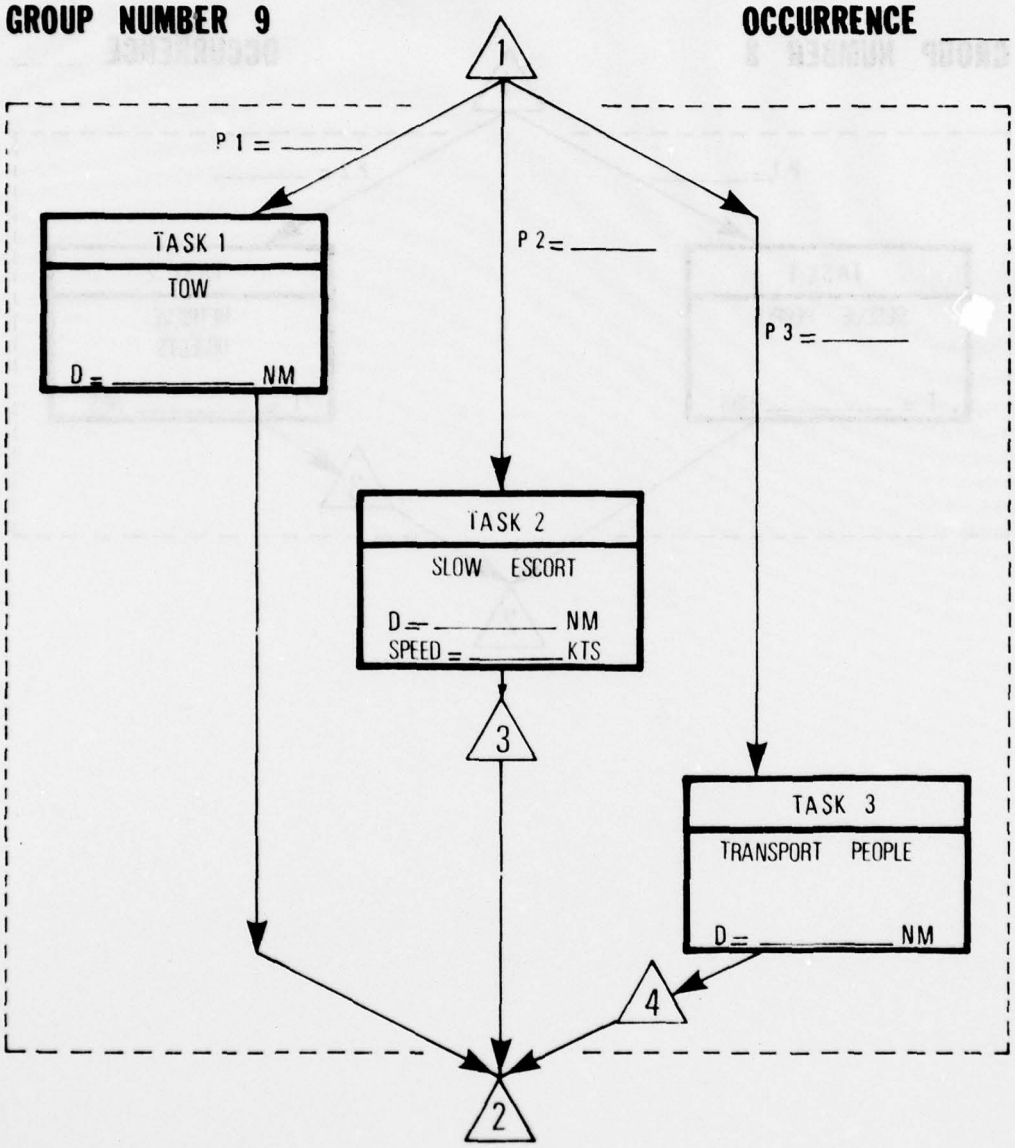
RESCUE GROUP

GROUP NUMBER 8 **OCCURRENCE ____**

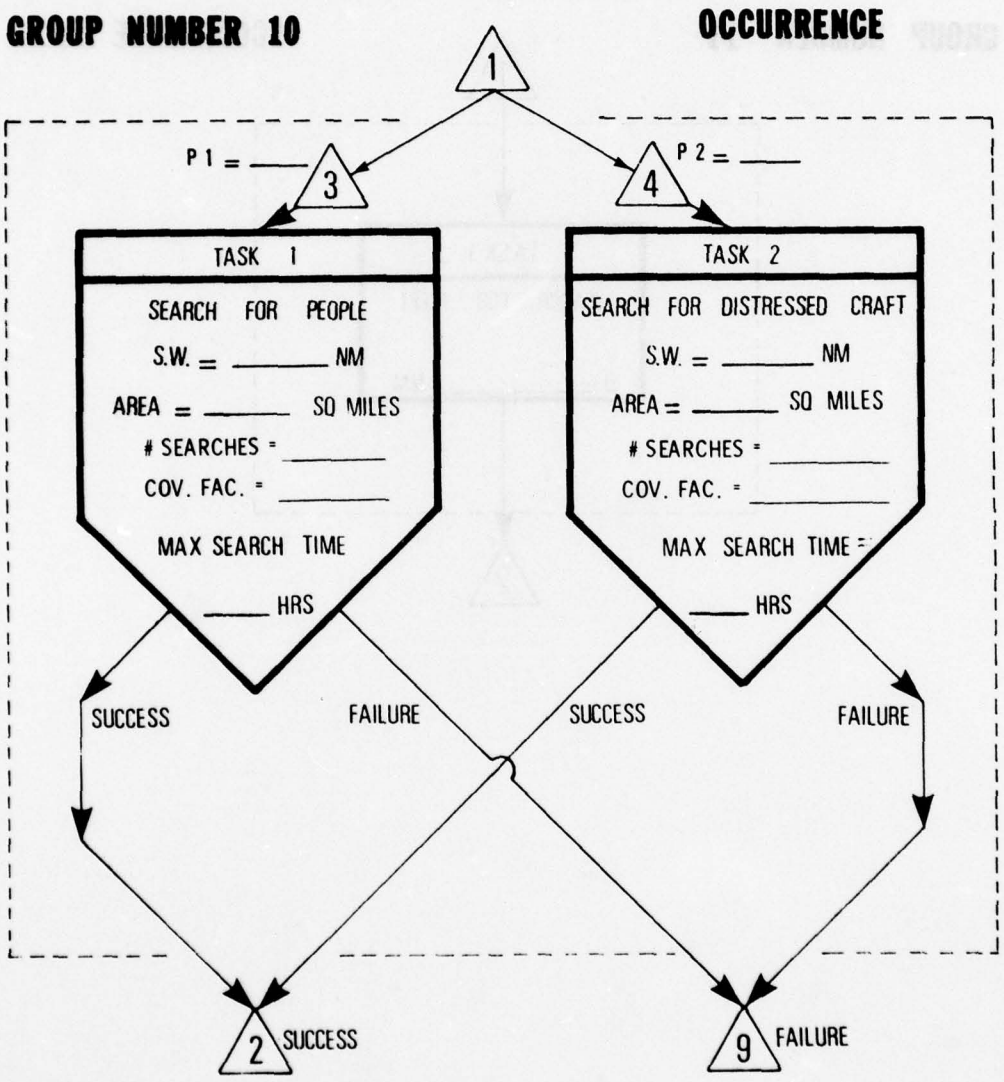


RESCUE RETURN GROUP

GROUP NUMBER 9 **OCCURRENCE ____**

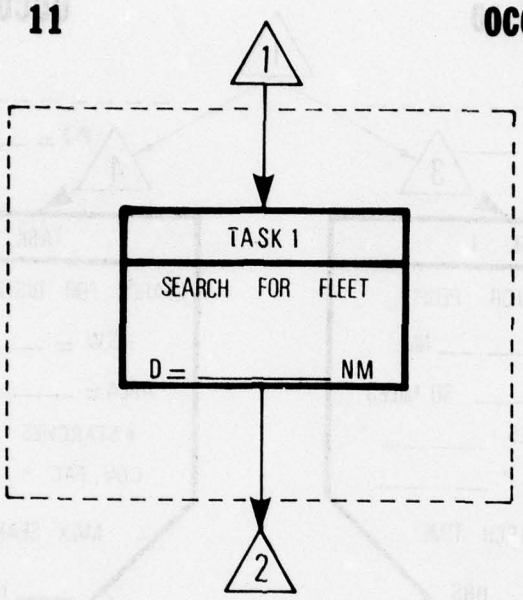


SAR SEARCH GROUP



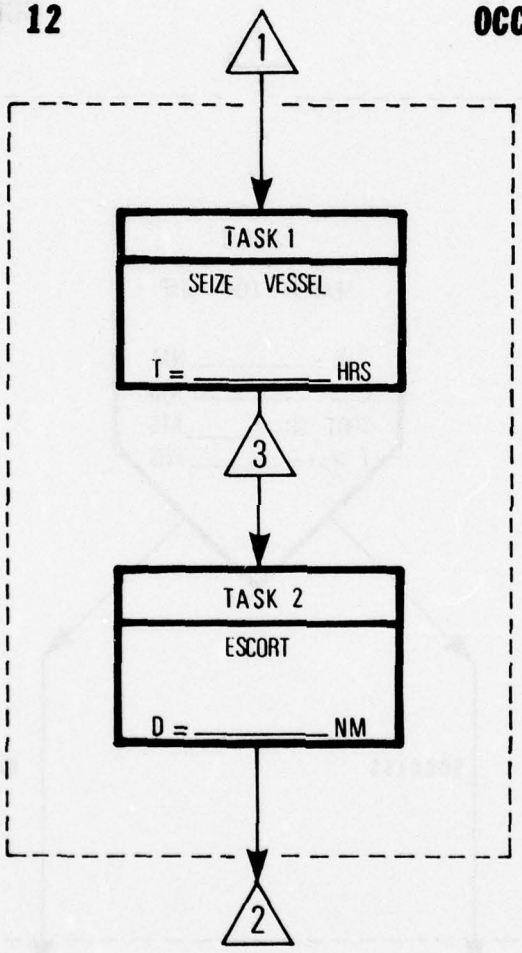
SEARCH FLEET GROUP

GROUP NUMBER 11 **OCCURRENCE ____**



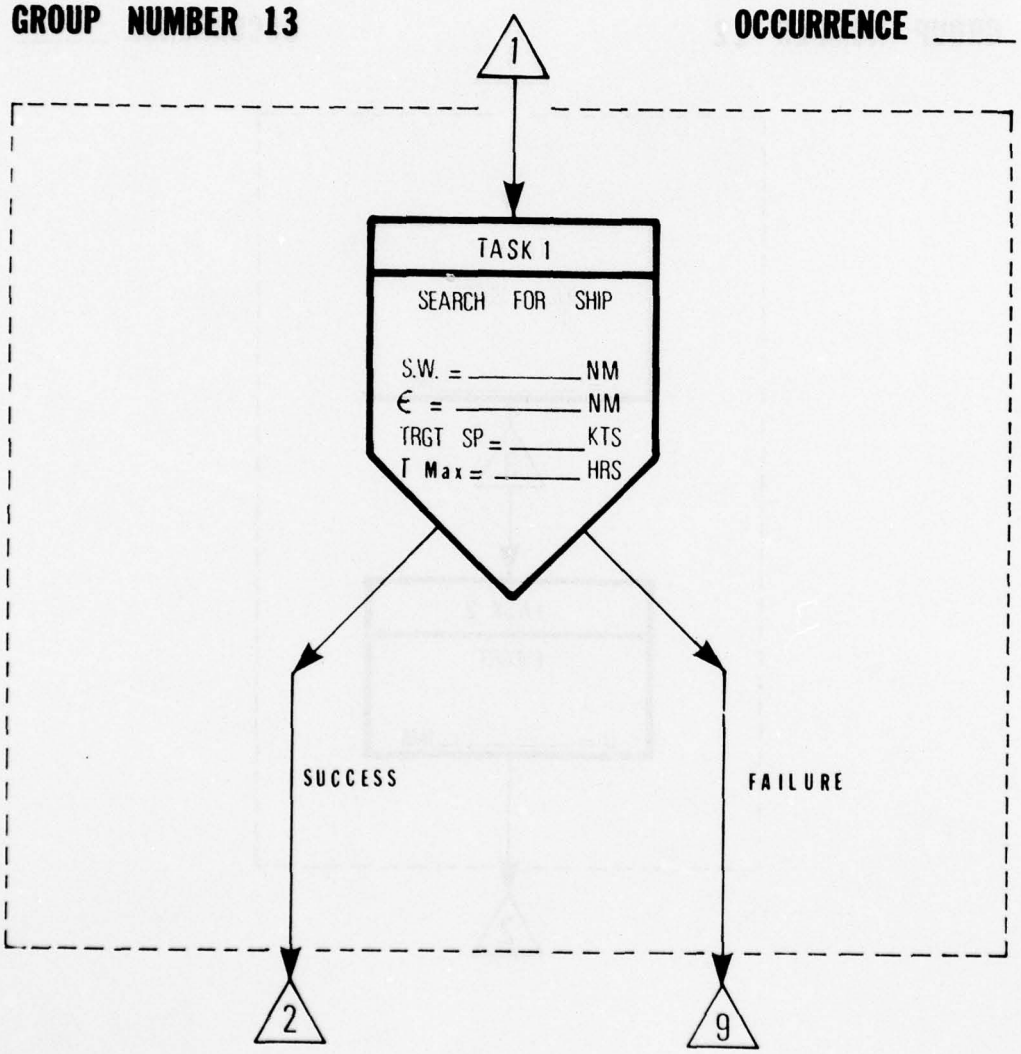
SEIZE GROUP

GROUP NUMBER 12 **OCCURRENCE _____**



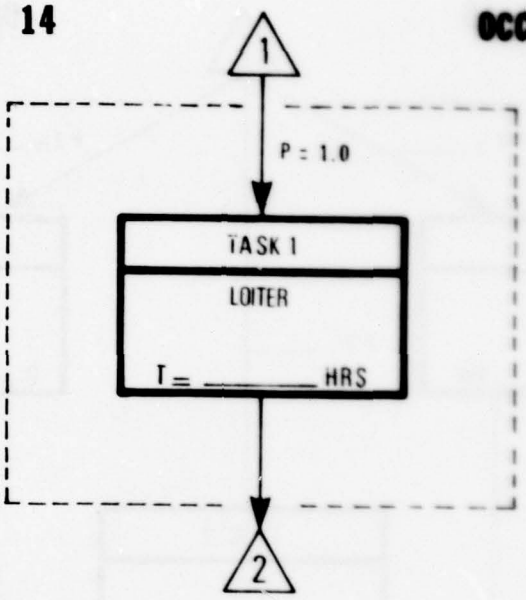
SENSOR SEARCH GROUP

GROUP NUMBER 13 **OCCURRENCE ____**



STANDBY GROUP

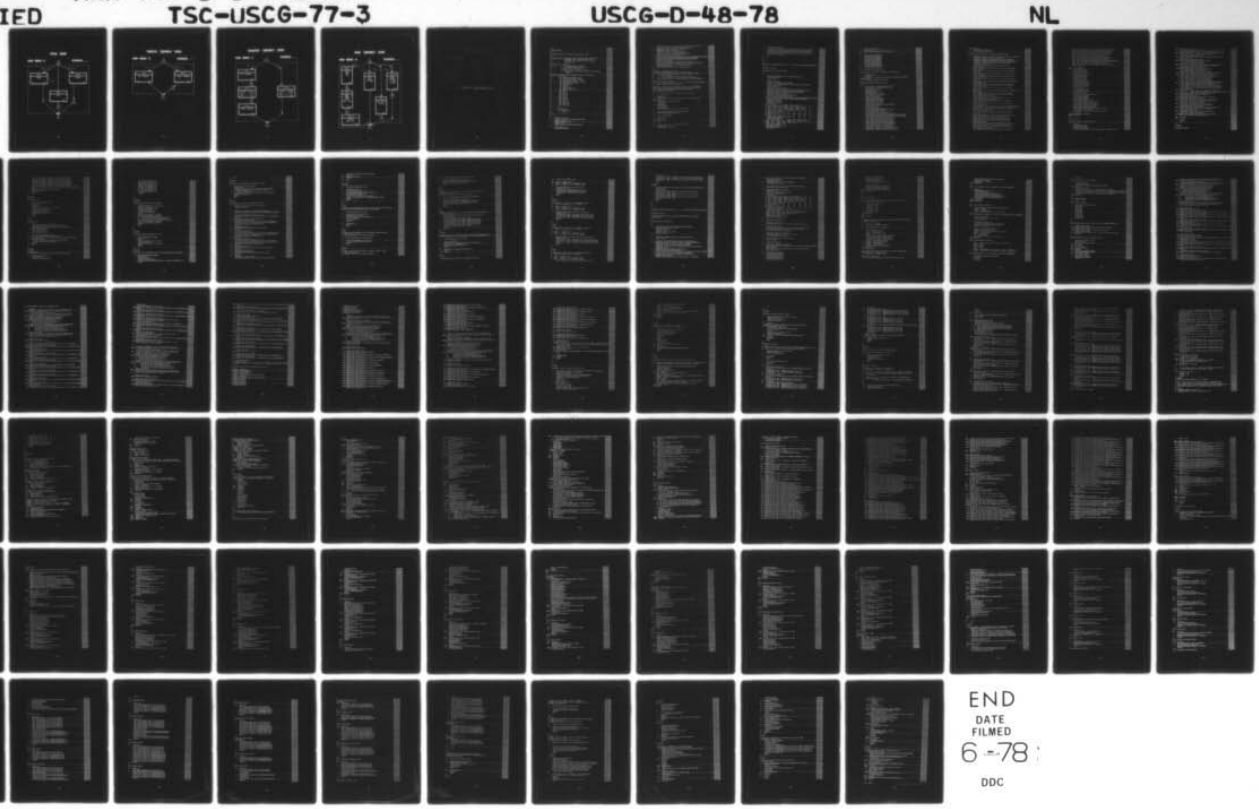
GROUP NUMBER 14 **OCCURRENCE _____**



AD-A053 644 TRANSPORTATION SYSTEMS CENTER CAMBRIDGE MASS F/G 13/10
THE CUTTER RESOURCE EFFECTIVENESS EVALUATION (CREE) PROGRAM -- --ETC(U)
MAR 78 D S PRERAU

