

LEVEL III

12

AD A059538

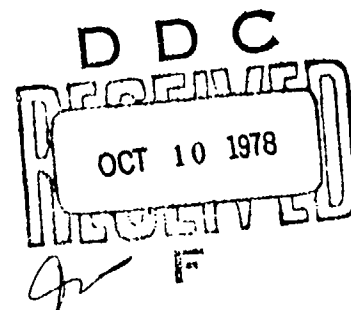
DDC FILE COPY

STRUCTURAL AREA INSPECTION FREQUENCY EVALUATION (SAIFE)

Volume IV. Software Documentation and User's Manual

Book 1. Initial Program

Carter J. Dinkeloo
Martin S. Moran



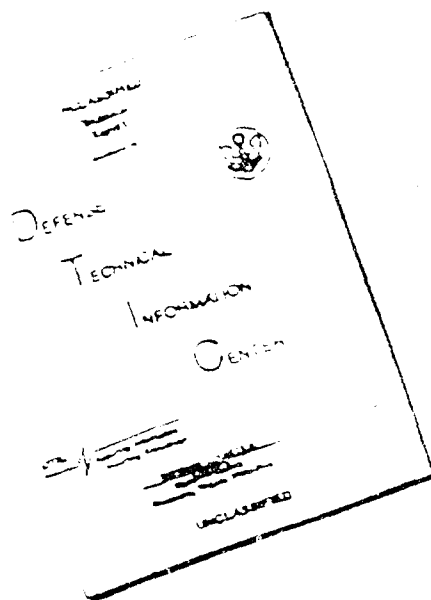
APRIL 1978
FINAL REPORT

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

Prepared for

U.S. DEPARTMENT OF TRANSPORTATION
FEDERAL AVIATION ADMINISTRATION
Systems Research & Development Service
Washington, D.C. 20590

DISCLAIMER NOTICE



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

REPRODUCED FROM
BEST AVAILABLE COPY

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.

1. Report No. FAA-RD-78-29, IV, Book 1		2. Government Accession No. (SAIFE)		3. Recipient's Catalog No.	
4. Title and Subtitle STRUCTURAL AREA INSPECTION FREQUENCY EVALUATION Volume IV. Software Documentation and User's Manual. Book 1 - Initial Program,				5. Report Date April 1978	
7. Author(s) Carter J. Dinkeloo, Martin S. Moran				6. Performing Organization Code	
9. Performing Organization Name and Address Technology Incorporated P.O. Box 3036, Overlook Branch Dayton, Ohio 45431				8. Performing Organization Report No.	
12. Sponsoring Agency Name and Address Department of Transportation Federal Aviation Administration Systems Research and Development Service Washington, D.C. 20590				10. Work Unit No. (TRAIS)	
15. Supplementary Notes				11. Contract or Grant No. DOT-FA74WA-3493	
16. Abstract To assist in the evaluation of proposed structural inspection programs for commercial jet transport aircraft, a logic was developed to simulate structural defects, failures, and inspections. This logic was incorporated in a computer program entitled Structural Area Inspection Frequency Evaluation (SAIFE). With the objective of quantifying the evaluation process currently used to establish and modify inspection intervals, SAIFE accounts for the following factors: (1) aircraft design analysis; (2) fatigue testing; (3) production, service, and corrosion defects; (4) probability of crack or corrosion detection; and (5) aircraft modification economics. As a five-volume document, this report covers the initial contract effort plus a subsequent parametric analysis as follows: Volume I (entitled Executive Summary) presents the SAIFE logic and documents the methodology for the decision-making processes in the simulation logic. Volume II (entitled Description of Simulation Logic) details the SAIFE simulation logic, presents the background data for the analytical functions and decision-making processes, and includes data for a typical simulation problem. Volume III (entitled Demonstration Input, Inspection Survey, and MRR Data) presents data tabulations derived from historical trends and design input data for a SAIFE demonstration problem. As the user's manual for the SAIFE computer program, Volume IV contains detailed computer logic flow diagrams and a complete listing of the program which is written in SIMSCRIPT II.5. Volume V (entitled Results of Model Demonstration) presents the results of the program application to a hypothetical aircraft and compares these results with the service experience of operational aircraft.				13. Type of Report and Period Covered Final Report	
17. Key Words Aircraft experience simulation; aircraft inspection intervals; aircraft hazard rate and reliability; production, service, and corrosion defects; crack or corrosion detection probability; crack growth.				14. Sponsoring Agency Code	
18. Distribution Statement Document is available to the public through the National Technical Information Service, Springfield, Virginia 22151.				15. Supplementary Notes	
19. Security Classif. (of this report) UNCLASSIFIED		20. Security Classif. (of this page) UNCLASSIFIED		21. No. of Pages 209	
				22. Price	

METRIC CONVERSION FACTORS

Approximate Conversions to Metric Measures

Symbol When You Know Multiply by To Find Symbol

LENGTH

in Centimeters 2.5
ft Centimeters 30
yd meters 0.9
mi kilometers 1.6

AREA

sq in square centimeters 6.5
sq ft square meters 0.09
sq yd square meters 0.8
sq mi square kilometers 2.6
acres hectares 0.4

MASS (weight)

oz grams 28
lb kilograms 0.45
short tons (2000 lb) tonnes 0.9

VOLUME

tsp milliliters 5
 Tbsp milliliters 15
 fl oz milliliters 30
 c cups 0.24
 pt pints 0.47
 qt quarts 0.95
 gal gallons 3.8
 cu ft cubic feet 0.03
 cu yd cubic yards 0.76

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 inches 0.04
cm inches 0.4
meters feet 3.3
meters yards 1.1
kilometers miles 0.6

AREA

square centimeters square inches 1.76
square meters square yards 1.2
square kilometers square miles 0.4
hectares (10,000 m²) acres 2.5

MASS (weight)

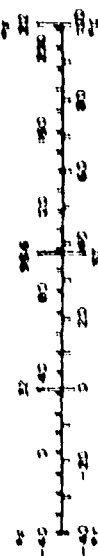
g grams 0.035
kg kilograms 2.2
tonnes (1000 kg) short tons 1.1

VOLUME

ml milliliters 0.03
l liters 0.26
cl centiliters 0.03
cubic meters cubic feet 35
cubic centimeters cubic inches 1.6

TEMPERATURE (approx)

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



* 1 in = 2.54 (exactly). For other exact conversions and more data visit us: www.nist.gov, P. 10-196.
Units of Weights and Measures, Price \$2.25, SO Catalog No. C1310 296.

PREFACE

Technology Incorporated prepared this fourth volume of a five-volume report to document the simulation logic for the Structural Area Inspection Frequency Evaluation (SAIFE) in accordance with Article II, paragraph B of Contract DOT-FA74WA-3493. (Volume IV along with Volume V completes the requirements of Phase III of the contract.) The effort is sponsored by the Aircraft Safety and Noise Abatement Division, Systems Research and Development Service of the Federal Aviation Administration.

The principal Technology Incorporated personnel engaged on this program were Mr. Carter J. Dinkeloo, project engineer, who served as principal investigator; Mr. Martin S. Moran, research engineer, who developed the model for the SAIFE computer program; and Mr. Ronald I. Rockafellow, program manager.

The contract monitors for the FAA were Messrs. Herbert Spicer and Charles Troha of the Aircraft Safety and Noise Abatement Division. The technical monitor was Mr. Arnold E. Anderjaska of the Flight Standards Division.

SEARCHED	☒
SERIALIZED	☐
INDEXED	☐
FILED	☐
FBI - NEW YORK	
JUN 23 1974	
A 23	

TABLE OF CONTENTS

<u>Section</u>	<u>Page</u>
I INTRODUCTION	1
II SYSTEM DESCRIPTION	3
III GENERAL PROGRAM DESCRIPTION	5
IV OPERATING PROCEDURES	6
1. Computer Requirements	6
2. Input	6
2.1 Aircraft Data	7
2.2 Long List Data	9
2.3 Element Data	10
2.4 Format Specifications	12
3. Output	13
3.1 Element Data	14
3.2 Summary Data	22
3.3 Long List Data	22
APPENDIX A. DETAILED PROGRAM DESCRIPTION	31
APPENDIX B. PROGRAM SOURCE LISTING	119

LIST OF ILLUSTRATIONS

<u>Figure</u>		<u>Page</u>
1	Approach to SAIPE Simulation	2
2	Major Aspects of SAIPE Logic	4
3	Sample Input Data to Produce Standard Output . .	13
4	Sample Input Data to Produce Long List Output .	14
5	Sample Standard Output Produced by Input Shown in Figure 3	15
6	Sample Long List Output Produced by Input Shown in Figure 4	24

LIST OF TABLES

<u>Table</u>		<u>Page</u>
1	Initial Random Roots	7

I. INTRODUCTION

It is the mutual goal of the FAA, airframe manufacturers, and air carriers to constantly improve the structural integrity and inspection efficiency of civil aircraft. The good safety record of U.S. air carriers indicates that the current process of establishing and modifying structural inspection programs has been successful. However, with the increasing size and complexity of second- and third-generation transport aircraft, there is a need to quantify more precisely the present, subjective evaluation process which relies heavily on reliability analyses of the new design and on operational experience of similar aircraft.

Because of the extreme complexity of the evaluation process, a computer simulation of all critical aircraft service life aspects was judged the most rational means for quantifying the process more exactly. As a five-volume document, this report documents the resultant Structural Area Inspection Frequency Evaluation (SAIFE) simulation logic. SAIFE accounts for the following factors: (1) aircraft design analysis; (2) component and full-scale fatigue testing; (3) production, service, and corrosion defects; (4) probability of crack or corrosion detection; and (5) aircraft modification economics. It treats these factors in a logical sequence that realistically represents the procedure currently used to establish and modify inspection intervals. Figure 1 illustrates the data sources and analytical functions that are integrated into the SAIFE logic. SAIFE is designed to provide a repeatable method for evaluating proposed inspection programs. However, it is not intended to supplant the Maintenance Review Board or the air carrier use of the Standard Operations Specification - Aircraft Maintenance.

As Volume IV, this user's manual for the SAIFE program contains a system description, a program description, operating procedures, a sample input and output, and a source listing of the program. The detailed description of the program events and routines is presented in Appendix A, and the program source listing is contained in Appendix B which because of its voluminous computer-generated data is on the included microfiche.

The original computer model developed during the initial contract has been modified by the Engineering and Manufacturing Branch, Flight Standards National Field Office for the parametric study and is documented in Book 2 of this volume.

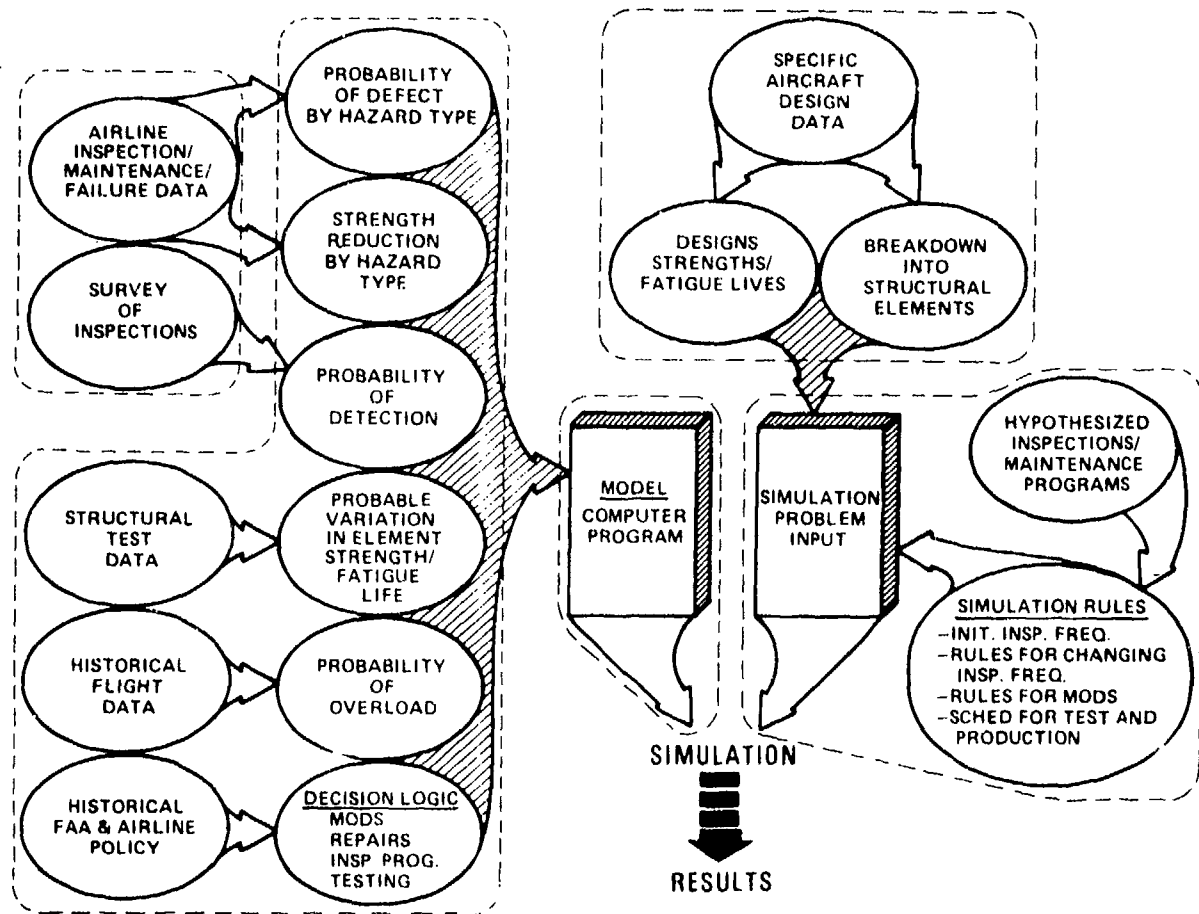


Figure 1. Approach to SAIPE Simulation

II. SYSTEM DESCRIPTION

The eight blocks in Figure 2 represent the major aspects of the SAIFE simulation logic. Block 1 accepts input data for the aircraft fleet and for each structural element in the aircraft. After determining whether element modifications are required because of the fatigue test results in Block 2, Block 1 assigns a fatigue life to each element in each aircraft. Block 3 determines whether production, service, or corrosion defects will occur; if it is determined that such defects will occur, Block 3 predicts the times when they will occur. After comparing the flight loads with the strength of each element, Block 4 predicts the time to failure for each element. Block 5 conducts the periodic inspections of each element. If defects are detected, Block 6 repairs the element and assigns it a new fatigue life. However, if an existing defect is allowed to grow until element failure, Block 5 deletes the aircraft from the fleet. Depending on the magnitude of the detected defects, special inspections and increased inspection frequencies may be called for in Block 7 and modifications may be instituted in Block 8. When all the aircraft have been deleted from the fleet or retired from service, the simulation is complete.

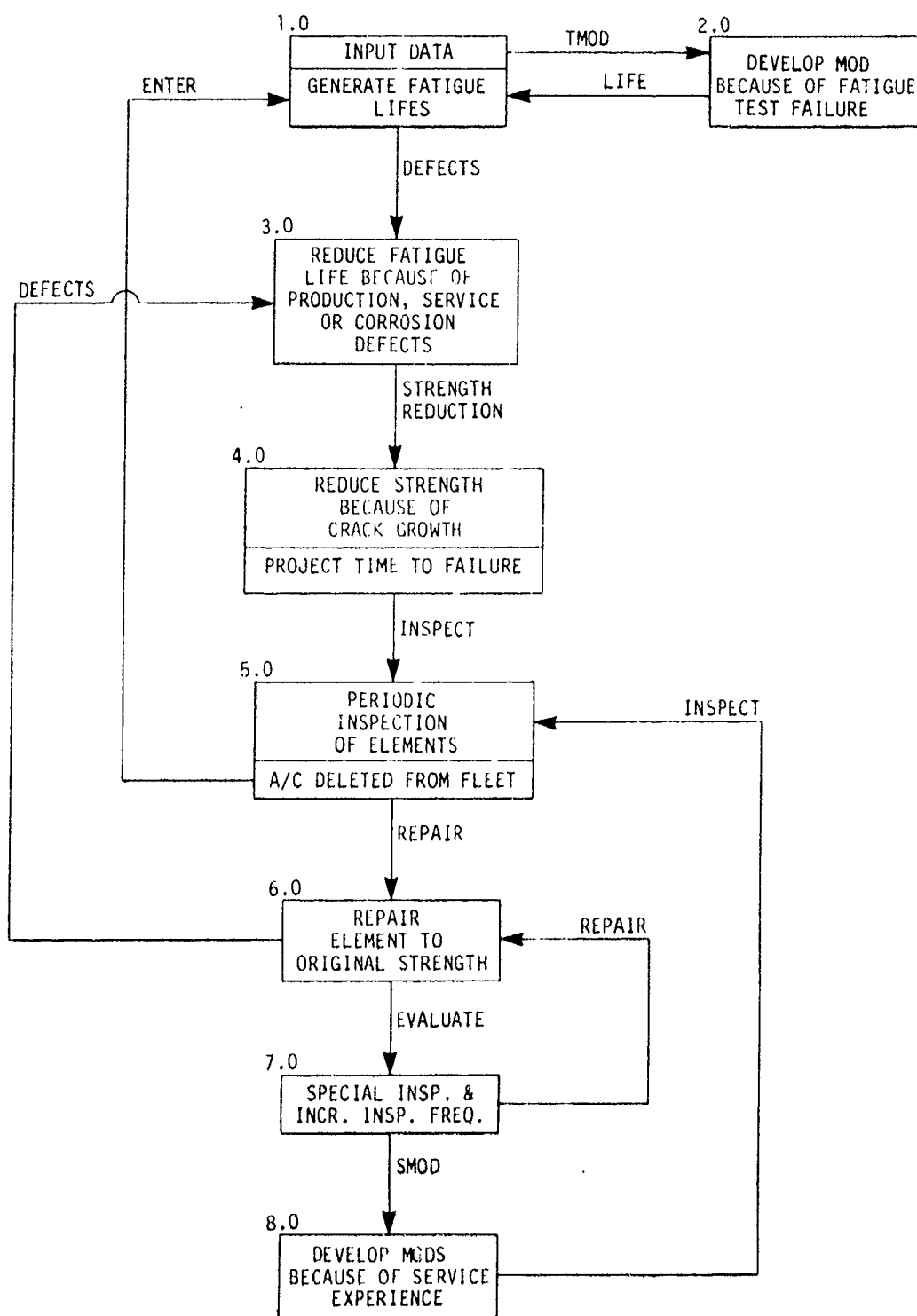


Figure 2. Major Aspects of SAIPE Logic

III. GENERAL PROGRAM DESCRIPTION

The SAIFE program is a large, complex math model designed to simulate the structural performance of aircraft in a fleet of aircraft and the effectiveness of the inspection program for the aircraft fleet. The aircraft model included in the math model is divided into structurally significant elements and the inspection program for each element is defined. Structural defects are classified as follows: fatigue and corrosion which are wear-out and aging phenomena; production or design defects; and operational or maintenance damage. These defects and the inspection program are treated as probabilistic phenomena interacting over time. If the simulation is to proceed properly, the passage of simulated time must be controlled. This control can be accomplished by a user-designed algorithm or it can be done automatically with a "simulation clock" in one of the special purpose simulation languages. SIMSCRIPT II.5 ^T (trademark, Consolidated Analysis Centers Inc., Los Angeles, California), a computer language designed for discrete-event simulation applications, was chosen for the SAIFE simulation. SIMSCRIPT II.5 is a large language designed to facilitate the simulation of large, complex systems, and to reduce the total time spent in designing, programming, and testing simulation models.

Because of the extensive detail, the program events and routines are described in Appendix A.

IV. OPERATING PROCEDURES

1. Computer Requirements

The SAIFE demonstration output presented in Volume V of this report was run on an IBM Model 360-65 computer using the SIMSCRIPT II.5 compiler, release 8F. The execution time per element ranged between 12 and 20 cpu seconds, and the compilation time was 205 cpu seconds. None of the demonstration runs required more than 325K bytes of core storage.

The run time and the core storage requirements depend on the values of the input variables. The variable that probably has the greatest effect on these requirements is the actual average fatigue life. The shorter the life, the greater the requirement: A shorter actual average fatigue life increases crack initiations across the fleet. These additional crack initiations, in turn, increase the inspection process, the repair activities, and the modification and interval change decisions. All these additional events require more execution time and storage space for event notices. The actual average fatigue life is shortened when the parameters involved in the distribution of the ratio of the actual to the predicted average fatigue life are so changed that the ratio is decreased.

Other variables which affect the run time and core storage requirements are as follows: with the actual average fatigue life established for all aircraft in the fleet, the requirements are increased if the parameters involved in the Weibull distribution of actual fatigue lives for individual aircraft are so changed that some lives are shortened and more crack initiations are introduced. In addition, regardless of the fatigue lives, the run time and storage requirements are increased if the defect occurrence rates are increased or the initial inspection intervals are decreased.

Since the seeds of the random number streams are initialized to the same values at the start of each execution, the user can reproduce the output, if desired, and more easily identify effects because of changes in the input parameters. Table 1 lists the ten seeds used by the IBM 360-65 to generate the demonstration output presented in Volume V.

2. Input

The program input consists of three parts. The first part contains input variables which pertain to the aircraft type under consideration. These variables are input only once per simulation run and are constant from element to element. The second part is optional and is explained in Section IV,2.2. The third part contains input variables whose values are unique to each element. These variables must be input in their entirety for each element being simulated.

TABLE 1. INITIAL RANDOM ROOTS

SEED.V(1) = 2116429302
SEED.V(2) = 683743814
SEED.V(3) = 964393174
SEED.V(4) = 1217426631
SEED.V(5) = 618433579
SEED.V(6) = 1157240309
SEED.V(7) = 15726055
SEED.V(8) = 48108509
SEED.V(9) = 1797920909
SEED.V(10) = 477424540

2.1 Aircraft Data

The input variables which pertain to the aircraft type are listed and described below in the order in which they are read in by SAIFE.

MODEL(*) - This one-dimensional alpha array of size two identifies the aircraft type under consideration. The total length of this identification cannot exceed eight characters.

SIZE.OF.FLEET - This integer variable is the number of aircraft in the fleet being simulated. The output format requires that this variable does not exceed 99999.

USAGE.LIFE - This real variable is the service life in flight hours of the aircraft being simulated. All aircraft in the fleet must have the same service life. The output format requires that this variable does not exceed 9999999.

BEGIN.PRODUCTION - This real variable is the time in flight hours relative to the start of the simulation when the first aircraft enters service. This variable in conjunction with the input variable START.TEST enables the user to start the fatigue test of the element before, after, or at the same time the first aircraft enters service.

PRODUCTION.TIME - This real variable defines the initial aircraft production rate. It is the time in flight hours between aircraft entering service.

2.PRODUCTION.TIME - This real variable defines the second aircraft production rate. It is the time in flight hours between aircraft entering service.

PRCHG - This real variable is the simulation time when the second aircraft production rate takes effect. Note that this time is measured from the time that the first aircraft enters service and not from the start of the simulation.

START.TEST - This real variable is the time in flight hours relative to the start of the simulation when the fatigue test of an element is begun. If no fatigue test is to be conducted, this variable is set to the machine upper limit.

TEST.ACCEL.FACT - This real variable is the fatigue test acceleration factor, that is, the quotient of the equivalent flight hours divided by the fatigue test hours.

C.GROWTH.RATE - This real variable is the corrosion area growth rate in sq. inches per hour for the aircraft being considered. The growth rate for each element in the aircraft is modified by its associated CRK (corrosion resistance rating).

C7 - If a modification is developed because of a fatigue test failure, this real variable is the percentage (expressed as a decimal fraction) of the test life when the inspection frequency is increased.

C28 - This real variable is the percentage (expressed as a decimal fraction) reduction in the remaining fatigue life of an element when corrosion occurs in a stress concentration.

C29 - This real variable is the percentage (expressed as a decimal fraction) reduction in the remaining fatigue life of an element when corrosion occurs outside a stress concentration.

MU.R - This real variable is the mean of the log-normal distribution of the ratio of the actual average fatigue life to the predicted average fatigue life.

SIG.R - This real variable is the standard deviation of the log-normal distribution of the ratio of the actual average fatigue life to the predicted average fatigue life.

A - This real variable is the result of fitting an exponential curve to flight load exceedance data. $A_{exp}[BS_a]$ is the number of flight loads per hour which exceed the load level S_a .

B - This real variable is the result of fitting an exponential curve to flight load exceedance data. $A_{exp}[BS_a]$ is the number of flight loads per hour which exceed the load level S_a .

LABCD(*) - This one-dimensional real array of size four contains the initial lengths in flight hours of the inspection intervals of the four levels of scheduled inspections. LABCD(1) corresponds to the A-level interval; LABCD(2) corresponds to the B-level interval; LABCD(3) corresponds to the C-level interval; and LABCD(4) corresponds to the D-level interval.

CABCD(*) - This one-dimensional real array of size four contains the inspection cost at each level of inspection. CABCD(1) corresponds to the A-level cost; CABCD(2) corresponds to the B-level cost; CABCD(3) corresponds to the C-level cost; and CABCD(4) corresponds to the D-level cost.

SAMPLING - This real variable is the percentage of the fleet to be sampled during a D-level inspection.

LONG.LIST - This alpha variable is read in as "YES" when the long list output option is desired; otherwise, it is read in as "NO."

PCCL - This real variable is the percentage (read in as a decimal fraction) of the element critical crack length at which a crack, which initiated internally, becomes external.

2.2 Long List Data

Occasionally in the standard output, elements will appear with unusually long fatigue cracks or early element failures. It is desirable to have a more complete service history of aircraft with these early element failures than that offered by the standard output. This service history is available through what is called the long list option. This output option is accessed by reading in alpha characters "YES" for the aircraft input variable LONG.LIST. After this input, the element description and identification numbers of the aircraft to be tracked are read in. The input variables for the long list option are listed and described below in the order in which they are read in by SAIFE.

NOE - This integer variable is the number of elements to be processed under the long list option.

ELID(*,*) - This two-dimensional alpha array of size four by NOE identifies each element to be processed. This identification must appear in the first sixteen columns of the data card and must be identical to the description read into the variable ELEMENT(*) described in Section IV, 2.3.

NOAC(*) - This one-dimensional integer array of size NOE is the number of aircraft to be tracked for each corresponding element.

TLID(*,*) - This two-dimensional integer array of size NOE by NOAC(*) contains the identification numbers of the aircraft to be tracked for a particular element.

The above data are read in immediately following the aircraft input data only when the long list option is desired. Normally, the long list option will be used only after a standard run has indicated a problem area. Since the SAIPE program depends on many sequences of random numbers, all elements in the long list run must be in the same sequence as those in the first run. When the long list option is in effect, the standard output is suppressed.

2.3 Element Data

The input variables which are unique to each element and must be read in for each element are listed and described below in the order in which they are read in by SAIPE.

ELEMENT(*) - This one-dimensional alpha array of size four identifies the element being simulated. The total length of this identification cannot exceed sixteen characters.

PREDICTED.LIFE - This real variable is the average element fatigue life in flight hours predicted by analysis. If the actual average fatigue life is known, this variable can be entered as zero. The output format requires that this variable does not exceed 999999.

ACTUAL.AVG.FAT.LIFE - This real variable is the actual average element fatigue life in flight hours determined by fatigue test. If this value is not known before running the simulation, input zero and SAIPE will determine it statistically.

M1.MEAN - This real variable is the average slow crack propagation rate for the element being simulated.

M2.MEAN - This real variable is the average fast crack propagation rate for the element being simulated.

LGHT.TO.FAILURE - This real variable contains the element crack length which corresponds to zero residual strength or level-flight structural failure.

CRIT.CRK.LGT - This real variable is the element critical crack length, that is, the crack length when the crack propagation rate changes from slow to fast.

FSAF.LGT - This real variable is the element fail-safe crack length.

BIRTH.DEFECT.PROBABILITY - This real variable is the probability (expressed as a decimal fraction) that the element has a production defect when the aircraft enters service.

CRR - This integer variable is the element Corrosion Resistance Rating.

SDM.OCCURRENCE.RATE - This real variable is the service damage occurrence rate in occurrences per element per aircraft per flight hour.

LEAD.TIME - This real variable is the time in flight hours between the time when the decision is made to develop a modification and the time when the modification is ready for implementation.

T.FREQ.CHG - This real variable is the factor (expressed as a decimal fraction) by which inspection intervals are decreased because of a fatigue test failure.

S.FREQ.CHG - This real variable is the factor (expressed as a decimal fraction) by which inspection intervals are decreased because of unfavorable service experience.

FREQ.DECREASE - This real variable is the factor (expressed as a decimal fraction) by which inspection intervals are increased because of favorable service experience.

1.PROB - Given that there is fatigue crack initiation, this real variable is the probability that the crack initiates internally. This variable applies to all three allowable fatigue cracks.

C.PROB - Given that there is corrosion initiation, this real variable is the probability that the corrosion initiates internally.

INT.LVL.INS - This alpha variable is the letter designation of the lowest internal inspection level.

EXT.LVL.INS - This alpha variable is the letter designation of the lowest external inspection level.

A.REPAIR.COST - This real variable is the repair cost of a defect detected during an A-level inspection.

B.REPAIR.COST - This real variable is the repair cost of a defect detected during a B-level inspection.

C.REPAIR.COST - This real variable is the repair cost of a defect detected during a C-level inspection.

D.REPAIR.COST - This real variable is the repair cost of a defect detected during a D-level inspection.

MOD.TESTED - This alpha variable indicates whether or not a structural modification is to be fatigue tested. The two acceptable input values are YES or NO.

1ST.TOOLING - This real variable is the tooling cost in the development of the first structural modification.

AD.TOOLING - This real variable is the tooling cost in the development of any additional structural modifications.

1ST.MD.COST - This real variable is the installation cost of the first structural modification.

AD.MD.COST - This real variable is the installation cost of any additional structural modifications.

S.REPAIR.COST - This real variable is the repair cost of a defect detected during a special inspection.

LOCATED.IN.STRESS.CON - This real variable is the probability (expressed as a decimal fraction) that there is corrosion in a stress concentration.

1.CDM.OCCURRENCE.RATE - This real variable is the initial corrosion occurrence rate in occurrences per element per aircraft per flight hour.

2.CDM.OCCURRENCE.RATE - This real variable is the second corrosion occurrence rate in occurrences per element per aircraft per flight hour.

CDM.RATE.CHANGE - This real variable is the aircraft service time in flight hours when the second corrosion occurrence rate takes effect.

2.4 Format Specifications

Most of the input data are entered into SAIFF by the free-form read statement. The program has only three formatted read statements.

The aircraft type identification, the alpha array MODEL, is entered under the format specification 2A4. This identification must be contained in the first eight columns of the first card of the Aircraft Input Data. All subsequent data in this section can appear in any columns and on as many cards as desired. All input values must be separated from one another by at least one blank column and a value cannot be continued on the next card.

The element identification alpha array ELID is entered under the format specification 4A4. This identification must be contained in the first sixteen columns of the long list element data card. Subsequent data can appear in any column and on as many cards as desired. When a second element is to be identified, its description must again appear in the first sixteen columns of the data card.

The element identification alpha array ELEMENT is entered under the format specification 4A4. This identification must be contained in the first sixteen columns of the first card of each set of Element Input Data. As in the Aircraft Input Data, all

subsequent data in this section can appear in any columns and on as many cards as desired.

Sample input data consisting of Aircraft Input Data and five sets of Element Input Data are illustrated in Figure 3. The aircraft type identification is HYBRID. The two cards immediately following the aircraft type card contain the Aircraft Input Data. The five elements shown in the sample are WSC-SWB-AFT-0000, WSC-SWB-AFT-0030, WSC-SWB-AFT-0060, WSC-SWB-AFT-0090, and WSC-SWB-AFT-0127. The Element Input Data for each element begins immediately after the element identification on the same card and terminates on the last card before the next element identification. The card following the last set of Element Input Data must contain EOD in the first four columns.

```
HYBRID
500 60000 150 50 100 5000 0 100 .002 .A .20 .40 1.00 .354 .284325
-8.80901 25 200 1000 12000 1440 7040 13563 19691 .25 NO 1.0
WSC-SWB-AFT-0000 745200 0 A.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 2
13.31E-9 1476 .800 .650 .250 .466 .667 D H 33 7A 144 20A.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
WSC-SWB-AFT-0030 690000 0 A.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 3
13.31E-9 1476 .800 .650 .250 .466 .667 D H 33 7A 144 20A.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
WSC-SWB-AFT-0060 662400 0 A.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 1
13.31E-9 1476 .800 .650 .250 .466 .667 D H 33 7A 144 20A.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
WSC-SWB-AFT-0090 593400 0 A.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 2
13.31E-9 1476 .800 .650 .250 .466 .667 D H 33 7A 144 20A.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
WSC-SWB-AFT-0127 607200 0 A.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 3
13.31E-9 1476 .800 .650 .250 .466 .667 D H 33 7A 144 20A.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
EOD
```

Figure 3. Sample Input Data to Produce Standard Output

Figure 4 illustrates the same data except that the variable LONG.LIST is now entered as "YES." The Long List Input data appears between the Aircraft Input Data and the Element Input Data. The Long List Input causes the program to track aircraft numbers 100, 200, and 300 for the element WSC-SWB-AFT-0000; aircraft numbers 86 and 497 for the element WSC-SWB-AFT-0030; aircraft number 1 for the element WSC-SWB-AFT-0060; aircraft numbers 9, 10, 11, and 12 for the element WSC-SWB-AFT-0090; and aircraft numbers 323, 456, and 472 for the element WSC-SWB-AFT-0127.

3. Output

Each element to be simulated by SAIFE is identified by three groups of alpha characters and one group of numeric characters. The alpha characters define the basic element type and general location on the aircraft, and the numeric characters define the specific location of the element by identifying the wing or fuselage station number. For example, an element identified as WNG-STR-CEN-396 would be a wing stringer located midway between the front and rear spars and centered at wing station 396.

```

HYBRID
500 60000 150 50 100 5000 0 100 .002 .8 .20 .40 1.00 .354 .284325
-8.80901 25 200 1000 12000 1440 7080 13563 19691 .25 YES 1.0
5
WSC-SWB-AFT-0000 3 100 200 300
WSC-SWB-AFT-0030 2 86 497
WSC-SWB-AFT-0060 1 1
WSC-SWB-AFT-0090 4 9 10 11 12
WSC-SWB-AFT-0127 3 323 456 472
WSC-SWB-AFT-0000 745200 0 8.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 2
13.31E-9 1476 .800 .650 .250 .466 .667 D R 33 78 144 208.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
WSC-SWB-AFT-0030 690000 0 8.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 3
13.31E-9 1476 .800 .650 .250 .466 .667 D R 33 78 144 208.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
WSC-SWB-AFT-0060 662400 0 8.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 1
13.31E-9 1476 .800 .650 .250 .466 .667 D R 33 78 144 208.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
WSC-SWB-AFT-0090 593400 0 8.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 2
13.31E-9 1476 .800 .650 .250 .466 .667 D R 33 78 144 208.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
WSC-SWB-AFT-0127 607200 0 8.00E-5 1.60E-3 140. 1.371 15.0 1.19E-5 3
13.31E-9 1476 .800 .650 .250 .466 .667 D R 33 78 144 208.5 YES
10000 10000 11584 11584 416 .056 9.154E-9 7.526E-8 7400
EOD

```

Figure 4. Sample Input Data to Produce Long List Output

The standard program output consists of two parts. The first part consists of the simulation results for each specific element. This part is printed for each set of Element Input Data. The second part consists of a summary of the first parts for an element type. In the example discussed above, WNG-STR-CEN identifies the element type. Whenever the program encounters a set of Element Input Data in which any single character of the three groups of alpha characters differs from those in the previous set of Element Input Data, a summary is printed.

A third output is available as an option. This long list option gives a more complete service history of certain selected aircraft and is discussed in Section IV,2.2. When the long list option is in effect, the standard output is suppressed.

3.1 Element Data

Figure 5 illustrates a sample output for the input shown in Figure 3. The aircraft type identification is the aircraft input array MODEL. The number of aircraft in the fleet is the aircraft input variable SIZE.OF.FLEET. The aircraft service life is the aircraft input variable USAGE.LIFE. The structural element identification is the element input array ELEMENT. The predicted average fatigue life is the element input variable PREDICTED.LIFE. The actual average fatigue life is the element input variable ACTUAL.AVG.FAT.LIFE.

AIRCRAFT TYPE: HYBRID
 AIRCRAFT SERVICE LIFE: 60000 HOURS
 NUMBER OF AIRCRAFT IN FLEET: 500
 STRUCTURAL ELEMENT: WSC-SWB-AFT-0000
 ACTUAL AVERAGE FATIGUE LIFE: 670108 HOURS
 PREDICTED AVERAGE FATIGUE LIFE: 745200 HOURS

NUMBER AND TIME TO INITIATION OF AIRCRAFT DEFECTS

	FIRST CRACK	CORROSION	SERVICE DAMAGE	PRODUCTION DEFECTS
OCCURRENCES	2	1	2	0
MIN(HRS)	32752	37720	32752	---
MAX(HRS)	59789	37720	59789	---
AVG(HRS)	46270	37720	46270	---

NUMBER AND LENGTH OF CRACKS DETECTED AT EACH LEVEL OF INSPECTION

	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
OCCURRENCES	0	1	0	0	0
MIN(IN)	0.	2.22	0.	0.	0.
MAX(IN)	0.	2.22	0.	0.	0.
AVG(IN)	0.	2.22	0.	0.	0.

NUMBER AND AREA OF CORROSION DEFECTS DETECTED AT EACH LEVEL OF INSPECTION

	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
OCCURRENCES	0	0	0	0	1
MIN(SQ.IN)	0.	0.	0.	0.	35.47
MAX(SQ.IN)	0.	0.	0.	0.	35.47
AVG(SQ.IN)	0.	0.	0.	0.	35.47

INSPECTION INTERVALS(HRS)

	INITIAL	25	200	1000	12000
2	25	200	1250	15000	
3	25	200	18750	18750	
4	25	200	1953	23438	
5	25	200	684	8203	
6	25	200	854	10254	
7	25	200	1068	12817	

NUMBER OF SPECIAL INSPECTIONS CONDUCTED: 1
 NUMBER OF STRUCTURAL MODIFICATIONS: 0
 FINAL ACTUAL AVERAGE MODIFIED FATIGUE LIFE: 670108 HOURS
 NUMBER OF AIRCRAFT MODIFIED IN SERVICE: 0

STRUCTURAL FAILURES
 AIRCRAFT NO. _____
 FLT. HOURS _____
 RESIDUAL STRENGTH EQUALS FAIL-SAFE STRENGTH
 AIRCRAFT NO. _____
 FLT. HOURS _____

Figure 5. Sample Standard Output Produced by Input Shown in Figure 3

AIRCRAFT TYPE: HYBRID		AIRCRAFT SERVICE LIFE: 60000 HOURS	
NUMBER OF AIRCRAFT IN FLEET: 500	STRUCTURAL ELEMENT: MSC-SWB-AFT-0030		
PREDICTED AVERAGE FATIGUE LIFE: 690000 HOURS	ACTUAL AVERAGE FATIGUE LIFE: 424261 HOURS		

NUMBER AND TIME TO INITIATION OF AIRCRAFT DEFECTS			
FIRST CRACK	CORROSION	SERVICE DAMAGE	PRODUCTION DEFECTS
-----	-----	-----	-----
1	1	0	0
51286	56541	0	-----
51286	56541	0	-----
51286	56541	0	-----

NUMBER AND LENGTH OF CRACKS DETECTED AT EACH LEVEL OF INSPECTION					
	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
-----	-----	-----	-----	-----	-----
0	0	0	0	0	0
0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.

NUMBER AND AREA OF CORROSION DEFECTS DETECTED AT EACH LEVEL OF INSPECTION					
	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
-----	-----	-----	-----	-----	-----
0	0	0	0	0	0
0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.

INSPECTION INTERVALS(HRS)	
INITIAL	25
2	200
3	200
4	200

NUMBER OF SPECIAL INSPECTIONS CONDUCTED: 0	12000
NUMBER OF STRUCTURAL MODIFICATIONS: 0	15000
FINAL ACTUAL AVERAGE MODIFIED FATIGUE LIFE: 424261 HOURS	18750
NUMBER OF AIRCRAFT MODIFIED IN SERVICE: 0	23438

NUMBER OF SPECIAL INSPECTIONS CONDUCTED: 0	1000
NUMBER OF STRUCTURAL MODIFICATIONS: 0	1250
FINAL ACTUAL AVERAGE MODIFIED FATIGUE LIFE: 424261 HOURS	1563
NUMBER OF AIRCRAFT MODIFIED IN SERVICE: 0	1953

STRUCTURAL FAILURES	RESIDUAL STRENGTH EQUALS FAIL-SAFE STRENGTH
AIRCRAFT NO. -----	AIRCRAFT NO. -----
FLT. HOURS -----	FLT. HOURS -----

Figure 5 - Continued

AIRCRAFT TYPE: HYBRID
 NUMBER OF AIRCRAFT IN FLEET: 500 AIRCRAFT SERVICE LIFE: 60000 HOURS
 STRUCTURAL ELEMENT: MSC-SMB-AFT-0060
 PREDICTED AVERAGE FATIGUE LIFE: 662400 HOURS ACTUAL AVERAGE FATIGUE LIFE: 547824 HOURS

NUMBER AND TIME TO INITIATION OF AIRCRAFT DEFECTS

	FIRST CRACK	CORROSION	SERVICE DAMAGE	PRODUCTION DEFECTS
	-----	-----	-----	-----
OCCURRENCES	2	2	0	0
MIN(MRS)	41301	21125	0	-----
MAX(MRS)	58746	38060	0	-----
AVG(MRS)	50043	29592	0	-----

NUMBER AND LENGTH OF CRACKS DETECTED AT EACH LEVEL OF INSPECTION

	A-LEVEL	H-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
	-----	-----	-----	-----	-----
OCCURRENCES	0	1	0	0	0
MIN(IN)	0.	1.12	0.	0.	0.
MAX(IN)	0.	1.12	0.	0.	0.
AVG(IN)	0.	1.12	0.	0.	0.