UNCLASSIFIED TSC-USCG-77-3 USCG-D-48-78 NL

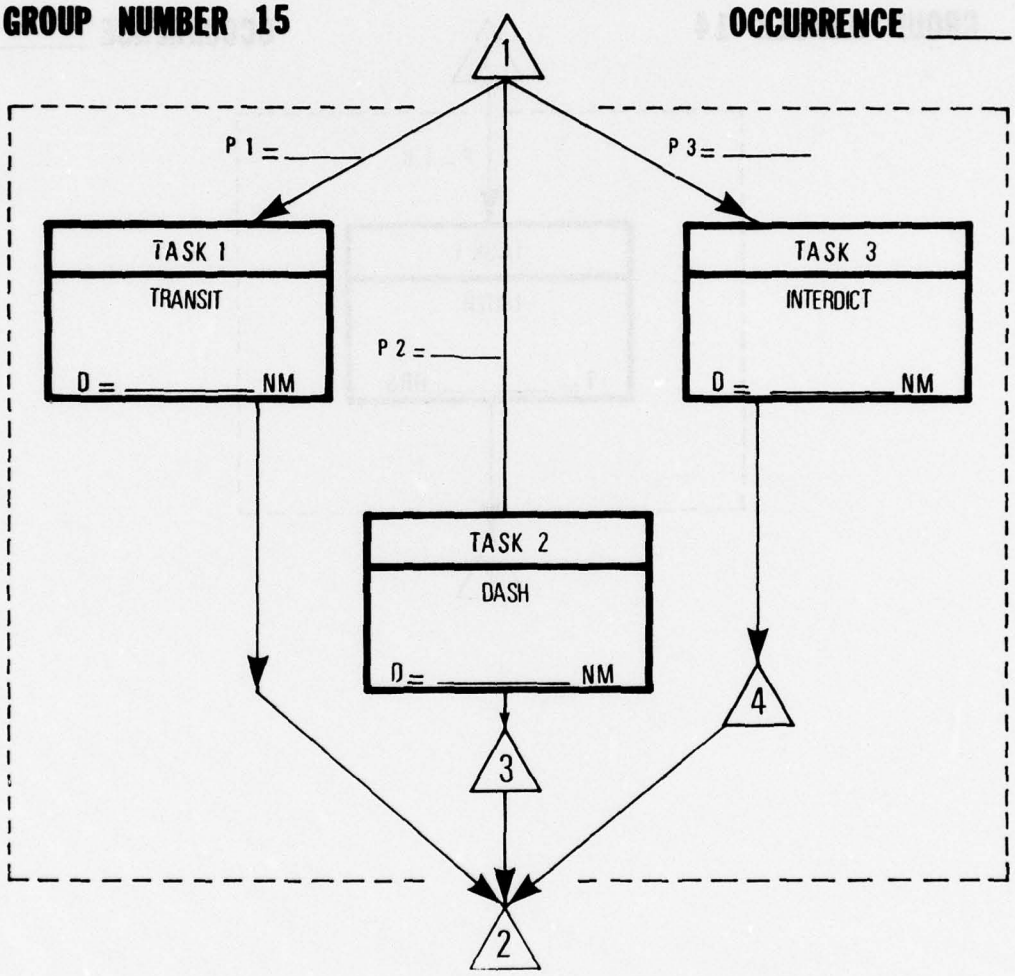
2 OF 2
AD
A053644



END
DATE
FILMED
6-78
DDC

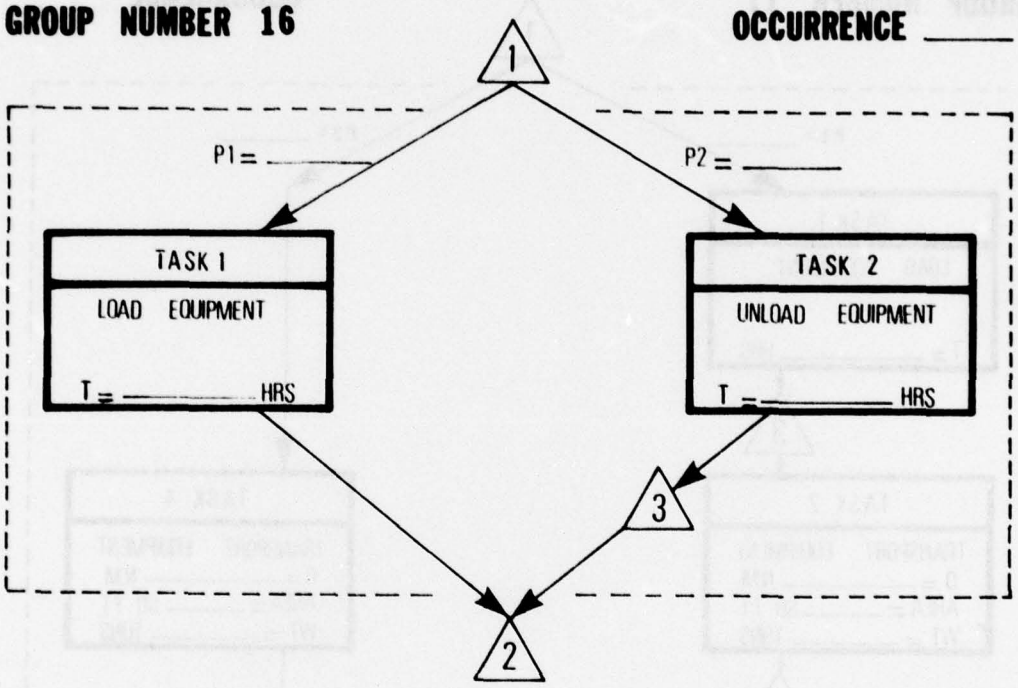
STEAM GROUP

GROUP NUMBER 15 **OCCURRENCE ____**



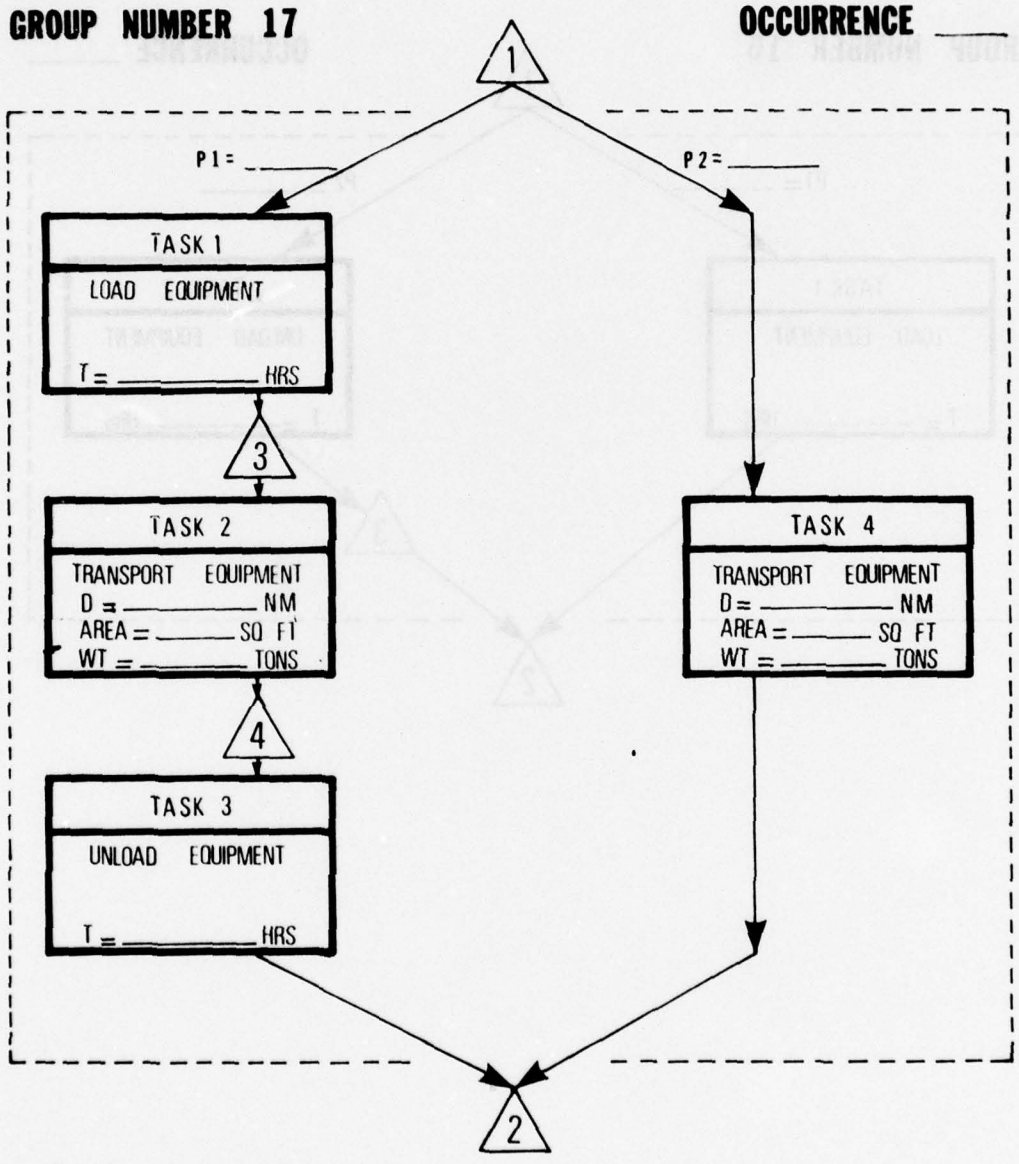
TRANSFER EQUIPMENT GROUP

GROUP NUMBER 16 **OCCURRENCE _____**



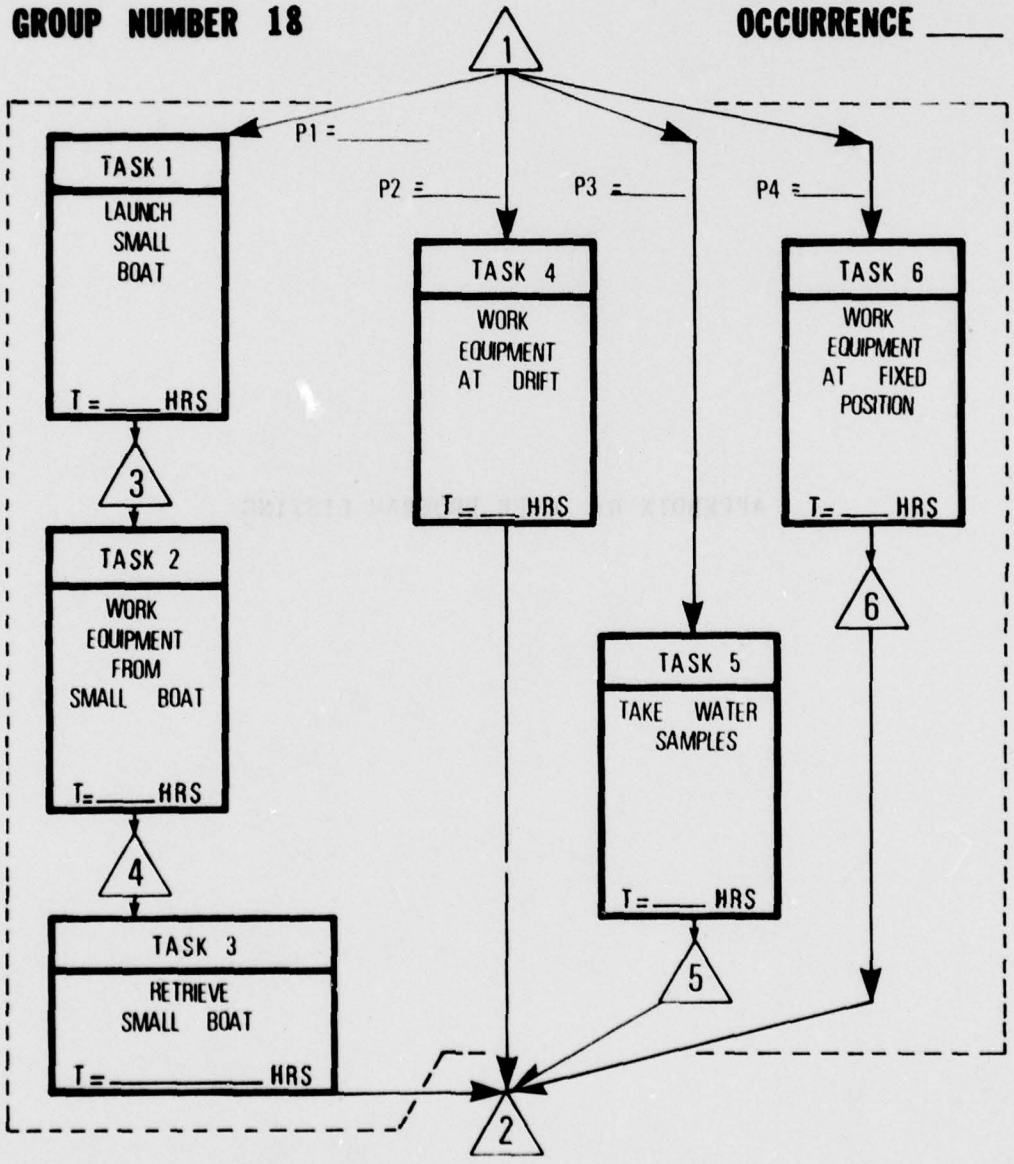
TRANSPORT EQUIPMENT GROUP

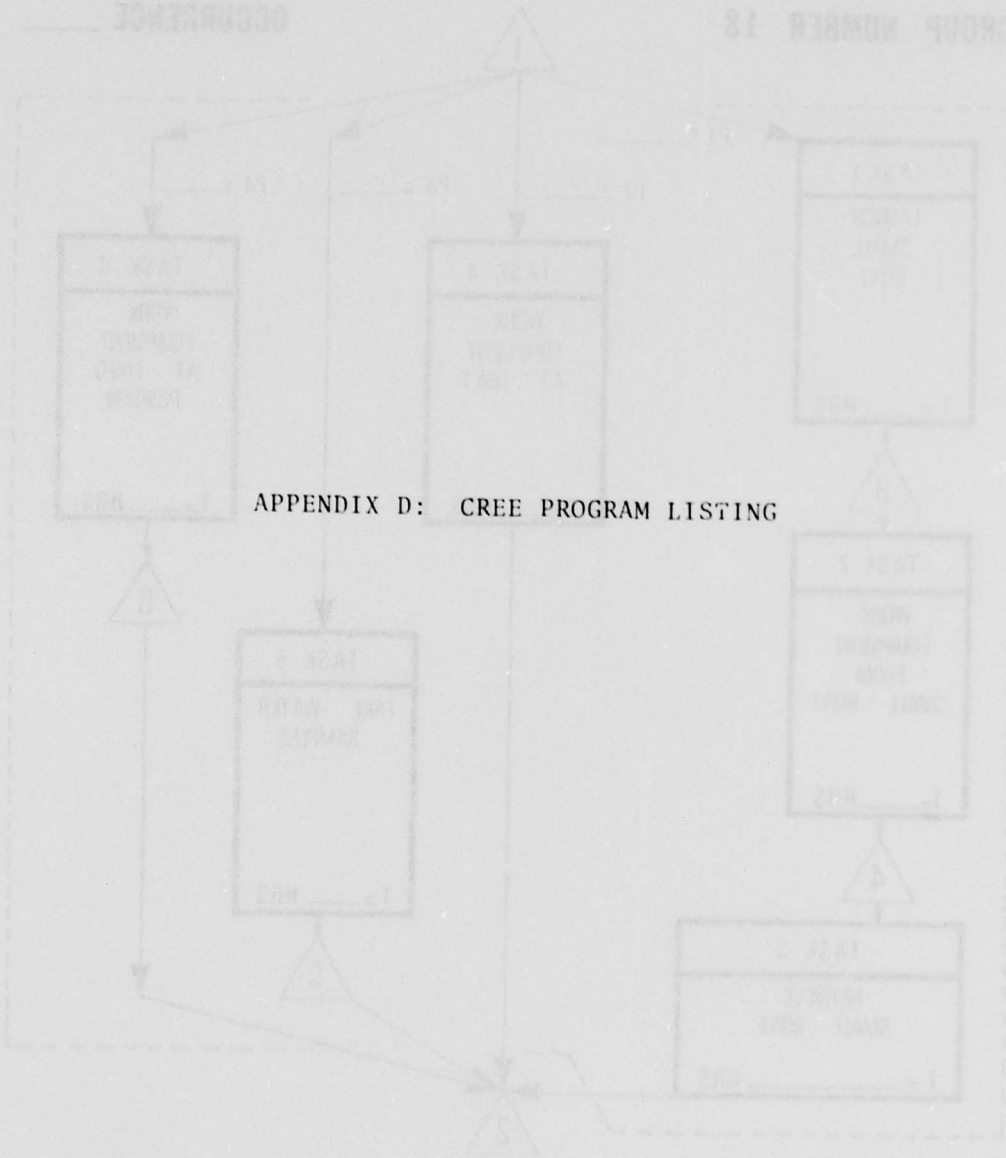
GROUP NUMBER 17 **OCCURRENCE ____**



WORK EQUIPMENT GROUP

GROUP NUMBER 18 **OCCURRENCE ____**





APPENDIX D: CREE PROGRAM LISTING

```
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                                                                       00000390
C        ENGINE CODES:                                                  00000400
C           1. GAS TURBINE                                              00000410
C           2. DIESEL                                                   00000420
C                                                                       00000430
C                                                                       00000440
C                                                                       00000450
C                                                                       00000460
      IMPLICIT REAL(A-Z)                                                00000470
      INTEGER I,J,K,INFILE,CASNUM,IDISP,IDSPD                           00000480
      INTEGER SS1,SS2                                                   00000490
      INTEGER TYPE,TYPE1,ENG                                            00000500
      INTEGER RATE                                                      00000510
      INTEGER VISDTB,TOWDTB,DPHDTB,SSPDTB                               00000520
C                                                                       00000530
      DATA INFILE/14/                                                   00000540
      DIMENSION SSPRBD(8)                                               00000570
```

```
      DIMENSION CWSPD(4),SFCENG(4),SFCCF(4),TOTSFC(4),SFCGAL(4)          00000580
      DIMENSION HPUTIL(4),FUELRT(4),ENDUR(4),RANGE(4)                    00000590
      DIMENSION FUELR2(4),ENG(4),SSPROB(8,10)                            00000600
      DIMENSION TWRAD(4),MOTION(4)                                       00000610
      DIMENSION SPEED(4),CFTNAM(8),MFULRT(4)                             00000620
      DIMENSION CC(19),DF(19),LS(19),MN(19),TW(19)                       00000630
      COMMON/CHAR/LTOB,BEAM,DTOL,DRAF,SSPROB,                            00000640
     1 DECK,USELD,FUELCP,CARGCP,TCWOSP,                                  00000650
     2 SURVIV,HPINST,HPPTON,HPTNKT,CWSPD,ENG,SFCENG,SFCCF,TOTSFC,SFCGAL,00000660
     3 HPUTIL,FUELRT,FUELR2,ENDUR,RANGE,MOTION,TWRAD,SSPDTB              00000670
      COMMON/PARAM/IDISP,IDSPD,CFTNAM,SSAVG,SPEED,MFULRT,                00000680
     1TCWSPD,CC,DF,LS,MN,TW                                              00000690
      COMMON/POS/ASST,BORD,MNAC,RTRV,WAIT,WFGD,WEGP,                     00000700
     1SDIU,SESC,SPED,SPAT,TOWS,ESCT,ICNT,PATI,STGT,TRPT,TRST,RSPD        00000710
      COMMON/MNCOM/LENG,FUFRAC,VISDTB,TOWDTB,DPHDTB                      00000711
      COMMON/CSSPRB/SSPROB                                               00000720
C     DATA SSPROB/ SEE BLOCK DATA STATEMENT BELOW                        00000730
C                                                                        00000740
C                                                                        00000750
C READ READ FIRST LINE OF INPUT                                          00000760
C  FORMAT: 'T=999,D=9999.9,L=9999.9,S=99.9,F=9.99'                       00000770
C       WHERE T=TYPE,D=DISPLACEMENT,L=LENGTH,S=DESIGN SPEED,             00000780
C         F=FUEL FRACTION                                                00000790
C     E.G.  'T= 20,D=0000.0,L= 100.0,S=60.0,F=0.50'                      00000880
C        (EITHER DISPLACEMENT OR LENGTH SHOULD BE 0000.0)                00000890
C     NOTE: FOR EXISTING COAST GUARD CRAFT D,L,S & F ARE IGNORED         00000910
C                                                                        00000920
C READ SECOND LINE OF INPUT                                              00000930
C           VISIBILITY,TCW,DEPTH, AND SEA STATE DISTRIBUTION NUMBERS     00000940
C   FORMAT: 'VS=99,TW=99,DF=99,SS=99'                                    00000950
C                                                                        00000960
1000  READ(INFILE,1001,END=1999)TYPE1,DISP1,LENG1,DSPD1,FUFRC1           00000970
1001  FORMAT(2X,I3,3X,2(F6.1,3X),F4.1,3X,F4.2,3X,I2,2(3X,I1))            00000990
      READ(INFILE,1020)VISDTB,TOWDTB,DPHDTB,SSPDTB                       00001000
 1020 FORMAT(3X,I2,3(4X,I2))                                             00001010
C                                                                        00001020
C     END DISCRETE SEA STATE PROBABILITY DISTRIBUTION                    00001030
C                                                                        00001040
      DO 6999 SS1=2,8                                                    00001050
      SS2=SS1-1                                                          00001060
      SSPROB(SS1)=.5*SSPROB(SS1,SSPDTB)+.5*SSPROB(SS2,SSPDTB)            00001070
6999   CONTINUE                                                          00001080
      SSPROB(1)=0.5*SSPROB(1,SSPDTB)                                     00001090
      TYPE=TYPE1                                                         00001130
      DISP=DISP1                                                         00001140
      LENG=LENG1                                                         00001150
      DSPEED=DSPD1                                                       00001160
      FUFRAC=FUFRC1                                                      00001170
      CALL SCHAR(TYPE,DISP,LENG,DSPEED,FUFRAC)                           00001180
C                                                                        00001190
      CALL SPTPOS(TYPE,DISP,LENG,DSPEED,FUFRAC,                          00001200
     1 VISDTB,TOWDTB,DPHDTB,SSPDTB)                                      00001210
C                                                                        00001220
      CALL SPRPOS(DUMMY)                                                 00001221
C                                                                        00001222
      GO TO 1000                                                         00001223
1999  CONTINUE                                                           00001500
      END                                                                00001510
C                                                                        00001511
C                                                                        00001512
C     BLOCK DATA ROUTINE TO INITIALIZE SSPROB                            00001520
      BLOCK DATA                                                         00001530
      DIMENSION SSPROB(8,10)                                             00001540
```