NUMBER AND AREA OF CORROSION DEFECTS DETECTED AT EACH LEVEL OF INSPECTION

	A-LEVEL	H-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
	-----	-----	-----	-----	-----
OCCURRENCES	0	1	0	1	0
MIN(SQ.IN)	0.	6.61	0.	17.63	0.
MAX(SQ.IN)	0.	6.61	0.	17.63	0.
AVG(SQ.IN)	0.	6.61	0.	17.63	0.

INSPECTION INTERVALS(MRS)

INITIAL	25	200	1000	12000
2	25	200	1250	15000
3	25	200	1563	18750
4	25	200	1953	23438
5	25	200	684	8203
6	25	200	854	10254
7	25	200	1068	12817
8	25	200	1335	16022
9	25	200	1669	20027

NUMBER OF SPECIAL INSPECTIONS CONDUCTED: 1
 NUMBER OF STRUCTURAL MODIFICATIONS: 0
 FINAL ACTUAL AVERAGE MODIFIED FATIGUE LIFE: 547824 HOURS
 NUMBER OF AIRCRAFT MODIFIED IN SERVICE: 0

STRUCTURAL FAILURES
 AIRCRAFT NO. -----
 RESIDUAL STRENGTH EQUALS FAIL-SAFE STRENGTH
 AIRCRAFT NO. -----
 FLT. HOURS -----

AIRCRAFT TYPE: HYBRID
 NUMBER OF AIRCRAFT IN FLEET: 500 AIRCRAFT SERVICE LIFE: 60000 HOURS
 STRUCTURAL ELEMENT: MSC-SWB-AFT-0090
 PREDICTED AVERAGE FATIGUE LIFE: 593400 HOURS ACTUAL AVERAGE FATIGUE LIFE: 539763 HOURS

NUMBER AND TIME TO INITIATION OF AIRCRAFT DEFECTS

	FIRST CRACK	CORROSION	SERVICE DAMAGE	PRODUCTION DEFECTS
OCCURRENCES	0	6	3	0
MIN(MRS)	1603	5542	1603	----
MAX(MRS)	48952	56707	48952	----
AVG(MRS)	33185	25377	23168	----

NUMBER AND LENGTH OF CRACKS DETECTED AT EACH LEVEL OF INSPECTION

	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
OCCURRENCES	0	2	2	1	0
MIN(IN)	0.	.51	.49	1.05	0.
MAX(IN)	0.	.65	.55	1.05	0.
AVG(IN)	0.	.58	.52	1.05	0.

NUMBER AND AREA OF CORROSION DEFECTS DETECTED AT EACH LEVEL OF INSPECTION

	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
OCCURRENCES	0	2	0	1	0
MIN(SQ.IN)	0.	1.65	0.	17.84	0.
MAX(SQ.IN)	0.	2.08	0.	17.84	0.
AVG(SQ.IN)	0.	1.86	0.	17.84	0.

INSPECTION INTERVALS(MRS)

INITIAL	25	200	1000	12000
2	25	200	1250	15000
3	25	200	1563	18750
4	25	200	1953	23438
5	25	200	684	8203
6	25	200	854	10254
7	25	200	1068	12817
8	25	200	1335	16022
9	25	200	1669	20027

NUMBER OF SPECIAL INSPECTIONS CONDUCTED: 1
 NUMBER OF STRUCTURAL MODIFICATIONS: 0
 FINAL ACTUAL AVERAGE MODIFIED FATIGUE LIFE: 539763 HOURS
 NUMBER OF AIRCRAFT MODIFIED IN SERVICE: 0

STRUCTURAL FAILURES	RESIDUAL STRENGTH EQUALS FAIL-SAFE STRENGTH
AIRCRAFT NO.	AIRCRAFT NO.
-----	-----
FLT. HOURS	FLT. HOURS
-----	-----

Figure 5 - Continued

AIRCRAFT TYPE: MYBRID
 NUMBER OF AIRCRAFT IN FLEET: 500 AIRCRAFT SERVICE LIFE: 60000 HOURS
 STRUCTURAL ELEMENT: MSC-SW8-AFT-0127
 PREDICTED AVERAGE FATIGUE LIFE: 607200 HOURS ACTUAL AVERAGE FATIGUE LIFE: 976224 HOURS

OCCURRENCES	NUMBER AND TIME TO INITIATION OF AIRCRAFT DEFECTS				PRODUCTION DEFECTS
	FIRST CRACK	CORROSION	SERVICE DAMAGE		
1	3	0	0	0	
MIN(MRS)	46529	25520	0	0	
MAX(MRS)	46529	59353	0	0	
AVG(MRS)	46529	43195	0	0	

OCCURRENCES	NUMBER AND LENGTH OF CRACKS DETECTED AT EACH LEVEL OF INSPECTION				SPECIAL
	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	
0	0	1	0	0	0
MIN(IN)	0.	3.33	0.	0.	0.
MAX(IN)	0.	3.33	0.	0.	0.
AVG(IN)	0.	3.33	0.	0.	0.

OCCURRENCES	NUMBER AND AREA OF CORROSION DEFECTS DETECTED AT EACH LEVEL OF INSPECTION				SPECIAL
	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	
0	0	1	0	1	0
MIN(SQ.IN)	0.	1.38	0.	2.96	0.
MAX(SQ.IN)	0.	1.38	0.	2.96	0.
AVG(SQ.IN)	0.	1.38	0.	2.96	0.

INSPECTION INTERVALS(MRS)	RESIDUAL STRENGTH EQUALS FAIL-SAFE STRENGTH			
	AIRCRAFT NO.	FLT. HOURS	AIRCRAFT NO.	FLT. HOURS
INITIAL	25	200	1000	12000
2	25	200	1250	15000
3	25	200	1563	18750
4	25	200	1953	23438
5	25	200	2684	3203
6	25	200	354	10254
7	25	200	1068	12817

NUMBER OF SPECIAL INSPECTIONS CONDUCTED: 1
 NUMBER OF STRUCTURAL MODIFICATIONS: 0
 FINAL ACTUAL AVERAGE MODIFIED FATIGUE LIFE: 976224 HOURS
 NUMBER OF AIRCRAFT MODIFIED IN SERVICE: 0

STRUCTURAL FAILURES
 AIRCRAFT NO. FLT. HOURS

Figure 5 - Continued

AIRCRAFT TYPE: A7E21D

NUMBER OF AIRCRAFT IN FLEET: 500 AIRCRAFT SERVICE LIFE: 80000 HOURS

SUMMARY OF STRUCTURAL ELEMENTS: ACC-500-AFT

NUMBER AND TIME TO INITIATION OF AIRCRAFT DEFECTS

FIRST CRACK	CORROSION	SERVICE DAMAGE	PRODUCTION DEFECTS
12	13	5	0
1603	5582	1483	---
50780	50353	50780	---
49706	33064	32430	---

NUMBER AND LENGTH OF CRACKS DETECTED AT EACH LEVEL OF INSPECTION

	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
OCCURRENCES	0	5	2	1	0
MIN(IN)	0.	.51	.60	1.05	0.
MAX(IN)	0.	3.33	.55	1.05	0.
AVG(IN)	0.	1.57	.52	1.05	0.

NUMBER AND AREA OF CORROSION DEFECTS DETECTED AT EACH LEVEL OF INSPECTION

	A-LEVEL	B-LEVEL	C-LEVEL	D-LEVEL	SPECIAL
OCCURRENCES	0	4	0	3	1
MIN(SQ.IN)	0.	1.38	0.	2.66	35.87
MAX(SQ.IN)	0.	6.61	0.	17.84	35.87
AVG(SQ.IN)	0.	2.93	0.	12.81	35.87

INSPECTION INTERVALS(HRS)

INITIAL	25	1000	12000
SHORTEST	25	684	8203
LONGEST	25	1953	23458

NUMBER OF SPECIAL INSPECTIONS CONDUCTED: 4

NUMBER OF STRUCTURAL MODIFICATIONS: 0

NUMBER OF AIRCRAFT MODIFIED IN SERVICE: 0

AIRCRAFT NO.	STRUCTURAL FAILURES FLT. HOURS	STA. NO.	RESIDUAL STRENGTH EQUALS FAIL-SAFE STRENGTH AIRCRAFT NO.	FLT. HOURS	STA. NO.
--------------	-----------------------------------	----------	---	------------	----------

Figure 5 - Concluded

The number of occurrences in the fleet and the times to initiation of the four types of aircraft defects considered by SAIFE are displayed next. Whenever there is a fatigue crack initiation and there are no other cracks in the element, a first crack is said to have occurred. A single element can experience more than one first crack in its lifetime by having a crack initiation after a repair. Similarly, there can be more than one occurrence of corrosion and service damage, although the program does not allow more than one corrosion or service damage defect to exist simultaneously in the same element.

Production defects are one-time occurrences unless there is a structural modification installed. These, too, can have production defects. The times to defect initiation are measured from the time when the aircraft enters service for the initial defects and from the time when the aircraft was last repaired for subsequent defects.

Next, the number and lengths of cracks detected in the fleet at each level of inspection are printed. These numbers include second and third crack detections. Following the crack detection output are the number and areas of corrosion defects detected in the fleet at each level of inspection. A history of the inspection interval changes is printed next. Each time that the aircraft service experience indicates that either an interval increase or an interval decrease is needed, the new interval values are printed. Although the number of interval changes allowed in the simulation is unlimited, the output array size limits the number printed to 30.

The number of fleet-wide special inspections performed is printed next. Each special inspection is always preceded by a decrease in inspection intervals. Defects detected during a special inspection can cause an additional decrease in inspection intervals. Next, the number of structural modifications developed is printed. This number includes modifications because of fatigue test failures or aircraft service experience. The final actual average fatigue life is printed next. If there have been no modifications, this number will be the same as that at the top of the page. If there have been modifications, this number is the actual average fatigue life of the most recently developed modification.

Shown next is the number of aircraft modified in service. If the only modification developed was due to a fatigue test failure, this number can be zero if the test life was such that retrofits were not required. If there were more than one modification requiring retrofits, this number can be greater than the size of the fleet. Finally, each time an aircraft experiences structural failure or its residual strength reaches its fail-safe strength, the aircraft number and the number of accrued flight hours are printed. The aircraft number is assigned by its relative time of entry into service. Aircraft No. 1 is the first aircraft to enter service.

3.2 Summary Data

The last section of Figure 5 illustrates a sample summary for the element type WSC-SWB-AFT. All the numbers represent a summary of all the specific elements of this type. Except for the shortest and longest inspection intervals, each number in the summary will appear in one of the specific element outputs. As indicated earlier, the number of interval changes allowed in the simulation is unlimited, and the number printed for a specific element is limited to 30. The shortest and longest intervals printed in the summary are determined from the unlimited number of changes occurring in the simulation.

3.3 Long List Data

Figure 6 illustrates a sample long list output for the input shown in Figure 4. For each element the long list headings contain the following: aircraft description, number of aircraft in the fleet, aircraft service life, element description, predicted average fatigue life of the element, actual average fatigue life of the element, and the initial inspection intervals. For each aircraft being tracked, the long list option causes selected information to be printed each time the program control passes to certain events and routines. These events and routines along with the information printed are the following:

Event ENTER.SERVICE - Prints aircraft identification number, number of hours from start of simulation, projected flight hours until crack initiations, and the slow and fast crack growth rates.

Routine INSTALL.MODIFICATION - Prints aircraft identification number, flight time on aircraft, flight hours until crack initiations, and the slow and fast crack growth rates.

Event IN.SERVICE.DAMAGE - Prints aircraft identification number and flight time on aircraft.

Event CORROSION - Prints aircraft identification number, flight time on aircraft, revised slow and fast crack growth rates, revised times until crack initiations, and revised time until failure.

Event 1.STRENGTH.REDUCTION - Prints aircraft identification number, flight time on aircraft, and projected flight hours until element failure.

Event 2.STRENGTH.REDUCTION - Prints aircraft identification number, flight time on aircraft, and projected flight hours until element failure.

Event 3.STRENGTH.REDUCTION - Prints aircraft identification number, flight time on aircraft, and projected flight hours until element failure.

Event 1.ITE - Prints aircraft identification number, length of crack, and flight time on aircraft.

Event 2.ITE - Prints aircraft identification number, length of crack, and flight time on aircraft.

Event 3.ITE - Prints aircraft identification number, length of crack, and flight time on aircraft.

Event D.LEVEL.INSPECTION - Prints aircraft identification number and flight time on aircraft. If an inspection interval increase is implemented at this time, revised intervals are also printed.

Routine EXAMINE - For each defect found, prints size of defect, level of inspection, aircraft identification number, and flight time on aircraft.

Event REACH.FAIL.SAFE.LGT - Prints aircraft identification number and flight time on aircraft.

Event FAILURE - Prints aircraft identification number, flight time on aircraft, sum of crack lengths, and element residual strength.

Event RETIRE.FROM.SERVICE - Prints aircraft identification number and flight time on aircraft.

Event REPAIR - Prints aircraft identification number and projected times to crack initiations.

Event INCREASE.INSPECTION.FREQUENCY - Prints revised C-level and D-level inspection intervals.

Event IMMEDIATE.FLEET.INSPECTION - For each defect found, prints type of defect, size of defect, aircraft identification number, and flight time on aircraft.

AIRCRAFT TYPE: HYBRID

NUMBER OF AIRCRAFT IN FLEET: 500 AIRCRAFT SERVICE LIFE: 60000 HOURS

STRUCTURAL ELEMENT: WSC-SWB-AFT-0000

PREDICTED AVERAGE FATIGUE LIFE: 745200 HOURS ACTUAL AVERAGE FATIGUE LIFE: 670104 HOURS

INITIAL INSPECTION INTERVALS

A-LEVEL	25 HOURS
B-LEVEL	200 HOURS
C-LEVEL	1000 HOURS
D-LEVEL	12000 HOURS

A/C NO. 100 ENTERS SERVICE 4100 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 655534 FLIGHT HOURS
 2ND CRACK INITIATION PROJECTED AT 848614 FLIGHT HOURS
 3RD CRACK INITIATION PROJECTED AT 950200 FLIGHT HOURS
 SLOW CRACK GROWTH RATE = .000000 INCHES/HOUR
 FAST CRACK GROWTH RATE = .001605 INCHES/HOUR

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1250 HOURS
 D-LEVEL INTERVAL NOW 15000 HOURS

A/C NO. 200 ENTERS SERVICE 15050 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 786644 FLIGHT HOURS
 2ND CRACK INITIATION PROJECTED AT 1101450 FLIGHT HOURS
 3RD CRACK INITIATION PROJECTED AT 1312006 FLIGHT HOURS
 SLOW CRACK GROWTH RATE = .000070 INCHES/HOUR
 FAST CRACK GROWTH RATE = .001490 INCHES/HOUR

D-LEVEL INSPECTION PERFORMED ON A/C NO. 100 AT 12000 HOURS

A/C NO. 300 ENTERS SERVICE 24050 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 354101 FLIGHT HOURS
 2ND CRACK INITIATION PROJECTED AT 453140 FLIGHT HOURS
 3RD CRACK INITIATION PROJECTED AT 1209263 FLIGHT HOURS
 SLOW CRACK GROWTH RATE = .000090 INCHES/HOUR
 FAST CRACK GROWTH RATE = .001463 INCHES/HOUR

D-LEVEL INSPECTION PERFORMED ON A/C NO. 200 AT 12000 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1543 HOURS
 D-LEVEL INTERVAL NOW 18750 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 100 AT 27000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 300 AT 12000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 200 AT 24000 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1953 HOURS
 D-LEVEL INTERVAL NOW 24000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 100 AT 45750 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 300 AT 30750 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 200 AT 42750 HOURS

A/C NO. 100 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS

A/C NO. 200 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 300 AT 54188 HOURS

A/C NO. 300 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS

INSPECTION INTERVAL DECREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 684 HOURS
 D-LEVEL INTERVAL NOW 8203 HOURS

a. Structural Element: WSC-SWB-AFT-0000

Figure 6. Sample Long List Output Produced by Input Shown in Figure 4

ALERT 41 - SPECIAL INSPECTION PERFORMED

INSPECTION INTERVAL INCREASE IMPLEMENTED
C-LEVEL INTERVAL NOW 850 HOURS
D-LEVEL INTERVAL NOW 10250 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
C-LEVEL INTERVAL NOW 1000 HOURS
D-LEVEL INTERVAL NOW 12017 HOURS

a. Structural Element: WSC-SWB-AFT-0000 (Concluded)

AIRCRAFT TYPE: HYBRID

NUMBER OF AIRCRAFT IN FLEET: 500

AIRCRAFT SERVICE LIFE: 60000 HOURS

STRUCTURAL ELEMENT: WSC-SWB-AFT-0030

PREDICTED AVERAGE FATIGUE LIFE: 60000 HOURS

ACTUAL AVERAGE FATIGUE LIFE: 424261 HOURS

INITIAL INSPECTION INTERVALS

A-LEVEL 25 HOURS
B-LEVEL 200 HOURS
C-LEVEL 1000 HOURS
D-LEVEL 12000 HOURS

A/C NO. 86 ENTERS SERVICE 4400 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 514080 FLIGHT HOURS
2ND CRACK INITIATION PROJECTED AT 644584 FLIGHT HOURS
3RD CRACK INITIATION PROJECTED AT 691441 FLIGHT HOURS
SLOW CRACK GROWTH RATE = .000073 INCHES/HOUR
FAST CRACK GROWTH RATE = .001468 INCHES/HOUR

INSPECTION INTERVAL INCREASE IMPLEMENTED
C-LEVEL INTERVAL NOW 1250 HOURS
D-LEVEL INTERVAL NOW 15000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 86 AT 12000 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
C-LEVEL INTERVAL NOW 1563 HOURS
D-LEVEL INTERVAL NOW 18750 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 86 AT 27000 HOURS

A/C NO. 497 ENTERS SERVICE 44750 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 514414 FLIGHT HOURS
2ND CRACK INITIATION PROJECTED AT 548364 FLIGHT HOURS
3RD CRACK INITIATION PROJECTED AT 663849 FLIGHT HOURS
SLOW CRACK GROWTH RATE = .000073 INCHES/HOUR
FAST CRACK GROWTH RATE = .001468 INCHES/HOUR

INSPECTION INTERVAL INCREASE IMPLEMENTED
C-LEVEL INTERVAL NOW 1953 HOURS
D-LEVEL INTERVAL NOW 23438 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 86 AT 45750 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 497 AT 12000 HOURS

A/C NO. 86 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 497 AT 35438 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 497 AT 58875 HOURS

A/C NO. 497 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS

b. Structural Element: WSC-SWB-AFT-0030

Figure 6 - Continued

AIRCRAFT TYPE: HYABID

NUMBER OF AIRCRAFT IN FLEET: 500 AIRCRAFT SERVICE LIFE: 60000 HOURS

STRUCTURAL ELEMENT: WSC-SWB-AFT-0060

PREDICTED AVERAGE FATIGUE LIFE: 662400 HOURS ACTUAL AVERAGE FATIGUE LIFE: 547424 HOURS

INITIAL INSPECTION INTERVALS

A-LEVEL	25 HOURS
H-LEVEL	200 HOURS
C-LEVEL	1000 HOURS
D-LEVEL	12000 HOURS

A/C NO. 1 ENTERS SERVICE 150 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 379222 FLIGHT HOURS
 2ND CRACK INITIATION PROJECTED AT 1099412 FLIGHT HOURS
 3RD CRACK INITIATION PROJECTED AT 1158443 FLIGHT HOURS
 SLOW CRACK GROWTH RATE = .000089 INCHES/HOUR
 FAST CRACK GROWTH RATE = .001774 INCHES/HOUR

D-LEVEL INSPECTION PERFORMED ON A/C NO. 1 AT 12000 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1250 HOURS
 D-LEVEL INTERVAL NOW 15000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 1 AT 24000 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1563 HOURS
 D-LEVEL INTERVAL NOW 18750 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 1 AT 39000 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1953 HOURS
 D-LEVEL INTERVAL NOW 23438 HOURS

INSPECTION INTERVAL DECREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 684 HOURS
 D-LEVEL INTERVAL NOW 8203 HOURS

FLEET WIDE SPECIAL INSPECTION PERFORMED

D-LEVEL INSPECTION PERFORMED ON A/C NO. 1 AT 53806 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 854 HOURS
 D-LEVEL INTERVAL NOW 10254 HOURS

A/C NO. 1 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1068 HOURS
 D-LEVEL INTERVAL NOW 12817 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1335 HOURS
 D-LEVEL INTERVAL NOW 16022 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1669 HOURS
 D-LEVEL INTERVAL NOW 20027 HOURS

c. Structural Element: WSC-SWB-AFT-0060

Figure 6 - Continued

AIRCRAFT TYPE: MYRIID
 NUMBER OF AIRCRAFT IN FLEET: 500
 AIRCRAFT SERVICE LIFE: 60000 HOURS
 STRUCTURAL ELEMENT: WSC-SWB-AFT-0090
 PREDICTED AVERAGE FATIGUE LIFE: 543000 HOURS
 ACTUAL AVERAGE FATIGUE LIFE: 519763 HOURS