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

```
     1 'GAS ','TURB','INE ',                                              00000570
     2 'DIES','EL  ','    '/                                              00000580
      DATA TYPLST/10,11,20,21,30,40,50,60,70/                             00000590
C                                                                         00000600
C                                                                         00000610
      IF(TYPE.EQ.10)TYPNUM=1                                              00000620
      IF(TYPE.EQ.11)TYPNUM=2                                              00000630
      IF(TYPE.EQ.20)TYPNUM=3                                              00000640
      IF(TYPE.EQ.21)TYPNUM=4                                              00000650
      IF(TYPE.EQ.30)TYPNUM=5                                              00000660
      IF(TYPE.EQ.40)TYPNUM=6                                              00000670
      IF(TYPE.EQ.50)TYPNUM=7                                              00000680
      IF(TYPE.EQ.60)TYPNUM=8                                              00000690
      IF(TYPE.EQ.70)TYPNUM=9                                              00000700
      IF(TYPE.EQ.80)TYPNUM=10                                             00000710
C                                                                         00000720
      IF(TYPE.GE.100)GOTO 5001                                            00000730
C                                                                         00000740
C FIND DISCRETE SEA STATE PROBABILITY DISTRIBUTION                        00000750
C                                                                         00000760
      DO 6999 SS1=2,8                                                     00000770
      SS2=SS1-1                                                           00000780
      SSPRBD(SS1)=.5*SSPROB(SS1,SSPDTB)+.5*SSPROB(SS2,SSPDTB)             00000790
 6999 CONTINUE                                                            00000800
      SSPRBD(1)=0.5*SSPROB(1,SSPDTB)                                      00000810
C                                                                         00000820
C COMPUTE CHARACTERISTICS FOR HPWC                                        00000830
C                                                                         00000840
C                                                                         00000850
      IF(LENG.EQ.0.) LENG=$LENG(TYPE,DISP)                                00000860
      IF(DISP.EQ.0.) DISP=$DISP(TYPE,LENG)                                00000870
      LTOB=$LTOB(TYPE,LENG)                                               00000880
      BEAM=LENG/LTOB                                                      00000890
      DTOL=$DTOL(TYPE,LENG)                                               00000900
      DRAF=DTOL*LENG                                                      00000910
      DECK=$DECK(TYPE,LENG,BEAM)                                          00000920
      USELD=$USFLD(TYPE,DISP)                                             00000930
      FUELCP=FUFRAC*USELD                                                 00000940
      CARGCP=(1-FUFRAC)*USELD                                             00000950
      TOWDSP=$TOWDS(TYPE,DISP)                                            00000960
      SURVIV=$SURVI(TYPE,LENG)                                            00000970
      ISURVV=SURVIV + 0.5001                                              00000980
      HPBINS=$HPBIN(TYPE,DISP)                                            00000990
      HPINST=(DSPEED/$BSSPD(TYPE))**3*HPBINS                              00001000
      HPPTON=HPINST/DISP                                                  00001010
      DO 2001 RATE=1,4                                                    00001020
      ENG(RATE)=$ENG(TYPE,RATE)                                           00001030
      CWSPD(RATE)=$CWSPD(TYPE,RATE,DSPEED)                                00001040
      FCTDSP=CWSPD(RATE)/DSPEED                                           00001050
      FCTBSP=CWSPD(RATE)/$BSSPD(TYPE)                                     00001060
      HPEFCTU=$HPEFCT(TYPE,RATE,FCTDSP,FCTBSP)                            00001070
      IF(RATE.EQ.1.OR.RATE.EQ.2)HPUTIL(RATE)=HPEFCTU*HPINST               00001080
      IF(RATE.EQ.3.OR.RATE.EQ.4)HPUTIL(RATE)=HPEFCTU*HPBINS               00001090
      SFCENG(RATE)=$SFCEN(ENG(RATE),HPINST)                               00001100
      SFCCF(RATE)=$SFCCF(ENG(RATE),HPEFCTU)                               00001110
      TOTSFC(RATE)=SFCENG(RATE)*SFCCF(RATE)                               00001120
      SFCGAL(RATE)=TOTSFC(RATE)*335./2240.                                00001130
      FUELRT(RATE)=HPUTIL(RATE)*SFCGAL(RATE)                              00001140
      FUELR2(RATE)=FUELRT(RATE)/CWSPD(RATE)                               00001150
      ENDUR(RATE)=FUELCP/(FUELRT(RATE)/335.)                              00001160
      RANGE(RATE)=ENDUR(RATE)*CWSPD(RATE)                                 00001170
      TNRAD(RATE)=(STNRAD(TYPE,CWSPD(RATE)))/3.                           00001180
      MOTION(RATE)=$MWTAV(SSPRBD,TYPE,DISP,RATE)                          00001190
```

```
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
     111500.,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
     11.15,.29,.20,.42/                                                 00001630
      DATA ACWSPD/25.,17.5,12.,5.,25.,18.,12.,5.,26.,18.,12.,5.,        00001640
     119.,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,                     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
     111600.,640.,320.,160.,2324.,1662.,581.,232.,5000.,                00001740
     113000.,2000.,500.,7000.,3000.,1750.,700.,36000.,                  00001750
     117000.,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.,23.4,23.4,23.4,10.,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
```

```
     11.7,1.,.2,2,2.08,2.0(,2.,.(.12,2.12,2.12,2.,.3.5,2.8,2.,.1.,
     13.7,3.,1,3.1,1.,.4.09,3.2,2.5,1.54,6.5,5.6,3.,.1.5,7.5,
     17.2,6.7,9.4,19.5,10.2,6.4,12.5,107.,25.,20.8,30./
      DATA AENDUR/4.,10.,16.6,33.3,4.7,9.7,16.6,100.,9.1,
     116.6,32.5,128.,10.7,13.2,13.2,33.3,45.,45.,45.,
     1105.,11.,16.,35.,169.,16.,25.,25.,186.,20.8,
     136.7,66.6,260.,23.,33.8,83.3,400.,341.,470.,587.,
     11000.,215.,533.,816.,1300.,87.,655.,1048.,1746./
      DATA ARANGE/100.,175.,200.,166.,117.,175.,200.,
     1500.,237.,300.,390.,644.,150.,160.,160.,166.,495.,495.,495.,
     1525.,241.,300.,421.,644.,250.,300.,400.,933.,490.,
     1624.,800.,1300.,460.,540.,1000.,2000.,6266.,
     16580.,7000.,5600.,4200.,8000.,9800.,6500.,2445.,10480.,
     112576.,8733./
C
5001  CGTYPE=TYPE-100
      LENG=ALENG(CGTYPE)
      DISP=ADISP(CGTYPE)
      DSPEED=ADSPEE(CGTYPE)
      FUERAC=AFUERA(CGTYPE)
      LTOB=ALTOB(CGTYPE)
      BEAM=ABEAM(CGTYPE)
      DTOL=ALTOL(CGTYPE)
      DRAF=ADRAF(CGTYPE)
      DECK=ADECK(CGTYPE)
      USELD=AUSFLD(CGTYPE)
      FUELCP=AFUELC(CGTYPE)
      CARGCP=ACARGC(CGTYPE)
      TOWDSP=ATOWLS(CGTYPE)
      ISURVV=ASURVI(CGTYPE)
      SURVIV=ISURVV
      HPINST=AHPINS(CGTYPE)
      HPETON=AHPPTO(CGTYPE)
      HPTURT=AHPTUR(CGTYPE)
      DO 5000 RATE=1,4
      CWSPD(RATE)=ACWSPD(RATE,CGTYPE)
      SFCENG(RATE)=9999999.
      SFCCF(RATE)=9999999.
      TOTSFC(RATE)=9999999.
      SFCGAL(RATE)=9999999.
      ENG(RATE)=AENG(RATE,CGTYPE)
       HPUTIL(RATE)=AHPUTI(RATE,CGTYPE)
      FUELRT(RATE)=AFUERT(RATE,CGTYPE)
      FUELR2(RATE)=AFUER2(RATE,CGTYPE)
      ENDUR(RATE)=AENDUR(RATE,CGTYPE)
      RANGE(RATE)=ARANGE(RATE,CGTYPE)
      MOTION(RATE) = $MWTAV(SSPRBD,TYPE,DISP,RATE)
      IF(RATE.NE.2) GO TO 5002
      TNRAD(RATE) = ($TNRAD(TYPE,CWSPD(RATE)))/3.
      GO TO 5000
 5002 TNRAD(RATE) = 9999999.
5000  CONTINUE
C
C
 1008 CONTINUE
C
C OUTPUT FORMATTED CHARACTERISTICS
C
      OUTFIL=6
      IDISP=DISP+.500001
      ILENG=LENG+.500001
      IDSPD=DSPEED+.500001
      IF(TYPE.LT.100)WRITE(OUTFIL,3000)(CHFNM(I,TYPNUM),I=1,8)
```

```
3000  FORMAT('1'/15X,'C R A F T  C H A R A C T E R I S T I C S'
     1,3(/),18X,'CRAFT TYPE',9X,8A4)
      IF(TYPE.GE.100)WRITE(OUTFIL,3025)(CWCRNM(I,CGTYPE),I=1,2)
3025  FORMAT('1'/15X,'C R A F T  C H A R A C T E R I S T I C S'
     1 ,3(/),18X,'CRAFT TYPE',9X,'COAST GUARD ',2A4)
      WRITE(OUTFIL,3026)IDISP,ILENG
3026   FORMAT(18X,'DISPLACEMENT',4X,I6,1X,'TONS'/18X,'LENGTH',
     110X,I6,1X,'FEET')
      WRITE(OUTFIL,3022)IUSPD,FUFRAC
3022  FORMA (18X,'DESIGN SPEED',4X,I6,' KNOTS'/
     1    18X,'FUEL FRACTION',3X,F7.2)
      WRITE(OUTFIL,3001)LENG,BEAM,DRAF
3001  FORMAT(4(/),13X,'LENGTH',22X,F8.1,2X,'FEET'/13X,'BEAM',
     1 24X,F8.1,2X,'FEET'/13X,'DRAFT',23X,F8.1,2X,'FEET')
      WRITE(OUTFIL,3002)LTOB,DTOL,DISP
 3002 FORMAT(13X,'LENGTH/BEAM RATIO',12X,F8.2
     1/13X,'DRAFT/LENGTH RATIO',11X,F8.2
     4 /13X,'DISPLACEMENT',16X,F8.1,2X,'TONS')
      WRITE(OUTFIL,3003)ISURVV,TOWDSP,DECK,CARGCP,FUELCP,USFUL
3003  FORMAT (13X,'SURVIVABILITY',14X,I7,4X,'SEA STATE'
     1 /13X,'TOWS VESSELS UP TO',10X,F7.0,3X,'TONS'
     2 /13X,'USEABLE DECK AREA',11X,F7.0,3X,'SQUARE FEET'
     4 /13X,'CARGO CAPACITY',14X,F8.1,2X,'TONS'
     5 /13X,'FUEL CAPACITY',15X,F8.1,2X,'TONS'
     3 /13X,'USEFUL PAYLOAD',14X,F8.1,2X,'TONS')
      WRITE (OUTFIL,3004)HPTNST,HPPTON,HPTNKT,RANGE(2),ENDUR(2)
3004  FORMAT(13X,'INSTALLED POWER',13X,F7.0,3X,'HORSEPOWER'
     1 /13X,'POWER TO WEIGHT',13X,F8.1,2X,'HP/TON'
     2 /13X,'TRANSPORT EFFICIENCY',9X,F8.2,1X,'HP/TON-KNOT'
     4 /13X,'RANGE AT CRUISE SPEED',7X,F7.0,3X,'NAUTICAL MILES'
     5 /13X,'ENDURANCE AT CRUISE SPEED',3X,F8.1,2X,'HOURS')
      WRITE(OUTFIL,3010)
3010  FORMAT(5(/),27X,'  FLANK  ',1X,'CRUISE ',1X,'REDUCED',
     1 1X,'  ON    '/29X,'SPEED',3X,'SPEED',3X,' SPEED ',1X,' SCENE
      WRITE(OUTFIL,3023)(ENGNAM(ENG(RATE)),RATE=1,4)
3023  FORMAT(10X,'ENGINE TYPE',5X,4(4X,A4)/)
      WRITE(OUTFIL,3011)CWSPD
 3011 FORMAT(10X,'CALM WATER SPEED',4F8.1,4X,'KNOTS')
      WRITE(OUTFIL,3012)TOTSFC,SFCGAL
3012  FORMAT(10X,'SFC (WEIGHT) ',4X,4F8.2,3X,'LBS/HP-HR'
     1 /10X,'SFC (VOLUME) ',4X,4F8.2,3X,'GAL/HP-HR')
      WRITE(OUTFIL,3013)HPUTIL,FUELRT
3013  FORMAT(10X,'HP UTILIZED',5X,4F8.1,4X,'HP'
     1 /10X,'FUEL CONSUMPTION',4F8.1,4X,'GAL/HR')
      WRITE(OUTFIL,3014)FUELR2,ENDUR
3014  FORMAT(10X,'FUEL CONSUMPTION',4F8.1,4X,'GAL/NAUT MI'
     1 /10X,'ENDURANCE (FUEL)',4F8.1,4X,'HOURS')
      WRITE(OUTFIL,3015)RANGE
3015  FORMAT(10X,'RANGE',11X,4F8.1,4X,'NAUTICAL MI')
      WRITE(OUTFIL,3031) TNRAD,MOTION
 3031 FORMAT(10X,'TURNING RADIUS',2X,4F8.1,4X,'YARDS'
     1/10X,'CRAFT MOTION',4X,4F8.1,4X,'G')
C
2021  CONTINUE
C
      RETURN
      END
C
C
C $LENG
C
C LENGTH(IN FEET)
C
```

```
      FUNCTION $LENG(TYPE,DISP)                                          00003610
      INTEGER TYPE                                                       00003620
      IF(TYPE.EQ.10)$LENG=.418*DISP+45.                                  00003630
      IF(TYPE.EQ.11)$LENG=.414*DISP+60.                                  00003640
      IF(TYPE.EQ.20)$LENG=$$(DISP,15.,65.,200.,135.)                     00003650
      IF(TYPE.EQ.21)$LENG=$$(DISP,15.,50.,175.,99.)                      00003660
      IF(TYPE.EQ.30)$LENG=$$(DISP,90.,100.,180.,126.)                    00003670
      IF(TYPE.EQ.40)$LENG=.289*DISP+78.3                                 00003680
      IF(TYPE.EQ.50.OR.TYPE.EQ.70)$LENG=$$(DISP,10.,40.,185.,155.)       00003690
      IF(TYPE.EQ.60)$LENG=$$(DISP,150.,85.,1000.,200.,2000.,250.,        00003700
     13500.,300.)                                                        00003710
      IF(TYPE.EQ.80.AND.DISP.LE.1000)$LENG=110.*ALOG10(DISP)-120.        00003720
      IF(TYPE.EQ.80.AND.DISP.GT.1000)$LENG=356.4*ALOG10(DISP)-859.2      00003730
      RETURN                                                             00003740
      END                                                                00003750
C                                                                        00003760
C                                                                        00003770
C $DISP                                                                  00003780
C                                                                        00003790
C DISPLACEMENT(IN TONS)                                                  00003800
C                                                                        00003810
      FUNCTION $DISP(TYPE,LENG)                                          00003820
      REAL LENG                                                          00003830
      INTEGER TYPE                                                       00003840
      IF(TYPE.EQ.10)$DISP=(LENG-45.)/.418                                00003850
      IF(TYPE.EQ.11)$DISP=(LENG-60.)/.414                                00003860
      IF(TYPE.EQ.20)$DISP=$$(LENG,65.,15.,135.,200.)                     00003870
      IF(TYPE.EQ.21)$DISP=$$(LENG,50.,15.,99.,175.)                      00003880
      IF(TYPE.EQ.30)$DISP=$$(LENG,100.,90.,126.,180.)                    00003890
      IF(TYPE.EQ.40)$DISP=(LENG-70.)/.314                                00003900
      IF(TYPE.EQ.50.OR.TYPE.EQ.70)$DISP=$$(LENG,40.,10.,155.,185.)       00003910
      IF(TYPE.EQ.60)$DISP=$$4(LENG,85.,150.,200.,1000.,250.,2000.,300.,  00003920
     13500.)                                                             00003930
      IF(TYPE.EQ.80.AND.LENG.LE.210.)$DISP=10**((LENG+120.)/110)         00003940
      IF(TYPE.EQ.80.AND.LENG.GT.210.)$DISP=10**((LENG+859.2)/356.4)      00003950
      RETURN                                                             00003960
      END                                                                00003970
C                                                                        00003980
C                                                                        00003990
C $LTOB                                                                  00004000
C                                                                        00004010
C LENGTH TO BEAM RATIO                                                   00004020
C                                                                        00004030
      FUNCTION $LTOB(TYPE,LENG)                                          00004040
      REAL LENG                                                          00004050
      INTEGER TYPE                                                       00004060
      IF(TYPE.EQ.10)$LTOB=4.0                                            00004070
      IF(TYPE.EQ.11)$LTOB=4.5                                            00004080
      IF(TYPE.EQ.20.OR.TYPE.EQ.21)$LTOB=2.                               00004090
      IF(TYPE.EQ.30)$LTOB=3.                                             00004100
      IF(TYPE.EQ.40)$LTOB=5.5                                            00004110
      IF(TYPE.EQ.50)$LTOB=2.5                                            00004120
      IF(TYPE.EQ.60)$LTOB=3.0                                            00004130
      IF(TYPE.EQ.70)$LTOB=3.                                             00004140
      IF(TYPE.EQ.80)$LTOB=5.                                             00004150
      RETURN                                                             00004160
      END                                                                00004170
C                                                                        00004180
C                                                                        00004190
C $DTOL                                                                  00004200
C                                                                        00004210
C DRAFT TO LENGTH RATIO                                                  00004220
C                                                                        00004230
```

```
      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)$HPBI=$$(DISP,20.,1000.,250.,14000.)                00004870
      IF(TYPE.EQ.20)$HPBI=$$(DISP,8.,1570.,80.,4606.)                   00004871
      IF(TYPE.EQ.21)$HPBI=$$(DISP,25.,2500.,150.,17000.)                00004890
      IF(TYPE.EQ.30)$HPBI=$$(DISP,0.,4000.,180.,10000.)                 00004900
      IF(TYPE.EQ.40)$HPBI=$$(DISP,35.,3000.,155.,14500.)                00004910
      IF(TYPE.EQ.50)$HPBI=$$(DISP,69.,3400.,74.,3600.)                  00004920
      IF(TYPE.EQ.70)$HPBI=$$(DISP,25.,762.,200.,6845.)                  00004930
      IF(TYPE.EQ.60)$HPBI=$$$(DISP,200.,2000.,1000.,6800.,4000.,        00004940
     112800.)                                                           00004950
      IF(TYPE.EQ.80)$HPBI=10**(.659*ALOG10                              00004960
     1(DISP)+2.2648)                                                    00004970
      RETURN                                                            00004980
      END                                                               00004990
C                                                                       00005000
C                                                                       00005010
C $SURVI                                                                00005020
C                                                                       00005030
C SURVIVABILITY                                                         00005040
C                                                                       00005050
      FUNCTION $SURVI(TYPE,LENG)                                        00005060
      INTEGER TYPE                                                      00005070
      REAL LENG                                                         00005080
      SSUR=3.0                                                          00005090
      IF(TYPE.EQ.20.OR.TYPE.EQ.21)SSUR=2.0                              00005100
      IF(TYPE.EQ.50)SSUR=3.5                                            00005110
      IF(TYPE.EQ.60)SSUR=4.0                                            00005120
      $SURVI=.02*LENG+SSUR                                              00005130
      IF($SURVI.GT.7.)$SURVI=7.                                         00005140
      RETURN                                                            00005150
      END                                                               00005160
C                                                                       00005170
C                                                                       00005180
C $ENG                                                                  00005190
C                                                                       00005200
C ENGINE TYPE                                                           00005210
C                                                                       00005220
      INTEGER FUNCTION $ENG(TYPE,RATE)                                  00005230
      INTEGER TYPE,RATE,ALLDSL,ALLGT,GT3DSL,GT1DSL                      00005240
      DIMENSION ALLDSL(4),ALLGT(4),GT3DSL(4),GT1DSL(4)                  00005250
      DATA ALLDSL/2,2,2,2/                                              00005260
      DATA ALLGT/1,1,1,1/                                               00005270
      DATA GT3DSL/1,1,1,2/                                              00005280
      DATA GT1DSL/1,2,2,2/                                              00005290
C                                                                       00005300
      IF(TYPE.EQ.11.OR.TYPE.EQ.50                                       00005310
     1   .OR.TYPE.EQ.70)$ENG=ALLDSL(RATE)                               00005320
      IF(TYPE.EQ.20.OR.TYPE.EQ.21.OR.TYPE.EQ.60.OR.TYPE.EQ.30)          00005330
     1$ENG=ALLGT(RATE)                                                  00005340
      IF (TYPE.EQ.10.OR.TYPE.EQ.40)$ENG=GT3DSL(RATE)                    00005350
      IF(TYPE.EQ.80)$ENG=GT1DSL(RATE)                                   00005360
      RETURN                                                            00005370
      END                                                               00005380
C                                                                       00005390
C                                                                       00005400
C                                                                       00005410
C $BSSPD                                                                00005420
C                                                                       00005430
C BASE SPEED                                                            00005440
C  (USED FOR BASE CURVES FOR HPINST,ACCEL,BRAKG,& TURN)                 00005450
C                                                                       00005460
      FUNCTION $BSSPD(TYPE)                                             00005470
      INTEGER TYPE                                                      00005480
      IF(TYPE.EQ.10)$BSSPD=50.                                          00005490
```

```
      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
```

```
      END                                                              00006130
C                                                                      00006140
C                                                                      00006150
C $SFCCF                                                               00006160
C                                                                      00006170
C SPECIFIC FUEL CONSUMPTION CORRECTION FACTOR                          00006180
C                                                                      00006190
      FUNCTION $SFCCF(ENG,HPFCTU)                                      00006200
      INTEGER ENG                                                      00006210
      IF(ENG.EQ.1.AND.HPFCTU.GT..5)$SFCCF=-.4*HPFCTU+1.4               00006220
      IF(ENG.EQ.1.AND.HPFCTU.GT..25.AND.HPFCTU.LE..5)                  00006230
     1    $SFCCF=-1.6*HPFCTU+2.                                        00006240
      IF(ENG.EQ.1.AND.HPFCTU.LE..25)$SFCCF=-3.2*HPFCTU+2.4             00006250
      IF(ENG.EQ.2)$SFCCF=1.                                            00006260
      RETURN                                                           00006270
      END                                                              00006280
C                                                                      00006290
C                                                                      00006300
C $HPFCT                                                               00006310
C                                                                      00006320
C FRACTION OF INSTALLED HORSEPOWER UTILIZED                            00006330
C                                                                      00006340
      FUNCTION $HPFCT(TYPE,RATE,PCTDSP,PCTBSP)                         00006350
       INTEGER TYPE,RATE                                               00006360
C                                                                      00006370
C                                                                      00006380
      IF(RATE.EQ.1.OR.RATE.EQ.2)PCTDSP=100.*PCTDSP                     00006390
      IF(RATE.EQ.3.OR.RATE.EQ.4.OR.RATE.EQ.0)PCTDSP=100.*PCTBSP        00006400
C                                                                      00006410
      IF(TYPE.NE.10.AND.TYPE.NE.11)GO TO 20                            00006420
      HPPCT=$$5(PCTDSP,0.,.5,.20,.14,.40,.48,.85,.75,.100,.100.)       00006430
      GOTO 99                                                          00006440
20    IF(TYPE.NE.20.AND.TYPE.NE.21)GO TO 30                            00006450
      IF(PCTDSP.LE.10.)HPPCT=10.                                       00006460
      IF(PCTDSP.LE.30..AND.PCTDSP.GT.10.)HPPCT=2.*PCTDSP-10.           00006470
      IF(PCTDSP.LE.80..AND.PCTDSP.GT.30.)HPPCT=.20*PCTDSP+44.          00006480
      IF(PCTDSP.GT.80.)HPPCT=2.*PCTDSP-100.                            00006490
      GOTO 99                                                          00006500
30    IF(TYPE.NE.30)GO TO 40                                           00006510
      HPPCT=$$3(PCTDSP,0.,10.,80.,60.,100.,100.)                       00006520
      GOTO 99                                                          00006530
40    IF(TYPE.NE.40)GO TO 50                                           00006540
      IF(PCTDSP.LE.20)HPPCT=5.                                         00006550
      IF(PCTDSP.LE.40..AND.PCTDSP.GT.20.)HPPCT=2.75*PCTDSP-50.         00006560
      IF(PCTDSP.LE.80..AND.PCTDSP.GT.40.)HPPCT=.25*PCTDSP+50.          00006570
      IF(PCTDSP.GT.80)HPPCT=1.5*PCTDSP-50.                             00006580
      GOTO 99                                                          00006590
50    IF(TYPE.NE.50)GO TO 60                                           00006600
      IF(PCTDSP.LE.10.)HPPCT=.50*PCTDSP+5.                             00006610
      IF(PCTDSP.LE.30..AND.PCTDSP.GT.10.)HPPCT=2.5*PCTDSP-15.          00006620
      IF(PCTDSP.LE.80..AND.PCTDSP.GT.30.)HPPCT=.2*PCTDSP+54.           00006630
      IF(PCTDSP.GT.80.)HPPCT=1.5*PCTDSP-50.                            00006631
      GOTO 99                                                          00006640
60    IF(TYPE.NE.60)GO TO 70                                           00006650
      HPPCT=$$4(PCTDSP,0.,.5,.30.,.12.,.70.,.45.,.100.,.100.)          00006660
      GOTO 99                                                          00006670
70    IF(TYPE.NE.70) GO TO 80                                          00006680
      IF(PCTDSP.LE.10.)HPPCT=5.                                        00006690
      IF(PCTDSP.GT.10.)HPPCT=1.05556*PCTDSP-5.55556                    00006700
      GOTO 99                                                          00006710
80    CONTINUE                                                         00006720
      IF(PCTDSP.LE.20.)HPPCT=10.                                       00006730
      IF(PCTDSP.LE.60.AND.PCTDSP.GT.20)HPPCT=.25*PCTDSP+5.             00006740
```

```
      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
```

```
      FUNCTION $$4(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4)                          00007380
C                                                                      00007390
      IF(X.LE.X2)$$4=$$(X,X1,Y1,X2,Y2)                                 00007400
      IF(X.GT.X2.AND.X.LE.X3)$$4=$$(X,X2,Y2,X3,Y3)                     00007410
      IF(X.GT.X3)$$4=$$(X,X3,Y3,X4,Y4)                                 00007420
      RETURN                                                           00007430
      END                                                              00007440
C                                                                      00007450
C                                                                      00007460
C $$5                                                                  00007470
C                                                                      00007480
C FINDS Y VALUE ON BROKEN LINE OF 5 POINTS, GIVEN X VALUE              00007490
C   AND THE 5 POINTS                                                   00007500
C (ASSUMING ENDS OF LINE EXTEND INFINITELY)                            00007510
C                                                                      00007520
      FUNCTION $$5(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5)                    00007530
C                                                                      00007540
      IF(X.LE.X2)$$5=$$(X,X1,Y1,X2,Y2)                                 00007550
      IF(X.GT.X2.AND.X.LE.X3)$$5=$$(X,X2,Y2,X3,Y3)                     00007560
      IF(X.GT.X3.AND.X.LE.X4)$$5=$$(X,X3,Y3,X4,Y4)                     00007570
      IF(X.GT.X4)$$5=$$(X,X4,Y4,X5,Y5)                                 00007580
      RETURN                                                           00007590
      END                                                              00007600
C                                                                      00007610
C                                                                      00007620
C                                                                      00007630
C $$8                                                                  00007640
C FINDS Y VALUE ON BROKEN LINE OF 8 POINTS, GIVEN X VALUES AND         00007650
C   THE EIGHT POINTS                                                   00007660
      FUNCTION $$8(X,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5,X6,Y6,             00007670
     1X7,Y7,X8,Y8)                                                     00007680
      IF(X.LE.X2) $$8 = $$(X,X1,Y1,X2,Y2)                              00007690
      IF(X.GT.X2.AND.X.LE.X3) $$8 = $$(X,X2,Y2,X3,Y3)                  00007700
      IF(X.GT.X3.AND.X.LE.X4) $$8 = $$(X,X3,Y3,X4,Y4)                  00007710
      IF(X.GT.X4.AND.X.LE.X5) $$8 = $$(X,X4,Y4,X5,Y5)                  00007720
      IF(X.GT.X5.AND.X.LE.X6) $$8 = $$(X,X5,Y5,X6,Y6)                  00007730
      IF(X.GT.X6.AND.X.LE.X7) $$8 = $$(X,X6,Y6,X7,Y7)                  00007740
      IF(X.GT.X7) $$8 = $$(X,X7,Y7,X8,Y8)                              00007750
      RETURN                                                           00007760
      END                                                              00007770
C                                                                      00007780
C                                                                      00007781
C $$LGLG                                                               00007782
C                                                                      00007783
C FINDS Y VALUE ON A STRAIGHT LINE ON LOG-LOG PAPER,GIVEN X VALUE      00007784
C AND 2 POINTS ON THE LINE (ASSUMING LINE EXTENDS INFINITELY)          00007785
C                                                                      00007786
      FUNCTION $$LGLG(X,X1,Y1,X2,Y2)                                   00007790
C                                                                      00007800
      IF(X1.EQ.X2)GOTO 1                                               00007810
      SLOPE=(ALOG(Y2)-ALOG(Y1))/(ALOG(X2)-ALOG(X1))                    00007820
      B= ALOG(Y1) - SLOPE*ALOG(X1)                                     00007830
      $$LGLG= EXP(SLOPE*ALOG(X) + B)                                   00007840
      RETURN                                                           00007850
C                                                                      00007860
1     $$LGLG= EXP( (ALOG(Y1)+ALOG(Y2))/2. )                            00007870
      RETURN                                                           00007880
      END                                                              00007890
C                                                                      00007900
C                                                                      00007910
C $$$                                                                  00007920
C                                                                      00007930
C  READ CURVE Y VS X,OR X VS Y DEPENDING UPON FLAG                     00007940
```

```
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
```

```
      INTEGER FLAG                                                      00008570
      IF(FLAG.EQ.0) $$$5 = $$5(XORY,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5)      00008580
      IF(FLAG.EQ.1) $$$5 = $$5(XORY,Y5,X5,Y4,X4,Y3,X3,Y2,X2,Y1,X1)      00008590
      IF(FLAG.EQ.2) $$$5 = $$5(XORY,Y1,X1,Y2,X2,Y3,X3,Y4,X4,Y5,X5)      00008591
      RETURN                                                            00008600
      END                                                               00008610
C                                                                       00008620
C $$$8                                                                  00008630
C                                                                       00008640
      FUNCTION $$$8(XORY,FLAG,X1,Y1,X2,Y2,X3,Y3,X4,Y4,X5,Y5,X6,Y6,      00008650
     1X7,Y7,X8,Y8)                                                      00008660
      INTEGER FLAG                                                      00008670
      IF(FLAG.EQ.0) $$$8 = $$8(XORY,X1,Y1,X2,Y2,X3,Y3,X4,Y4,            00008680
     1X5,Y5,X6,Y6,X7,Y7,X8,Y8)                                          00008690
      IF(FLAG.EQ.1) $$$8 = $$8(XORY,Y8,X8,Y7,X7,Y6,X6,Y5,X5,            00008700
     1Y4,X4,Y3,X3,Y2,X2,Y1,X1)                                          00008710
      IF(FLAG.EQ.2) $$$8 = $$8(XORY,Y1,X1,Y2,X2,Y3,X3,Y4,X4,            00008711
     1Y5,X5,Y6,X6,Y7,X7,Y8,X8)                                          00008712
      RETURN                                                            00008720
      END                                                               00008730
C                                                                       00000010
C                                                                       00000020
C                                                                       00000030
C                                                                       00000040
C                                                                       00000050
C                                                                       00000060
C***************************** S P T P O S ******************************00000061
C                                                                       00000080
C                                                                       00000090
C SPTPOS SUBROUTINE                                                     00000100
C                                                                       00000110
C COMPUTES CRAFT PARAMETERS AND TASK PROBABILITIES OF SUCCESS           00000120
C FOR A CRAFT                                                           00000130
C                                                                       00000140
C                                                                       00000150
C TO FIND CRAFT PARAMETERS:                                             00000160
C                                                                       00000170
C                                                                       00000180
C                                                                       00000190
      SUBROUTINE SPTPOS(TYPE,DISP,LENG,DSPEED,FUFRAC,                   00000200
     1 VISDTB,TOWDTB,DPHDTB,SSPDTB)                                     00000210
      IMPLICIT REAL(A-Z)                                                00000220
      INTEGER I,J,K                                                     00000230
      INTEGER TYPE,ENG,RG,JTPOS                                         00000240
      INTEGER TYPNUM                                                    00000250
C                                                                       00000260
C                                                                       00000270
      DIMENSION CFTNAM(8),CRAFT(8),SPEED(4),MFULRT(4)                   00000280
      DATA CRAFT/8*'    '/                                              00000290
C                                                                       00000300
      INTEGER IDISP,ILENG,IDSPD,IFILE,RATE,ISURVV,SS1,SS2               00000310
      DIMENSION CWSPD(4),SFCENG(4),SFCCF(4),TOTSFC(4),SFCGAL(4)         00000320
      DIMENSION HPUTIL(4),FUELRT(4),ENDUR(4),RANGE(4)                   00000330
      DIMENSION FUELR2(4),ENG(4),TNRAD(4),MOTION(4)                     00000340
      DIMENSION CC(19),DF(19),LS(19),MN(19),MO(19),TW(19),TPOS(19)      00000350
      DIMENSION SK(19)                                                  00000360
      COMMON/CHAR/LTOB,BEAM,DTOL,DRAF,SSPRBD,                           00000370
     1 DECK,USELD,FUELCP,CARGCP,TOWDSP,                                 00000380
     2 SURVIV,HPINST,HPPTON,HPTNKT,CWSPD,ENG,SFCENG,SFCCF,TOTSFC,SFCGAL,00000390
     3 HPUTIL,FUELRT,FUELR2,ENDUR,RANGE,MOTION,TNRAD                    00000400
      COMMON/PARAM/IDISP,IDSPD,CFTNAM,SSAVG,SPEED,MFULRT,               00000410
     1TOWSPD,CC,DF,LS,MN,TW                                             00000420
      COMMON/POS/ASST,BORD,MNAC,RTRV,WAIT,WEQD,WEQP,SDIU,SESC,          00000430
```

```
     1SPEO,SPAT,TOWS,ESCT,IDNT,PATL,STGT,TRPT,TRST,RSPD                  00000440
      COMMON/CSSPRB/SSPROB                                              00000450
      DIMENSION GO(4)                                                   00000460
      DIMENSION MNACC(4),MNBRKG(4),MNTUR(4)                             00000470
      DIMENSION SO(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','CLF-','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
     1 'CONV','ENTI','ONAL',' CRA','FT  ','    ','    ','    '/        00000670
      DIMENSION CGCRNM(2,12)                                            00000680
      DATA CGCRNM/'MRB ','    ','FWB ','32  ','UTB ','41  ',           00000690
     1'MLB ','44  ','MLB ','52  ','ANB ','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)','(DE)'/                                        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
```

```
      IF(TYPE.EQ.*0)TYPNUM=7                                            00001020
      IF(TYPE.EQ.*0)TYPNUM=8                                            00001030
      IF(TYPE.EQ.70)TYPNUM=9                                            00001040
      IF(TYPE.EQ.80)TYPNUM=10                                           00001050
      IF(TYPE.GE.100)CGTYPE=TYPE-100                                    00001060
C                                                                       00001070
      DO 4200 RATE=1,4                                                  00001080
      IF(TYPE.GE.100)SFCENG(RATE)=0.0                                   00001090
      IF(TYPE.GE.100)SFCCF(RATE)=0.0                                    00001100
      IF(TYPE.GE.100)TOTSFC(RATE)=0.0                                   00001110
      IF(TYPE.GE.100)SFCGAL(RATE)=0.0                                   00001120
 4200 CONTINUE                                                          00001130
C                                                                       00001140
C                                                                       00001150
C                                                                       00001160
C                                                                       00001170
C FIND PARAMETERS                                                       00001180
C                                                                       00001190
C   ALL PARAMETER PROBABILITIES SET EQUAL TO 1.0 EXCEPT                 00001200
C     WHERE RECALCULATED BELOW                                          00001210
C                                                                       00001220
      DO 9000 JTPOS=1,19                                                00001230
      CC(JTPOS) = 1.0                                                   00001240
      DF(JTPOS) = 1.0                                                   00001250
      LS(JTPOS) = 1.0                                                   00001260
      MN(JTPOS) = 1.0                                                   00001270
      SK(JTPOS) = 1.0                                                   00001280
      TW(JTPOS) = 1.0                                                   00001290
 9000 CONTINUE                                                          00001300
C                                                                       00001310
 4660 CONTINUE                                                          00001320
C                                                                       00001330
C                                                                       00001340
C  AVERAGE SPEED AND AVERAGE FUEL RATE (IN EXPECTED SEA STATES          00001350
C   AND VISIBILITIES)                                                   00001360