INITIAL INSPECTION INTERVALS

A-LEVEL 25 HOURS
 B-LEVEL 200 HOURS
 C-LEVEL 1000 HOURS
 D-LEVEL 12000 HOURS

A/C NO. 9 ENTERS SERVICE 550 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 479151 FLIGHT HOURS
 2ND CRACK INITIATION PROJECTED AT 1077594 FLIGHT HOURS
 3RD CRACK INITIATION PROJECTED AT 1249581 FLIGHT HOURS
 SLOW CRACK GROWTH RATE = .000083 INCHES/HOUR
 FAST CRACK GROWTH RATE = .001657 INCHES/HOUR

A/C NO. 10 ENTERS SERVICE 600 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 454008 FLIGHT HOURS
 2ND CRACK INITIATION PROJECTED AT 704099 FLIGHT HOURS
 3RD CRACK INITIATION PROJECTED AT 840003 FLIGHT HOURS
 SLOW CRACK GROWTH RATE = .000074 INCHES/HOUR
 FAST CRACK GROWTH RATE = .001445 INCHES/HOUR

A/C NO. 11 ENTERS SERVICE 650 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 370220 FLIGHT HOURS
 2ND CRACK INITIATION PROJECTED AT 898443 FLIGHT HOURS
 3RD CRACK INITIATION PROJECTED AT 114317 FLIGHT HOURS
 SLOW CRACK GROWTH RATE = .000089 INCHES/HOUR
 FAST CRACK GROWTH RATE = .001782 INCHES/HOUR

A/C NO. 12 ENTERS SERVICE 700 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 604293 FLIGHT HOURS
 2ND CRACK INITIATION PROJECTED AT 661954 FLIGHT HOURS
 3RD CRACK INITIATION PROJECTED AT 724727 FLIGHT HOURS
 SLOW CRACK GROWTH RATE = .000074 INCHES/HOUR
 FAST CRACK GROWTH RATE = .001474 INCHES/HOUR

D-LEVEL INSPECTION PERFORMED ON A/C NO. 9 AT 12000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 10 AT 12000 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED

C-LEVEL INTERVAL NOW 1250 HOURS
 D-LEVEL INTERVAL NOW 15000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 11 AT 12000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 12 AT 12000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 9 AT 24000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 10 AT 27000 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED

C-LEVEL INTERVAL NOW 1563 HOURS
 D-LEVEL INTERVAL NOW 18750 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 11 AT 27000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 12 AT 27000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 9 AT 39000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 10 AT 45750 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED

C-LEVEL INTERVAL NOW 1953 HOURS
 D-LEVEL INTERVAL NOW 23418 HOURS

d. Structural Element: WSC-SWB-AFT-0090

Figure 6 - Continued

D-LEVEL INSPECTION PERFORMED ON A/C NO. 11 AT 45750 HOURS
 D-LEVEL INSPECTION PERFORMED ON A/C NO. 12 AT 45750 HOURS
 INSPECTION INTERVAL DECREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 654 HOURS
 D-LEVEL INTERVAL NOW 8203 HOURS
 FIRST AIDE SPECIAL INSPECTION PERFORMED
 D-LEVEL INSPECTION PERFORMED ON A/C NO. 9 AT 46500 HOURS
 D-LEVEL INSPECTION PERFORMED ON A/C NO. 10 AT 53453 HOURS
 INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 854 HOURS
 D-LEVEL INTERVAL NOW 10254 HOURS
 D-LEVEL INSPECTION PERFORMED ON A/C NO. 11 AT 54951 HOURS
 D-LEVEL INSPECTION PERFORMED ON A/C NO. 12 AT 54951 HOURS
 D-LEVEL INSPECTION PERFORMED ON A/C NO. 9 AT 54703 HOURS
 A/C NO. 9 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS
 A/C NO. 10 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS
 A/C NO. 11 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS
 A/C NO. 12 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS
 INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1068 HOURS
 D-LEVEL INTERVAL NOW 12817 HOURS
 INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1335 HOURS
 D-LEVEL INTERVAL NOW 14022 HOURS
 INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1669 HOURS
 D-LEVEL INTERVAL NOW 20027 HOURS

d. Structural Element: WSC-SWB-AFT-0090 (Concluded)

AIRCRAFT TYPE: MYH10
 NUMBER OF AIRCRAFT IN FLEET: 500 AIRCRAFT SERVICE LIFE: 60000 HOURS
 STRUCTURAL ELEMENT: WSC-SWB-AFT-0127
 PREDICTED AVERAGE FATIGUE LIFE: 607200 HOURS ACTUAL AVERAGE FATIGUE LIFE: 976224 HOURS
 INITIAL INSPECTION INTERVALS
 A-LEVEL 25 HOURS
 H-LEVEL 200 HOURS
 C-LEVEL 1000 HOURS
 D-LEVEL 12000 HOURS
 INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1250 HOURS
 D-LEVEL INTERVAL NOW 15000 HOURS
 A/C NO. 121 ENTERS SERVICE 27350 HOURS FROM START OF SIMULATION
 1ST CRACK INITIATION PROJECTED AT 1151052 FLIGHT HOURS
 2ND CRACK INITIATION PROJECTED AT 1562644 FLIGHT HOURS
 3RD CRACK INITIATION PROJECTED AT 1783574 FLIGHT HOURS
 SLOW CRACK GROWTH RATE = .000074 INCHES/HOUR
 FAST CRACK GROWTH RATE = .001487 INCHES/HOUR
 INSPECTION INTERVAL INCREASE IMPLEMENTED
 C-LEVEL INTERVAL NOW 1563 HOURS
 D-LEVEL INTERVAL NOW 18750 HOURS

e. Structural Element: WSC-SWB-AFT-0127

Figure 6 - Continued

D-LEVEL INSPECTION PERFORMED ON A/C NO. 323 AT 12000 HOURS

A/C NO. 456 ENTERS SERVICE 40650 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 448494 FLIGHT HOURS
2ND CRACK INITIATION PROJECTED AT 1391054 FLIGHT HOURS
3RD CRACK INITIATION PROJECTED AT 1602494 FLIGHT HOURS
SLOW CRACK GROWTH RATE = .000094 INCHES/HOUR
FAST CRACK GROWTH RATE = .001907 INCHES/HOUR

A/C NO. 472 ENTERS SERVICE 42250 HOURS FROM START OF SIMULATION

1ST CRACK INITIATION PROJECTED AT 1436062 FLIGHT HOURS
2ND CRACK INITIATION PROJECTED AT 2101312 FLIGHT HOURS
3RD CRACK INITIATION PROJECTED AT 2294671 FLIGHT HOURS
SLOW CRACK GROWTH RATE = .000066 INCHES/HOUR
FAST CRACK GROWTH RATE = .001319 INCHES/HOUR

INSPECTION INTERVAL INCREASE IMPLEMENTED

C-LEVEL INTERVAL NOW 1453 HOURS

D-LEVEL INTERVAL NOW 23434 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 456 AT 12000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 472 AT 12000 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 323 AT 30750 HOURS

INSPECTION INTERVAL DECREASE IMPLEMENTED

C-LEVEL INTERVAL NOW 684 HOURS

D-LEVEL INTERVAL NOW 4203 HOURS

FLEET WIDE SPECIAL INSPECTION PERFORMED

INSPECTION INTERVAL INCREASE IMPLEMENTED

C-LEVEL INTERVAL NOW 854 HOURS

D-LEVEL INTERVAL NOW 10254 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 323 AT 45041 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 456 AT 31741 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 472 AT 30141 HOURS

INSPECTION INTERVAL INCREASE IMPLEMENTED

C-LEVEL INTERVAL NOW 1068 HOURS

D-LEVEL INTERVAL NOW 12817 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 323 AT 55295 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 456 AT 41995 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 472 AT 40395 HOURS

A/C NO. 323 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 456 AT 54812 HOURS

D-LEVEL INSPECTION PERFORMED ON A/C NO. 472 AT 53212 HOURS

A/C NO. 456 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS

A/C NO. 472 RETIRED FROM SERVICE AT 60000 FLIGHT HOURS

e. Structural Element: WSC-SWB-AFT-0127 (Concluded)

Figure 6 - Concluded

APPENDIX A

DETAILED PROGRAM DESCRIPTION

In the following detailed description of the SAIFE program, each event and routine in the program is presented separately. Except for the PREAMBLE, each presentation consists of a description, the definition of the local variables, if any, and a flow chart to illustrate the logic of the event or routine.

APPENDIX A

TABLE OF CONTENTS

<u>Section</u>		<u>Page</u>
1.	PREAMBLE	33
2.	MAIN	40
3.	INITIALIZATION	44
4.	SUM.INITIALIZE	45
5.	REAL.LIFE	46
6.	ENTER.SERVICE	48
7.	FATIGUE.LIFE.SCATTER	51
8.	INSTALL.MODIFICATION	53
9.	IN.SERVICE.DAMAGE	55
10.	T.IMPLEMENT.MOD	57
11.	PREEDICT.SERVICE.DAMAGE	58
12.	PREDICT.CORROSION	59
13.	COROSION	61
14.	RATE	63
15.	1.STRENGTH.REDUCTION	65
16.	2.STRENGTH.REDUCTION	69
17.	3.STRENGTH.REDUCTION	74
18.	1.ITE	79
19.	2.ITE	80
20.	3.ITE	81
21.	INSPECTION.SCHEDULER	82
22.	A.LEVEL.INSPECTION	84
23.	B.LEVEL.INSPECTION	85
24.	C.LEVEL.INSPECTION	87
25.	D.LEVEL.INSPECTION	89
26.	EXAMINE	92
27.	PODD	96
28.	CANCEL.SCHEDULED.INSPECTIONS	97
29.	REACH.FAIL.SAFE.LGT	99
30.	FAILURE	100
31.	RETIRE.FROM.SERVICE	101
32.	REPAIR	102
33.	T.INSPECTION.INCREASE	108
34.	INCREASE.INSPECTION.FREQUENCY	109
35.	IMMEDIATE.FLEET.INSPECTION	110
36.	DECISION.ON.MOD	112
37.	IMPLEMENT.MODIFICATION	115
38.	DISPLAY.OUTPUT	117
39.	SUMMARY	118

1. PREAMBLE

The PREAMBLE is the definition section of a SIMSCRIPT program. All global variables and global arrays are defined. Temporary entities are defined and tally statistics are identified. Event notices and functions are defined and an event priority order is set. The global real variables are listed and described below in the order in which they appear in the PREAMBLE. Global real variables which are input variables are not included here but can be found in the input section.

Global Real Variables

FLEET.STR.RED - This variable is the sum of crack lengths found in the fleet since the last inspection frequency change.

1AAFL - This variable is the actual average fatigue life of the element design determined in the MAIN program.

CRRF - Assigned a value in routine INITIALIZATION according to the element corrosion resistance rating, this variable is multiplied by the aircraft corrosion growth rate to give the element corrosion growth rate.

COST.OF.REPAIRS - This variable is the sum of repair costs for the fleet since the last modification.

FIXIT.COST - This variable is the cost of repairing a defect found at a particular inspection level. Its value is set in the inspection events.

CHG.FREQ.TIME - This variable is set equal to TIME.V whenever an inspection interval change is scheduled in the event REPAIR.

1CRKT - Each time a first crack occurs, this variable is set equal to the service time on the aircraft.

1CORT - Each time corrosion occurs, this variable is set equal to the service time on the aircraft.

1SDT - Each time service damage occurs, this variable is set equal to the service time on the aircraft.

ACRKL, BCRKL, CCRKL, DCRKL, SCRKL - Each time a crack is found during an A-level, B-level, C-level, D-level, or Special inspection, the corresponding variable is set equal to the crack length.

ACA, BCA, CCA, DCA, SCA - Each time corrosion is found during an A-level, B-level, C-level, D-level, or Special inspection, the corresponding variable is set equal to the corrosion area.

AIRFRAME.TIME - This variable is the number of flight hours accumulated since the last modification for aircraft no longer in service.

G1CRK - Each time a first crack occurs, this variable is set equal to the service time on the aircraft.

G1COR - Each time corrosion occurs, this variable is set equal to the service time on the aircraft.

G1SD - Each time service damage occurs, this variable is set equal to the service time on the aircraft.

GACRK, GBCRK, GCCRK, GDCRK, GSCRK - Each time a crack is found during an A-level, B-level, C-level, D-level, or Special inspection, the corresponding variable is set equal to the crack length.

GACA, GBCA, GCCA, GDCA, GSCA - Each time corrosion is found during an A-level, B-level, C-level, D-level, or Special inspection, the corresponding variable is set equal to the corrosion area.

CINSL, DINSL - Each time there is an inspection interval change, these variables are set equal to the C-level and D-level intervals, respectively.

KSMP - This variable is set equal to 1.0 in the A-level, B-level, and C-level inspection events and set equal to the D-level sampling percentage in the D-level inspection event.

The global integer variables are listed next. Again, input variables are not included in this list.

Global Integer Variables

ID - In each event and routine, this variable is the identification number of the aircraft being processed.

IDCK - This variable is initialized to zero and incremented by one each time an aircraft enters service.

I - This variable is used as a local index or array subscript in different locations in the program.

COUNT.ELEMENT - Each time new element data is read in, this variable is incremented by one.

NICHG - This variable is the number of times that the inspection intervals have changed.

LHTA - This variable is the identification number of the aircraft among the ten high-time aircraft with the fewest flight hours.

1, NUM, OF, RETIRE - This variable is the number of aircraft that have been retired from service.

2, NUM, OF, CRASH - This variable is the number of aircraft which have been removed from service because of structural failure.

ITRN1 - This variable is the numeric identification of the lowest internal level of inspection.

EXT, INSP, LEVEL - This variable is the numeric identification of the lowest external level of inspection.

ILL, ILL - These variables are the numeric identifications of the lowest internal and external levels of inspection, respectively. If either of these variables is less than three, it is set equal to three.

TO, RE, MODIFIED - This variable is the number of aircraft with a pending retrofit modification.

BLN, MODIFIED - This variable is the number of aircraft that have had a current retrofit modification installed.

FDCK - This variable is the number of aircraft in service when a modification is implemented because of a fatigue test failure.

OCR, OCOR, OSDM, OPD - These variables are the number of occurrences of first cracks, corrosion, service damage, and production defects, respectively, for a particular element.

OSCR, OSCO - These variables are the number of cracks and corrosion defects, respectively, detected during a special inspection for a particular element.

NSIC - This variable is the number of special inspections conducted for a particular element.

NSMD - This variable is the number of aircraft modified in service for a particular element.

NSFL - This variable is the number of aircraft experiencing structural failure for a particular element.

NMD - This variable is the number of structural modifications made on a particular element.

NRFS - This variable is the number of aircraft with the residual strength for a particular element reaching the fail-safe strength.

SNRES - This variable is the number of aircraft with the residual strength for a particular element type reaching the fail-safe strength.

J - This variable is used as a local index or array subscript in different locations in the program.

LDA - If the long list option is in effect, this variable is the ascending sequential position of the element being processed among those elements read in under the long list option.

GOLCR, GOCOR, GOSDM - These variables are the number of occurrences of first cracks, corrosion damage, and service damage, respectively, for a particular element type.

GOSCR, GOSCO - These variables are the number of cracks and corrosion defects, respectively, detected during a special inspection for a particular element type.

GOPD - This variable is the number of occurrences of production defects for a particular element type.

SNSIC - This variable is the number of special inspections conducted for a particular element type.

SNMD - This variable is the number of structural modifications made on a particular element type.

SNSMD - This variable is the number of aircraft modified in service for a particular element type.

SNSFL - This variable is the number of aircraft experiencing structural failure for a particular element type.

The real arrays are listed next. Unless otherwise noted, all arrays are 1-dimensional. As before, input arrays are not included in this list.

Real Arrays

C.INTERVAL, D.INTERVAL - The elements of these arrays are the current C-level and D-level inspection intervals for each aircraft in the fleet.

ABCD - This array is of size four and contains the most recent intervals for each of the four levels of inspection.

CKREP.TIME - This array is the simulation time of the most recent crack repair for each aircraft.

CORRP.TIME - This array is the simulation time of the most recent corrosion repair for each aircraft.

LAST.SD - This array is the simulation time of the most recent occurrence of service damage for each aircraft.

OCCUR.MOD - This array is the simulation time when the most recent modification was installed for each aircraft.

MSR, MFR - These arrays are the slow and fast crack growth rates, respectively, for each aircraft.

SC, SD - These arrays contain each of the inspection interval changes for the C-level and D-level, respectively.

NRN - This array is the random number selected to calculate the time until structural failure for each aircraft.

MRDD - This array is the simulation time of the most recently detected defect at either a C-level or D-level inspection for each of the ten high-time aircraft.

GCRI - This array is the corrosion multiplying factor for each aircraft.

The following are the integer arrays. Again, unless otherwise noted, all arrays are 1-dimensional and input arrays are not included.

Integer Arrays

A1SR, A2SR, A3SR - These arrays contain the event notice identification numbers for each aircraft for the events 1.STRENGTH.REDUCTION, 2.STRENGTH.REDUCTION, and 3.STRENGTH.REDUCTION, respectively.

AF - This array is the event notice identification number for each aircraft for event FAILURE.

AIRPLANE - This array is the temporary entity identification number for each aircraft.

AAL, ABL, ACL, ADL - These arrays are the event notice identification numbers for each aircraft for the events A.LEVEL.INSPECTION, B.LEVEL.INSPECTION, C.LEVEL.INSPECTION, and D.LEVEL.INSPECTION, respectively.

AC, ATII - These arrays are the event notice identification numbers for each aircraft for events COROSION and T.INSPECTION. INCREASE, respectively.

ACID - This array contains the identification numbers of those aircraft experiencing structural failure for a particular element.

OICR, OICO - These arrays are the number of cracks and corrosion defects, respectively, detected at each of the four levels of inspection for a particular element.

SACID - This array contains the identification numbers of those aircraft experiencing structural failure for a particular element type.

GOICR, GOICO - These arrays are the number of cracks and corrosion defects, respectively, detected at each of the four levels of inspection for a particular element type.

HLTIME, ACRFT - This array contains the identification numbers of the ten high-time aircraft.

APID - This array contains the identification numbers of those aircraft with a particular element whose residual strength has reached the fail-safe strength.

SAPID - This array contains the identification numbers of those aircraft with a particular element type whose residual strength has reached the fail-safe strength.

STIM - This array contains the flight hours on each aircraft when the residual strength for a particular element reaches the fail-safe strength.

SSTIM - This array contains the flight hours on each aircraft when the residual strength for a particular element type reaches the fail-safe strength.

FLTHR - This array contains the flight hours on each aircraft when structural failure occurs for a particular element.

SFLTHR - This array contains the flight hours on each aircraft when structural failure occurs for a particular element type.

ARFSL - This array is the event notice identification number for each aircraft for event REACH.FAIL.SAFE.1GT.

A1E, A2E, A3E - These arrays are the event notice identification numbers for each aircraft for events 1.ITE, 2.ITE, and 3.ITE, respectively.

The global alpha arrays are listed next. As before, input arrays are not included in this list.

Global Alpha Arrays

1.CR.EXISTS, 2.CR.EXISTS, 3.CR.EXISTS - The elements of these arrays are set equal to "YES" for each aircraft whenever there is a first, second, and third crack initiation, respectively.

CO.EXISTS - This array is set equal to "YES" for each aircraft when it has corrosion initiation.

SD.SCH - This array is set equal to "YES" for each aircraft that has event IN.SERVICE.DAMAGE scheduled.

SSTAN - This array is the station number which identifies each aircraft experiencing the failure of a particular element type.

SELNB - This array is the station number which identifies each aircraft with a particular element type whose residual strength has reached the fail-safe strength.

AIL, FSH - This array is set equal to "YES" for each aircraft when events REACH.FAIL.SAFE.LGT and FAILURE, respectively, are scheduled.

IE1, IE2, IE3 - This array is set equal to "YES" for each aircraft that has events 1.IE, 2.IE, and 3.IE, respectively, scheduled.

TMOD.PENDING - This array is set equal to "YES" for each aircraft that has a modification pending because of a fatigue test failure.

SMOD.PENDING - This array is set equal to "YES" for each aircraft that has a modification pending because of service experience.

INSP.SCH - This array is set equal to "YES" for each aircraft that has inspections below the overhaul level scheduled.

1.INT, 2.INT, 3.INT - These arrays are set equal to "YES" for each aircraft that has a first crack, second crack, or third crack, respectively, initiated internally.

C.INT - This array is set equal to "YES" for each aircraft that has corrosion initiated internally.

The temporary entity definitions and tally statements are self-explanatory. The events, functions, and routines are described in detail in the following sections.

2. MAIN

2.1 Description

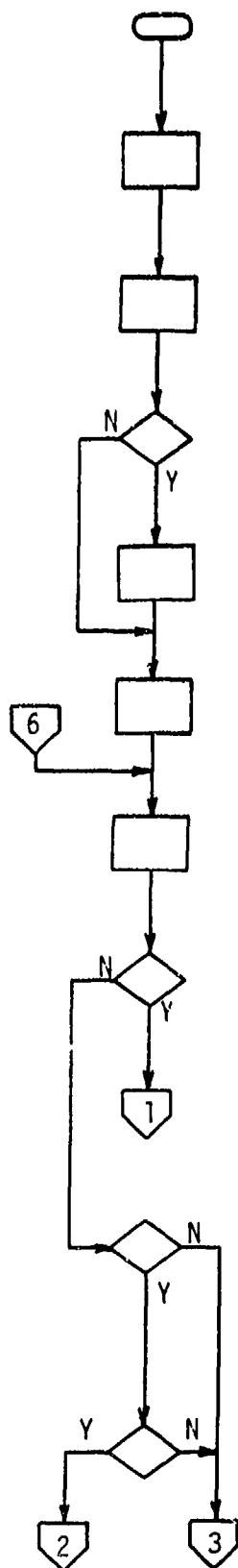
In the MAIN program, space is reserved for all global arrays. The following operations are performed in the order given: all input data is read in; the actual average fatigue life of the element type is calculated; the necessity of a structural modification because of a fatigue test failure is determined; the first event ENTER.SERVICE is scheduled; and the simulation is initiated.

2.2 Local Variables

NETS - This real variable is the time in flight hours from when the second production rate goes into effect to when the last aircraft enters service.

SATL - This real variable is the earliest simulation time at which a structural modification because of a fatigue test failure is ready for installation.

2.3 Flow Chart



MAIN.

Reserve aircraft arrays.

Read aircraft data.

Does LONG.LIST="YES" ?

Read in elements and aircraft to be tracked.

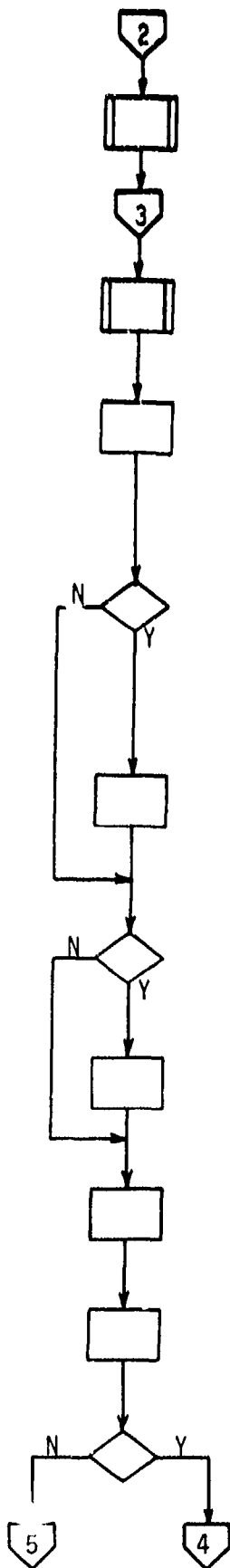
Reserve element arrays.

Read element data.

Does ELEMENT(1)="EOD" ?

Is this element a different element type
than previous element?

Does LONG.LIST="NO" ?



Call routine SUMMARY.

Call routine INITIALIZATION.

Calculate actual average fatigue life of element type.

Does fatigue test failure occur at less than twice the aircraft service life?

Schedule a structural modification.

Is long list option in effect?

Print long list headings.

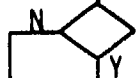
Schedule event ENTER.SERVICE.

Start simulation.

Does LONG.LIST="NO"?



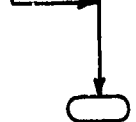
Call routine DISPLAY.OUTPUT.



Does LONG.LIST="NO" ?



Call routine SUMMARY.



END.

3. INITIALIZATION

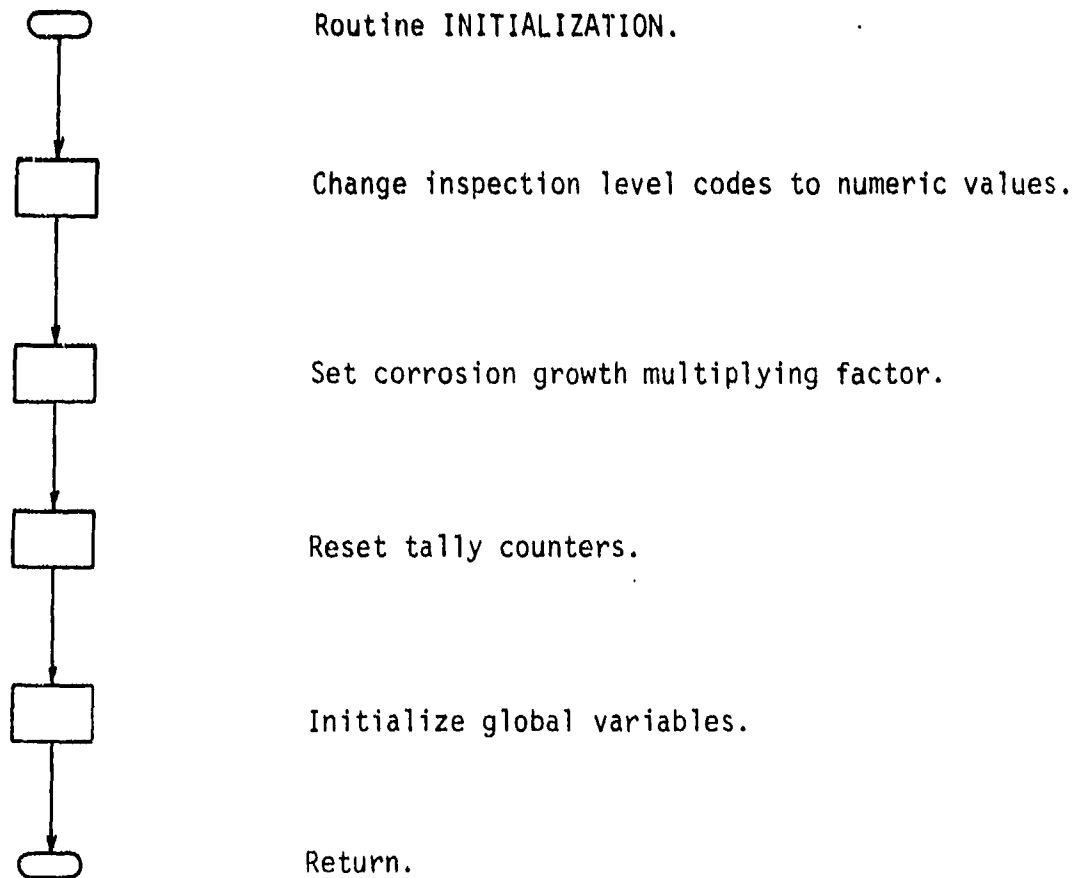
3.1 Description

This routine is called immediately after reading each new set of element input data. This routine changes the inspection level codes to numeric values, sets the corrosion growth multiplying factor based on the corrosion resistance rating, and resets the tally counters. It also initializes all the element global variables which are not part of the input. This routine is called from the MAIN program.

3.2 Local Variables

There are no local variables in this routine.

3.3 Flow Chart



4. SUM.INITIALIZE

4.1 Description

This routine is called each time a new element type is read in. The element type is identified by the first twelve characters of the element identification. This routine initializes the global variables and resets the tally counters. This routine is called from the MAIN program.

4.2 Local Variables

There are no local variables in this routine.

4.3 Flow Chart



Routine SUM.INITIALIZE.

Initialize global variables.

Reset tally counters.

Return.

5. REAL.LIFE

5.1 Description

This routine accepts (1) the predicted average fatigue life of a particular element design and (2) the mean and standard deviation of the log-normal distribution of the ratio of the actual average fatigue life to the predicted average fatigue life. A random selection is made from the distribution and multiplied by the predicted average fatigue life. The resulting actual average fatigue life is returned to the calling routine. REAL.LIFE can be called from the MAIN program and events IMPLEMENT.MODIFICATION and T.IMPLEMENT.MOD.

5.2 Local Variables

MEAN - This real variable, whose value is passed from the calling routine, is the mean of the ratio distribution.

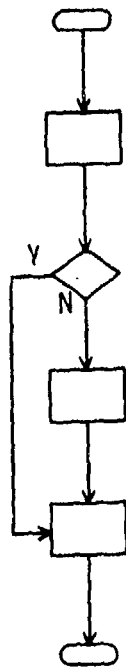
RATIO - This real variable, determined to be log-normally distributed, is the ratio of the actual fatigue life of an element design to its predicted fatigue life.

STD.DEV - This real variable, whose value is passed from the calling routine, is the standard deviation of the ratio distribution.

PDL - This real variable is the design predicted average fatigue life passed from the calling routine.

RFL - This real variable is the element actual average fatigue life which is returned to the calling routine.

5.3 Flow Chart



Routine REAL.LIFE.

Randomly select RATIO from distribution.

Is RATIO within allowable limits?

Set RATIO to limit.

Calculate actual average fatigue life.

Return with actual average fatigue life.

6. ENTER.SERVICE

6.1 Description

This event represents the entry into service of a new aircraft. The temporary entity AIRCRAFT is created and identified by the variable AIRPLANE(ID). The entity attributes TAIL.ID and ENTRY.TIME are defined and the AIRPLANE is filed in the set ACTIVE.FLEET. The routine FATIGUE.LIFE.SCATTER is called to determine the times to first, second, and third crack initiations. The slow and fast crack growth rates are calculated. The times to corrosion initiation and service damage are calculated. If either of these times is less than the service life of the aircraft, the corresponding defect is scheduled. If there is a production defect, the time to first crack initiation is replaced by a time drawn from a distribution of times to crack initiation of aircraft with production defects. If the long list option is in effect for each aircraft being tracked, this routine prints the following: (1) aircraft identification and time it enters service, (2) times to crack initiations, and (3) slow and fast crack growth rates. Crack initiations, D-level inspection, and retirement from service are also scheduled. If the present aircraft is not the last aircraft of the fleet, another ENTER.SERVICE is scheduled. This event can only be scheduled in the MAIN program and within itself.

6.2 Local Variables

DEFECT.LIFE - This real variable is the time to first crack initiation when the aircraft has a production defect.

HOURS.TO.CORROSION - This real variable is the time to corrosion initiation.

SECOND.LIFE - This real variable is the time to second crack initiation.

STD.SLOW - This real variable is the standard deviation of the distribution of slow crack growth rates.

FIRST.LIFE - This real variable is the time to first crack initiation when the aircraft has no production defect.

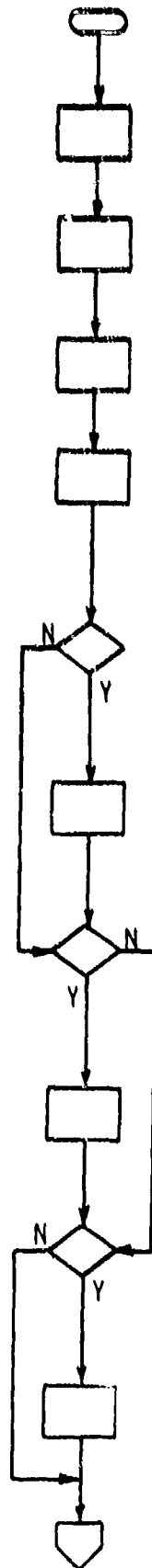
OURS.TO.SERVICE.DAMAGE - This real variable is the time to service damage occurrence.

RN - This real variable is a uniformly distributed random number between zero and one.

STD.FAST - This real variable is the standard deviation of the distribution of fast crack growth rates.

THIRD.LIFE - This real variable is the time to third crack initiation.

6.3 Flow Chart



Event ENTER SERVICE.

Create an AIRCRAFT and file in ACTIVE.FLEET.

Determine times to first three crack initiations.

Calculate slow and fast crack growth rates.

Determine times to corrosion initiation and service damage.

Is time to corrosion initiation less than aircraft service life?

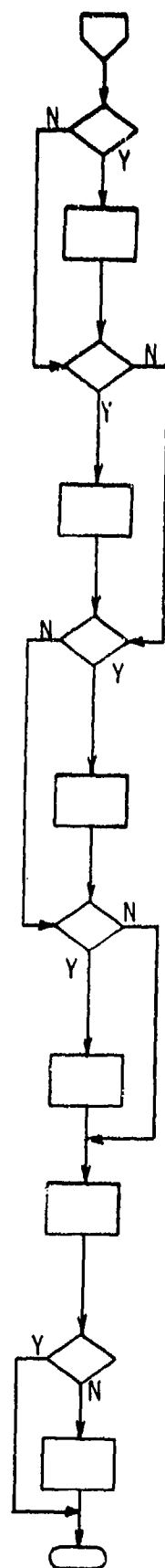
Schedule a corrosion initiation.

Is time to service damage less than aircraft service life?

Schedule a service damage occurrence.

Is random number less than probability of a production defect?

Determine new time to first crack initiation.



Is long list option in effect?

Print long list output if aircraft is one being tracked.

Is time to first crack initiation less than aircraft service life or is corrosion initiation scheduled?

Schedule first crack initiation.

Is time to second crack initiation less than aircraft service life or is corrosion initiation scheduled?

Schedule second crack initiation.

Is time to third crack initiation less than aircraft service life or is corrosion initiation scheduled?

Schedule third crack initiation.

Schedule first D-level inspection and aircraft retirement.

Is this last aircraft to enter service?

Schedule event ENTER.SERVICE.

Return.

7. FATIGUE.LIFE.SCATTER

7.1 Description

This routine receives the actual average fatigue life of the element design from the calling routine and returns the times to crack initiation of the first three fatigue cracks for the element in a particular aircraft. These times are random selections from a two-parameter Weibull distribution. This routine can be called from routine INSTALL.MODIFICATION and events ENTER.SERVICE and REPAIR.

7.2 Local Variables

ALPHA - This real variable is the shape parameter of the fatigue life distribution.

FIRST.LIFE - This real variable is the time to first crack initiation. This time is returned to the calling routine.

N - This integer variable, passed from the calling routine, identifies the random number stream to be used.

RN - This real variable is a uniformly distributed random number.

THIRD.LIFE - This real variable is the time to third crack initiation. This time is returned to the calling routine.

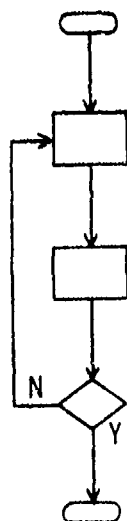
BETA - This real variable is the scale parameter of the fatigue life distribution.

LIFE - This real array of length three is used to temporarily hold the times to crack initiation of the three fatigue cracks.

RFL - This real variable is the element actual average fatigue life passed from the calling routine.

SECOND.LIFE - This real variable is the time to second crack initiation. This time is returned to the calling routine.

7.3 Flow Chart



Routine FATIGUE LIFE SCATTER.

Draw uniformly distributed random number.

Calculate time to crack initiation.

Times calculated for these racks?

Return.

8. INSTALL.MODIFICATION

8.1 Description

This routine represents the installation of a structural modification caused by a fatigue test failure or by aircraft service experience. The modification is installed during a repair or a D-level inspection. All previously scheduled defect initiations are cancelled, and new times to defect initiations are calculated for each aircraft when it is modified. This routine can be called from the events REPAIR and D.LEVEL.INSPECTION.

8.2 Local Variables

DEFECT.LIFE - This real variable is the time to crack initiation drawn from a distribution of fatigue lives of elements having production defects.

HOURS.TO.CORROSION - This real variable contains the value returned by routine PREDICT.CORROSION.

RST - This real variable is the remaining service time to retirement of the aircraft being considered.

STD.FAST - This real variable is the standard deviation of the distribution of fast crack growth rates.

THIRD.LIFE - This real variable is the time to third crack initiation returned by routine FATIGUE.LIFE.SCATTER.

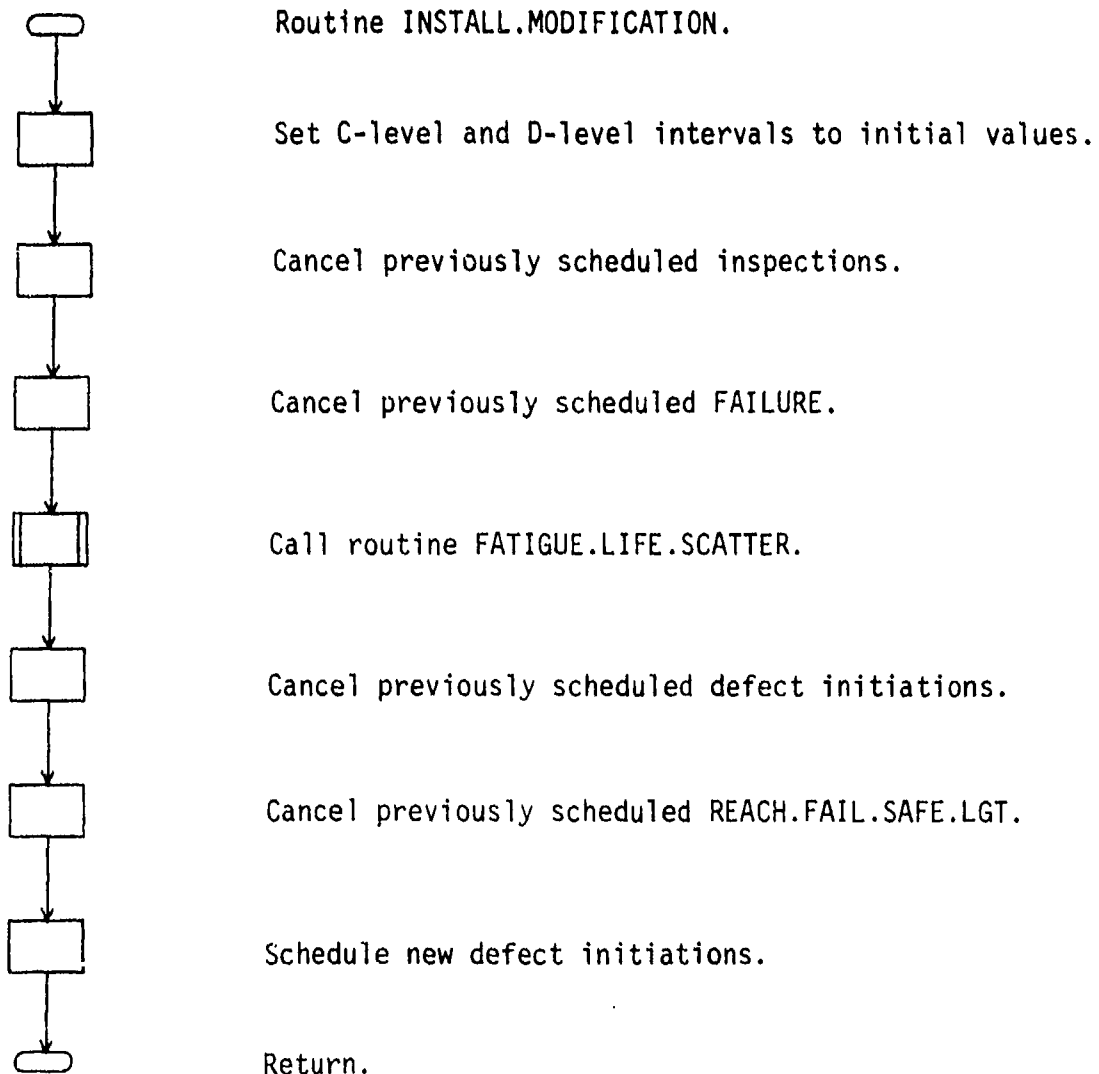
FIRST.LIFE - This real variable is the time to first crack initiation returned by routine FATIGUE.LIFE.SCATTER.

RN - This real variable is drawn from a uniform distribution of random numbers between 0 and 1.

SECOND.LIFE - This real variable is the time to second crack initiation returned by routine FATIGUE.LIFE.SCATTER.

STD.SLOW - This real variable is the standard deviation of the distribution of slow crack growth rates.

8.3 Flow Chart



9. IN.SERVICE.DAMAGE

9.1 Description

This event represents the occurrence of a service damage defect. This occurrence results in the immediate initiation of the next scheduled crack. A new time to service damage is determined. If the new time is less than the remaining time in service of the aircraft, this event is scheduled once again. This event can be scheduled from within itself or in event ENTER.SERVICE.

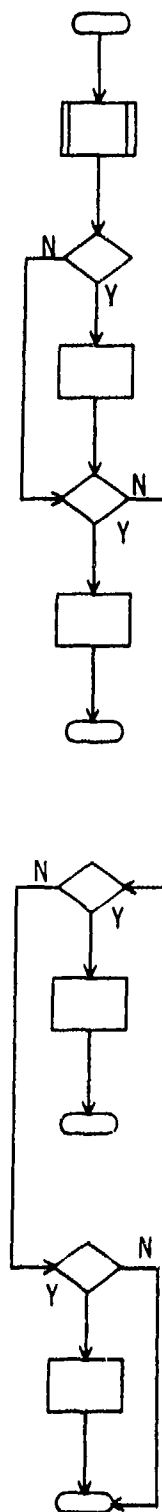
9.2 Local Variables

IDSDM - This integer variable is the identification number of the aircraft for which the event was scheduled.

RST - This real variable is the remaining service time to retirement of the aircraft being considered.