C                                                                       00001370
      VISFUF(1) = 99999.                                                00001380
C    FUEL RATE IN LIMITED VISIBILITY                                    00001390
C                                                                       00001391
      IF(TYPE.GT.100) GO TO 8900                                        00001392
C                                                                       00001393
      DO 4988 VISTYP = 2,3,1                                            00001400
      VVIS = VMXVIS(VISTYP)                                             00001410
      FCTRSP = VVIS/$RSSPD(TYPE)                                        00001420
      RATE = 0                                                          00001430
      HPFCTU = $HPFCTI(TYPE,RATE+0.,FCTRSP)                             00001440
      ZHPTIL = HPFCTU*$HPBIN(TYPE,DISP)                                 00001450
      IF(VISTYP.EQ.2)ZENG = $ENG(TYPE,2)                                00001460
      IF(VISTYP.EQ.3)ZENG = $ENG(TYPE,3)                                00001470
      ZSFCEN = $SFCEN(ZENG,HPINST)                                      00001480
      ZSFCCF = $SFCCF(ZENG,HPFCTU)                                      00001490
      ZTTSFC = ZSFCEN*ZSFCCF                                            00001500
      ZSFCGL = ZTTSFC*335./2240.                                        00001510
      VISFUF(VISTYP) = ZHPTIL*ZSFCGL                                    00001520
 4988 CONTINUE                                                          00001530
      GO TO 8910                                                        00001531
C                                                                       00001540
C  FUEL RATES FOR COAST GUARD CRAFT IN LIMITED VISIBILITY               00001541
 8900 VISFUF(2) = CGFR20(CGTYPE)                                        00001542
      VISFUF(3) = CGFR10(CGTYPE)                                        00001543
C                                                                       00001544
 8910 DO 711 RATE = 1,4                                                 00001550
      CALL VWTAV(SSPRBD,VISDIS,VISDTB,VMXVIS,TYPE,DISP,                 00001560
```

```
     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=$SSPDS(TYPE,DISP,RATE,DSPEED,PCDSPD)                             00001700
      GO(RATE)=$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(12) = 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
```

```
      SU(4)=SUO                                                          00002200
C                                                                        00002210
C LS: LIMITING SEA STATE PARAMETER                                       00002220
C                                                                        00002230
      DO 4710 RATE=1,4                                                   00002240
      LLS(RATE)=GO(RATE)                                                 00002250
      IF(SU(RATE).LT.GO(RATE))LLS(RATE)=SU(RATE)                         00002260
4710  CONTINUE                                                           00002270
      DO 9002 JTPOS=1,19                                                 00002280
      IF(JTPOS.LE.7) LS(JTPOS) = AMIN1(LLS(4),SK(JTPOS))                 00002290
      IF(JTPOS.GE.8.AND.JTPOS.LE.12) LS(JTPOS) = AMIN1(LLS(3),SK(JTPOS))00002300
      IF(JTPOS.GE.13.AND.JTPOS.LE.18) LS(JTPOS) =AMIN1(LLS(2),SK(JTPOS))00002310
      IF(JTPOS.EQ.19) LS(JTPOS) = AMIN1(LLS(1),SK(JTPOS))                00002320
 9002 CONTINUE                                                           00002330
C                                                                        00002340
C                                                                        00002350
C CC: CARGO CAPACITY                                                     00002360
      CC(17) = 999.                                                      00002370
C                                                                        00002380
C DF: DRAFT PARAMETER                                                    00002390
C                                                                        00002400
      DFO=1.-$PDPTH(DPHDTB,DRAF)                                         00002410
      DF(1)=DFO                                                          00002420
      DF(2)=DFO                                                          00002430
      DF(3)=DFO                                                          00002440
      DF(4)=DFO                                                          00002450
      DF(6)=DFO                                                          00002460
      DF(7)=DFO                                                          00002470
      DF(8)=DFO                                                          00002480
      DF(10)=DFO                                                         00002490
      DF(11)=DFO                                                         00002500
      DF(16)=DFO                                                         00002510
C                                                                        00002520
C                                                                        00002530
C                                                                        00002540
C PRINT MFULRT,SPEED AND TOWSPD AT END OF CHARACTERISTICS LIST           00002550
C                                                                        00002560
      WRITE(6,3041) MFULRT                                               00002570
 3041 FORMAT(10X,'AVG FUEL RATE',3X,4F8.1,4X,'GAL/HR')                   00002580
      WRITE(6,3042) SPEED                                                00002590
 3042 FORMAT(10X,'AVG SPEED',7X,4F8.1,4X,'KNOTS')                        00002600
      WRITE(6,3043) TOWSPD                                               00002610
 3043 FORMAT(10X,'TOW SPEED',13X,'-',7X,'-',1X,F8.1,6X,'-',5X,           00002620
     1'KNOTS')                                                           00002630
C                                                                        00002640
C                                                                        00002650
C PRINT PARAMETER VALUES FOR MASTER TASKS                                00002660
C                                                                        00002670
      IF(TYPE.GE.100)GO TO 4792                                          00002680
      DO 4791 I=1,8                                                      00002690
      CFTNAM(I)=CRFNM(I,TYPNUM)                                          00002700
4791  CONTINUE                                                           00002710
      GO TO 4795                                                         00002720
4792  DO 4793 I=1,8                                                      00002730
      CFTNAM(I)=CRAFT(I)                                                 00002740
4793  CONTINUE                                                           00002750
      CFTNAM(1)=CGCRNM(1,CGTYPE)                                         00002760
      CFTNAM(2)=CGCRNM(2,CGTYPE)                                         00002770
4795  SSAVG=AVESS(SSPDTB)                                                00002780
      IDISP=DISP+.500001                                                 00002790
      ILENG=LENG+.500001                                                 00002800
      IDSPD=DSPEED+.500001                                               00002810
      IFILE=6                                                            00002860
```

```
      IF(TYPE.LT.100)WRITE(IFILE,4901)(CRFNM(I,TYPNUM),I=1,8)
 4901 FORMAT('1'/10X,12X,'C R A F T   P A R A M E T E R S'/
     1   //11X,11X,'CRAFT TYPE',5X,8A4)
      IF(TYPE.GE.100)WRITE(IFILE,4936)(CGCRNM(I,CGTYPE),I=1,2)
 4936  FORMAT('1'/10X,12X,'C R A F T   P A R A M E T E R S'/
     1//11X,11X,'CRAFT TYPE',5X,'COAST GUARD ',2A4)
       WRITE(IFILE,4937)IDISP,ILENG,IDSPD,FUFRAC
 4937 FORMAT(11X,11X,'DISPLACEMENT',16,2X,'TONS'/
     3   11X,11X,'LENGTH',6X,I6,2X,'FEET'/
     4   11X,11X,'DESIGN SPEED',I6,2X,'KNOTS'/
     3   11X,11X,'FUEL FRACTION',F6.2/)
      WRITE(IFILE,4945)VISDTB,TOWDTB,DPHDTB,SSPDTB,AVESS(SSPDTB)
 4945 FORMAT(11X,11X,4X,'VISIBILITY DISTRIBUTION NO.',I2/
     2        11X,11X,4X,'TOW DISTRIBUTION NO.',I2/
     2        11X,11X,4X,'DEPTH DISTRIBUTION NO.',I2/
     3        11X,11X,4X,'SEA STATE DISTRIBUTION NO.',I2/
     4        11X,11X,4X,'(AVERAGE SEA STATE=',F3.1,')')
      WRITE(IFILE,5001)
   01 FORMAT(//14X,'TASK',2X,'CARGO',1X,'DRAFT',1X,'MANEUV',2X,
     1 'SEA',3X,'TOW'/14X,'CODE',2X,'CPCTY',14X,'STATE'//22X,
     2 'CC',4X,'DF',4X,'MN',4X,'LS',4X,'TW')
C
      WRITE(IFILE,5002)
 5002 FORMAT(//10X,'ON SCENE:')
      WRITE(IFILE,5003)DF(1),MN(1),LS(1)
 5003 FORMAT(14X,'ASST',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'ASSIST')
      WRITE(IFILE,5006) DF(2),MN(2),LS(2)
 5006 FORMAT(14X,'BORD',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'BOARD')
      WRITE(IFILE,5005)DF(3),MN(3),LS(3)
 5005 FORMAT(14X,'MNAC',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'MOTTOW ACT
     1IVITIES')
      WRITE(IFILE,5007)DF(4),MN(4),LS(4)
 5007 FORMAT(14X,'RTRV',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'RETRIEVE')
      WRITE(IFILE,5004)LS(5)
 5004 FORMAT(14X,'WAIT',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'WAIT')
      WRITE(IFILE,5008)DF(6),LS(6)
 5008 FORMAT(14X,'WFOD',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'WORK EQUIP
     1ENT @ DRIFT')
      WRITE(IFILE,5009)DF(7),MN(7),LS(7)
 5009 FORMAT(14X,'WFOP',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'WORK EQUIP
     1ENT @ POSITION')
C
      WRITE(IFILE,5010)
 5010 FORMAT(/10X,'REDUCED SPEED:')
      WRITE(IFILE,5013)DF(8),LS(8)
 5013 FORMAT(14X,'SDIU',3X,2(1X,'--',1X,2X,F4.2,2X),1X,'--',3X,'SEARCH F
     1OR DISTRESSED UNIT')
      WRITE(IFILE,5012)LS(9)
 5012 FORMAT(14X,'SESC',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'SLOW ESCORT')
      WRITE(IFILE,5015)DF(10),LS(10)
 5015 FORMAT(14X,'SPAT',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'SLOW PATROL
     1')
      WRITE(IFILE,5014)DF(11),LS(11)
 5014 FORMAT(14X,'SPEO',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'SEARCH FOR
     1PEOPLE')
      WRITE(IFILE,5011)MN(12),LS(12),TW(12)
 5011 FORMAT(14X,'TOWS',3X,2(1X,'--',3X),3(F4.2,2X),'TOWS')
C
      WRITE(IFILE,5016)
 5016 FORMAT(/10X,'CRUISE SPEED:')
      WRITE(IFILE,5030)LS(13)
 5030 FORMAT(14X,'ESCT',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'ESCORT')
      WRITE(IFILE,5017)MN(14),LS(14)
```

```
 5017 FORMAT(14X,'IDNT',3X,2(1X,'---',3X),2(F4.2,2X),1X,'---',3X,'IDENTIFY00003460
     1')                                                                00003461
      WRITE(IFILE,5018)LS(15)                                           00003470
 5018 FORMAT(14X,'PATL',3X,3(1X,'---',3X),F4.2,3X,'---',3X,'PATROL')    00003480
      WRITE(IFILE,5019)DF(16),LS(16)                                    00003490
 5019 FORMAT(14X,'STGT',3X,2(1X,'---',3X,F4.2,2X),1X,'---',3X,'SEARCH FOR 00003500
     1TARGET')                                                          00003501
      WRITE(IFILE,5021)CC(17),LS(17)                                    00003510
 5021 FORMAT(14X,'TRPT',3X,F4.2,2X,2(1X,'---',3X),F4.2,3X,'---',3X,'TRANSP00003520
     1ORT')                                                             00003521
      WRITE(IFILE,5020)LS(18)                                           00003530
 5020 FORMAT(14X,'TRST',3X,3(1X,'---',3X),F4.2,3X,'---',3X,'TRANSIT')   00003540
C                                                                       00003550
      WRITE(IFILE,5022)                                                 00003560
 5022 FORMAT(/10X,'FLANK SPEED:')                                       00003570
      WRITE(IFILE,5023) LS(19)                                          00003580
 5023 FORMAT(14X,'RSPD',3X,3(1X,'---',3X),F4.2,3X,'---',3X,'RESPOND')   00003590
      WRITE(IFILE,4989)                                                 00003600
 4989 FORMAT(//10X,                                                     00003610
     1'**** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AND WEIGHT OF CARGO00003620
     2)')                                                               00003630
C                                                                       00003650
C                                                                       00003660
C                                                                       00003670
C                                                                       00003680
C TO FIND TASK PROBABILITIES OF SUCCESS:                                00003690
C                                                                       00003700
C                                                                       00003710
      DO 9058 JTPOS=1,19                                                00003720
      IF(JTPOS.EQ.17) GO TO 9060                                        00003730
      TPOS(JTPOS) = CC(JTPOS)*DF(JTPOS)*LS(JTPOS)*MN(JTPOS)             00003740
     1*TW(JTPOS)                                                        00003750
      GO TO 9059                                                        00003760
 9060 TPOS(JTPOS) = 999.                                                00003770
 9059 CONTINUE                                                          00003780
 9058 CONTINUE                                                          00003790
C                                                                       00003800
C                                                                       00003810
C                                                                       00003820
C PRINT TASK PROBABILITIES OF SUCCESS                                   00003830
C                                                                       00003840
      IFILE=6                                                           00003870
      IF(TYPE.LT.100)WRITE(IFILE,6031)(CRFNM(I,TYPNUM),I=1,8)           00003900
 6031 FORMAT('1'/5X,8X,'T A S K  P R O B A B I L I T I E S  O F  ',     00003910
     1'S U C C E S S '                                                  00003920
     1//2X,15X,'CRAFT TYPE',5X,8A4)                                     00003930
      IF(TYPE.GE.100)WRITE(IFILE,6032)(CGCRNM(I,CGTYPE),I=1,2)          00003940
 6032 FORMAT('1'/5X,8X,'T A S K  P R O B A B I L I T I E S  O F  ',     00003950
     1'S U C C E S S '/                                                 00003960
     1//2X,15X,'CRAFT TYPE',5X,'COAST GUARD ',2A4)                      00003970
      WRITE(IFILE,6033)IDISP,ILENG,IDSPD,FUFRAC                         00003980
 6033 FORMAT(2X,15X,'DISPLACEMENT',I6,2X,'TONS'/                        00003990
     3   2X,15X,'LENGTH',6X,I6,' FEET'/                                 00004000
     4   2X,15X,'DESIGN SPEED',I6,2X,'KNOTS'/                           00004010
     5   2X,15X,'FUEL FRACTION',F6.2/)                                  00004020
      WRITE(IFILE,6112)VISDTB,TOWDTB,DPHDTB,SSPDTB,AVESS(SSPDTB)        00004030
 6112 FORMAT(2X,15X,4X,'VISIBILITY DISTRIBUTION NO.',I2/                00004040
     2       2X,15X,4X,'TOW DISTRIBUTION NO.',I2/                       00004050
     2       2X,15X,4X,'DEPTH DISTRIBUTION NO.',I2/                     00004060
     3       2X,15X,4X,'SEA STATE DISTRIBUTION NO.',I2/                 00004070
     4       2X,15X,4X,'(AVERAGE SEA STATE=',F3.1,')')                  00004080
      WRITE(IFILE,6001)                                                 00004090
 6001 FORMAT(//14X,'TASK',3X,'TASK PROB.',4X,'TASK'/14X,'CODE',         00004100
```

```
     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
```

```
C  PRINT PARAMETER VALUES FOR EXPANDED TASKS                                00004750
C                                                                          00004760
      IFILE=6                                                              00004790
      IF(TYPE.LT.100)WRITE(IFILE,2031)(CRFNM(I,TYPNUM),I=1,8)              00004820
2031  FORMAT('1'/14X,12X,'C R A F T   P A R A M E T E R S'/                00004830
     1   //11X,15X,'CRAFT TYPE',5X,8A4)                                    00004840
      IF(TYPE.GE.100)WRITE(IFILE,2032)(CGCRNM(I,CGTYPE),I=1,2)             00004850
2032   FORMAT('1'/14X,12X,'C R A F T   P A R A M E T E R S '/              00004860
     1//11X,15X,'CRAFT TYPE',5X,'COAST GUARD ',2A4)                        00004870
      WRITE(IFILE,2033)IDISP,ILENG,IDSPD,FUFRAC                            00004880
2033  FORMAT(11X,15X,'DISPLACEMENT',I6,2X,'TONS'/                          00004890
     3   11X,15X,'LENGTH',6X,I6,2X,'FEET'/                                 00004900
     4   11X,15X,'DESIGN SPEED',I6,2X,'KNOTS'/                             00004910
     3   11X,15X,'FUEL FRACTION',F6.2/)                                    00004920
      WRITE(IFILE,2112)VISDTB,TOWDTB,DPHDTB,SSPDTB,AVESS(SSPDTB)           00004930
2112  FORMAT(11X,15X,4X,'VISIBILITY DISTRIBUTION NO.',I2/                  00004940
     2        11X,15X,4X,'TOW DISTRIBUTION NO.',I2/                        00004950
     2        11X,15X,4X,'DEPTH DISTRIBUTION NO.',I2/                      00004960
     3        11X,15X,4X,'SEA STATE DISTRIBUTION NO.',I2/                  00004970
     4        11X,15X,4X,'(AVERAGE SEA STATE=',F3.1,')')                   00004980
      WRITE(IFILE,2001)                                                    00004990
2001  FORMAT(//14X,'TASK',2X,'CARGO',1X,'DRAFT',1X,'MANEUV',2X,            00005000
     1 'SEA',3X,'TOW'/14X,'CODE',2X,'CPCTY',14X,'STATE'//22X,              00005010
     2 'CC',4X,'DF',4X,'MN',4X,'LS',4X,'TW')                               00005020
C                                                                          00005030
      WRITE(IFILE,2002)                                                    00005040
 2002 FORMAT(//10X,'ON SCENE:')                                            00005050
C                                                                          00005060
      WRITE(IFILE,2003) DF(2),MN(2),LS(2)                                  00005070
 2003 FORMAT(14X,'BRD',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'BOARD')        00005080
      WRITE(IFILE,2004)DF(7),MN(7),LS(7)                                   00005090
 2004 FORMAT(14X,'FFF',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'FIGHT FIRE F00005100
     1ROM CG VESSEL')                                                      00005101
      WRITE(IFILE,2005)LS(5)                                               00005110
 2005 FORMAT(14X,'FFO',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'FIGHT FIRE ON A00005120
     1NOTHER VESSEL')                                                      00005121
      WRITE(IFILE,2006)DF(1),MN(1),LS(1)                                   00005130
 2006 FORMAT(14X,'GAS',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'GENERAL ASSI00005140
     1STANCE')                                                             00005141
      WRITE(IFILE,2007)LS(5)                                               00005150
 2007 FORMAT(14X,'INS',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'INSPECTION')      00005160
      WRITE(IFILE,2008)DF(7),MN(7),LS(7)                                   00005170
 2008 FORMAT(14X,'LEQ',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'LOAD EQUIPME00005180
     1NT')                                                                 00005181
      WRITE(IFILE,2009)LS(5)                                               00005190
 2009 FORMAT(14X,'LOI',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'LOITER')          00005200
      WRITE(IFILE,2010)DF(7),MN(7),LS(7)                                   00005210
 2010 FORMAT(14X,'LSB',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'LAUNCH SMALL00005220
     1 BOAT')                                                              00005221
      WRITE(IFILE,2011)DF(3),MN(3),LS(3)                                   00005230
 2011 FORMAT(14X,'MAC',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'MONITOR ACTI00005240
     1VITIES')                                                             00005241
      WRITE(IFILE,2012)DF(3),MN(3),LS(3)                                   00005250
 2012 FORMAT(14X,'MOS',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'MONITOR OIL 00005260
     1SPILL')                                                              00005261
      WRITE(IFILE,2013)LS(5)                                               00005270
 2013 FORMAT(14X,'OBA',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'ON BOARD ASSIST00005280
     1ANCE')                                                               00005281
      WRITE(IFILE,2015)LS(5)                                               00005290
 2015 FORMAT(14X,'OSC',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'ON SCENE COMMAN00005300
     1DER(GENERAL)')                                                       00005301
      WRITE(IFILE,2016) DF(2),MN(2),LS(2)                                  00005310
 2016 FORMAT(14X,'RRP',3X,1X,'--',3X,3(F4.2,2X),1X,'--',3X,'RETRIEVE BOA00005320
```

```
     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,'S/E',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'SETZE')       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 EQUIPME00005520
     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                                                                      00005570
C                                                                      00005580
      IF(TYPE.LT.100)WRITE(IFILE,3031)(CRFNM(I,TYPNUM),I=1,8)          00005640
 3031 FORMAT('1'/14X,12X,'C R A F T  P A R A M E T E R S'/             00005650
     1   //11X,15X,'CRAFT TYPE',5X,8A4)                               00005660
      IF(TYPE.GE.100)WRITE(IFILE,3032)(CGCRNM(I,CGTYPE),I=1,2)         00005670
 3032  FORMAT('1'/14X,12X,'C R A F T  P A R A M E T E R S '/           00005680
     1//11X,15X,'CRAFT TYPE',5X,'COAST GUARD ',2A4)                    00005690
      WRITE(IFILE,3033)IDISP,ILENG,IUSPD,FUFRAC                        00005700
 3033 FORMAT(11X,15X,'DISPLACEMENT',I6,2X,'TONS'/                      00005710
     3   11X,15X,'LENGTH',6X,I6,2X,'FEET'/                             00005720
     4   11X,15X,'DESIGN SPEED',I6,2X,'KNOTS'/                         00005730
     3   11X,15X,'FUEL FRACTION',F6.2/)                                00005740
      WRITE(IFILE,3112)VISDTB,TOWDTB,DPHDTB,SSPDTB,AVESS(SSPDTB)       00005750
 3112 FORMAT(11X,15X,4X,'VISIBILITY DISTRIBUTION NO.',I2/              00005760
     2       11X,15X,4X,'TOW DISTRIBUTION NO.',I2/                     00005770
     2       11X,15X,4X,'DEPTH DISTRIBUTION NO.',I2/                   00005780
     3       11X,15X,4X,'SEA STATE DISTRIBUTION NO.',I2/               00005790
     4       11X,15X,4X,'(AVERAGE SEA STATE=',F3.1,')')                00005800
      WRITE(IFILE,3001)                                                00005810
 3001 FORMAT(//14X,'TASK',2X,'CARGO',1X,'DRAFT',1X,'MANEUV',2X,        00005820
     1 'SEA',3X,'TOW'/14X,'CODE',2X,'CPCTY',14X,'STATE'//22X,          00005830
     2 'CC',4X,'DF',4X,'MN',4X,'LS',4X,'TW')                           00005840
C                                                                      00005850
      WRITE(IFILE,3002)                                                00005860
 3002 FORMAT(//10X,'REDUCED SPEED:')                                   00005870
C                                                                      00005880
      WRITE(IFILE,3004)DF(8),LS(8)                                     00005890
 3004 FORMAT(14X,'SDU',3X,2(1X,'--',1X,2X,F4.2,2X),1X,'--',1X,2X,'SEARCH00005900
     1 FOR DISTRESSED UNIT')                                           00005901
      WRITE(IFILE,3003)LS(9)                                           00005910
 3003 FORMAT(14X,'SES',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'SLOW ESCORT')  00005920
      WRITE(IFILE,3005)DF(11),LS(11)                                   00005930
```

```
 3005 FORMAT(14X,'SPE',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'SEARCH FOR P 00005940
     1EOPLE')                                                            00005941
      WRITE(IFILE,3006)DF(10),LS(10)                                     00005950
 3006 FORMAT(14X,'SPT',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'SLOW PATROL' 00005960
     1)                                                                  00005961
      WRITE(IFILE,3007)MN(12),LS(12),TW(12)                              00005970
 3007 FORMAT(14X,'TOW',3X,2(1X,'--',3X),3(F4.2,2X),'TOW')                00005980
C                                                                        00005990
      WRITE(IFILE,3008)                                                  00006000
 3008 FORMAT(/10X,'CRUISE SPEED:')                                       00006010
      WRITE(IFILE,3030)LS(13)                                            00006020
 3030 FORMAT(14X,'ESC',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'ESCORT')        00006030
      WRITE(IFILE,3009)MN(14),LS(14)                                     00006040
 3009 FORMAT(14X,'IDC',3X,2(1X,'--',3X),2(F4.2,2X),1X,'--',3X,'IDENTIFY  00006050
     1CRAFT')                                                            00006051
      WRITE(IFILE,3010)MN(14),LS(14)                                     00006060
 3010 FORMAT(14X,'IDF',3X,2(1X,'--',3X),2(F4.2,2X),1X,'--',3X,'IDENTIFY  00006070
     1')                                                                 00006071
      WRITE(IFILE,3011)LS(15)                                            00006080
 3011 FORMAT(14X,'PAT',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'PATROL')        00006090
      WRITE(IFILE,3035)LS(18)                                            00006100
 3035 FORMAT(14X,'SFL',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'SEARCH FOR FLEE 00006110
     1T')                                                                00006111
      WRITE(IFILE,3014)DF(16),LS(16)                                     00006120
 3014 FORMAT(14X,'SSH',3X,2(1X,'--',3X,F4.2,2X),1X,'--',3X,'SEARCH FOR S 00006130
     1HIP')                                                              00006131
      WRITE(IFILE,3015) CC(17),LS(17)                                    00006140
 3015 FORMAT(14X,'TEG',3X,F4.2,2X,2(1X,'--',3X),F4.2,3X,'--',3X,'TRANSPO 00006150
     1RT')                                                               00006151
      WRITE(IFILE,3016)LS(18)                                            00006160
 3016 FORMAT(14X,'TPE',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'TRANSPORT PEOPL 00006170
     1E')                                                                00006171
      WRITE(IFILE,3017)LS(18)                                            00006180
 3017 FORMAT(14X,'TRA',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'TRANSIT')       00006190
C                                                                        00006200
      WRITE(IFILE,3018)                                                  00006210
 3018 FORMAT(/10X,'FLANK SPEED:')                                        00006220
C                                                                        00006230
      WRITE(IFILE,3019)LS(19)                                            00006240
 3019 FORMAT(14X,'DSH',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'DASH')          00006250
      WRITE(IFILE,3020)LS(19)                                            00006260
 3020 FORMAT(14X,'INT',3X,3(1X,'--',3X),F4.2,3X,'--',3X,'INTERDICT')     00006270
C                                                                        00006280
      WRITE(IFILE,3021)                                                  00006290
 3021 FORMAT(//10X,                                                      00006300
     1'***** DEPENDENT UPON SCENARIO (E.G., FOOTPRINT AND WEIGHT OF CARGO 00006310
     2)')                                                                00006320
C                                                                        00006340
C PRINT EXPANDED TASK PROBABILITIES OF SUCCESS                           00006350
C                                                                        00006360
C TPOS(1) =ASST=GAS                                                      00006370
C TPOS(2) =BORD=BRD=RBP                                                  00006380
C TPOS(3) =MNAC=SSI=MAC=MOS                                              00006390
C TPOS(4) =RTRV=RCB=RPE                                                  00006400
C TPOS(5) =WAIT=FFO=INS=LOI=OBA=SZE=OSC=WOB                              00006410
C TPOS(6) =WEQD=WGD                                                      00006420
C TPOS(7) =WEQP=FFF=LEQ=LSB=RSB=TWS=ULQ=WQF                              00006430
C TPOS(8) =SDIU=SDU                                                      00006440
C TPOS(9) =SESC=SES                                                      00006450
C TPOS(10)=SPAT=SPT                                                      00006460
C TPOS(11)=SPEO=SPE                                                      00006470
C TPOS(12)=TOWS=TOW                                                      00006480
C TPOS(13)=ESCT=ESC                                                      00006490
```

```
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=TPF=TKA                                             00006540
C TPOS(19)=WSPD=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 E S  O F  ',     00006630
     1'S U C C E 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 E S  O F  ',     00006670
     1'S U C C E 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
```

```
 7018 FORMAT(14X,'RPE',6X,F5.3,5X,'RESCUE PEOPLE')                      00007170
      WRITE(IFILE,7019)TPOS(7)                                          00007180
 7019 FORMAT(14X,'RSB',6X,F5.3,5X,'RETRIEVE SMALL BOAT')                00007190
      WRITE(IFILE,7070)TPOS(3)                                          00007200
 7070 FORMAT(14X,'SSI',6X,F5.3,5X,                                      00007210
     1'STAKEOUT SPECIAL INTEREST VESSEL')                               00007220
      WRITE(IFILE,7021)TPOS(5)                                          00007230
 7021 FORMAT(14X,'SZE',6X,F5.3,5X,'SEIZE')                              00007240
      WRITE(IFILE,7022)TPOS(7)                                          00007250
 7022 FORMAT(14X,'TWS',6X,F5.3,5X,'TAKE WATER SAMPLE')                  00007260
      WRITE(IFILE,7023)TPOS(7)                                          00007270
 7023 FORMAT(14X,'ULQ',6X,F5.3,5X,'UNLOAD EQUIPMENT')                   00007280
      WRITE(IFILE,7026)TPOS(5)                                          00007290
 7026 FORMAT(14X,'WQB',6X,F5.3,5X,                                      00007300
     1'WORK EQUIPMENT FROM SMALL BOAT')                                 00007310
      WRITE(IFILE,7024)TPOS(6)                                          00007320
 7024 FORMAT(14X,'WQD',6X,F5.3,5X,'WORK EQUIPMENT @ DRIFT')             00007330
      WRITE(IFILE,7025)TPOS(7)                                          00007340
 7025 FORMAT(14X,'WQF',6X,F5.3,5X,                                      00007350
     1'WORK EQUIPMENT @ FIXED POSITION')                                00007360
C                                                                       00007380
C                                                                       00007390
      IF(TYPE.LT.100)WRITE(IFILE,8031)(CRFNM(I,TYPNUM),I=1,8)           00007450
 8031 FORMAT('1'/5X,8X,'T A S K  P R O B A B I L I T I F S  O F  ',     00007460
     1'S U C C E S S '                                                  00007470
     1//2X,15X,'CRAFT TYPE',5X,8A4)                                     00007480
      IF(TYPE.GE.100)WRITE(IFILE,8032)(CGCRNM(I,CGTYPE),I=1,2)          00007490
 8032 FORMAT('1'/5X,8X,'T A S K  P R O B A B I L I T I F S  O F  ',     00007500
     1'S U C C E S S '/                                                 00007510
     1//2X,15X,'CRAFT TYPE',5X,'COAST GUARD ',2A4)                      00007520
      WRITE(IFILE,8033)IDISP,ILENG,IDSPD,FUFRAC                         00007530
 8033 FORMAT(2X,15X,'DISPLACEMENT',I6,2X,'TONS'/                        00007540
     3   2X,15X,'LENGTH',6X,I6,' FEET'/                                 00007550
     4   2X,15X,'DESIGN SPEED',I6,2X,'KNOTS'/                           00007560
     5   2X,15X,'FUEL FRACTION',F6.2/)                                  00007570
      WRITE(IFILE,8112)VISDTB,TOWDTB,DPHDTB,SSPDTB,AVESS(SSPDTB)        00007580
 8112 FORMAT(2X,15X,4X,'VISIBILITY DISTRIBUTION NO.',I2/                00007590
     2       2X,15X,4X,'TOW DISTRIBUTION NO.',I2/                       00007600
     2       2X,15X,4X,'DEPTH DISTRIBUTION NO.',I2/                     00007610
     3       2X,15X,4X,'SEA STATE DISTRIBUTION NO.',I2/                 00007620
     4       2X,15X,4X,'(AVERAGE SEA STATE=',F3.1,')')                  00007630
      WRITE(IFILE,8001)                                                 00007640
 8001 FORMAT(//14X,'TASK',3X,'TASK PROB.',4X,'TASK'/14X,'CODE',         00007650
     1 3X,'OF SUCCESS')                                                 00007660
C                                                                       00007670
      WRITE(IFILE,8002)                                                 00007680
 8002 FORMAT(//10X,'REDUCED SPEED:')                                    00007690
C                                                                       00007700
      WRITE(IFILE,8004)TPOS(8)                                          00007710
 8004 FORMAT(14X,'SDU',6X,F5.3,'*',4X,'SEARCH FOR DISTRESSED UNIT')     00007720
      WRITE(IFILE,8003)TPOS(9)                                          00007730
 8003 FORMAT(14X,'SES',6X,F5.3,5X,'SLOW ESCORT')                        00007740
      WRITE(IFILE,8005)TPOS(11)                                         00007750
 8005 FORMAT(14X,'SPE',6X,F5.3,'*',4X,'SEARCH FOR PEOPLE')              00007760
      WRITE(IFILE,8006)TPOS(10)                                         00007770
 8006 FORMAT(14X,'SPT',6X,F5.3,5X,'SLOW PATROL')                        00007780
      WRITE(IFILE,8007)TPOS(12)                                         00007790
 8007 FORMAT(14X,'TOW',6X,F5.3,5X,'TOW')                                00007800
C                                                                       00007810
      WRITE(IFILE,8008)                                                 00007820
 8008 FORMAT(/10X,'CRUISE SPEED:')                                      00007830
      WRITE(IFILE,8080)TPOS(13)                                         00007840
 8080 FORMAT(14X,'ESC',6X,F5.3,5X,'ESCORT')                             00007850
```

```
      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,'TEQ',6X,F5.3,5X,'TRANSPORT EQUIPMENT')                 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('1')                                                        00008240
      RETURN                                                             00008250
      END                                                                00008260
C                                                                        00008270
C                                                                        00008280
C                                                                        00008290
C                                                                        00008300
C VWTAV                                                                  00008310
C                                                                        00008320
C WEIGHTED AVERAGE VELOCITY AND FUEL RATE                                00008330
C                                                                        00008340
      SUBROUTINE VWTAV(SSPEED,VISDIS,VISDTR,VMXVIS,TYPE,DISP,RATE,       00008350
     1DSPEED,FUELRT,VISFUE,VAVG,AVFURT)                                  00008360
      IMPLICIT REAL(A-Z)                                                 00008370
      INTEGER TYPE,RATE,SS1,DELTA,VISTYP,SSD,VISDTR                      00008380
      DIMENSION SSPEED(8),VMXVIS(3),VISDIS(3,3)                          00008390
      DIMENSION FUELRT(4),VISFUE(3)                                      00008400
      VAVG = 0.                                                          00008410
      AVFURT = 0.                                                        00008420
      DO 10 SS1 = 1,8                                                    00008430
      SSD = SS1 - 1                                                      00008440
      DO 100 DELTA = 1,9                                                 00008450
      SS = SSD + DELTA/10                                                00008460
      DO 200 VISTYP = 1,3                                                00008470
      PVINSS = $PVSSS(TYPE,DISP,RATE,DSPEED,SS)                          00008480
      VINSS = DSPEED*PVINSS/100.                                         00008490
```

```
      VWVIS = AMIN1(VINSS,VMXVIS(VISTYP))                                00008500
      POFSS = SSPROB(SS1)/9.                                             00008510
      POFVIS = VISDIS(VISTYP,VISDTB)                                     00008520
      VAVG = VAVG + VWVIS*POFSS*POFVIS                                   00008530
      FUEUSE = FUELRT(RATE)                                              00008540
      IF(VMXVIS(VISTYP).LT.VINSS) FUEUSE = VISFUE(VISTYP)                00008550
      AVFURT = AVFURT + FUEUSE*POFSS*POFVIS                              00008560
  200 CONTINUE                                                           00008570
  100 CONTINUE                                                           00008580
   10 CONTINUE                                                           00008590
      RETURN                                                             00008600
      END                                                                00008610
C                                                                        00008620
C                                                                        00008630
C $CPBSS                                                                 00008640
C                                                                        00008650
C CUMULATIVE PROBABILITY OF SEA STATE                                    00008660
C   $CPBSS(SS)= PROB THAT SEA STATE < OR = SS                            00008670
C                                                                        00008680
      FUNCTION $CPBSS(SSPROB,SSPDTB,SS)                                  00008690
      IMPLICIT REAL(A-Z)                                                 00008700
      INTEGER SSPDTB,J,ISSMX,ISS                                         00008710
      DIMENSION SSPROB(8,10)                                             00008720
      IF(SS.LT.8.)GO TO 50                                               00008730
      $CPBSS=1.                                                          00008740
      RETURN                                                             00008750
   50 ISS=SS                                                             00008760
      INTRP=SS-ISS                                                       00008770
      PRBSUM=0.                                                          00008780
      DO 100 J=1,ISS                                                     00008790
      PRBSUM=PRBSUM+SSPROB(J,SSPDTB)                                     00008800
  100 CONTINUE                                                           00008810
      INTSS=ISS+1                                                        00008820
      $CPBSS=PRBSUM+INTRP*SSPROB(INTSS,SSPDTB)                           00008830
      RETURN                                                             00008840