OURS.TO.SERVICE.DAMAGE - This real variable is the value returned by routine PREEDICT.SERVICE.DAMAGE.

9.3 Flow Chart



Event IN.SERVICE.DAMAGE.

Call routine PREEDICT.SERVICE.DAMAGE.

Is time to service damage less than remaining service time of aircraft?

Schedule event IN.SERVICE.DAMAGE.

Is first crack initiation scheduled?

Reschedule first crack to occur immediately.

Return.

Is second crack initiation scheduled?

Reschedule second crack to occur immediately.

Return.

Is third crack initiation scheduled?

Reschedule third crack to occur immediately.

Return.

10. T.IMPLEMENT.MOD

10.1 Description

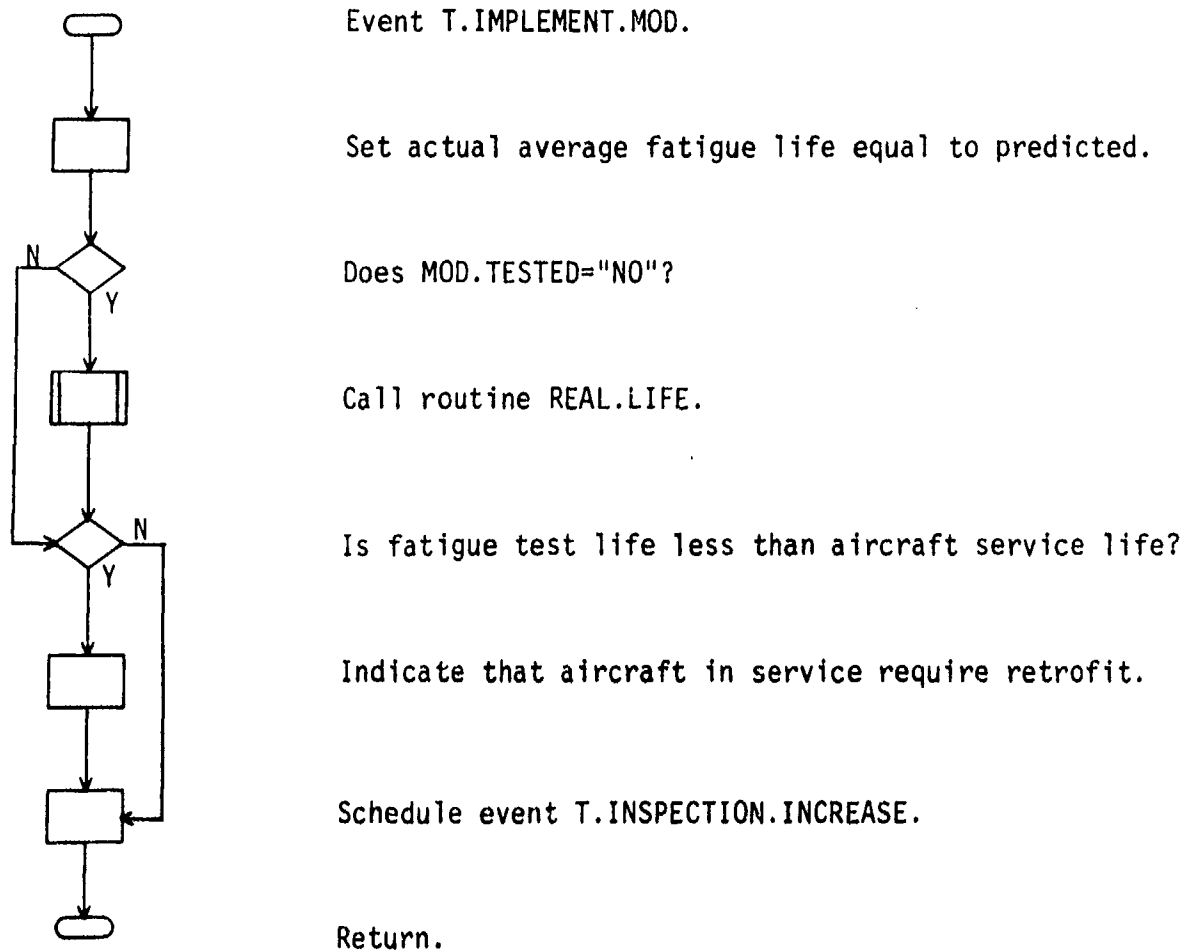
This event represents the development of a structural modification because of a fatigue test failure. This event determines the actual average fatigue life of the modification and schedules an increase in inspection frequencies at some percentage of the fatigue test life. This event is scheduled in the MAIN program.

10.2 Local Variables

NSIG - This real variable is $(\text{SIG.R})(.85)$ and is based on the assumption that a modification usually improves the actual average fatigue life of a particular design.

NMU - This real variable is $\text{MU.R} + 0.15(1.0 - \text{MU.R})$ and is also based on the foregoing assumption for NSIG.

10.3 Flow Chart



11. PREEDICT.SERVICE.DAMAGE

11.1 Description

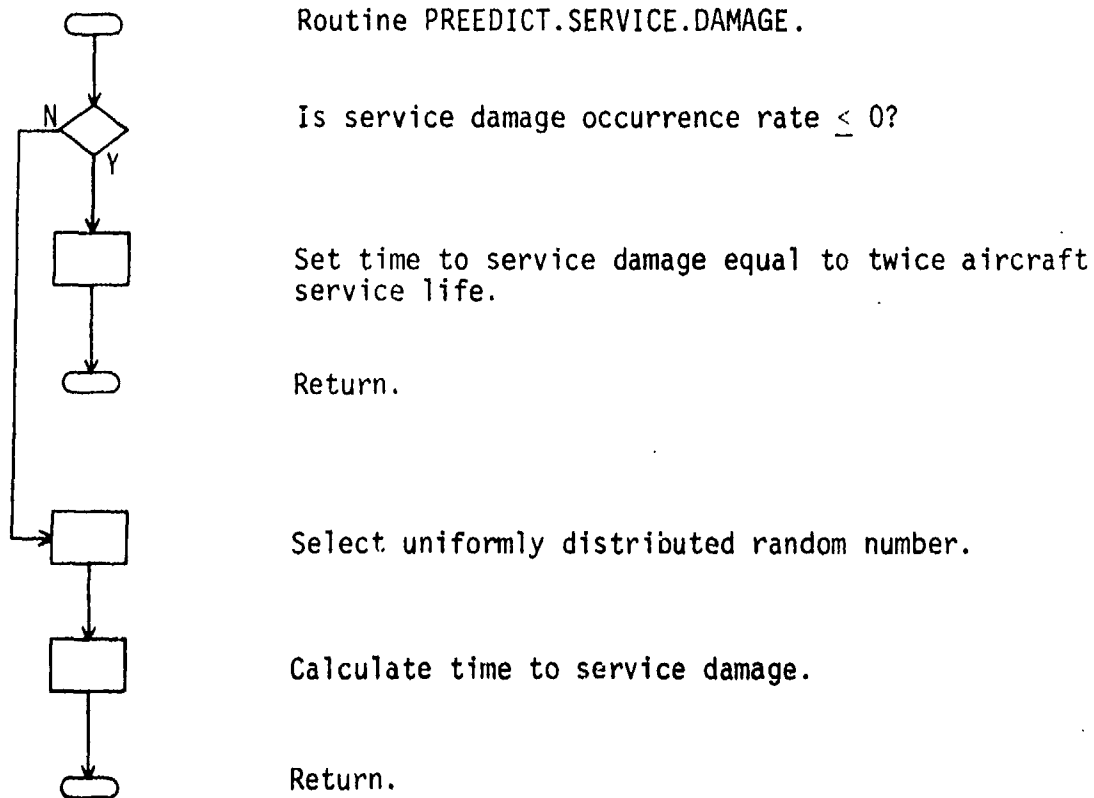
This routine generates the time to service damage occurrence for a given aircraft from a constant service damage occurrence rate. If the service damage occurrence rate is zero in the input, the routine sets the time to service damage occurrence as twice that of the aircraft service life. This routine can be called from events ENTER.SERVICE and IN.SERVICE.DAMAGE.

11.2 Local Variables

OURS.TO.SERVICE.DAMAGE - This real variable is the time to service damage occurrence in flight hours. This time is returned by the routine.

RN - This real variable is a uniformly distributed random number between 0 and 1.

11.3 Flow Chart



12. PREDICT.CORROSION

12.1 Description

This routine generates time to corrosion initiation for a given aircraft from a time-dependent occurrence rate approximated by two constant rates. The first constant occurrence rate, the second constant occurrence rate, and the service time on the aircraft when the second rate goes into effect are all input variables. This routine can be called from the routine INSTALL, MODIFICATION and events ENTER.SERVICE and REPAIR.

12.2 Local Variables

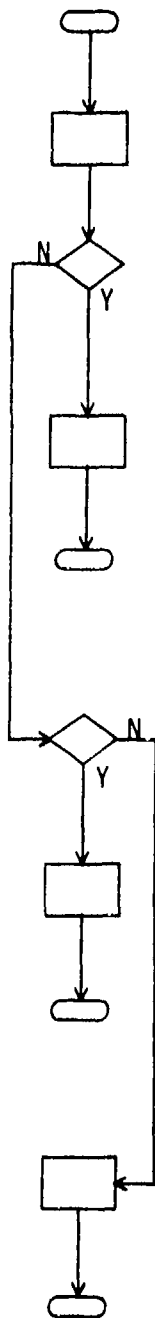
CRCT - This real variable is the remaining time in flight hours until the second corrosion occurrence rate goes into effect. This variable can be negative indicating that the second rate is already in effect.

LD - This real variable is used to hold an intermediate value during the calculation of time to corrosion initiation. The calculation uses a combination of both corrosion occurrence rates.

HOURS.TO.CORROSION - This real variable is the flight time until corrosion initiation. This time is returned to the calling routine.

RN - This real available is a uniformly distributed random number.

12.3 Flow Chart



Routine PREDICT.CORROSION.

Select uniformly distributed random number.

Is time on aircraft greater than time at which second occurrence rate goes into effect?

Calculate time to corrosion initiation using second rate.

Return.

Is selected random number > random number required to yield corrosion initiation at time of occurrence rate change?

Calculate time to corrosion initiation using first rate.

Return.

Calculate time to corrosion initiation using combination of both occurrence rates.

Return.

13. COROSION

13.1 Description

This event represents the initiation of a corrosion defect. The remaining time to crack initiation of all scheduled cracks is reduced by a corrosion damage factor. If either of the events FAILURE or REACH.FAIL.SAFE.LGT is scheduled, its remaining time until occurrence is also reduced by the corrosion damage factor. This event can be scheduled in the routine INSTALL. MODIFICATION and in events ENTER.SERVICE and REPAIR.

13.2 Local Variables

CDM.MULTIPLYING.FACTOR - This real variable is the factor which when multiplied by the remaining time to crack initiation accounts for the shortening effect of corrosion on fatigue lives.

NFTM - If a FAILURE has been scheduled, this real variable is the remaining time until its occurrence.

REMAINING.LIFE - This real variable is the remaining time until a scheduled crack initiation.

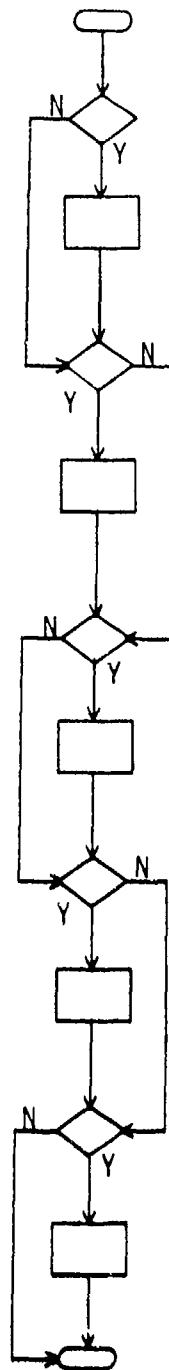
TRT - If a REACH.FAIL.SAFE.LGT has been scheduled, this real variable is the remaining time until its occurrence multiplied by the corrosion damage factor.

IDCO - This integer variable contains the identification number of the aircraft for which the event CORROSION was scheduled.

REDUCED.REMAINING.LIFE - This real variable is the REMAINING.LIFE multiplied by the corrosion damage factor.

RST - This real variable is the remaining service time of the aircraft under consideration.

13.3 Flow Chart



Event COROSION .

Is event REACH.FAIL.SAFE.LGT scheduled?

Reduce remaining time to occurrence of REACH.FAIL.SAFE.LGT by corrosion damage factor.

Is event FAILURE scheduled?

Reduce remaining time to occurrence of FAILURE by corrosion damage factor.

Is first crack initiation scheduled?

Reduce remaining time to first crack initiation by corrosion damage factor.

Is second crack initiation scheduled?

Reduce remaining time to second crack initiation by corrosion damage factor.

Is third crack initiation scheduled?

Reduce remaining time to third crack initiation by corrosion damage factor.

Return.

14. RATE

14.1 Description

This routine statistically generates element crack growth rates which reflect variation in material properties and load environment. The growth rates are randomly drawn from a normal distribution which is defined by a mean growth rate and a standard deviation passed from the calling routine. If a random draw yields a negative growth rate, the rate is set equal to the mean growth rate minus four standard deviations. Thus, the user must be sure that the standard deviation is always less than one-fourth of the mean. This routine is defined as a function in the routine PREAMBLE and is used in event ENTER.SERVICE and routine INSTALL.MODIFICATION.

14.2 Local Variables

G1 - This real variable is used to hold intermediate values in the calculation of the crack growth rate.

M - This real variable is the mean crack growth rate passed from the calling routine.

RN - This real variable is a uniformly distributed random number between 0 and 1.

S - This real variable is the crack growth rate standard deviation passed from the calling routine.

Z - This real variable is the element crack growth rate returned to the calling routine.

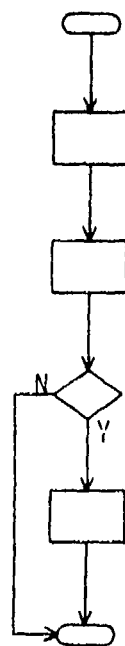
G2 - This real variable is used to hold intermediate values in the calculation of the crack growth rate.

N - This integer variable, passed from the calling routine, identifies the random number stream to be used.

RNI - This real variable, equal to $1.0 - RN$, is a uniformly distributed random number between 0 and 1.

W - This real variable is used to hold intermediate values in the calculation of the crack growth rate.

14.3 Flow Chart



Routine RATE.

Select uniformly distributed random number.

Calculate crack growth rate.

Is calculated growth rate negative?

Set growth rate equal to mean minus four standard deviations.

Return.

15. 1.STRENGTH.REDUCTION

15.1 Description

This event represents the initiation of the first crack. If the long list option is in effect, the aircraft identification number and flight hours are printed for those aircraft being tracked. A uniform random number is compared with the probability of internal cracking to determine whether this crack initiates internally. If it does initiate internally, the time until it becomes external is calculated and the event 1.ITE is scheduled. Next, the time to structural failure is calculated by using the three-part residual strength curve described in Vol. II. If this time is less than the remaining service life of the aircraft, the event FAILURE is scheduled. The time until the residual strength of the element reaches the fail-safe strength is calculated, and event REACH.FAIL.SAFE.LGT is scheduled. This event can be scheduled in events ENTER.SERVICE, INSTALL.MODIFICATION, and REPAIR.

15.2 Local Variables

ARG - This real variable is used as an intermediate value in the calculation of time until structural failure.

GR2 - This real variable is the fast crack growth rate.

ID1SR - This integer variable is the aircraft identification number.

K1, K11, K13, K4, K9, LG - These real variables are used as intermediate values in the calculation of time until structural failure.

LIST - This real variable is set equal to 1.0 when the long list option is in effect and the aircraft being processed is one of those being tracked.

R2 - This real variable is the strength degradation rate when the crack length is between the critical crack length and the fail-safe length.

SF - This real variable is the element fail-safe strength.

S1 - This real variable is the element residual strength when the crack is at the critical crack length.

TAR - This real variable is the simulation time at which the aircraft being processed retires from service.

T1 - This real variable is the time in flight hours for the crack to grow from its initiation to the critical crack length.

GR1 - This real variable is the element slow crack growth rate.

K10, K12, K2, K8, LGK5 - These real variables are used as intermediate values in the calculation of time until structural failure.

R1 - This real variable is the strength degradation rate from crack initiation to critical crack length.

R3 - This real variable is the strength degradation rate from fail-safe strength to structural failure.

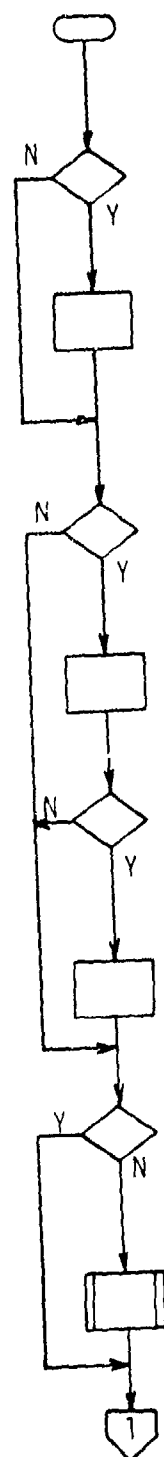
SU - This real variable is the ultimate strength of the element.

T - This real variable is the time in flight hours until a crack initiated internally becomes external.

TTF - This real variable is the time in flight hours until element failure.

T2 - This real variable is the time in flight hours until the fail-safe strength is reached.

15.3 Flow Chart



Event 1.STRENGTH.REDUCTION.

Is long list option in effect?

Print ID number and flight hours for aircraft being tracked.

Does crack initiate internally?

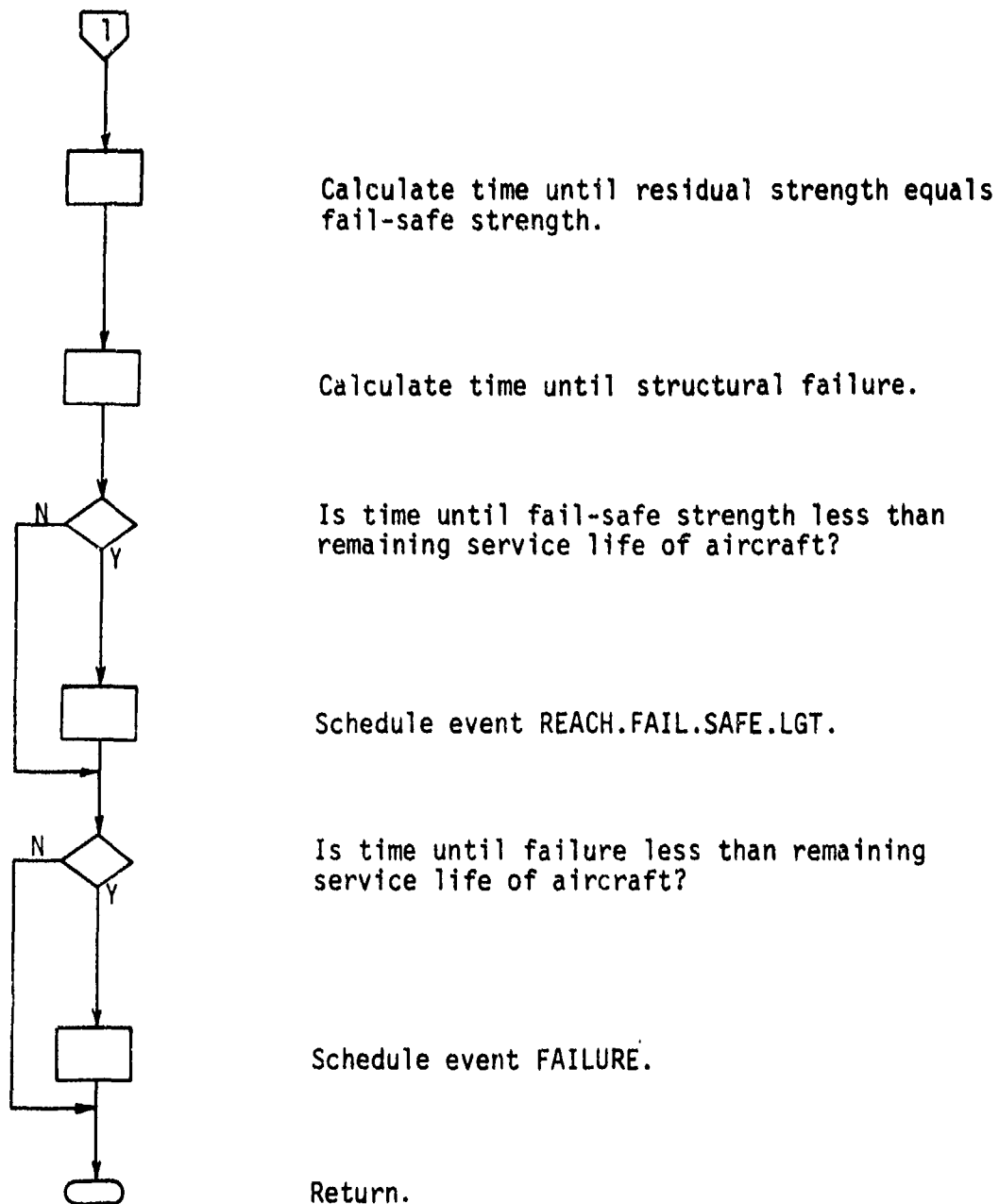
Set 1.INT to "YES".