      END                                                                00008850
C                                                                        00008860
C                                                                        00008870
C                                                                        00008880
C                                                                        00008890
C PTWD                                                                   00008900
C                                                                        00008910
C TOW DISPLACEMENT CUMULATIVE PROBABILITY DISTRIBUTION                   00008920
C                                                                        00008930
C    PTOWD(D)= PROBABILITY THAT CRAFT TO BE TOWED HAS DISPLACEMENT < D   00008940
C    AVTWDS = AVERAGE DISPLACEMENT VALUE THAT CAN BE TOWED               00008950
C                                                                        00008960
      SUBROUTINE PTWD(TOWDIS,TOWDTB,TOWDSP,PTOWD,AVTWDS)                 00008970
      IMPLICIT REAL(A-Z)                                                 00008980
      INTEGER TOWDTB,I                                                   00008990
      DIMENSION TOWDIS(6,5)                                              00009000
      IN = TOWDSP                                                        00009010
   10 OUT = $$8(IN,0.,.0,.TOWDIS(1,TOWDTB),0.,                           00009020
     1TOWDIS(2,TOWDTB),.2,                                               00009030
     1TOWDIS(3,TOWDTB),.4,TOWDIS(4,TOWDTB),.6,TOWDIS(5,TOWDTB),.8,       00009040
     2TOWDIS(6,TOWDTB),1.,.999999,.1.)                                   00009050
      PTOWD = OUT                                                        00009060
      IN = PTOWD/2.                                                      00009070
      OUT = $$8(IN,0.,.0,.0,.TOWDIS(1,TOWDTB),.2,TOWDIS(2,TOWDTB),       00009080
     1.4,TOWDIS(3,TOWDTB),.6,TOWDIS(4,TOWDTB),                           00009090
     2.8,TOWDIS(5,TOWDTB),1.,TOWDIS(6,TOWDTB),1.,.999999.)               00009100
   60 AVTWDS = OUT                                                       00009110
  100 RETURN                                                             00009120
```

```
      END                                                                00009130
C                                                                        00009140
C                                                                        00009150
C-----------------------------------------------------------------       00009160
C $MWTAV                                                                 00009170
C                                                                        00009180
C WEIGHTED AVERAGE MOTION OF CRAFT                                       00009190
C                                                                        00009200
      FUNCTION $MWTAV(SSPRBD,TYPE,DISP,RATE)                             00009210
      IMPLICIT REAL(A-Z)                                                 00009220
      INTEGER TYPE,RATE,SS1,SS                                           00009230
      DIMENSION SSPRBD(8)                                                00009240
C                                                                        00009250
      SUM=0.                                                             00009260
C LAMBDA CONVERTS WAVE HEIGHT FROM CRAFT DISPLACEMENT TO                 00009270
C BASE DISPLACEMENT (=100 TONS)                                          00009280
      LAMBDA=(100/DISP)**.333                                            00009290
      IF(TYPE.EQ.60)LAMBDA=(1500/DISP)**.333                             00009300
      DO 100 SS1=1,8                                                     00009310
      SS=SS1-1                                                           00009320
      WVHTCF=.5*(-1.+2.5*EXP(.4*SS))                                     00009330
      WVHTBS=LAMBDA*WVHTCF                                               00009340
      SUM=SUM + SSPRBD(SS1)*$MVSWH(TYPE,RATE,WVHTBS)                     00009350
100   CONTINUE                                                           00009360
      $MWTAV=SUM                                                         00009370
      RETURN                                                             00009380
      END                                                                00009390
C-----------------------------------------------------------------       00009400
C                                                                        00009410
C $MVSWH AND $WHVSM                                                      00009420
C                                                                        00009430
C  MOTION OF BASE CRAFT VS. WAVE HEIGHT (FOR DISPLACEMENT=100 TONS       00009440
C  EXCEPT TYPE 60 DISPLACEMENT=1500 TONS) AND REVERSE                    00009450
C-----------------------------------------------------------------       00009460
      FUNCTION $MVSWH(TYPE,RATE,WVHTBS)                                  00009470
      IMPLICIT REAL(A-Z)                                                 00009480
      INTEGER TYPE,RATE,FLAG                                             00009490
C                                                                        00009500
      IN = WVHTBS                                                        00009510
      FLAG = 0                                                           00009520
      GO TO 1                                                            00009530
C                                                                        00009540
      ENTRY $WHVSM(TYPE,RATE,MTN)                                        00009550
      IN = MTN                                                           00009560
      FLAG = 2                                                           00009570
C-----------------------------------------------------------------       00009580
    1 IF(TYPE.NE.10)GO TO 11                                             00009590
      IF(RATE.EQ.1.OR.RATE.EQ.2) OUT = $$$3(IN,FLAG,0.,0.,15.,.5,        00009600
     118.,1.0)                                                           00009610
      IF(RATE.EQ.3) OUT = $$$(IN,FLAG,0.,0.,8.,1.0)                      00009620
      IF(RATE.EQ.4) OUT = $$$(IN,FLAG,0.,0.,12.,1.0)                     00009630
      GO TO 999                                                          00009640
11    IF(TYPE.NE.11)GO TO 20                                             00009650
      IF(RATE.EQ.1.OR.RATE.EQ.2) OUT = $$$3(IN,FLAG,0.,0.,5.,.25,        00009660
     18.,1.0)                                                            00009670
      IF(RATE.EQ.3) OUT = $$$(IN,FLAG,0.,0.,10.,1.0)                     00009680
      IF(RATE.EQ.4) OUT = $$$(IN,FLAG,0.,0.,14.,1.0)                     00009690
      GO TO 999                                                          00009700
20    IF(TYPE.NE.20.OR.TYPE.NE.21)GO TO 40                               00009710
      IF(RATE.EQ.1) OUT = $$$3(IN,FLAG,0.,0.,4.,2.5,1.0)                 00009720
      IF(RATE.EQ.2) OUT = $$$3(IN,FLAG,0.,0.,4.,2.6,1.0)                 00009730
      IF(RATE.EQ.3) OUT = $$$3(IN,FLAG,0.,0.,4.,2.7,1.)                  00009740
      IF(RATE.EQ.4) OUT = $$$3(IN,FLAG,0.,0.,4.,2.10.,1.0)               00009750
```

```
      GO TO 999                                                          00009760
40    IF(TYPE.NE.30)GO TO 50                                             00009770
      IF(RATE.EQ.1) OUT = $$$3(IN,FLAG,0.,0.,5.,.2,7.,1.0)               00009780
      IF(RATE.EQ.2) OUT = $$$3(IN,FLAG,0.,0.,5.,.2,8.,1.0)               00009790
      IF(RATE.EQ.3) OUT = $$$3(IN,FLAG,0.,0.,5.,.2,10.,1.0)              00009800
      IF(RATE.EQ.4) OUT = $$$3(IN,FLAG,0.,0.,5.,.2,12.,1.0)              00009810
      GO TO 999                                                          00009820
 50   IF(TYPE.NE.40) GO TO 60                                            00009830
      IF(RATE.EQ.1) OUT = $$$(IN,FLAG,0.,0.,3.,1.0)                      00009840
      IF(RATE.EQ.2) OUT = $$$(IN,FLAG,0.,0.,4.,1.0)                      00009850
      IF(RATE.EQ.3) OUT = $$$(IN,FLAG,0.,0.,8.,1.0)                      00009860
      IF(RATE.EQ.4) OUT = $$$(IN,FLAG,0.,0.,12.,1.0)                     00009870
      GO TO 999                                                          00009880
60    IF(TYPE.NE.50.OR.TYPE.NE.70.OR.TYPE.NE.80.OR..NOT.(TYPE.GT.100))   00009890
     1GO TO 80                                                           00009891
      IF(RATE.EQ.1) OUT = $$$(IN,FLAG,0.,0.,5.,1.0)                      00009900
      IF(RATE.EQ.2) OUT = $$$(IN,FLAG,0.,0.,6.,1.0)                      00009910
      IF(RATE.EQ.3) OUT = $$$(IN,FLAG,0.,0.,8.,1.0)                      00009920
      IF(RATE.EQ.4) OUT = $$$(IN,FLAG,0.,0.,12.,1.0)                     00009930
      GO TO 999                                                          00009940
C  FOR TYPE = 60                                                         00009941
   80 OUT = $$$3(IN,FLAG,0.,0.,17.143,.343,21.01,1.0)                    00009970
 999  $MVSWH = OUT                                                       00009990
      IF($MVSWH.GT.1.) $MVSWH = 1.                                       00010000
      $WHVSM = OUT                                                       00010010
      RETURN                                                             00010020
      END                                                                00010030
C                                                                        00010200
C                                                                        00010210
C                                                                        00010220
C $PDPTH                                                                 00010230
C                                                                        00010240
C DEPTH CUMULATIVE PROBABILITY DISTRIBUTION                              00010250
C                                                                        00010260
C   $PDPTH(D)=PROBABILITY THAT DEPTH < D                                 00010270
C                                                                        00010280
      FUNCTION $PDPTH(DPHDTB,DEPTH)                                      00010290
      IMPLICIT REAL(A-Z)                                                 00010300
      INTEGER DPHDTB                                                     00010310
      IF(DPHDTB.EQ.1)$PDPTH=0.                                           00010320
      RETURN                                                             00010330
      END                                                                00010340
C                                                                        00010350
C                                                                        00010360
C $PDSSS AND $SSPDS                                                      00010370
C                                                                        00010380
C  PERCENT DESIGN SPEED VS SEA STATE AND                                 00010390
C  SEA STATE VS PERCENT DESIGN SPEED                                     00010400
C                                                                        00010410
C   FOR CRUISE : MAXIMUM IS LIMITED BY LINE PARALLEL TO INITIAL          00010420
C     FLANK   LINE AND STARTING FROM PERCENT DESIGN SPEED AXIS           00010430
C    AT CWSPD(2)                                                         00010440
C                                                                        00010450
C   FOR REDUCED SPEED: MAXIMUM IS LIMITED BY LINE PARALLEL TO            00010460
C   SEA STATE AXIS AT                                                    00010470
C              PERCENT DESIGN SPEED = CWSPD(3)/DESIGN SPEED*100          00010480
C                                                                        00010490
C                                                                        00010500
C                                                                        00010510
      FUNCTION $PDSSS(TYPE,DISP,RATE,DSPEED,SS)                          00010520
      IMPLICIT REAL(A-Z)                                                 00010530
      INTEGER TYPE,RATE,FLAG                                             00010540
C                                                                        00010550
```

```
      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.,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.GT.150..AND.DISP.LE.200.) OUT = $$$4(IN,FLAG,0.,        00011190
     1100.,3.,100.,4.,80.,5.,0.)                                      00011200
      IF(DISP.GT.200.) OUT = $$$4(IN,FLAG,0.,100.,3.5,100.,           00011210
     14.5,60.,6.0,0.)                                                 00011220
      GO TO 991                                                       00011230
 1020 IF(DISP.LE.20.) OUT = $$$3(IN,FLAG,0.,85.,45,85.,3.,0.)         00011240
      IF(DISP.GT.20..AND.DISP.LE.50.) OUT = $$$4(IN,FLAG,0.,85.,      00011250
     12.25,85.,2.5,80.,4.0,0.)                                        00011260
      IF(DISP.GT.50..AND.DISP.LE.100.) OUT =$$$4(IN,FLAG,0.,85.,      00011270
     12.75,85.,3.,80.,4.5,0.)                                         00011280
      IF(DISP.GT.100..AND.DISP.LE.150.) OUT = $$$4(IN,FLAG,0.,85.,    00011290
     13.25,85.,3.5,80.,5.,0.)                                         00011300
      IF(DISP.GT.150..AND.DISP.LE.200.) OUT = $$$4(IN,FLAG,0.,        00011310
     185.,3.75,85.,4.,80.,5.5,0.)                                     00011320
      IF(DISP.GT.200.) OUT = $$$4(IN,FLAG,0.,85.,4.25,85.,4.5,80.,    00011330
     16.,0.)                                                          00011340
      GO TO 999                                                       00011350
C                                                                     00011360
  30  IF(RATE.EQ.2) GO TO 1030                                        00011370
      IF(DISP.LE.150.) OUT = $$$3(IN,FLAG,0.,100.,.5,100.,.5.5,0.)    00011380
      IF(DISP.GT.150.) OUT = $$$3(IN,FLAG,0.,100.,1.,100.,6.,0.)      00011390
      GO TO 991                                                       00011400
 1030 IF(DISP.LE.150.) OUT = $$$3(IN,FLAG,0.,87.5,1.125,87.5,         00011410
     15.5,0.)                                                         00011420
      IF(DISP.GT.150.)OUT = $$$3(IN,FLAG,0.,87.5,1.625,87.5,6.,0.)    00011430
      GO TO 999                                                       00011440
C                                                                     00011450
  40  IF(RATE.EQ.2) GO TO 1040                                        00011460
      IF(DISP.LE.5.) OUT = $$$3(IN,FLAG,0.,100.,1.72,92.,5.,0.)       00011470
      IF(DISP.LE.20.) OUT = $$$3(IN,FLAG,0.,100.,2.28,89.,5.5,0.)     00011480
      IF(DISP.LE.50.) OUT = $$$3(IN,FLAG,0.,100.,2.75,85.5,6.,0.)     00011490
      IF(DISP.LE.100.) OUT = $$$3(IN,FLAG,0.,100.,3.5,82.5,6.5,0.)    00011500
      IF(DISP.LE.200.) OUT = $$$3(IN,FLAG,0.,100.,4.1,79.8,7.,0.)     00011510
      IF(DISP.GT.200.) OUT =$$$3(IN,FLAG,0.,100.,4.65,77.5,7.5,0.)    00011520
      GO TO 991                                                       00011530
C                                                                     00011540
 1040 IF(DISP.LE.5.) OUT = $$$3(IN,FLAG,0.,87.5,2.2,76.5,5.,0.)       00011550
      IF(DISP.LE.20.) OUT = $$$3(IN,FLAG,0.,87.5,2.8,73.5,5.5,0.)     00011560
      IF(DISP.LE.50.) OUT = $$$3(IN,FLAG,0.,87.5,3.4,71.0,6.,0.)      00011570
      IF(DISP.LE.100.) OUT = $$$3(IN,FLAG,0.,87.5,4.05,67.,6.5,0.)    00011580
      IF(DISP.LE.200.) OUT = $$$3(IN,FLAG,0.,87.5,4.65,64.,7.,0.)     00011590
      IF(DISP.GT.200.) OUT =$$$3(IN,FLAG,0.,87.5,5.15,61.5,7.5,0.)    00011600
      GO TO 999                                                       00011610
C                                                                     00011620
  50  IF(RATE.EQ.2) GO TO 1050                                        00011630
      IF(DISP.LE.100.) OUT = $$$3(IN,FLAG,0.,100.,4.,74.,6.,0.)       00011640
      IF(DISP.GT.100.) OUT = $$$3(IN,FLAG,0.,100.,5.3,65.5,7.,0.)     00011650
      GO TO 991                                                       00011660
 1050 IF(DISP.LE.100.) OUT = $$$3(IN,FLAG,0.,87.5,4.4,59.,6.,0.)      00011670
      IF(DISP.GT.100.) OUT = $$$3(IN,FLAG,0.,87.5,5.7,50.5,7.,0.)     00011680
      GO TO 999                                                       00011690
C                                                                     00011700
  60  IF(RATE.EQ.2) GO TO 1060                                        00011710
      IF(DISP.LE.500.) OUT = $$$4(IN,FLAG,0.,100.,3.,100.,5.,90.,     00011720
     18.,0.)                                                          00011730
      IF(DISP.GT.500.) OUT = $$$4(IN,FLAG,0.,100.,4.,100.,6.,90.,     00011740
     19.,0.)                                                          00011750
      GO TO 991                                                       00011760
 1060 IF(DISP.LE.500.) OUT = $$$3(IN,FLAG,0.,60.,6.,60.,8.,0.)        00011770
      IF(DISP.GT.500.) OUT = $$$3(IN,FLAG,0.,60.,7.,60.,9.,0.)        00011780
      GO TO 999                                                       00011790
C                                                                     00011800
  80  IF(RATE.EQ.2) GO TO 1080                                        00011810
```

```
      IF(DISP.LE.5.) OUT = $$$3(IN,FLAG,0.,100.,2.,83.,5.,0.)
      IF(DISP.GT.5..AND.DISP.LE.20.) OUT = $$$3(IN,FLAG,0.,100.,
     12,67,77.,5,5,0.)
      IF(DISP.GT.20..AND.DISP.LE.50.) OUT = $$$3(IN,FLAG,0.,100.,
     13,34,72.,6.,0.)
      IF(DISP.GT.50..AND.DISP.LE.100.) OUT = $$$3(IN,FLAG,0.,100.,
     14.,65.,6.5,0.)
      IF(DISP.GT.100..AND.DISP.LE.200.) OUT = $$$3(IN,FLAG,0.,
     1100.,4,67,60.,7.,0.)
      IF(DISP.GT.200.) OUT = $$$3(IN,FLAG,0.,100.,5,34,54.,7.5,0.)
      GO TO 991
 1080 IF(DISP.LE.5.) OUT = $$$3(IN,FLAG,0.,60.,4,1,25,6,5.,0.)
      IF(DISP.GT.5..AND.DISP.LE.20.) OUT = $$$3(IN,FLAG,
     10.,60.,4,85,19,5,5,5,0.)
      IF(DISP.GT.20..AND.DISP.LE.50.) OUT = $$$3(IN,FLAG,
     10.,60.,5,58,13,0,6.,0.)
      IF(DISP.GT.50..AND.DISP.LE.100.) OUT = $$$3(IN,FLAG,0.,60.,
     16,3,6,4,6,5,0.)
      IF(DISP.GT.100..AND.DISP.LE.200.) OUT = $$$(IN,FLAG,0.,60.,
     17.,0.)
      IF(DISP.GT.200.) OUT = $$$3(IN,FLAG,0.,60.,7.,0.,7.5,0.)
      GO TO 999
  101 IF(RATE.EQ.2) GO TO 1101
      IF(DISP.LE.10.) OUT = $$$3(IN,FLAG,0.,100.,1.,96,7.5.,0.)
      IF(DISP.GT.10..AND.DISP.LE.25.) OUT = $$$3(IN,FLAG,0.,100.,
     12.,93,3,6.,0.)
      IF(DISP.GT.25.) OUT = $$$3(IN,FLAG,0.,100.,3.,90.,7.,0.)
      GO TO 991
 1101 IF(DISP.LE.10.) OUT = $$$3(IN,FLAG,0.,70.,2,45,62.,5.,0.)
      IF(DISP.GT.10..AND.DISP.LE.25.) OUT = $$$3(IN,FLAG,0.,85.,
     12,75,76.,6.,0.)
      IF(DISP.GT.25.) OUT = $$$3(IN,FLAG,0.,100.,3.,90.,6.5,0.)
      GO TO 999
C
  991 IF(RATE.EQ.1) GO TO 999
      IF(RATE.EQ.3) GO TO 993
      IF(RATE.EQ.4) GO TO 994
  993 PCDSMX = 12./DSPEED*100.
  995 IF(FLAG.EQ.0.AND.OUT.GT.PCDSMX) OUT = PCDSMX
      IF(FLAG.EQ.1.AND.IN.GT.PCDSMX) OUT = 0.
      GO TO 999
  994 PCDSMX = 5./DSPEED*100.
      GO TO 995
C
C     IF CALCULATED OUTPUT OF SEA STATE VS PERCENT DESIGN SPEED
C      IS LESS THAN ZERO ,SET VALUE EQUAL TO ZERO
C
  999 IF(OUT.LT.0.) OUT = 0.
      $PDSSS = OUT
      $SSPDS = OUT
      RETURN
      END
C PROPOS
C FINDS THE PROGRAM PROBABILITY OF SUCCESS
C
C FOR A GIVEN FLOW CHART OF GROUPS, FINDS ALL POSSIBLE PATHS
C  (I.E., SEQUENCES OF TASKS) THAT DO NOT VIOLATE TIME AND
C  FUEL MAXIMA. FINDS THE PATH PROBABILITY OF SUCCESS (PTHPOS) FOR
C  EACH PATH. COMBINES THE PATH POS'S TO PRODUCE THE PROPOS.
C
C LIMITS:
C   MAXIMUM NUMBER OF GROUPS = 40
C   MAXIMUM NUMBER OF NODES IN FLOWCHART = 50
```

```
C   MAXIMUM NUMBER OF SORTIES COMPLETED = 1000                              00000120
C   MAXIMUM NUMBER OF NODES IN A SINGLE SORTIE= 100                         00000130
C                                                                           00000140
C                                                                           00000150
C                                                                           00000160
      SUBROUTINE SPKPOS(DUMMY)                                              00000170
      IMPLICIT REAL(A-Z)                                                    00000180
      INTEGER I,J,K,PROVNO,PRGPNO,NXOVNO,NXGPNO                             00000190
      INTEGER LSOVGO,LSGPGO,BEGIN,NNODOV,LNKSGP,NUMNOS                      00000200
      INTEGER NNODE,PTR,PSHLST,NPATH,GPPLMX,POPPED,MXGPLK                   00000210
      INTEGER $PACK,$GROUP,$INST,$NODE,$POP,$TCP,$MOVNO                     00000220
      INTEGER GROUP,INST,NODE,GPDAT1,IFULST,IPHFUE,NTASK                    00000230
      INTEGER LOCATN,NODE1,NODE2,TASKNO,RATE,GROUP1                         00000240
      INTEGER IDISP,CASNUM,IDSPD,IPTHFU                                     00000250
      INTEGER NPRTED,NPTOUT,BASGRP                                          00000260
      INTEGER VISDTB,TOWDTB,DPHDTB,SSPDTB                                   00000270
      INTEGER MASTER,MASTSK,NMSTSK                                          00000280
      INTEGER NDAYS,IMRATE,IMTSKN,IMPTSK,NMIMTK                             00000290
      INTEGER COUNT,MTASK,FLAG,IM,RATE1,TASKN1                              00000300
      DATA NMSTSK/19/                                                       00000310
      DIMENSION VISDS1(3,3),VISDS2(3,3)                                     00000320
      DIMENSION SSPRBD(8),CWSPD(4),SFCENG(4),SFCCF(4),TOTSFC(4)             00000321
      DIMENSION SFCGAL(4),HPUTIL(4),FUELRT(4),ENDUR(4),RANGE(4)             00000322
      DIMENSION MOTION(4),TNRAD(4),FUELR2(4),ENG(4)                         00000323
      DIMENSION SPEED(4),CFTNAM(8),MFULRT(4)                                00000330
      DIMENSION CC(19),DF(19),LS(19),MN(19),TW(19)                          00000331
      DIMENSION NNODE(22)                                                   00000340
      DIMENSION UNAJPB(1000),PRPOS(1000),PATHTM(1000),IPTHFU(1000)          00000350
      DIMENSION DATA(20,40)                                                 00000360
      DIMENSION TPOSMX(25)                                                  00000370
      DIMENSION MINTIM(50),MINFUE(50)                                       00000380
      COMMON/PSHDWN/PTR,PSHLST(100),TIMLST(100),FUELST(100),PRBLST(100)     00000390
      COMMON/GPDATA/GPDAT1(40,2),GPDAT2(40,18)                              00000400
      COMMON/VZ/VISDIS(3,3)                                                 00000410
      COMMON/MNCOM/LENG,FUERAC,VISDTB,TOWDTB,DPHDTB                         00000430
      COMMON/PARAM/IDISP,IDSPD,CFTNAM,SSAVG,SPEED,MFULRT,TOWSPD,            00000431
     1  CC,DF,LS,MN,TW                                                     00000432
      COMMON/CHAR/LTOR,BEAM,DTOL,DRAF,SSPRBD,DECK,USELD,                    00000433
     1 FUELCP,CARGCP,TOWDSP,SURVIV,HPINST,HPPTON,HPTNKT,                    00000434
     2 CWSPD,ENG,SFCENG,SFCCF,TOTSFC,SFCGAL,HPUTIL,                         00000435
     3 FUELRT,FUELR2,ENDUR,RANGE,MOTION,TNRAD,SSPDTB                        00000436
      DATA VISDS1/.9,.1,0.,.7,.2,.1,.5,.3,.2/                               00000450
      DATA VISDS2/'VERY','GOOD','GOOD',' GOO','    ',                       00000460
     X  ' TO ','D   ','    ','FAIR'/                                        00000470
      DIMENSION OVCNMX(50,50),GPPLMX(50,50)                                 00000480
      DATA OVCNMX/2500*0./                                                  00000490
      DATA TOTPRB/0./                                                       00000500
      DATA NNODE/7,3,4,3,6,5,3,3,4,4,2,3,2,                                 00000510
     X  2,4,3,4,6/                                                          00000520
      DIMENSION MASTER(25,4),COUNT(25,4),TOTCNT(25,4)                       00000530
      DIMENSION IM(25,4),IMRATE(40),IMTSKN(40),IMPTSK(40,40)                00000540
      DIMENSION MTASK(40)                                                   00000550
      DATA MASTER/19,19,23*0, 13,14,14,                                     00000560
     X  15,18,16,17,18,18,16*0, 8,9,                                        00000570
     Y  11,10,12,20*0, 2,7,5,1,5,7,                                         00000580
     Z  5,7,3,3,5,5,2,4,4,7,3,5,7,7,5,6,7/                                  00000590
      NAMELIST/GROUP/DATA                                                   00000600
      DATA DATA/800*0./                                                     00000610
      DIMENSION TASKNM(6,25,4)                                              00000620
      DIMENSION TASKN2(6,25,2)                                              00000630
      DIMENSION TASKN3(6,25,1)                                              00000640
      DIMENSION TASKN4(6,13,1)                                              00000650
      EQUIVALENCE (TASKNM(301),TASKN2(1))                                   00000660
```

```
      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',' OBJ','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
```

```
     1 'SEAR','CH F','LEFT','     ',                                  00001300
     1 'SEIZ','E    ','     ','     ',                              00001310
     1 'SENS','OR S','EARC','H   ',                                  00001320
     1 'STAN','DBY ','     ','     ',                               00001330
     1 'STEA','M   ','     ','     ',                               00001340
     1 'TRAN','SFER',' EQU','IP  ',                                  00001350
     1 'TRAN','SPOR','T EQ','UIP ',                                  00001360
     1 'WORK',' EQU',' IPME','NT  ',                                 00001370
     1 8*0./                                                         00001380
C                                                                    00001390
C                                                                    00001400
C                                                                    00001410
C                                                                    00001420
C                                                                    00001430
C GET PATHS                                                          00001440
C                                                                    00001450
C                                                                    00001460
C                                                                    00001470
C READ CG PROGRAM AND SCENARIO NUMBER                                00001480
C   FORMAT ' CG PROGRAM= FFF'                                        00001490
C          ' SCENARIO NO.=NN'                                        00001500
      READ(13,130)PROGRM,SCENNO                                      00001510
130   FORMAT(12X,A4/14X,I2)                                          00001520
      WRITE(6,131)PROGRM,SCENNO                                      00001530
131   FORMAT('1','** SCENARIO DATA **'//1X,'CG PROGRAM=',A4/         00001540
     1  1X,'SCENARIO NO.=',I2)                                       00001550
C                                                                    00001560
C READ MAXIMUM TIME                                                  00001570
C   FORMAT ' MAXIMUM TIME=FFFF.F'                                    00001580
C      WHERE TIME IS IN HOURS                                        00001590
      READ(13,125)MXTIME                                             00001600
125   FORMAT(1X,13X,F6.1)                                            00001610
      WRITE(6,127)MXTIME                                             00001620
127   FORMAT(1X,'MAXIMUM TIME=',F6.1)                                00001630
C                                                                    00001640
C READ FRACTION OF RANGE THAT CAN BE USED                            00001650
C   FORMAT ' RANGE FRACTION=F.FF'                                    00001660
      READ(13,135)RANGFR                                             00001670
135   FORMAT(1X,15X,F4.2)                                            00001680
      WRITE(6,137)RANGFR                                             00001690
137   FORMAT(1X,'RANGE FRACTION=',F4.2)                              00001700
C                                                                    00001710
C READ NUMBER OF DAYS IN OPERATION                                   00001720
C  FORMAT ' NO. DAYS OF OPERATION=NNNN'                              00001730
      READ(13,505)NDAYS                                              00001740
505   FORMAT(23X,I4)                                                 00001750
      WRITE(6,506)NDAYS                                              00001760
506   FORMAT(' NO. DAYS OF OPERATION=',I4)                           00001770
C                                                                    00001780
C READ THE NUMBER OF IMPORTANT TASKS AND THE IMPORTANT TASKS         00001790
C FORMAT ' NUMBER OF IMPORTANT TASKS=NN'                             00001800
C FORMAT ' RTI RTI RTI.........                                      00001810
C R=RATE OF TASK,T=TASK NUMBER; USER CAN INPUT IMPORTANT             00001820
C TASKS (10/LINE) AFTER HE INPUTS THE NUMBER OF IMPORTANT            00001830
C TASKS.                                                             00001840
      READ(13,801)NMIMTK                                             00001850
801   FORMAT(27X,I2)                                                 00001860
      READ(13,507)(IMRATE(I),IMTSKN(I),I=1,NMIMTK)                   00001870
507   FORMAT((10(1X,I1,I2)))                                         00001880
      DO 520 I=1,NMIMTK                                              00001890
520   MTASK(I)=IMRATE(I)*100+IMTSKN(I)                               00001900
      WRITE(6,508)NMIMTK                                             00001910
508   FORMAT(' NUMBER OF IMPORTANT TASKS=',I2)                       00001920
```

```
      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                                                                       00002560
C READ NUMBER OF PRINTOUTS                                              00002570
C FORMAT ' NUMBER OF PRINTOUTS=NN'                                      00002580
      READ(13,140)NPTOUT                                                00002590
140   FORMAT(21X,I2)                                                    00002600
      WRITE(6,141)NPTOUT                                                00002610
141   FORMAT(' NUMBER OF PRINTOUTS=',I2)                                00002620
C READ OUTPUT FORMAT                                                    00002630
C FORMAT ' OUTPUT FORMAT=N'                                             00002640
C FLAG=1 FOR FULL OUTPUT                                                00002650
C FLAG=2 FOR PARTIAL OUTPUT:NO SORTIES PRINTED                          00002660
      READ(13,142)FLAG                                                  00002670
142   FORMAT(15X,I1)                                                    00002680
      WRITE(6,143)FLAG                                                  00002690
143   FORMAT(' OUTPUT FORMAT=',I1)                                      00002700
      REWIND 13                                                         00002701
C THIS REWIND ALLOWS USER TO RUN MODEL FOR                              00002702
C MANY CRAFT FOR THE SAME SCENARIO                                      00002703
C CONVERT FUEL(TONS) TO FUEL(GALLONS)                                   00002820
      MXFUEL=FUELCP                                                     00002821
      MXGALS=MXFUEL*335.*RANGER                                         00002830
C CALCULATE TPOS(TASK PROBABILITY OF SUCCESS)                           00002840
      DO 292 I=1,NMSTSK                                                 00002841
      TPOSMX(I)=CC(I)*DF(I)*MN(I)*LS(I)*TW(I)                           00002842
292   CONTINUE                                                          00002843
C                                                                       00003210
C                                                                       00003220
C                                                                       00003230
C INITIALIZE                                                            00003240
      NPATH=0                                                           00003250
C ZERO OUT TIME, FUEL AND TASK COUNTERS AFTER A CRAFT                   00003260
C HAS COMPLETED ALL POSSIBLE SORTIES IN A SCENARIO                      00003270
      TOTIM=0.                                                          00003280
      TOTFUE=0.                                                         00003290
      DO 423 I=1,25                                                     00003300
      DO 424 J=1,4                                                      00003310
      TOTCNT(I,J)=0.                                                    00003320
424   CONTINUE                                                          00003330
423    CONTINUE                                                         00003340
      PTHTIM=0.                                                         00003350
      PTHFUE=0.                                                         00003360
      PTHPRB=1.                                                         00003370
      PTR=0                                                             00003380
      TOTPRB=0.                                                         00003390
      DO 11 I=1,100                                                     00003400
      PSHLST(I)=0                                                       00003410
      TIMLST(I)=0.                                                      00003420
      FUELST(I)=0.                                                      00003430
      PRBLST(I)=0.                                                      00003440
11    CONTINUE                                                          00003450
      DO 12 I=1,3                                                       00003460
      DO 12 J=1,3                                                       00003470
      VISDIS(I,J)=VISDS1(I,J)                                           00003480
12    CONTINUE                                                          00003490
C                                                                       00003500
C                                                                       00003510
C GET MINIMUM FUEL AND TIME PATHS                                       00003520
      CALL $MINPH(NNODOV,OVCNMX,GPPLMX,MINTIM,MINFUE)                   00003530
C                                                                       00003540
C                                                                       00003550
C                                                                       00003560
C                                                                       00003570
C FIND PATHS (SORTIES) THROUGH THE FLOWCHART                            00003580
```

```
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
```

```
C FOUND A COMPLETE PATH                                                  00004220
42    CALL $PUSH(000002,0.,0.,OVPROB)                                    00004230
      PTHPRB=PTHPRB*OVPROB                                               00004240
      GOTO 200                                                           00004250
201   CALL $POP(POPPED,POPTIM,POPFUE,POPPRB)                             00004260
      PTHPRB=PTHPRB/POPPRB                                               00004270
44    IF($TOP(0).GE.10000)GOTO 82                                        00004280
C WENT BACK TO OVERALL NODE                                              00004290
      LSOVGO=NXOVND                                                      00004300
      GOTO 10                                                            00004310
C WENT BACK TO GROUP NODE.                                               00004320
82     PROVND=$NOVND(0)                                                  00004330
      GO TO 80                                                           00004340
C TOP IS A GROUP NODE, WANT TO POP IT                                    00004350
80    CALL $POP(POPPED,POPTIM,POPFUE,POPPRB)                             00004360
      PTHTIM=PTHTIM-POPTIM                                               00004370
      PTHFUE=PTHFUE-POPFUE                                               00004380
      PTHPRB=PTHPRB/POPPRB                                               00004390
      IF($NODF(POPPED).EQ.1)GOTO 44                                      00004400
      LSGPGO=POPPED                                                      00004410
      PRGPND=$TOP(0)                                                     00004420
      GOTO 60                                                            00004430
C AT OVERALL NODE WITH NO OVERALL NODES TO GO TO -OR-                    00004440
C WITH NOT ENOUGH TIME OR FUEL LEFT: REMOVE TOP OVERALL NODE             00004450
90    CALL $POP(POPPED,POPTIM,POPFUE,POPPRB)                             00004460
      PTHPRB=PTHPRB/POPPRB                                               00004470
      IF(POPPED.EQ.000001)GO TO 9999                                     00004480
      LSOVGO=POPPED                                                      00004490
      NXOVND=POPPED                                                      00004500
      IF($TOP(0).GE.10000)GOTO 82                                        00004510
      PROVND=$TOP(0)                                                     00004520
      GOTO 10                                                            00004530
C                                                                        00004540
C                                                                        00004550
C FOUND A PATH: FIND PATH DATA AND PRINT IT OUT                          00004560
C                                                                        00004570
200   NPATH=NPATH+1                                                      00004580
      IPHFUE=PTHFUE                                                      00004590
      UNADJP=1.                                                          00004600
      DO 209 I=1,PTR                                                     00004610
      UNADJP=UNADJP*PRBLST(I)                                            00004620
209   CONTINUE                                                           00004630
      UNAJPB(NPATH)=UNADJP                                               00004640
      TOTPRB=TOTPRB+UNADJP                                               00004650
C STORE SORTIE TIME AND FUEL IN ARRAYS                                   00004660
      PATHTM(NPATH)=PTHTIM                                               00004670
      IPTHFU(NPATH)=IPHFUE                                               00004680
      DO 3000 NPRTED=1,NPTOUT                                            00004690
C WRITE HEADING                                                          00004700
      IF(FLAG.EQ.1)WRITE(6,210)PROGRM,SCENNO                             00004710
210   FORMAT('1'////30X,A4,' SCENARIO ',I2)                              00004720
      IF(FLAG.EQ.1)WRITE(6,45)NPATH                                      00004730
45    FORMAT(1X,28X,'SORTIE NUMBER ',I4/)                                00004740
      IF(FLAG.EQ.1)WRITE(6,211)MXTIME,CFTNAM,RANGER,IDISP,               00004750
     X (VISDS2(VISDTB,I),I=1,3),IDSPD,SSAVG,FUFRAC                       00004760
211   FORMAT(/13X,'  OPERATIONAL REQUIREMENTS:',11X,                     00004770
     1   'SELECTED CRAFT:'//13X,'  MAXIMUM DURATION ',F6.1,' HOURS',     00004780
     2   7X,8A4/13X,                                                     00004790
     3   '  RANGE FRACTION ',F4.2,17X,'DISPLACEMENT',I5,' TONS'/13X,     00004800
     4   '  VISIBILITY ',3A4,13X,                                        00004810
     5    'DESIGN SPEED ',I2,' KNOTS'/13X,                              00004820
     6   '  AVERAGE SEA STATE ',F3.1,15X,'FUEL FRACTION ',F4.2)          00004830
      IF(FLAG.EQ.1)WRITE(6,202)                                          00004840
```

```
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
```

```
      IF(FLAG.EQ.1)WRITE(6,206)(GRPNM(J,BASGRP),J=1,4),LOCATN                00005480
206   FORMAT(1X,10X,4A4,17X,I6)                                              00005490
      NODE2=1                                                                00005500
      GOTO 47                                                                00005510
204   IF(FLAG.EQ.1)WRITE(6,207)LOCATN                                        00005520
207   FORMAT(1X,43X,I6)                                                      00005530
47    CONTINUE                                                               00005540
C CALCULATE PATH PROBABILITY OF SUCCESS                                      00005550
      PTHPOS=CCMIN*DFMIN*MNMIN*LSMIN*TWMIN                                   00005560
      IF(FLAG.EQ.1)WRITE(6,208)PTHTIM                                        00005570
208   FORMAT(//1X,10X,'TIME TO COMPLETE SORTIE (HRS)',12X,F5.1)              00005580
      IF(FLAG.EQ.1)WRITE(6,401)IPHFUE                                        00005590
401   FORMAT(/1X,10X,'FUEL CONSUMED IN SORTIE (GALS)',18X,I6)                00005600
      IF(FLAG.EQ.1)WRITE(6,402)PTHPOS                                        00005610
402   FORMAT(//1X,                                                           00005620
     1 17X,'SORTIE PROBABILITY OF SUCCESS',22X,                              00005630
     1 F6.4)                                                                 00005640
      IF(FLAG.EQ.1)WRITE(6,403)UNAJPB(NPATH)                                 00005650
403   FORMAT(/1X,                                                            00005660
     1 17X,'SORTIE FREQUENCY OF OCCURRENCE',21X,F6.4)                        00005670
      PT=PTHPOS*UNAJPB(NPATH)                                                00005680
      TOTIM=TOTIM+PTHTIM*PT                                                  00005690
      TOTFUE=TOTFUE+PT*PTHFUE                                                00005700
C LOOP TO INCREMENT TOTAL COUNTER FOR TASKS                                  00005710
      DO 490 I=1,25                                                          00005720
      DO 491 J=1,4                                                           00005730
      TOTCNT(I,J)=TOTCNT(I,J)+PT*COUNT(I,J)                                  00005740
491   CONTINUE                                                               00005750
490   CONTINUE                                                               00005760
3000  CONTINUE                                                               00005770
      PHPOS(NPATH)=PTHPOS                                                    00005780
      GOTO 201                                                               00005790
C                                                                            00005800
C                                                                            00005810
C                                                                            00005820
C PRINT SORTIE SUMMARY                                                       00005830
C                                                                            00005840
9999  CONTINUE                                                               00005850
      DO 3001 NPRTED=1,NPTOUT                                                00005860
      WRITE(6,9992)PROGRM,SCENNO                                             00005870
9992  FORMAT('1'////                                                         00005880
     X 18X,'************** SORTIE SUMMARY **************'//                 00005890
     1   30X,A4,' SCENARIO ',I2/)                                           00005900
      WRITE(6,211)MXTIME,CFTNAM,RANGFR,IDISP,                                00005910
     X (VISDS2(VISDTB,I),I=1,3),IDSPD,SSAVG,FUFRAC                          00005920
      IF(NPATH.EQ.0)GOTO 9994                                                00005930
      WRITE(6,9997)TOTPRB                                                    00005940
9997  FORMAT(//                                                              00005950
     1  1X,17X,5X,'FRACTION OF SCENARIO COMPLETED  ',F6.4//                  00005960
     1  13X,'SORTIE',2X,'SORTIE',2X,'SORTIE',2X,'FREQUENCY',                 00005970
     1  8X,'SORTIE',8X,'SORTIE'/15X,'NO.',4X,'TIME',4X,'FUEL',               00005980
     1  7X,'OF',8X,'PROBABILITY',4X,'SUCCESSFUL'/22X,'(HRS)',                00005990
     1  2X,'(GALS)',2X,'OCCURRENCE',4X,'OF',2X,'SUCCESS',                    00006000
     1  4X,'OCCURRENCE'//)                                                   00006010
      PROPOS=0.                                                              00006020
      DO 9996 I=1,NPATH                                                      00006030
      ADJPRB=UNAJPB(I)                                                       00006040
      PTHPOS=PHPOS(I)                                                        00006050
      CONTRB=PTHPOS*ADJPRB                                                   00006060
      PROPOS=PROPOS+CONTRB                                                   00006070
      WRITE(6,9998)I,PATHTM(I),IPTHFU(I),ADJPRB,PTHPOS,CONTRB                00006080
9998  FORMAT(13X,I4,4X,F5.1,3X,I6,5X,F6.4,8X,F6.4,8X,F6.4)                   00006090
9996  CONTINUE                                                               00006100
```

```
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,IDISP,                           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
```

```
      IF(TOTCNT(15,4).GT.0.)WRITE(6,7018)TOTCNT(15,4)                    00006740
 7018 FORMAT(1X,22X,'RPE',6X,F5.2,5X,'RESCUE PEOPLE')                    00006750
      IF(TOTCNT(16,4).GT.0.)WRITE(6,7019)TOTCNT(16,4)                    00006760
 7019 FORMAT(1X,22X,'RSB',6X,F5.2,5X,'RETRIEVE SMALL BOAT')              00006770
      IF(TOTCNT(17,4).GT.0.)WRITE(6,7070)TOTCNT(17,4)                    00006780
 7070 FORMAT(1X,22X,'SSI',6X,F5.2,5X,                                    00006790
     1'STAKEOUT SPECIAL INTEREST VESSEL')                                00006800
      IF(TOTCNT(18,4).GT.0.)WRITE(6,7021)TOTCNT(18,4)                    00006810
 7021 FORMAT(1X,22X,'SZE',6X,F5.2,5X,'SEIZE')                            00006820
      IF(TOTCNT(19,4).GT.0.)WRITE(6,7022)TOTCNT(19,4)                    00006830
 7022 FORMAT(1X,22X,'TWS',6X,F5.2,5X,'TAKE WATER SAMPLE')                00006840
      IF(TOTCNT(20,4).GT.0.)WRITE(6,7023)TOTCNT(20,4)                    00006850
 7023 FORMAT(1X,22X,'ULG',6X,F5.2,5X,'UNLOAD EQUIPMENT')                 00006860
      IF(TOTCNT(21,4).GT.0.)WRITE(6,7026)TOTCNT(21,4)                    00006870
 7026 FORMAT(1X,22X,'WQB',6X,F5.2,5X,                                    00006880
     1'WORK EQUIPMENT FROM SMALL BOAT')                                  00006890
      IF(TOTCNT(22,4).GT.0.)WRITE(6,7024)TOTCNT(22,4)                    00006900
 7024 FORMAT(1X,22X,'WQD',6X,F5.2,5X,'WORK EQUIPMENT & DRIFT')           00006910
      IF(TOTCNT(23,4).GT.0.)WRITE(6,7025)TOTCNT(23,4)                    00006920
 7025 FORMAT(1X,22X,'WQF',6X,F5.2,5X,                                    00006930
     1'WORK EQUIPMENT & FIXED POSITION')                                 00006940
      DO 492 I=1,25                                                      00006950
      IF(TOTCNT(I,4).GT.0.)GO TO 494                                     00006960
 492  CONTINUE                                                           00006970
      WRITE(6,7027)                                                      00006980
 7027 FORMAT(1X,22X,'NO TASKS')                                          00006990
C                                                                       00007000
 494  WRITE(6,8002)                                                      00007010
 8002 FORMAT(/1X,18X,'REDUCED SPEED:')                                   00007020
      IF(TOTCNT(1,3).GT.0.)WRITE(6,8004)TOTCNT(1,3)                      00007030
 8004 FORMAT(23X,'SDU',6X,F5.2,5X,'SEARCH FOR DISTRESSED UNIT: FOUND')   00007040
      IF(TOTCNT(20,1).GT.0.)WRITE(6,3033)TOTCNT(20,1)                    00007050
 3033 FORMAT(23X,'SDU',6X,F5.2,5X,'SEARCH FOR DISTRESSED UNIT: FAILED')  00007060
      IF(TOTCNT(2,3).GT.0.)WRITE(6,8003)TOTCNT(2,3)                      00007070
 8003 FORMAT(1X,22X,'SES',6X,F5.2,5X,'SLOW ESCORT')                      00007080
      IF(TOTCNT(3,3).GT.0.)WRITE(6,8005)TOTCNT(3,3)                      00007090
 8005 FORMAT(1X,22X,'SPE',6X,F5.2,5X,'SEARCH FOR PEOPLE: FOUND')         00007100
      IF(TOTCNT(22,1).GT.0.)WRITE(6,3031)TOTCNT(22,1)                    00007110
 3031 FORMAT(1X,22X,'SPE',6X,F5.2,5X,'SEARCH FOR PEOPLE: FAILED')        00007120
      IF(TOTCNT(4,3).GT.0.)WRITE(6,8006)TOTCNT(4,3)                      00007130
 8006 FORMAT(1X,22X,'SPT',6X,F5.2,5X,'SLOW PATROL')                      00007140
      IF(TOTCNT(5,3).GT.0.)WRITE(6,8007)TOTCNT(5,3)                      00007150
 8007 FORMAT(1X,22X,'TOW',6X,F5.2,5X,'TOW')                              00007160
      DO 493 I=1,5                                                       00007170
      IF(TOTCNT(I,3).GT.0.)GO TO 495                                     00007180
 493  CONTINUE                                                           00007190
      WRITE(6,7027)                                                      00007200
C                                                                       00007210
 495  WRITE(6,8008)                                                      00007220
 8008 FORMAT(/1X,18X,'CRUISE SPEED:')                                    00007230
      IF(TOTCNT(1,2).GT.0.)WRITE(6,8080)TOTCNT(1,2)                      00007240
 8080 FORMAT(1X,22X,'ESC',6X,F5.2,5X,'ESCORT')                           00007250
      IF(TOTCNT(2,2).GT.0.)WRITE(6,8009)TOTCNT(2,2)                      00007260
 8009 FORMAT(1X,22X,'IDC',6X,F5.2,5X,'IDENTIFY CRAFT')                   00007270
      IF(TOTCNT(3,2).GT.0.)WRITE(6,8010)TOTCNT(3,2)                      00007280
 8010 FORMAT(1X,22X,'IDF',6X,F5.2,5X,'IDENTIFY FLEET')                   00007290
      IF(TOTCNT(4,2).GT.0.)WRITE(6,8011)TOTCNT(4,2)                      00007300
 8011 FORMAT(1X,22X,'PAT',6X,F5.2,5X,'PATROL')                           00007310
      IF(TOTCNT(5,2).GT.0.)WRITE(6,8035)TOTCNT(5,2)                      00007320
 8035 FORMAT(1X,22X,'SFL',6X,F5.2,5X,'SEARCH FOR FLEET')                 00007330
      IF(TOTCNT(6,2).GT.0.)WRITE(6,8014)TOTCNT(6,2)                      00007340
 8014 FORMAT(1X,22X,'SSH',6X,F5.2,5X,'SEARCH FOR SHIP: FOUND')           00007350
      IF(TOTCNT(25,1).GT.0.)WRITE(6,3032)TOTCNT(25,1)                    00007360
```

```
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,RANGFR,IDISP,                            00007670
     1 (VISDS2(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
```

```
2008 FORMAT(1X,22X,'LEQ',6X,I5,5X,'LOAD EQUIPMENT')                    00008000
     IF(IM(7,4).GE.0..AND.IMPTSK(7,4).EQ.1)WRITE(6,2009)IM(7,4)        00008010
2009 FORMAT(1X,22X,'LOI',6X,I5,5X,'LOITER')                            00008020
     IF(IM(8,4).GE.0..AND.IMPTSK(8,4).EQ.1)WRITE(6,2010)IM(8,4)        00008030
2010 FORMAT(1X,22X,'LSB',6X,I5,5X,'LAUNCH SMALL BOAT')                 00008040
     IF(IM(9,4).GE.0..AND.IMPTSK(9,4).EQ.1)WRITE(6,2011)IM(9,4)        00008050
2011 FORMAT(1X,22X,'MAC',6X,I5,5X,'MONITOR ACTIVITIES')                00008060
     IF(IM(10,4).GE.0..AND.IMPTSK(10,4).EQ.1)WRITE(6,2012)IM(10,4)     00008070
2012 FORMAT(1X,22X,'MOS',6X,I5,5X,'MONITOR OIL SPILL')                 00008080
     IF(IM(11,4).GE.0..AND.IMPTSK(11,4).EQ.1)WRITE(6,2013)IM(11,4)     00008090
2013 FORMAT(1X,22X,'OBA',6X,I5,5X,'ON BOARD ASSISTANCE')               00008100
     IF(IM(12,4).GE.0..AND.IMPTSK(12,4).EQ.1)WRITE(6,2015)IM(12,4)     00008110
2015 FORMAT(1X,22X,'OSC',6X,I5,5X,'ON SCENE COMMANDER(GENERAL)')       00008120
     IF(IM(13,4).GE.0..AND.IMPTSK(13,4).EQ.1)WRITE(6,2016) IM(13,4)    00008130
2016 FORMAT(1X,22X,'RBP',6X,I5,5X,'RETRIEVE BOARDING PARTY')           00008140
     IF(IM(14,4).GE.0..AND.IMPTSK(14,4).EQ.1)WRITE(6,2017)IM(14,4)     00008150
2017 FORMAT(1X,22X,'ROB',6X,I5,5X,'RETRIEVE OBJECTS')                  00008160
     IF(IM(15,4).GE.0..AND.IMPTSK(15,4).EQ.1)WRITE(6,2018)IM(15,4)     00008170
2018 FORMAT(1X,22X,'RPE',6X,I5,5X,'RESCUE PEOPLE')                     00008180
     IF(IM(16,4).GE.0..AND.IMPTSK(16,4).EQ.1)WRITE(6,2019)IM(16,4)     00008190
2019 FORMAT(1X,22X,'RSB',6X,I5,5X,'RETRIEVE SMALL BOAT')               00008200
     IF(IM(17,4).GE.0..AND.IMPTSK(17,4).EQ.1)WRITE(6,2020)IM(17,4)     00008210
2020 FORMAT(1X,22X,'SSI',6X,I5,5X,                                     00008220
    1'STAKEOUT SPECIAL INTEREST VESSEL')                               00008230
     IF(IM(18,4).GE.0..AND.IMPTSK(18,4).EQ.1)WRITE(6,2021)IM(18,4)     00008240
2021 FORMAT(1X,22X,'SZE',6X,I5,5X,'SEIZE')                             00008250
     IF(IM(19,4).GE.0..AND.IMPTSK(19,4).EQ.1)WRITE(6,2022)IM(19,4)     00008260
2022 FORMAT(1X,22X,'TWS',6X,I5,5X,'TAKE WATER SAMPLE')                 00008270
     IF(IM(20,4).GE.0..AND.IMPTSK(20,4).EQ.1)WRITE(6,2023)IM(20,4)     00008280
2023 FORMAT(1X,22X,'ULQ',6X,I5,5X,'UNLOAD EQUIPMENT')                  00008290
     IF(IM(21,4).GE.0..AND.IMPTSK(21,4).EQ.1)WRITE(6,2026)IM(21,4)     00008300
2026 FORMAT(1X,22X,'WQB',6X,I5,5X,                                     00008310
    1'WORK EQUIPMENT FROM SMALL BOAT')                                 00008320
     IF(IM(22,4).GE.0..AND.IMPTSK(22,4).EQ.1)WRITE(6,2024)IM(22,4)     00008330
2024 FORMAT(1X,22X,'WQD',6X,I5,5X,'WORK EQUIPMENT @ DRIFT')            00008340
     IF(IM(23,4).GE.0..AND.IMPTSK(23,4).EQ.1)WRITE(6,2025)IM(23,4)     00008350
2025 FORMAT(1X,22X,'WQF',6X,I5,5X,                                     00008360
    1'WORK EQUIPMENT @ FIXED POSITION')                                00008370
     DO 692 I=1,25                                                     00008380
     IF(IM(I,4).GE.0..AND.IMPTSK(I,4).EQ.1)GO TO 694                   00008390
692  CONTINUE                                                          00008400
     WRITE(6,2027)                                                     00008410
2027 FORMAT(1X,22X,'NO IMPORTANT TASKS SPECIFIED')                     00008420
C                                                                      00008430
694  WRITE(6,3002)                                                     00008440
3002 FORMAT(/1X,18X,'REDUCED SPEED:')                                  00008450
     IF(IM(1,3).GE.0..AND.IMPTSK(1,3).EQ.1)WRITE(6,3004)IM(1,3),       00008460
    1 IM(20,1)                                                         00008461
3004 FORMAT(1X,22X,'SDU',6X,I5,5X,'SEARCH FOR DISTRESSED UNIT: FOUND'/ 00008470
    1 1X,22X,'SDU',6X,I5,5X,'SEARCH FOR DISTRESSED UNIT: FAILED')      00008471
     IF(IM(2,3).GE.0..AND.IMPTSK(2,3).EQ.1)WRITE(6,3003)IM(2,3)        00008480
3003 FORMAT(1X,22X,'SES',6X,I5,5X,'SLOW ESCORT')                       00008490
     IF(IM(3,3).GE.0..AND.IMPTSK(3,3).EQ.1)WRITE(6,3005)IM(3,3),       00008500
    X IM(22,1)                                                         00008501
3005 FORMAT(1X,22X,'SPE',6X,I5,5X,'SEARCH FOR PEOPLE: FOUND'/          00008510
    X 1X,22X,'SPF',6X,I5,5X,'SEARCH FOR PEOPLE: FAILED')              00008511
     IF(IM(4,3).GE.0..AND.IMPTSK(4,3).EQ.1)WRITE(6,3006)IM(4,3)        00008520
3006 FORMAT(1X,22X,'SPT',6X,I5,5X,'SLOW PATROL')                       00008530
     IF(IM(5,3).GE.0..AND.IMPTSK(5,3).EQ.1)WRITE(6,3007)IM(5,3)        00008540
3007 FORMAT(1X,22X,'TOW',6X,I5,5X,'TOW')                               00008550
     DO 693 I=1,5                                                      00008560
     IF(IM(I,3).GE.0..AND.IMPTSK(I,3).EQ.1)GO TO 695                   00008570
693  CONTINUE                                                          00008580
```

```
      WRITE(6,2027)
C
695   WRITE(6,3008)
 3008 FORMAT(/1X,18X,'CRUISE SPEED:')
      IF(IM(1,2).GE.0..AND.IMPTSK(1,2).EQ.1)WRITE(6,3030)IM(1,2)
 3030 FORMAT(1X,22X,'ESC',6X,I5,5X,'ESCORT')
      IF(IM(2,2).GE.0..AND.IMPTSK(2,2).EQ.1)WRITE(6,3009)IM(2,2)
 3009 FORMAT(1X,22X,'IDC',6X,I5,5X,'IDENTIFY CRAFT')
      IF(IM(3,2).GE.0..AND.IMPTSK(3,2).EQ.1)WRITE(6,3010)IM(3,2)
 3010 FORMAT(1X,22X,'IDF',6X,I5,5X,'IDENTIFY FLEET')
      IF(IM(4,2).GE.0..AND.IMPTSK(4,2).EQ.1)WRITE(6,3011)IM(4,2)
 3011 FORMAT(1X,22X,'PAT',6X,I5,5X,'PATROL')
      IF(IM(5,2).GE.0..AND.IMPTSK(5,2).EQ.1)WRITE(6,3035)IM(5,2)
 3035 FORMAT(1X,22X,'SFL',6X,I5,5X,'SEARCH FOR FLEET')
      IF(IM(6,2).GE.0..AND.IMPTSK(6,2).EQ.1)WRITE(6,3014)IM(6,2),
     X IM(25,1)
 3014 FORMAT(1X,22X,'SSH',6X,I5,5X,'SEARCH FOR SHIP: FOUND'/
     X 1X,22X,'SSH',6X,I5,5X,'SEARCH FOR SHIP: FAILED')
      IF(IM(7,2).GE.0..AND.IMPTSK(7,2).EQ.1)WRITE(6,3015) IM(7,2)
 3015 FORMAT(1X,22X,'TEQ',6X,I5,5X,'TRANSPORT EQUIPMENT')
      IF(IM(8,2).GE.0..AND.IMPTSK(8,2).EQ.1)WRITE(6,3016)IM(8,2)
 3016 FORMAT(1X,22X,'TPE',6X,I5,5X,'TRANSPORT PEOPLE')
      IF(IM(9,2).GE.0..AND.IMPTSK(9,2).EQ.1)WRITE(6,3017)IM(9,2)
 3017 FORMAT(1X,22X,'TRA',6X,I5,5X,'TRANSIT')
      DO 696 I=1,9
      IF(IM(I,2).GE.0..AND.IMPTSK(I,2).EQ.1)GO TO 697
696   CONTINUE
      WRITE(6,2027)
C
697   WRITE(6,3018)
 3018 FORMAT(/1X,18X,'FLANK SPEED:')
      IF(IM(1,1).GE.0..AND.IMPTSK(1,1).EQ.1)WRITE(6,3019)IM(1,1)
 3019 FORMAT(1X,22X,'DSH',6X,I5,5X,'DASH')
      IF(IM(2,1).GE.0..AND.IMPTSK(2,1).EQ.1)WRITE(6,3020)IM(2,1)
 3020 FORMAT(1X,22X,'INT',6X,I5,5X,'INTERDICT')
      DO 701 I=1,2
      IF(IM(I,1).GE.0..AND.IMPTSK(I,1).EQ.1)GO TO 702
701   CONTINUE
      WRITE(6,2027)
702   CONTINUE
481   GOTO 9990
9994  WRITE(6,9993)
9993  FORMAT(//
     1   1X,23X,'NO SORTIES CAN BE COMPLETED')
C
9990  CONTINUE
3001  CONTINUE
C
4999  CONTINUE
      RETURN
      END
C
C PUSHDOWN LIST SUBROUTINES
C
C $PUSH
C
C PUT A NUMBER ON THE PUSHDOWN LIST
      SUBROUTINE $PUSH(ENTRY,LKTIME,LKFLEET,LKPROF)
      IMPLICIT REAL(A-Z)
      INTEGER PSHLST,PTR,I,ENTRY
      COMMON/PSHDWN/PTR,PSHLST(100),TIMLST(100),FLTLST(100),PRBLST(100)
      PTR=PTR+1
      PSHLST(PTR)=ENTRY
```

```
      TIMLST(PTR)=LKTIME                                                00000130
      FUELST(PTR)=LKFUEL                                                00000140
      PRBLST(PTR)=LKPROB                                                00000150
C     IF(PTR/5)*5.EQ.PTR)WRITE(14,8888)PTR,(PSHLST(I),I=1,PTR)          00000160
8888  FORMAT(1X,'P=',I3,5X,8(I6,1X)/13(1X,I6))                          00000170
C     WRITE(6,8801)PTR,(TIMLST(I),I=1,PTR)                              00000180
C     WRITE(6,8801)PTR,(FUELST(I),I=1,PTR)                              00000190
8801  FORMAT(1X,'**',I3,5X,8(F6.1,1X)/10(1X,F6.1))                      00000200
      RETURN                                                            00000210
      END                                                               00000220
C                                                                       00000230
C $POP                                                                  00000240
C                                                                       00000250
C POP A NUMBER OFF THE PUSHDOWN LIST                                    00000260
C                                                                       00000270
      SUBROUTINE $POP(POPPED,POPTIM,POPFUE,POPPRB)                      00000280
      IMPLICIT REAL(A-Z)                                                00000290
      INTEGER PSHLST,PTR,POPPED                                         00000300
      COMMON/PSHDWN/PTR,PSHLST(100),TIMLST(100),FUELST(100),PRBLST(100) 00000310
      IF(PTR.LE.0)GO TO 9001                                            00000320
      POPPED=PSHLST(PTR)                                                00000330
      PSHLST(PTR)=0                                                     00000340
      POPTIM=TIMLST(PTR)                                                00000350
      TIMLST(PTR)=0.                                                    00000360
      POPFUE=FUELST(PTR)                                                00000370
      FUELST(PTR)=0.                                                    00000380
      POPPRB=PRBLST(PTR)                                                00000390
      PRBLST(PTR)=0.                                                    00000400
      PTR=PTR-1                                                         00000410
      RETURN                                                            00000420
9001  POPPED=-1                                                         00000430
      PTR=0                                                             00000440
      RETURN                                                            00000450
      END                                                               00000460
C                                                                       00000470
C $TOP                                                                  00000480
C                                                                       00000490
C READ TOP NUMBER FROM PUSHDOWN LIST                                    00000500
      INTEGER FUNCTION $TOP(DUMMY)                                      00000510
      IMPLICIT INTEGER(A-Z)                                             00000520
      COMMON/PSHDWN/PTR,PSHLST(100)                                     00000530
      $TOP=PSHLST(PTR)                                                  00000540
      RETURN                                                            00000550
      END                                                               00000560
C $PACK                                                                 00000570
C PACKS THE TWO-DIGIT GROUP,INSTANCE, AND NODE INTO ONE                 00000580
C SIX-DIGIT NUMBER OF THE FORM: 'GGIINN'.                               00000590
C                                                                       00000600
      INTEGER FUNCTION $PACK(GROUP,INST,NODE)                           00000610
      IMPLICIT INTEGER(A-Z)                                             00000620
      $PACK=10000*GROUP+100*INST+NODE                                   00000630
      RETURN                                                            00000640
      END                                                               00000650
C                                                                       00000660
C $GROUP                                                                00000670
C                                                                       00000680
C                                                                       00000690
C                                                                       00000700
      INTEGER FUNCTION $GROUP(Z)                                        00000710
      IMPLICIT INTEGER(A-Z)                                             00000720
      $GROUP=Z/10000                                                    00000730
      RETURN                                                            00000740
      END                                                               00000750
```

```
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.10000)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
```

```
C LINK DATA                                                              00001360
C                                                                        00001370
C FOR A GROUP LINK: RETURNS PROBABILITY, TIME, FUEL USED.                00001380
C                                                                        00001390
      SUBROUTINE $LKDAT(GROUP,INST,NODE1,NODE2,LKPROB,LKTIME,LKFUEL)     00001400
      IMPLICIT REAL(A-Z)                                                 00001410
      INTEGER I,J,K                                                      00001420
      INTEGER NODE1,NODE2,GROUP,INST,ROW,GPDAT1,GROUP1                   00001430
      INTEGER NSEAR1,NSEAR2,IDISP,IDSPD                                  00001440
      DIMENSION GP1PB(7,7),GP2PB(3,3),GP3PB(4,4),GP4PB(3,3)              00001450
      DIMENSION GP7PB(3,3),GP8PB(3,3),GP5PB(6,6),GP6PB(5,5)              00001460
      DIMENSION GP11PB(2,2),GP12PB(3,3),GP9PB(4,4),GP10PB(4,4)           00001470
      DIMENSION GP15PB(4,4),GP16PB(3,3),GP13PB(2,2),GP14PB(2,2)          00001480
      DIMENSION GP17PB(4,4),GP18PB(6,6),GP90PB(9,9),GP93PB(9,9)          00001490
      DIMENSION SPEED(4),CFTNAM(8),MFULRT(4)                             00001491
      COMMON/GPDATA/GPDAT1(40,2),GPDAT2(40,18)                           00001500
      COMMON/PARAM/IDISP,IDSPD,CFTNAM,SSAVG,SPEED,MFULRT,TOWSPD          00001501
C                                                                        00001510
C GET ROW OF GROUP AND INSTANCE                                          00001520
      GROUP1=GROUP                                                       00001530
      IF(GROUP.EQ.90)GROUP1=10                                           00001540
      IF(GROUP.EQ.93)GROUP1=13                                           00001550
      DO 100 ROW=1,100                                                   00001560
      IF(GPDAT1(ROW,1).EQ.GROUP1.AND. GPDAT1(ROW,2).EQ.INST)GOTO 200     00001570
100   CONTINUE                                                           00001580
C                                                                        00001590
200   LKTIME=0.                                                          00001600
      LKFUEL=0.                                                          00001610
      IF(GROUP.EQ.90)GO TO 90                                            00001620
      IF(GROUP.EQ.93)GO TO 93                                            00001630
      GOTO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18),GROUP          00001640
C                                                                        00001650
C                                                                        00001660
C                                                                        00001670
C 1. ASSIST GROUP                                                        00001680
C                                                                        00001690
1     CONTINUE                                                           00001700
      DATA GP1PB/7*0.,.92,.2*0.,.2*1.,.0,.1,.91,.6*0.,                   00001710
     X2*0.,.1,.4*0.,.93,.6*0.,.94,.6*0.,.5*0.,.1,.0./                    00001720
      GP1PB(1,2)=GPDAT2(ROW,2)                                           00001730
      GP1PB(1,3)=GPDAT2(ROW,1)                                           00001740
      GP1PB(1,5)=GPDAT2(ROW,3)                                           00001750
      GP1PB(1,6)=GPDAT2(ROW,4)                                           00001760
      LKPROB=GP1PB(NODE1,NODE2)                                          00001770
      IF(LKPROB.EQ.0.)RETURN                                             00001780
112   IF(NODE1.NE.1.OR.NODE2.NE.2) GOTO 113                              00001790
      T4=GPDAT2(ROW,6)                                                   00001800
      LKTIME=T4*$SKTIM(1)                                                00001810
      LKFUEL=LKTIME*MFULRT(4)                                            00001820
      RETURN                                                             00001830
113   IF(NODE1.NE.1.OR.NODE2.NE.3) GOTO 115                              00001840
      T1=GPDAT2(ROW,5)                                                   00001850
      LKTIME=T1*$SKTIM(2)*$MNTIM(2)                                      00001860
      LKFUEL=LKTIME*MFULRT(4)                                            00001870
      RETURN                                                             00001880
115   IF(NODE1.NE.1.OR.NODE2.NE.5) GOTO 116                              00001890
      T5=GPDAT2(ROW,9)                                                   00001900
      LKTIME=T5*$SKTIM(4)*$MNTIM(4)                                      00001910
      LKFUEL=LKTIME*MFULRT(4)                                            00001920
      RETURN                                                             00001930
116   IF(NODE1.NE.1.OR.NODE2.NE.6) GOTO 134                              00001940
      T6=GPDAT2(ROW,10)                                                  00001950
      LKTIME=T6*$SKTIM(7)*$MNTIM(7)                                      00001960
```

```
      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*(..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
```

```
      T1=GPDAT2(ROW,3)                                                  00002600
      LKTIME=T1*$SKTIM(2)*$MNTIM(2)                                     00002610
      LKFUEL=LKTIME*MFULRT(4)                                           00002620
      RETURN                                                            00002630
334   IF(NODE1.NE.3 .OR. NODE2.NE.4)GO TO 342                           00002640
      T2=GPDAT2(ROW,4)                                                  00002650
      LKTIME=T2                                                         00002660
      LKFUEL=LKTIME*MFULRT(4)                                           00002670
      RETURN                                                            00002680
342   IF(NODE1.GE.4 .OR. NODE2.NE.2)GO TO 399                           00002690
      T3=GPDAT2(ROW,5)                                                  00002700
      LKTIME=T3*$SKTIM(2)*$MNTIM(2)                                     00002710
      LKFUEL=LKTIME*MFULRT(4)                                           00002720
399   RETURN                                                            00002730
C                                                                       00002740
C                                                                       00002750
C                                                                       00002760
C 4. IDENTIFY GROUP                                                     00002770
C                                                                       00002780
4     CONTINUE                                                          00002790
      DATA GP4PB/3*0.,.91,.0,.1,.92,.2*0./                              00002800
      GP4PB(1,2)=GPDAT2(ROW,1)                                          00002810
      GP4PB(1,3)=GPDAT2(ROW,2)                                          00002820
      LKPROB=GP4PB(NODE1,NODE2)                                         00002830
      IF(LKPROB.EQ.0.)RETURN                                            00002840
412   IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 413                           00002850
      TSHIP1=GPDAT2(ROW,3)                                              00002860
      DSHIP1=GPDAT2(ROW,4)                                              00002870
      NSHIP1=GPDAT2(ROW,5)                                              00002880
      IDTIME=NSHIP1*TSHIP1*$VZTIM(14)                                   00002890
      TRAVEL=(DSHIP1-1.)*TSHIP1/SPEED(2)                                00002900
      LKTIME=IDTIME+TRAVEL                                              00002910
      LKFUEL=LKTIME*MFULRT(2)                                           00002920
      RETURN                                                            00002930
413   IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 499                           00002940
      T2=GPDAT2(ROW,6)                                                  00002950
      LKTIME=T2*$SKTIM(14)*$VZTIM(14)                                   00002960
      LKFUEL=LKTIME*MFULRT(2)                                           00002970
      RETURN                                                            00002980
499   RETURN                                                            00002990
C                                                                       00003000
C                                                                       00003010
C                                                                       00003020
C 5. INSPECT GROUP                                                      00003030
C                                                                       00003040
5     CONTINUE                                                          00003050
      DATA GP5PB/6*0.,.3*0.,.1.,.0.,.1.,.91.,.5*0.,.2*0.,.1.,.3*0.,.   00003060
     X   92.,.5*0.,.4*0.,.1.,.0./                                       00003070
      GP5PB(1,3)=GPDAT2(ROW,1)                                          00003080
      GP5PB(1,5)=GPDAT2(ROW,2)                                          00003090
      LKPROB=GP5PB(NODE1,NODE2)                                         00003100
      IF(LKPROB.EQ.0.)RETURN                                            00003110
513   IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 515                           00003120
      T1=GPDAT2(ROW,3)                                                  00003130
      LKTIME=T1*$SKTIM(7)*$MNTIM(7)                                     00003140
      LKFUEL=LKTIME*MFULRT(4)                                           00003150
      RETURN                                                            00003160
515   IF(NODE1.NE.1 .OR. NODE2.NE.5)GO TO 534                           00003170
      T4=GPDAT2(ROW,6)                                                  00003180
      LKTIME=T4*$SKTIM(2)*$MNTIM(2)                                     00003190
      LKFUEL=LKTIME*MFULRT(4)                                           00003200
      RETURN                                                            00003210
534   IF(NODE1.NE.3 .OR. NODE2.NE.4)GO TO 542                           00003220
```

```
      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
```

```
      GP7PB(1,3)=GPDAT2(ROW,2)                                          00003860
      LKPROB=GP7PB(NODE1,NODE2)                                         00003870
      IF(LKPROB.EQ.0.)RETURN                                            00003880
712   IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 713                           00003890
      D1=GPDAT2(ROW,3)                                                  00003900
      V1=GPDAT2(ROW,4)                                                  00003910
      LKTIME=D1/V1                                                      00003920
      LKFUEL=LKTIME*MFULRT(3)                                           00003930
      RETURN                                                            00003940
713   IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 799                           00003950
      D2=GPDAT2(ROW,5)                                                  00003960
      LKTIME=D2/SPEED(2)                                                00003970
      LKFUEL=LKTIME*MFULRT(2)                                           00003980
      RETURN                                                            00003990
799   RETURN                                                            00004000
C                                                                       00004010
C                                                                       00004020
C                                                                       00004030
C 8. RESCUE GROUP                                                       00004040
C                                                                       00004050
8     CONTINUE                                                          00004060
      DATA GP8PB/3*0.,  91.,0.,1.,  92.,2*0./                           00004070
      GP8PB(1,2)=GPDAT2(ROW,1)                                          00004080
      GP8PB(1,3)=GPDAT2(ROW,2)                                          00004090
      LKPROB=GP8PB(NODE1,NODE2)                                         00004100
      IF(LKPROB.EQ.0.)RETURN                                            00004110
812   IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 813                           00004120
      T1=GPDAT2(ROW,3)                                                  00004130
      LKTIME=T1*$SKTIM(4)+$MNTIM(4)                                     00004140
      LKFUEL=LKTIME*MFULRT(4)                                           00004150
      RETURN                                                            00004160
813   IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 899                           00004170
      T2=GPDAT2(ROW,4)                                                  00004180
      LKTIME=T2*$SKTIM(4)+$MNTIM(4)                                     00004190
      LKFUEL=LKTIME*MFULRT(4)                                           00004200
      RETURN                                                            00004210
899   RETURN                                                            00004220
C                                                                       00004230
C                                                                       00004240
C                                                                       00004250
C 9. RESCUE RETURN GROUP                                                00004260
C                                                                       00004270
9     CONTINUE                                                          00004280
      DATA GP9PB/4*0.,  91.,0.,2*1.,  92.,3*0.,  93.,3*0./              00004290
      GP9PB(1,2)=GPDAT2(ROW,1)                                          00004300
      GP9PB(1,3)=GPDAT2(ROW,2)                                          00004310
      GP9PB(1,4)=GPDAT2(ROW,3)                                          00004320
      LKPROB=GP9PB(NODE1,NODE2)                                         00004330
      IF(LKPROB.EQ.0.)RETURN                                            00004340
912   IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 913                           00004350
      D1=GPDAT2(ROW,4)                                                  00004360
      LKTIME=D1/TOWSPD                                                  00004370
      LKFUEL=LKTIME*MFULRT(3)                                           00004380
      RETURN                                                            00004390
913   IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 914                           00004400
      D2=GPDAT2(ROW,5)                                                  00004410
      V2=GPDAT2(ROW,6)                                                  00004420
      LKTIME=D2/V2                                                      00004430
      LKFUEL=LKTIME*MFULRT(3)                                           00004440
      RETURN                                                            00004450
914   IF(NODE1.NE.1 .OR. NODE2.NE.4)GO TO 999                           00004460
      D3=GPDAT2(ROW,7)                                                  00004470
      LKTIME=D3/SPEED(2)                                                00004480
```

```
      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                                                                      00004880
C 11. SEARCH FLEET GROUP                                               00004890
C                                                                      00004900
11    CONTINUE                                                         00004910
      DATA GP11PB/2*0.,.1.,0./                                         00004920
      LKPROB=GP11PB(NODE1,NODE2)                                       00004930