Is time until crack becomes external less than remaining service life of aircraft?

Schedule event 1.ITE.

Is aircraft presently inspected below overhaul level?

Call routine INSPECTION.SCHEDULER.



16. 2.STRENGTH.REDUCTION

16.1 Description

This event represents the second crack initiation. If the long list option is in effect, the aircraft identification number and flight hours are printed for those aircraft being tracked. A uniform random number is compared with the probability of internal cracking to determine whether this crack initiates internally. If it does initiate internally, the time until it becomes external is calculated and the event 2.ITE is scheduled. Next, the time to structural failure is calculated by using the same three-part residual strength curve as in event 1.STRENGTH.REDUCTION. However, the crack growth rate is now one calculated by a least-squares fit of points determined from the sum of crack lengths of the two cracks. This calculation is described in Vol. II. If the time until failure is less than the remaining service life of the aircraft, the event FAILURE is scheduled. The time until the residual strength of the element reaches the fail-safe strength is calculated and event REACH.FAIL.SAFE.LGT is scheduled. This event can be scheduled in events ENTER.SERVICE, INSTALL.MODIFICATION, and REPAIR.

16.2 Local Variables

ARG - This real variable is used as an intermediate value in the calculation of time until element failure.

CL - This real variable is the crack length when the crack initiated.

GR1 - This real variable is the slow crack growth rate.

K10, K12, K2, K8, LGK5, N - These real variables are used as intermediate values in the calculation of time until element failure.

R2 - This real variable is the strength degradation rate when the composite crack length is between the critical crack length and the fail-safe length.

SF - This real variable is the fail-safe strength of the element.

SU - This real variable is the ultimate strength of the element.

T - This real variable is the time in flight hours until a crack initiated internally becomes external.

TAR - This real variable is the simulation time at which the aircraft being processed retires from service.

TCL - This real variable is the time it takes a single crack to grow from its initiation to the critical crack length.

T1 - This real variable is the time until the first crack reaches its critical crack length.

W, WXS, WY, Y - These real variables are used as intermediate values in the calculation of time until structural failure.

CCL - This real variable is the critical crack length of the element.

DL - This real variable is the length of the first crack at the time of corrosion initiation.

GR2 - This real variable is the fast crack growth rate of the element.

ID2SR - This integer variable is the aircraft identification number.

K1, K11, K13, K4, K9, LG - These real variables are used as intermediate values in the calculation of time until structural failure.

LIST - This real variable is set equal to 1.0 when the long list option is in effect and the aircraft being processed is one of those being tracked.

R1 - This real variable is the strength degradation rate between the time that a first crack initiates until the time that this crack reaches its critical length.

R3 - This real variable is the strength degradation rate from fail-safe strength until structural failure.

SMW - This real variable is used as an intermediate value in the calculation of time until structural failure.

S1 - This real variable is the element residual strength when the first crack is at the critical crack length.

TAC - This real variable is the time of corrosion initiation.

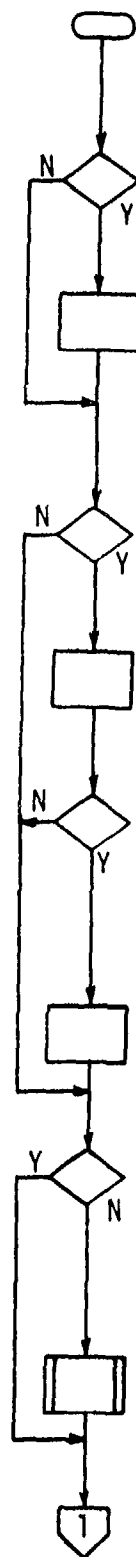
TA1 - This real variable is the time of first crack initiation.

TTF - This real variable is the time in flight hours until structural failure.

T2 - This real variable is the time in flight hours from when an element has a residual strength until when it has a fail-safe strength.

WX, WXY, X - These real variables are used as intermediate values in the calculation of time until structural failure.

16.3 Flow Chart



Event 2.STRENGTH.REDUCTION .

Is long list option in effect?

Print ID number and flight hours for aircraft being tracked.

Does crack initiate internally?

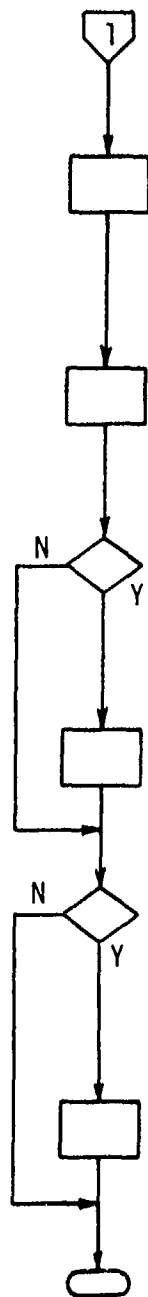
Set 2.INT to "YES".

Is time until crack becomes external less than remaining service life of aircraft?

Schedule event 2.ITE.

Is aircraft presently inspected below overhaul level?

Call routine INSPECTION.SCHEDULER.



Calculate time until residual strength equals fail-safe strength.

Calculate time until structural failure.

Is time until fail-safe strength less than remaining service life of aircraft?

Schedule event REACH.FAIL.SAFE.LGT.

Is time until failure less than remaining service life of aircraft?

Schedule event FAILURE.

Return.

17. 3.STRENGTH.REDUCTION

17.1 Description

This event represents the third crack initiation. If the long list option is in effect, the aircraft identification number and flight hours are printed for those aircraft being tracked. A uniform random number is compared with the probability of internal cracking to determine whether this crack initiates internally. If it does initiate internally, the time until it becomes external is calculated and the event 3.ITE is scheduled. Next, the time to structural failure is calculated by using the same three-part residual strength curve as in event 1.STRENGTH.REDUCTION. However, the crack growth rate is now calculated by a least-squares fit of points determined from the sum of crack lengths of the three cracks. This calculation is described in Vol. II. If the time until failure is less than the remaining service life of the aircraft, the event FAILURE is scheduled. The time until the residual strength of the element reaches the fail-safe strength is calculated and event REACH.FAIL.SAFE.LGT is scheduled. This event can be scheduled in events ENTER.SERVICE, INSTALL.MODIFICATION, and REPAIR.

17.2 Local Variables

ARG - This real variable is used as an intermediate value in the calculation of time until structural failure.

CL - This real variable is the crack length when the crack initiated.

GR1 - This real variable is the slow crack growth rate.

K10, K12, K2, K8, LGK5, N - These real variables are used as intermediate values in the calculation of time until structural failure.

R2 - This real variable is the strength degradation rate when the composite crack length is between the critical crack length and the fail-safe length.

SF - This real variable is the fail-safe strength of the element.

SU - This real variable is the ultimate strength of the element.

T - This real variable is the time in flight hours until a crack initiated internally becomes external.

TAR - This real variable is the simulation time at which the aircraft being processed retires from service.

TA2 - This real variable is the time of the second crack initiation.

TTF - This real variable is the time in flight hours until structural failure.

T2 - This real variable is the time in flight hours from when the element has a residual strength until it reaches a fail-safe strength.

WX, WXY, X, Y - These real variables are used as intermediate values in the calculation of time until structural failure.

1CL - This real variable is the crack length of the first crack at third crack initiation.

CCL - This real variable is the critical crack length of the element.

DL - This real variable is the length of the first crack at the time of corrosion initiation.

GR2 - This real variable is the fast crack growth rate of the element.

ID3SR - This integer variable is the aircraft identification number.

K1, K11, K13, K4, K9, LG - These real variables are used as intermediate values in the calculation of time until structural failure.

LIST - This real variable is set equal to 1.0 when the long list option is in effect and the aircraft being processed is one of those being tracked.

R1 - This real variable is the strength degradation rate from when the first crack initiates until it reaches its critical crack length.

R3 - This real variable is the strength degradation rate from fail-safe strength until structural failure.

SMW - This real variable is used as an intermediate value in the calculation of time until structural failure.

S1 - This real variable is the element residual strength when the first crack is at the critical crack length.

TAC - This real variable is the time of corrosion initiation.

TAl - This real variable is the time of first crack initiation.

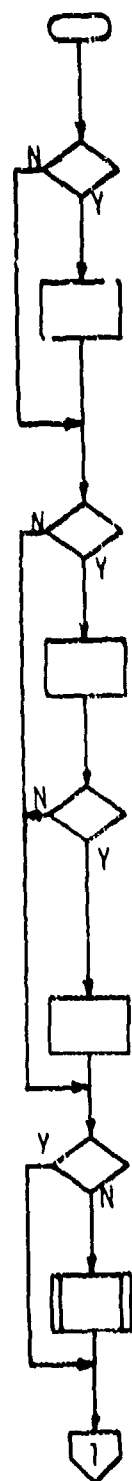
TCL - This real variable is the time it takes a single crack to grow from its initiation to the critical crack length.

Tl - This real variable is the time until the first crack reaches its critical crack length.

W, WXS, WY, XK, Y2 - These real variables are used as intermediate values in the calculation of time until structural failure.

2CL - This real variable is the length of the second crack at the time of the third crack initiation.

17.3 Flow Chart



Event 3. STRENGTH REDUCTION .

Is long list option in effect?

Print ID number and flight hours for aircraft being tracked.

Does crack initiate internally?

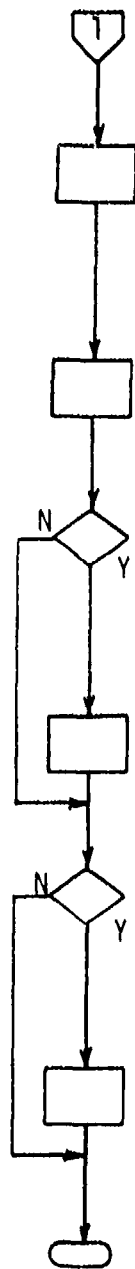
Set 3.INT to "YES".

Is time until crack becomes external less than remaining service life of aircraft?

Schedule event 3.ITE.

Is aircraft presently inspected below overhaul level?

Call routine INSPECTION.SCHEDULER.



Calculate time until residual strength equals fail-safe strength.

Calculate time until structural failure.

Is time until fail-safe strength less than remaining service life of aircraft?

Schedule event REACH.FAIL.SAFE.LGT.

Is time until failure less than remaining service life of aircraft?

Schedule event FAILURE.

Return.

18. 1.ITE

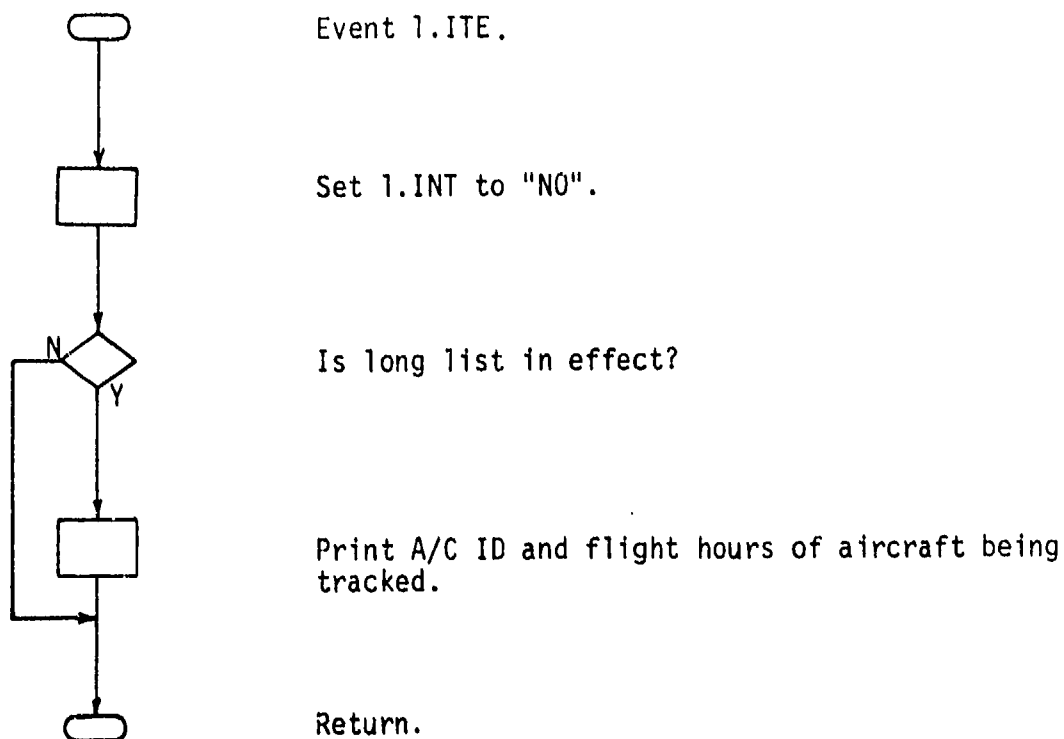
18.1 Description

This event represents the time when a first crack which initiated internally becomes external. This time is defined as the time when the element crack length reaches a percentage of the critical crack length. This percentage is an input parameter. At this time the appropriate element of the alpha array 1.INT is changed from "YES" to "NO". If the long list option is in effect, the aircraft identification and flight hours are printed for those aircraft being tracked. This event is scheduled in event 1.STRENGTH.REDUCTION.

18.2 Local Variables

ID1E - This integer variable is the aircraft identification number.

18.3 Flow Chart



19. 2.ITE

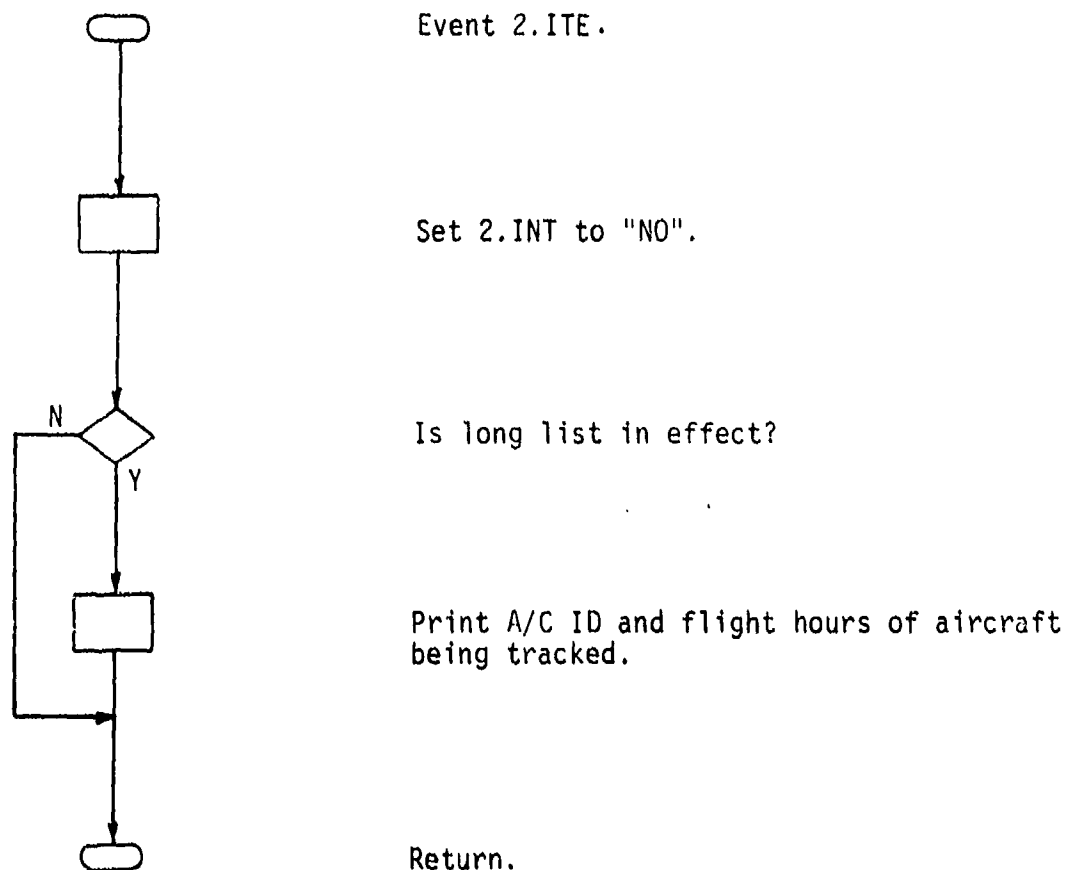
19.1 Description

This event represents the time when a second crack which initiated internally becomes external. This time is defined as the time when the element crack length reaches a percentage of the critical crack length. This percentage is an input parameter. At this time the appropriate element of the alpha array 2.INT is changed from "YES" to "NO". If the long list option is in effect, the aircraft identification and flight hours are printed for those aircraft being tracked. This event is scheduled in event 2.STRENGTH.REDUCTION.

19.2 Local Variables

ID2E - This integer variable is the aircraft identification number.

19.3 Flow Chart



20. 3.ITE

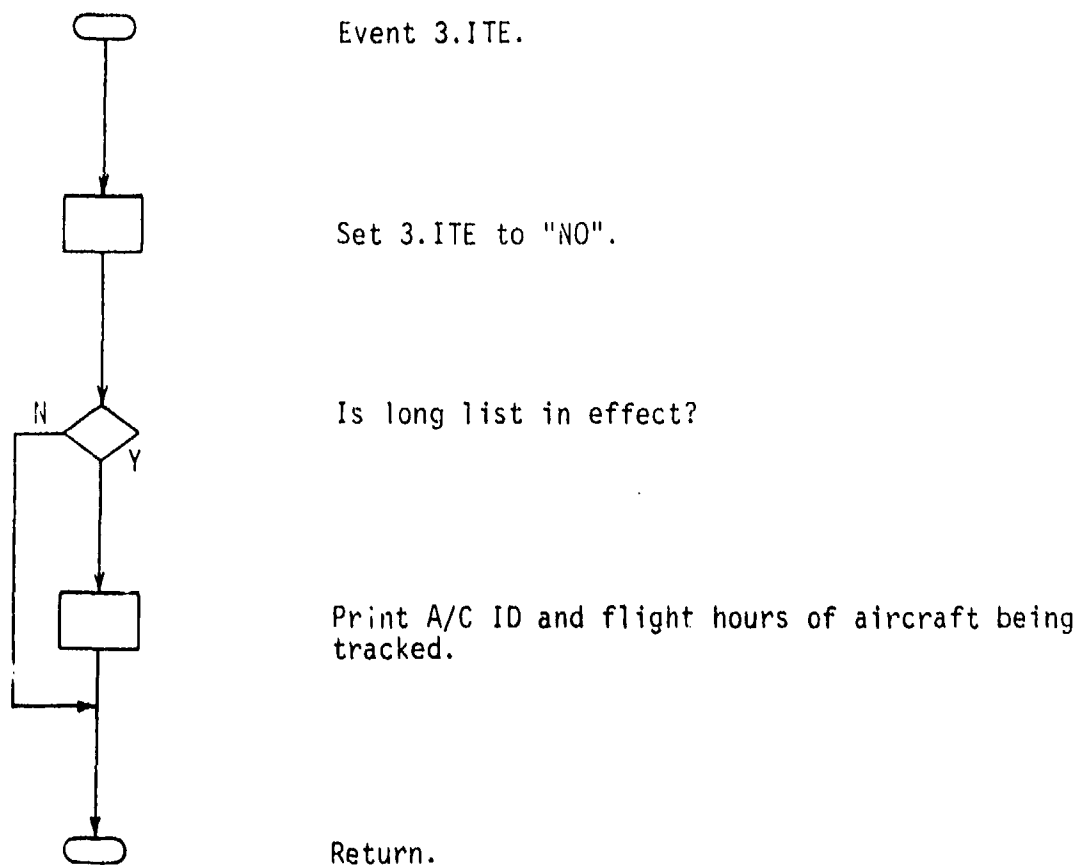
20.1 Description

This event represents the time when a first crack which initiated internally becomes external. This time is defined as the time when the element crack length reaches a percentage of the critical crack length. This percentage is an input parameter. At this time the appropriate element of the alpha array 3.INT is changed from "YES" to "NO". If the long list option is in effect, the aircraft identification and flight hours are printed for those aircraft being tracked. This event is scheduled in event 3.STRENGTH.REDUCTION.

20.2 Local Variables

ID3E - This integer variable is the aircraft identification number.

20.3 Flow Chart



21. INSPECTION.SCHEDULER

21.1 Description

This routine schedules all inspections below the overhaul level on a given aircraft. To conserve execution time, the inspections are scheduled so that the aircraft is not inspected before the defect reaches its minimum detectable size at each level of inspection. This routine is called from events 1. STRENGTH.REDUCTION and COROSION.

21.2 Local Variables

C1 - This real variable is the corrosion growth rate used to calculate the time to the minimum detectable corrosion area.