      IF(LKPROB.EQ.0.)RETURN                                           00004940
1112  IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 1199                         00004950
      D1=GPDAT2(ROW,1)                                                 00004960
      LKTIME=D1/SPEED(2)                                               00004970
      LKFUEL=LKTIME*MFULRT(2)                                          00004980
      RETURN                                                           00004990
1199  RETURN                                                           00005000
C                                                                      00005010
C                                                                      00005020
C                                                                      00005030
C 12. SEIZE GROUP                                                      00005040
C                                                                      00005050
12    CONTINUE                                                         00005060
      DATA GP12PB/3*0.,.2*0.,.1.,.1.,.2*0./                            00005070
      LKPROB=GP12PB(NODE1,NODE2)                                       00005080
      IF(LKPROB.EQ.0.)RETURN                                           00005090
1213  IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 1232                         00005100
      T1=GPDAT2(ROW,1)                                                 00005110
```

```
      LKTIME=T1                                                          00005120
      LKFUEL=LKTIME*MFULRT(4)                                            00005130
      RETURN                                                             00005140
1232  IF(NODE1.NE.3 .OR. NODE2.NE.2)GO TO 1299                           00005150
      D2=GPDAT2(ROW,2)                                                   00005160
      LKTIME=D2/SPEED(2)                                                 00005170
      LKFUEL=LKTIME*MFULRT(2)                                            00005180
      RETURN                                                             00005190
1299  RETURN                                                             00005200
C                                                                        00005210
C                                                                        00005220
C                                                                        00005230
C 13. SENSOR SEARCH GROUP                                                00005240
C THIS GROUP MUST ALWAYS FOLLOW A STEAM GROUP.                           00005250
C SUCCESS                                                                00005260
13    CONTINUE                                                           00005270
      DATA GP13PB/2*0.. 95..0./                                          00005280
      SW=GPDAT2(ROW,1)                                                   00005290
      E=GPDAT2(ROW,2)                                                    00005300
      VTAR=GPDAT2(ROW,3)                                                 00005310
      TMAX=GPDAT2(ROW,4)                                                 00005320
      TEEF=$PRVTM(0.)                                                    00005330
      CALL $SSHP(SPEED,TEEF,SW,E,VTAR,TMAX,PS,PF,TS,TF)                  00005340
      GP13PB(1,2)=PS                                                     00005350
      LKPROB=GP13PB(NODE1,NODE2)                                         00005360
      IF(LKPROB.EQ.0.)RETURN                                             00005370
      LKTIME=TS                                                          00005380
      LKFUEL=LKTIME*MFULRT(2)                                            00005390
1399  RETURN                                                             00005400
C                                                                        00005410
C                                                                        00005420
C                                                                        00005430
C 14. STANDBY GROUP                                                      00005440
C                                                                        00005450
14    CONTINUE                                                           00005460
      DATA GP14PB/2*0.. 1..0./                                           00005470
      LKPROB=GP14PB(NODE1,NODE2)                                         00005480
      IF(LKPROB.EQ.0.)RETURN                                             00005490
1412  IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 1499                           00005500
      T1=GPDAT2(ROW,1)                                                   00005510
      LKTIME=T1                                                          00005520
      LKFUEL=LKTIME*MFULRT(4)                                            00005530
      RETURN                                                             00005540
1499  RETURN                                                             00005550
C                                                                        00005560
C                                                                        00005570
C                                                                        00005580
C 15. STEAM GROUP                                                        00005590
C                                                                        00005600
15    CONTINUE                                                           00005610
      DATA GP15PB/4*0.. 91..0..2*1.. 92..3*0.. 93..3*0./                 00005620
      GP15PB(1,2)=GPDAT2(ROW,1)                                          00005630
      GP15PB(1,3)=GPDAT2(ROW,2)                                          00005640
      GP15PB(1,4)=GPDAT2(ROW,3)                                          00005650
      LKPROB=GP15PB(NODE1,NODE2)                                         00005660
      IF(LKPROB.EQ.0.)RETURN                                             00005670
1512  IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 1513                           00005680
      D1=GPDAT2(ROW,4)                                                   00005690
      LKTIME=D1/SPEED(2)                                                 00005700
      LKFUEL=LKTIME*MFULRT(2)                                            00005710
      RETURN                                                             00005720
1513  IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 1514                           00005730
      D2=GPDAT2(ROW,5)                                                   00005740
```

```
      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.,2*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*$SKTIM(7)*$WNTIM(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*$SKTIM(7)*$WNTIM(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*$SKTIM(7)*$WNTIM(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
```

```
      LKTIME=T3*$SKTIM(7)*$MNTIM(7)                                     00006380
      LKFUEL=LKTIME*MFULRT(4)                                           00006390
      RETURN                                                            00006400
1799  RETURN                                                            00006410
C                                                                       00006420
C                                                                       00006430
C                                                                       00006440
C 18. WORK EQUIPMENT GROUP                                              00006450
C                                                                       00006460
18    CONTINUE                                                          00006470
      DATA GP18PB/6*0.,.92.,2*0.,3*1.,.91.,5*0.,                        00006480
     X  2*0.,1.,3*0.,.93.,5*0.,.94.,5*0./                               00006490
      GP18PB(1,2)=GPDAT2(ROW,2)                                         00006500
      GP18PB(1,3)=GPDAT2(ROW,1)                                         00006510
      GP18PB(1,5)=GPDAT2(ROW,3)                                         00006520
      GP18PB(1,6)=GPDAT2(ROW,4)                                         00006530
      LKPROB=GP18PB(NODE1,NODE2)                                        00006540
      IF(LKPROB.EQ.0.)RETURN                                            00006550
1812  IF(NODE1.NE.1 .OR. NODE2.NE.2)GO TO 1813                          00006560
      T4=GPDAT2(ROW,8)                                                  00006570
      LKTIME=T4*$SKTIM(6)                                               00006580
      LKFUEL=LKTIME*MFULRT(4)                                           00006590
      RETURN                                                            00006600
1813  IF(NODE1.NE.1 .OR. NODE2.NE.3)GO TO 1815                          00006610
      T1=GPDAT2(ROW,5)                                                  00006620
      LKTIME=T1*$SKTIM(7)*$MNTIM(7)                                     00006630
      LKFUEL=LKTIME*MFULRT(4)                                           00006640
      RETURN                                                            00006650
1815  IF(NODE1.NE.1 .OR. NODE2.NE.5)GO TO 1816                          00006660
      T5=GPDAT2(ROW,9)                                                  00006670
      LKTIME=T5*$SKTIM(7)*$MNTIM(7)                                     00006680
      LKFUEL=LKTIME*MFULRT(4)                                           00006690
      RETURN                                                            00006700
1816  IF(NODE1.NE.1 .OR. NODE2.NE.6)GO TO 1834                          00006710
      T6=GPDAT2(ROW,10)                                                 00006720
      LKTIME=T6*$SKTIM(7)*$MNTIM(7)                                     00006730
      LKFUEL=LKTIME*MFULRT(4)                                           00006740
      RETURN                                                            00006750
1834  IF(NODE1.NE.3 .OR. NODE2.NE.4)GO TO 1842                          00006760
      T2=GPDAT2(ROW,6)                                                  00006770
      LKTIME=T2                                                         00006780
      LKFUEL=LKTIME*MFULRT(4)                                           00006790
      RETURN                                                            00006800
1842  IF(NODE1.NE.4 .OR. NODE2.NE.2)GO TO 1899                          00006810
      T3=GPDAT2(ROW,7)                                                  00006820
      LKTIME=T3*$SKTIM(7)*$MNTIM(7)                                     00006830
      LKFUEL=LKTIME*MFULRT(4)                                           00006840
      RETURN                                                            00006850
1899  RETURN                                                            00006860
C                                                                       00006870
C 90. SAR SEARCH GROUP                                                  00006880
C FAILURE                                                               00006890
90    CONTINUE                                                          00006900
      DATA GP90PB/9*0.,.9*0.,.91.,8*0.,.92.,8*0.,                       00006910
     X 9*0.,.9*0.,.9*0.,.9*0.,.2*0.,.917.,.927.,.5*0./                  00006920
      GP90PB(1,3)=GPDAT2(ROW,1)                                         00006930
      GP90PB(1,4)=GPDAT2(ROW,2)                                         00006940
      SW1=GPDAT2(ROW,3)                                                 00006950
      A1=GPDAT2(ROW,4)                                                  00006960
      NSEAR1=GPDAT2(ROW,5)                                              00006970
      CF1=GPDAT2(ROW,6)                                                 00006980
      TMAX1=GPDAT2(ROW,7)                                               00006990
      SW2=GPDAT2(ROW,8)                                                 00007000
```

```
      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 $SPEDU(SPEED,SW2,A2,NSEAR2,CF2,TMAX2,PS2,PF2,TS2,TF2)        00007060
      GP90PB(3,9)=PF1                                                   00007070
      GP90PB(4,9)=PF2                                                   00007080
      LKPROB=GP90PB(NODE1,NODE2)                                        00007090
      IF(LKPROB.EQ.0.)RETURN                                            00007100
9039  IF(NODE1.NE.3 .OR. NODE2.NE.9)GO TO 9049                          00007110
      LKTIME=TF1                                                        00007120
      LKFUEL=LKTIME*MFULRT(3)                                           00007130
      RETURN                                                            00007140
9049  IF(NODE1.NE.4 .OR. NODE2.NE.9)GO TO 9099                          00007150
      LKTIME=TF2                                                        00007160
      LKFUEL=LKTIME*MFULRT(3)                                           00007170
      RETURN                                                            00007180
9099  RETURN                                                            00007190
C                                                                       00007200
C 93. SENSOR SEARCH GROUP                                               00007210
C THIS GROUP MUST ALWAYS FOLLOW A STEAM GROUP                           00007220
C FAILURE                                                               00007230
93    CONTINUE                                                          00007240
      DATA GP93PB/72*0., 97.,8*0./                                      00007250
      SW=GPDAT2(ROW,1)                                                  00007260
      E=GPDAT2(ROW,2)                                                   00007270
      VTAR=GPDAT2(ROW,3)                                                00007280
      TMAX=GPDAT2(ROW,4)                                                00007290
      TBEF=$PRVTMT(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
```

```
      TASKNO=GP1TK(NODE1,NODE2) - RATE*100                              00007640
      RETURN                                                            00007650
C                                                                       00007660
C 2. ESCORT GROUP                                                       00007670
C                                                                       00007680
2     CONTINUE                                                          00007690
      DATA GP2TK/3*0, 302,0,0, 201,2*0/                                 00007700
      RATE=GP2TK(NODE1,NODE2)/100                                       00007710
      TASKNO=GP2TK(NODE1,NODE2) - RATE*100                              00007720
      RETURN                                                            00007730
C                                                                       00007740
C 3. FIGHT FIRE GROUP                                                   00007750
C                                                                       00007760
3     CONTINUE                                                          00007770
      DATA GP3TK/4*0, 402,2*0,413, 401,3*0, 2*0,403,0/                  00007780
      RATE=GP3TK(NODE1,NODE2)/100                                       00007790
      TASKNO=GP3TK(NODE1,NODE2) - RATE*100                              00007800
      RETURN                                                            00007810
C                                                                       00007820
C 4. IDENTIFY GROUP                                                     00007830
C                                                                       00007840
4     CONTINUE                                                          00007850
      DATA GP4TK/3*0, 203,0,0, 202,2*0/                                 00007860
      RATE=GP4TK(NODE1,NODE2)/100                                       00007870
      TASKNO=GP4TK(NODE1,NODE2) - RATE*100                              00007880
      RETURN                                                            00007890
C                                                                       00007900
C 5. INSPECT GROUP                                                      00007910
C                                                                       00007920
5     CONTINUE                                                          00007930
      DATA GP5TK/6*0, 3*0,416,0,413, 408,5*0,                           00007940
     X  2*0,405,3*0, 401,5*0, 4*0,405,0/                                00007950
      RATE=GP5TK(NODE1,NODE2)/100                                       00007960
      TASKNO=GP5TK(NODE1,NODE2) - RATE*100                              00007970
      RETURN                                                            00007980
C                                                                       00007990
C 6. MONITOR GROUP                                                      00008000
C                                                                       00008010
6     CONTINUE                                                          00008020
      DATA GP6TK/5*0, 409,0,3*0, 410,4*0, 412,4*0,                      00008030
     X 417,4*0/                                                         00008040
      RATE=GP6TK(NODE1,NODE2)/100                                       00008050
      TASKNO=GP6TK(NODE1,NODE2)-RATE*100                                00008060
      RETURN                                                            00008070
C                                                                       00008080
C 7. PATROL GROUP                                                       00008090
C                                                                       00008100
7     CONTINUE                                                          00008110
      DATA GP7TK/3*0, 304,0,0, 204,2*0/                                 00008120
      RATE=GP7TK(NODE1,NODE2)/100                                       00008130
      TASKNO=GP7TK(NODE1,NODE2) - RATE*100                              00008140
      RETURN                                                            00008150
C                                                                       00008160
C 8. RESCUE GROUP                                                       00008170
C                                                                       00008180
8     CONTINUE                                                          00008190
      DATA GP8TK/3*0, 415,0,0, 414,2*0/                                 00008200
      RATE=GP8TK(NODE1,NODE2)/100                                       00008210
      TASKNO=GP8TK(NODE1,NODE2) - RATE*100                              00008220
      RETURN                                                            00008230
C                                                                       00008240
C 9. RESCUE RETURN GROUP                                                00008250
                                                                        00008260
```

```
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
```

```
C                                                                       00008900
17    CONTINUE                                                          00008910
      DATA GP17TK/4*0, 207,2*0,420, 406,3*0,                            00008920
     X 2*0,207,0/                                                       00008930
      RATE=GP17TK(NODE1,NODE2)/100                                      00008940
      TASKNO=GP17TK(NODE1,NODE2) - RATE*100                             00008950
      RETURN                                                            00008960
C                                                                       00008970
C 18. WORK EQUIPMENT GROUP                                              00008980
C                                                                       00008990
18    CONTINUE                                                          00009000
      DATA GP18TK/6*0, 422,2*0,416,0,0, 408,5*0,                        00009010
     X 2*0,421,3*0, 419,5*0, 423,5*0/                                   00009020
      RATE=GP18TK(NODE1,NODE2)/100                                      00009030
      TASKNO=GP18TK(NODE1,NODE2) - RATE*100                             00009040
      RETURN                                                            00009050
C                                                                       00009060
C 90. SAR SEARCH GROUP                                                  00009070
C FAILURE                                                               00009080
90    CONTINUE                                                          00009090
      DATA GP90TK/9*0,9*0,9*0,9*0,9*0,9*0,9*0,                          00009100
     X 9*0,9*0, 2*0,303,301,5*0/                                        00009110
      RATE=GP90TK(NODE1,NODE2)/100                                      00009120
      TASKNO=GP90TK(NODE1,NODE2)-RATE*100                               00009130
      RETURN                                                            00009140
C                                                                       00009150
C 93. SENSOR SEARCH GROUP                                               00009160
C FAILURE                                                               00009170
93    CONTINUE                                                          00009180
      DATA GP93TK/72*0, 206,8*0/                                        00009190
      RATE=GP93TK(NODE1,NODE2)/100                                      00009200
      TASKNO=GP93TK(NODE1,NODE2)-RATE*100                               00009210
      RETURN                                                            00009220
C                                                                       00009230
      END                                                               00009240
C                                                                       00009250
C                                                                       00009260
C                                                                       00009270
C $MINPH                                                                00009280
C                                                                       00009290
C FINDS THE MINIMUM TIME AND FUEL PATHS FROM ANY OVERALL NODE TO        00009300
C THE END OF THE SCENARIO                                               00009310
C                                                                       00009320
      SUBROUTINE $MINPH(N,OVCNMX,GPPLMX,MINTIM,MINFUE)                  00009330
      IMPLICIT REAL (A-Z)                                               00009340
      INTEGER I,J,K,N,GPPLMX,GROUP,INST,FLAG,FLAGNW,AGAIN               00009350
      DIMENSION GPPLMX(50,50),OVCNMX(50,50),FLAG(50),FLAGNW(50)         00009360
      DIMENSION GPMNT(50,50),GPMNF(50,50),MINTIM(50),MINFUE(50)         00009370
C                                                                       00009380
C INITIALIZATION                                                        00009390
      DO 10 I=1,N                                                       00009400
      MINTIM(I)=99999.                                                  00009410
      MINFUE(I)=99999.                                                  00009420
      FLAG(I)=0                                                         00009430
      FLAGNW(I)=0                                                       00009440
      DO 10 J=1,N                                                       00009450
      GPMNT(I,J)=0.                                                     00009460
      GPMNF(I,J)=0.                                                     00009470
10    CONTINUE                                                          00009480
      AGAIN=0                                                           00009490
C                                                                       00009500
C FIND THE MINIMUM GROUP TIME PATH AND FUEL PATH FOR ALL LINKS          00009510
      DO 20 I=1,N                                                       00009520
```

```
      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
```

```
C                                                                       00010160
      SUBROUTINE $GPMIN(GROUP,INST,MINTIM,MINFUE)                       00010170
      IMPLICIT REAL(A-Z)                                                00010180
      INTEGER GROUP,INST                                                00010190
      MINTIM=99999.                                                     00010200
      MINFUE=99999.                                                     00010210
      IF(GROUP.EQ.90)GO TO 90                                           00010220
      IF(GROUP.EQ.93)GO TO 93                                           00010230
      GO TO(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18),GROUP         00010240
C                                                                       00010250
C                                                                       00010260
C                                                                       00010270
C 1. ASSIST GROUP                                                       00010280
C                                                                       00010290
1     CONTINUE                                                          00010300
      CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                           00010310
      CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                           00010320
      CALL $LKDAT(GROUP,INST,1,5,P15,T15,F15)                           00010330
      CALL $LKDAT(GROUP,INST,1,6,P16,T16,F16)                           00010340
      CALL $LKDAT(GROUP,INST,3,4,P34,T34,F34)                           00010350
      CALL $LKDAT(GROUP,INST,4,2,P42,T42,F42)                           00010360
      CALL $LKDAT(GROUP,INST,6,7,P67,T67,F67)                           00010370
      CALL $LKDAT(GROUP,INST,7,2,P72,T72,F72)                           00010380
      T1342=T13+T34+T42                                                 00010390
      T1672=T16+T67+T72                                                 00010400
      IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                         00010410
      IF(P13.GT.0..AND.T1342.LT.MINTIM)MINTIM=T1342                     00010420
      IF(P15.GT.0..AND.T15.LT.MINTIM)MINTIM=T15                         00010430
      IF(P16.GT.0..AND.T1672.LT.MINTIM)MINTIM=T1672                     00010440
      F1342=F13+F34+F42                                                 00010450
      F1672=F16+F67+F72                                                 00010460
      IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                         00010470
      IF(P13.GT.0..AND.F1342.LT.MINFUE)MINFUE=F1342                     00010480
      IF(P15.GT.0..AND.F15.LT.MINFUE)MINFUE=F15                         00010490
      IF(P16.GT.0..AND.F1672.LT.MINFUE)MINFUE=F1672                     00010500
      RETURN                                                            00010510
C                                                                       00010520
C                                                                       00010530
C 2. ESCORT GROUP                                                       00010540
C                                                                       00010550
2     CONTINUE                                                          00010560
      CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                           00010570
      CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                           00010580
      IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                         00010590
      IF(P13.GT.0..AND.T13.LT.MINTIM)MINTIM=T13                         00010600
      IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                         00010610
      IF(P13.GT.0..AND.F13.LT.MINFUE)MINFUE=F13                         00010620
      RETURN                                                            00010630
C                                                                       00010640
C                                                                       00010650
C 3. FIGHT FIRE GROUP                                                   00010660
C                                                                       00010670
3     CONTINUE                                                          00010680
      CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                           00010690
      CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                           00010700
      CALL $LKDAT(GROUP,INST,3,4,P34,T34,F34)                           00010710
      CALL $LKDAT(GROUP,INST,4,2,P42,T42,F42)                           00010720
      T1342=T13+T34+T42                                                 00010730
      IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                         00010740
      IF(P13.GT.0..AND.T1342.LT.MINTIM)MINTIM=T1342                     00010750
      F1342=F13+F34+F42                                                 00010760
      IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                         00010770
      IF(P13.GT.0..AND.F1342.LT.MINFUE)MINFUE=F1342                     00010780
```

```
      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                                                                       00011420
C                                                                       00011430
C 8. RESCUE GROUP                                                       00011440
C                                                                       00011450
8       CONTINUE                                                        00011460
        CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                         00011470
        CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                         00011480
        IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                       00011490
        IF(P13.GT.0..AND.T13.LT.MINTIM)MINTIM=T13                       00011500
        IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                       00011510
        IF(P13.GT.0..AND.F13.LT.MINFUE)MINFUE=F13                       00011520
        RETURN                                                          00011530
C                                                                       00011540
C                                                                       00011550
C 9. RESCUE RETURN GROUP                                                00011560
C                                                                       00011570
9       CONTINUE                                                        00011580
        CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                         00011590
        CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                         00011600
        CALL $LKDAT(GROUP,INST,1,4,P14,T14,F14)                         00011610
        IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                       00011620
        IF(P13.GT.0..AND.T13.LT.MINTIM)MINTIM=T13                       00011630
        IF(P14.GT.0..AND.T14.LT.MINTIM)MINTIM=T14                       00011640
        IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                       00011650
        IF(P13.GT.0..AND.F13.LT.MINFUE)MINFUE=F13                       00011660
        IF(P14.GT.0..AND.F14.LT.MINFUE)MINFUE=F14                       00011670
        RETURN                                                          00011680
C                                                                       00011690
C                                                                       00011700
C 10. SAR SEARCH GROUP                                                  00011710
C SUCCESS                                                               00011720
10      CONTINUE                                                        00011730
        CALL $LKDAT(GROUP,INST,3,2,P32,T32,F32)                         00011740
        CALL $LKDAT(GROUP,INST,4,2,P42,T42,F42)                         00011750
        IF(P32.GT.0..AND. T32.LT.MINTIM)MINTIM=T32                      00011760
        IF(P42.GT.0..AND.T42.LT.MINTIM)MINTIM=T42                       00011770
        IF(P32.GT.0..AND.F32.LT.MINFUE)MINFUE=F32                       00011780
        IF(P42.GT.0..AND.F42.LT.MINFUE)MINFUE=F42                       00011790
        RETURN                                                          00011800
C                                                                       00011810
C                                                                       00011820
C 11. SEARCH FLEET GROUP                                                00011830
C                                                                       00011840
11      CONTINUE                                                        00011850
        CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                         00011860
        IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                       00011870
        IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                       00011880
        RETURN                                                          00011890
C                                                                       00011900
C                                                                       00011910
C 12. SEIZE GROUP                                                       00011920
C                                                                       00011930
12      CONTINUE                                                        00011940
        CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                         00011950
        CALL $LKDAT(GROUP,INST,3,2,P32,T32,F32)                         00011960
        T132=T13+T32                                                    00011970
        P132=P13*P32                                                    00011980
        IF(P132.GT.0..AND.T132.LT.MINTIM)MINTIM=T132                    00011990
        F132=F13+F32                                                    00012000
        P132=P13*P32                                                    00012010
        IF(P132.GT.0..AND.F132.LT.MINFUE)MINFUE=F132                    00012020
        RETURN                                                          00012030
C                                                                       00012040
```

```
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                                                                       00012680
18      CONTINUE                                                        00012690
        CALL $LKDAT(GROUP,INST,1,2,P12,T12,F12)                         00012700
        CALL $LKDAT(GROUP,INST,1,3,P13,T13,F13)                         00012710
        CALL $LKDAT(GROUP,INST,1,5,P15,T15,F15)                         00012720
        CALL $LKDAT(GROUP,INST,1,6,P16,T16,F16)                         00012730
        CALL $LKDAT(GROUP,INST,3,4,P34,T34,F34)                         00012740
        CALL $LKDAT(GROUP,INST,4,2,P42,T42,F42)                         00012750
        T1342=T13+T34+T42                                               00012760
        IF(P12.GT.0..AND.T12.LT.MINTIM)MINTIM=T12                       00012770
        IF(P13.GT.0..AND.T1342.LT.MINTIM)MINTIM=T1342                   00012780
        IF(P15.GT.0..AND.T15.LT.MINTIM)MINTIM=T15                       00012790
        IF(P16.GT.0..AND.T16.LT.MINTIM)MINTIM=T16                       00012800
        F1342=F13+F34+F42                                               00012810
        IF(P12.GT.0..AND.F12.LT.MINFUE)MINFUE=F12                       00012820
        IF(P13.GT.0..AND.F1342.LT.MINFUE)MINFUE=F1342                   00012830
        IF(P15.GT.0..AND.F15.LT.MINFUE)MINFUE=F15                       00012840
        IF(P16.GT.0..AND.F16.LT.MINFUE)MINFUE=F16                       00012850
        RETURN                                                          00012860
C                                                                       00012870
C 90. SAR SEARCH GROUP                                                  00012880
C FAILURE                                                               00012890
90      CONTINUE                                                        00012900
        CALL $LKDAT(GROUP,INST,3,9,P39,T39,F39)                         00012910
        CALL $LKDAT(GROUP,INST,4,9,P49,T49,F49)                         00012920
        IF(P39.GT.0..AND. T39.LT.MINTIM)MINTIM=T39                      00012930
        IF(P49.GT.0..AND.T49.LT.MINTIM)MINTIM=T49                       00012940
        IF(P39.GT.0..AND.F39.LT.MINFUE)MINFUE=F39                       00012950
        IF(P49.GT.0..AND.F49.LT.MINFUE)MINFUE=F49                       00012960
        RETURN                                                          00012970
C                                                                       00012980
C 93. SENSOR SEARCH GROUP                                               00012990
C FAILURE                                                               00013000
93      CONTINUE                                                        00013010
        CALL $LKDAT(GROUP,INST,1,9,P19,T19,F19)                         00013020
        IF(P19.GT.0..AND.T19.LT.MINTIM)MINTIM=T19                       00013030
        IF(P19.GT.0..AND.F19.LT.MINFUE)MINFUE=F19                       00013040
        RETURN                                                          00013050
C                                                                       00013060
C                                                                       00013070
        END                                                             00013080
C                                                                       00013090
C ZZ                                                                    00013100
C                                                                       00013110
C FINDS Y VALUE ON A STRAIGHT LINE, GIVEN X                             00013120
C VALUE AND TWO POINTS ON THE LINE (ASSUMING LINE                       00013130
C EXTENDS INFINITELY)                                                   00013140
C                                                                       00013150
        FUNCTION ZZ(X,X1,Y1,X2,Y2)                                      00013160
C                                                                       00013170
        IF(ABS(X2-X1).LT..0001)GOTO 1                                   00013180
        SLOPE=(Y2-Y1)/(X2-X1)                                           00013190
        IF(ABS(Y2-Y1).LT..0001) SLOPE=0.                                00013200
        B=Y1-SLOPE*X1                                                   00013210
        ZZ=SLOPE*X+B                                                    00013220
        RETURN                                                          00013230
C                                                                       00013240
1       ZZ=(Y1+Y2)/2.                                                   00013250
        RETURN                                                          00013260
        END                                                             00013270
C                                                                       00013280
C                                                                       00013290
C ZZZ                                                                   00013300
```

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

```
      END                                                              00013880
C                                                                      00013890
C $MNTIM                                                               00013900
C                                                                      00013910
      FUNCTION $MNTIM(MASTSK)                                          00013920
      IMPLICIT REAL(A-Z)                                               00013930
      INTEGER MASTSK                                                   00013940
      COMMON/MNCOM/LENG                                                00013950
C      BORD,RTRV,WLOP                                                  00013960
C                                                                      00013970
      IF(MASTSK.EQ.2 .OR.MASTSK.EQ.4                                   00013980
     X  .OR. MASTSK.EQ.7)GO TO 1                                       00013990
      $MNTIM=1.                                                        00014000
      RETURN                                                           00014010
1     $MNTIM=$$4(LENG,0.,1., 100.,1., 400.,3., 1000.,3.)               00014020
      RETURN                                                           00014030
      END                                                              00014040
C                                                                      00014050
C $VZTIM                                                               00014060
C                                                                      00014070
      FUNCTION $VZTIM(MASTSK)                                          00014080
      IMPLICIT REAL(A-Z)                                               00014090
      INTEGER MASTSK,VISDTB                                            00014100
      COMMON/VZ/VISDIS(3,3)                                            00014110
      COMMON/MNCOM/LENG,FOFRAC,VISDTB                                  00014111
C      IDNT                                                            00014120
      IF(MASTSK.EQ.14)GO TO 14                                         00014130
      $VZTIM=1.                                                        00014140
      RETURN                                                           00014150
14    $VZTIM=VISDIS(1,VISDTB)+VISDIS(2,VISDTB)+                        00014160
     X 2.0*VISDIS(3,VISDTB)                                            00014170
      RETURN                                                           00014180
      END                                                              00014190
C                                                                      00014200
C $SPEDU                                                               00014210
C                                                                      00014220
C CALCULATES PROBABILITY AND TIME FOR SEARCH                           00014230
C FOR PEOPLE AND SEARCH FOR DIST. UNIT TASKS                           00014240
      SUBROUTINE $SPEDU(SPEED,SW,A,NMAX,CF,TMAX,PS,PF,TS,TF)           00014250
      IMPLICIT REAL(A-Z)                                               00014260
      INTEGER I,J,K,L,NMAX,NFULLS                                      00014270
      DIMENSION ALPHA(5),SPEED(4)                                      00014280
      DATA ALPHA/1.00,2.11,3.31,4.37,5.16/                             00014290
      IF(SW.LE.0..OR.A.LE.0..OR.NMAX.LE.0.OR.CF.LE.0..OR.              00014300
     X TMAX.LE.0.)GO TO 99                                             00014310
      GO TO 300                                                        00014320
99    PS=0.0                                                           00014330
      PF=1.0                                                           00014340
      TS=0.0                                                           00014350
      TF=0.0                                                           00014360
      RETURN                                                           00014370
300   NFULLS=NMAX                                                      00014380
      SFRAC=0.                                                         00014390
C     TEST IF NUMBER OF FULL SEARCHES TAKES TOO LONG                   00014400
301   TF=0.                                                            00014410
      DO 20 I=1,NFULLS                                                 00014420
      TNEXT=(CF*A*ALPHA(I))/(SPEED(3)*SW)                              00014430
      TF=TF+TNEXT                                                      00014440
      IF(TF.GT.TMAX)GO TO 32                                           00014450
20    CONTINUE                                                         00014460
      GO TO 33                                                         00014470
C     IF EXCEEDS TMAX                                                  00014480
32    NFULLS=I-1                                                       00014490
```

```
      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
```

```
      RETURN                                                            00015110
C INITIALIZE COUNTERS                                                   00015150
90    FPROD=1.                                                          00015160
      TNUMER=0.                                                         00015170
      TDENOM=0.                                                         00015180
      PSDELT=1.                                                         00015181
C START AT TIME ZERO                                                    00015190
      T=0.0                                                             00015200
C FIND TARGET'S AREA THAT IS STILL UNSEARCHED                           00015201
20    ATGTOT=PI*(RF+VTAR*(TOFF+T+DELTH))**2                             00015210
      ASRCHD=SW*SPEED(2)*(T+DELTH)                                      00015220
      ATGUNS=ATGTOT-ASRCHD                                              00015230
C TEST AREA UNSEARCHED:IF .LE. ASRCHD,THEN PS=1,ELSE CALCULATE          00015240
C  PROBABILITY OF SUCCESS IN NEXT DELTA T                               00015250
      IF(ATGUNS.LE.ASRCHD)GO TO 100                                     00015260
      PSDELT=(SW*SPEED(2)*DELTAT)/ATGUNS                                00015270
      FPROD=FPROD*(1.0-PSDELT)                                          00015280
      TNUMER=TNUMER+(T+DELTH)*PSDELT                                    00015290
      TDENOM=TDENOM+PSDELT                                              00015300
      T=T+DELTAT                                                        00015310
C TEST FOR TIME                                                         00015330
      IF(T.GE.TMAX)GO TO 101                                            00015340
      GO TO 20                                                          00015350
100   PS=1.                                                             00015360
      PF=0.                                                             00015370
      TNUMER=TNUMER+(T+DELTH)*PSDELT                                    00015380
      TDENOM=TDENOM+PSDELT                                              00015390
      TS=TNUMER/TDENOM                                                  00015400
      TF=TMAX                                                           00015410
      RETURN                                                            00015420
C TIME IS TMAX                                                          00015421
101   PS=1.0-FPROD                                                      00015430
      PF=FPROD                                                          00015440
      TS=TNUMER/TDENOM                                                  00015450
      TF=TMAX                                                           00015460
      RETURN                                                            00015470
      END                                                               00015480
C                                                                       00015490
C $CC17                                                                 00015500
C                                                                       00015510
C FINDS THE CARGO CARRYING PARAMETER FOR THE                            00015520
C  TRANSPORT MASTER TASK                                                00015530
      SUBROUTINE $CC17(GROUP,INST,NODE1,NODE2,DECK,CARGCP,CC0)          00015540
      IMPLICIT REAL(A-Z)                                                00015550
      INTEGER GROUP,INST,NODE1,NODE2,GROUP1,ROW,GPDAT1                  00015560
      COMMON/GPDATA/GPDAT1(40,2),GPDAT2(40,18)                          00015570
      GROUP1=GROUP                                                      00015580
      IF(GROUP.GE.90)GROUP1=GROUP-80                                    00015590
C GET ROW OF GROUP AND INSTANCE                                         00015600
      DO 100 ROW=1,100                                                  00015610
      IF(GPDAT1(ROW,1).EQ.GROUP1.AND.GPDAT1(ROW,2)                      00015620
     X .EQ.INST)GO TO 1712                                              00015630
100   CONTINUE                                                          00015640
1712  IF(NODE1.NE.1.AND.NODE2.NE.2)GO TO 1734                           00015650
      AREA=GPDAT2(ROW,9)                                                00015660
      WGHT=GPDAT2(ROW,10)                                               00015670
      GO TO 1700                                                        00015680
1734  IF(NODE1.NE.3.AND.NODE2.NE.4)GO TO 1799                           00015690
      AREA=GPDAT2(ROW,5)                                                00015700
      WGHT=GPDAT2(ROW,6)                                                00015710
1700  CC0=0.0                                                           00015720
      IF(AREA.LE.DECK .AND. WGHT.LE.CARGCP)CC0=1.0                      00015730
1799  RETURN                                                            00015740
      END                                                               00015750
```

300 copies

☆ U. S. GOVERNMENT PRINTING OFFICE: 1978--702-437--351