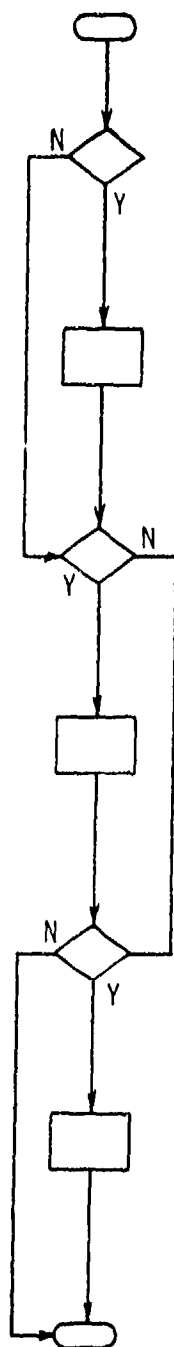
N - This integer variable indicates whether a crack initiation or a corrosion initiation caused this routine to be called.

TML - This real variable is the time to the minimum detectable defect size calculated for each level of inspection.

M1 - This real variable is the crack growth rate used to calculate the time to the minimum detectable crack length.

S.INSPE.AT - This real variable is the simulation time at which the first inspection at each level is scheduled.

21.3 Flow Chart



Routine INSPECTION.SCHEDULER.

Is lowest external inspection level less than or equal to A-level?

Calculate time to minimum detectable size.
Schedule A-level inspection.

Is lowest external inspection level less than or equal to B-level?

Calculate time to minimum detectable size.
Schedule B-level inspection.

Is lowest external inspection level less than or equal to C-level?

Calculate time to minimum detectable size.
Schedule C-level inspection.

Return.

22. A.LEVEL.INSPECTION

22.1 Description

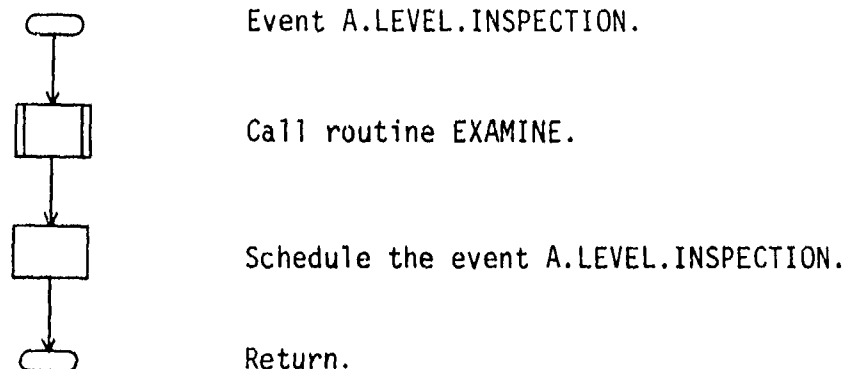
This event represents the performance of an A-level inspection. The constants which define the probability of detection equation at the A-level are passed to the routine EXAMINE which handles the actual inspection calculations for all levels of inspection. This event can be scheduled in the routine INSPECTION. SCHEDULER and the events A.LEVEL.INSPECTION, B.LEVEL.INSPECTION, C.LEVEL.INSPECTION, and D.LEVEL.INSPECTION.

22.2 Local Variables

FOUND - Not used in this event, this real variable is returned by routine EXAMINE with a non-zero value whenever a defect is found.

IDA - This integer variable is the identification number of the aircraft being inspected.

22.3 Flow Chart



23. B.LEVEL.INSPECTION

23.1 Description

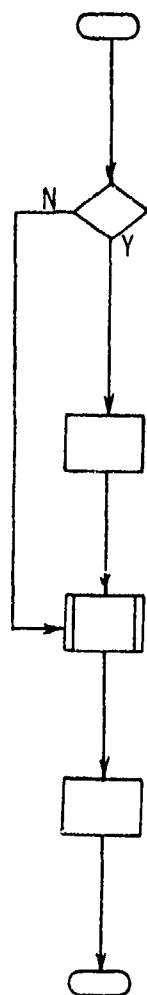
This event represents the performance of a B-level inspection. The constants which define the probability of detection equation at the B-level are passed to the routine EXAMINE. The inspection interval for each inspection level is not necessarily an even multiple of all lower level intervals. If an A-level inspection is scheduled, it is cancelled and rescheduled at present time plus one A-level interval later. This event can be scheduled in the routine INSPECTION.SCHEDULER and the events B.LEVEL.INSPECTION, C.LEVEL.INSPECTION, and D.LEVEL.INSPECTION.

23.2 Local Variables

FOUND - Not used in this event, this real variable is returned by routine EXAMINE with a non-zero value whenever a defect is found.

IDB - This integer variable is the identification number of the aircraft being inspected.

23.3 Flow Chart



Event B.LEVEL.INSPECTION.

Is lowest external inspection level less than
or equal to A-level?

Reschedule the A-level inspection.

Call routine EXAMINE.

Schedule the event B.LEVEL.INSPECTION.

Return.

24. C.LEVEL.INSPECTION

24.1 Description

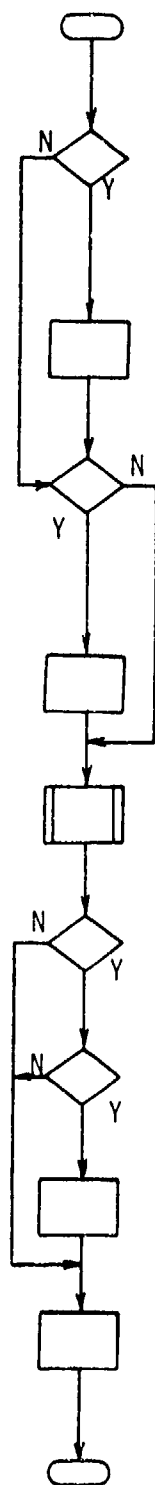
This event represents the performance of a C-level inspection. If there is either an A-level or a B-level inspection currently scheduled, it is cancelled and rescheduled at one A-level interval or B-level interval, respectively, later. The constants which define the probability of defect detection equation at the C-level are passed to the routine EXAMINE. If a crack is detected and the aircraft is one of the ten high-time aircraft, the time of detection is stored. This event can be scheduled in the routine INSPECTION.SCHEDULER and the events C.LEVEL.INSPECTION and D.LEVEL.INSPECTION.

24.2 Local Variables

FOUND - This real variable is returned by routine EXAMINE with a value of one whenever a crack is detected.

IDC - This integer variable is the identification number of the aircraft being inspected.

24.3 Flow Chart



Event C.LEVEL.INSPECTION.

Is lowest external inspection level less than or equal to B-level?

Reschedule B-level inspection.

Is lowest external inspection level less than or equal to A-level?

Reschedule A-level inspection.

Call routine EXAMINE.

Was crack detected?

Is aircraft one of ten high-time aircraft?

Retain time of detection.

Schedule event C.LEVEL.INSPECTION.

Return.

25. D.LEVEL.INSPECTION

25.1 Description

This event represents the performance of a D-level inspection. If the long list option is in effect on this element, the identification number and flight hours are printed for each aircraft being tracked. If there are any lower level inspections currently scheduled, they are cancelled and rescheduled at one inspection interval later. The constants which define the probability of defect detection equation at the D-level are passed to the routine EXAMINE. The defect histories of the ten high-time aircraft are now examined. If all of the ten high-time aircraft have gone one D-level interval without any cracks detected at either the C-level or the D-level, then the C-level and D-level inspection intervals are increased by the input factor `FREQ.` `DECREASE`.

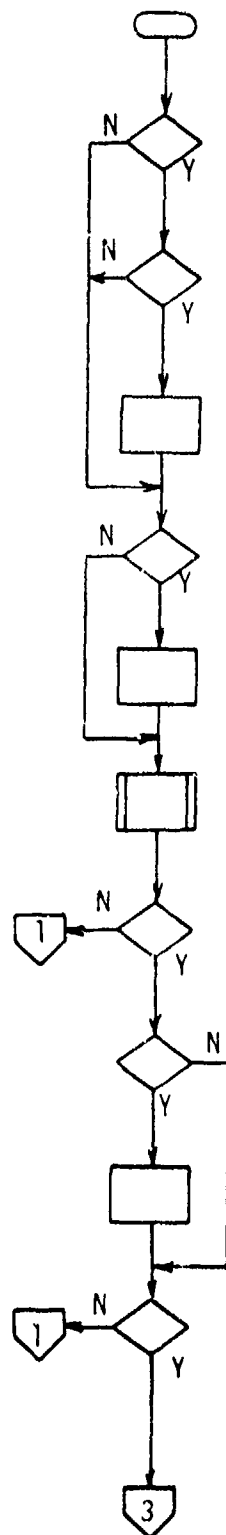
Any pending modifications are installed at this time. This event can be scheduled in the events `ENTER.SERVICE` and `D.LEVEL.INSPECTION`.

25.2 Local Variables

FOUND - This real variable is returned by routine EXAMINE with a value of one whenever a crack is detected.

IDD - This integer variable is the identification number of the aircraft being inspected.

25.3 Flow Chart



Event D.LEVEL.INSPECTION.

Is long list option in effect on this element?

Is this aircraft being tracked?

Print aircraft number and flight hours.

Are lower level inspections scheduled?

Reschedule lower level inspections.

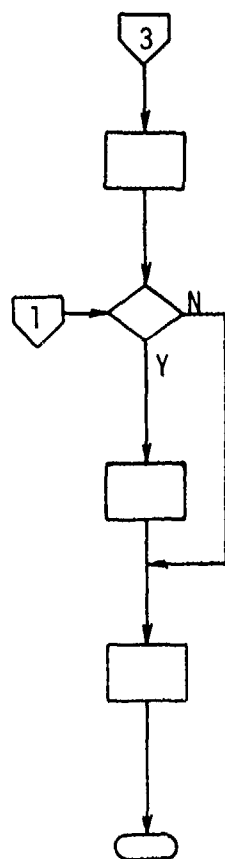
Call routine EXAMINE.

Is aircraft one of ten high-time aircraft?

Was crack detected?

Retain time of detection.

Have all ten high-time aircraft gone one D-level interval without any cracks detected at either C-level or D-level?



Increase C-level and D-level inspection intervals.

Is a modification pending on this aircraft?

Install modification.

Schedule event D.LEVEL.INSPECTION.

Return.

26. EXAMINE

26.1 Description

This routine performs the numerical comparison which determines whether a defect is detected at each level of inspection. The constants which define the probability of defect detection equation are passed to this routine from the events which represent the different levels of inspection. If the long list option is in effect, each time a defect is detected on one of the aircraft being tracked, the following are printed: the size of the defect, the inspection level, the aircraft identification number, and the number of flight hours on the aircraft. This routine can be called from events A.LEVEL.INSPECTION, B.LEVEL.INSPECTION, C.LEVEL.INSPECTION, and D.LEVEL.INSPECTION.

26.2 Local Variables

AREA - This real variable is the calculated area of the corrosion defect.

CL - This real variable is the calculated length of the fatigue cracks.

LIST - This real variable is set equal to 1.0 if the long list option is in effect and the aircraft being inspected is one of those being tracked.

M2 - This real variable is the fast crack growth rate for the aircraft being inspected.

TAC - This real variable is the simulation time of the corrosion initiation.

TA2 - This real variable is the simulation time of the second crack initiation.

XA - This real variable is one of the probability of detection equation constants passed by the calling event.

YA - This real variable is one of the probability of detection equation constants passed by the calling event.

Z - This alpha variable is the level of inspection being performed.

ZL - This real variable is one of the probability of detection equation constants passed by the calling event.

CCL - This real variable is the critical crack length of the element.

FOUND - This real variable is set equal to two whenever corrosion is detected and to one whenever a crack is detected.

M1 - This real variable is the slow crack growth rate for the aircraft being inspected.

N - This integer variable identifies the inspection level and is passed by the calling event.

TA1 - This real variable is the simulation time of the first crack initiation.

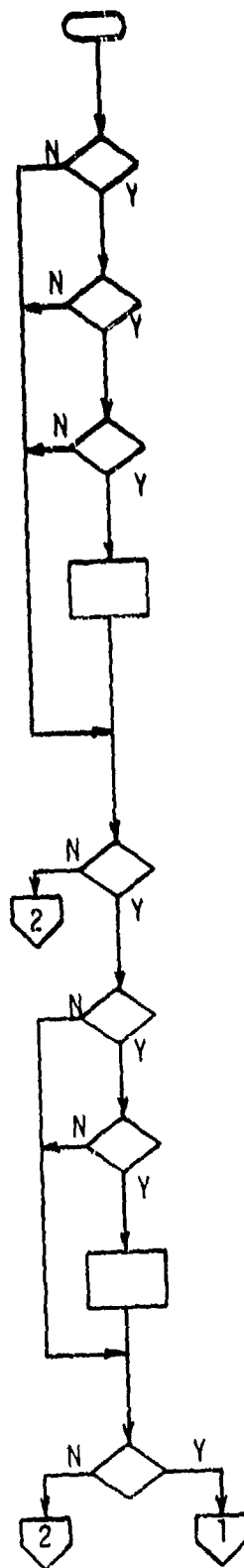
TA3 - This real variable is the simulation time of the third crack initiation.

XL - This real variable is one of the probability of detection equation constants passed by the calling event.

YL - This real variable is one of the probability of detection equation constants passed by the calling event.

ZA - This real variable is one of the probability of detection equation constants passed by the calling event.

20.3 Flow Chart



Routine EXAMINE.

Does corrosion exist?

Is the area inspected?

Is random number less than probability of detection?

Set tally counters to defect size.
Let FOUND=2.

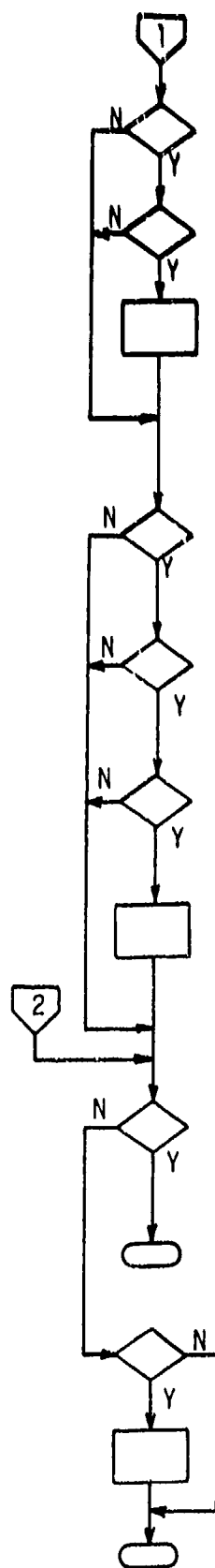
Does first crack exist?

Is the area inspected?

Is random number less than probability of detection?

Set tally counters to defect size.
Let FOUND=1.

Does second crack exist?



Is the area inspected?

Is random number less than probability of detection?

Set tally counters to defect size.
Let FOUND=1.

Does third crack exist?

Is the area inspected?

Is random number less than probability of detection?

Set tally counters to defect size.
Let FOUND=1.

Was routine called from D.LEVEL.INSPECTION?

Return.

Is FOUND greater than zero?

Schedule event REPAIR.

Return.

27. PODD

27.1 Description

This routine, defined as a function in the PREAMBLE, computes the probability of detecting a crack or corrosion defect of a given size. This probability is returned to the calling routine. PODD is called from routine EXAMINE and event IMMEDIATE. FLEET.INSPECTION.

27.2 Local Variables

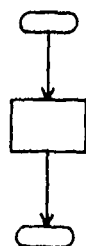
L - This real variable is the size of the defect under consideration.

Y - This real variable is an empirically determined equation constant for each level of inspection.

X - This real variable is the maximum probability of detection at a given inspection level.

Z - This real variable is the minimum defect size detectable at a given inspection level.

27.3 Flow Chart



Routine PODD.

Compute probability of detection.

Return.

28. CANCEL.SCHEDULED.INSPECTIONS

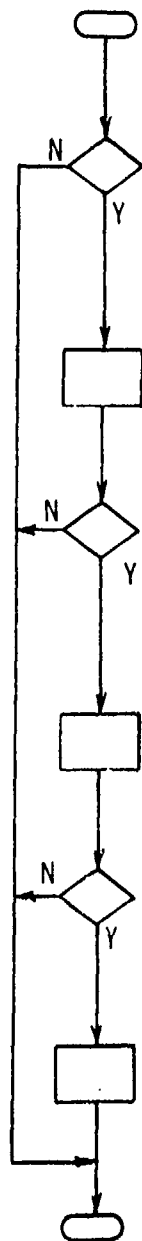
28.1 Description

This routine cancels all scheduled inspections below the overhaul level on a given aircraft. Whenever a defect is detected and repaired, it is assumed that all other defects existing on that particular element are also repaired. This routine is called at this time to cancel all subsequent inspections. Also, if an element fails or an aircraft with existing defects is retired, this routine is called to cancel all scheduled inspections. This routine can be called from events FAILURE, RETIRE, FROM.SERVICE, and REPAIR.

28.2 Local Variables

There are no local variables in this routine.

28.3 Flow Chart



Routine CANCEL.SCHEDULED.INSPECTIONS.

Is lowest level external inspection less than or equal to C-level?

Cancel the C-level inspection.

Is lowest level external inspection less than or equal to B-level?

Cancel the B-level inspection.

Is lowest level external inspection equal to A-level?

Cancel the A-level inspection.

Return.

29. REACH.FAIL.SAFE.LGT

29.1 Description

This event represents the time when the residual strength of the element has been reduced to the fail-safe strength. The time and aircraft identification number are saved as part of the output. The calculation of the strength reduction is based on the sum of all crack lengths in the element. This event can be scheduled in events 1.STRENGTH.REDUCTION, 2.STRENGTH.REDUCTION, and 3.STRENGTH.REDUCTION.

29.2 Local Variables

IDRFS - This integer variable is the identification number of the aircraft being processed.

29.3 Flow Chart



Event REACH.FAIL.SAFE.LGT.

Is long list option in effect?

Is this aircraft one of those being tracked?

Print aircraft number and flight hours.

Build output arrays.

Return.

30. FAILURE

30.1 Description

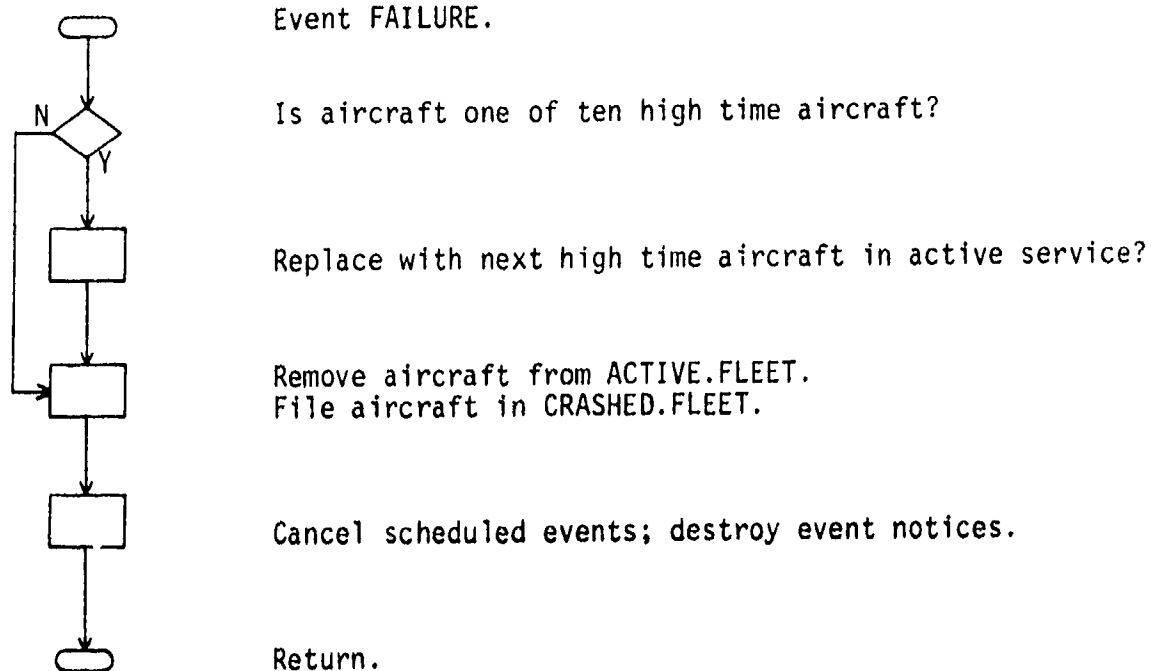
This event represents structural failure. When this event occurs, the aircraft is removed from the active fleet. If this aircraft was one of the ten high-time aircraft being monitored for the purpose of increasing inspection intervals, it is replaced by the next high-time aircraft in active service. Any remaining scheduled events are cancelled and their event notices destroyed. This event can be scheduled in events 1.STRENGTH.REDUCTION, 2.STRENGTH.REDUCTION, and 3.STRENGTH.REDUCTION.

30.2 Local Variables

HOLD - This integer variable serves as an intermediate storage for aircraft identification numbers when replacing one of the ten high-time aircraft.

IDFA - This integer variable is the identification number of the aircraft under consideration.

30.3 Flow Chart



31. RETIRE.FROM.SERVICE

31.1 Description

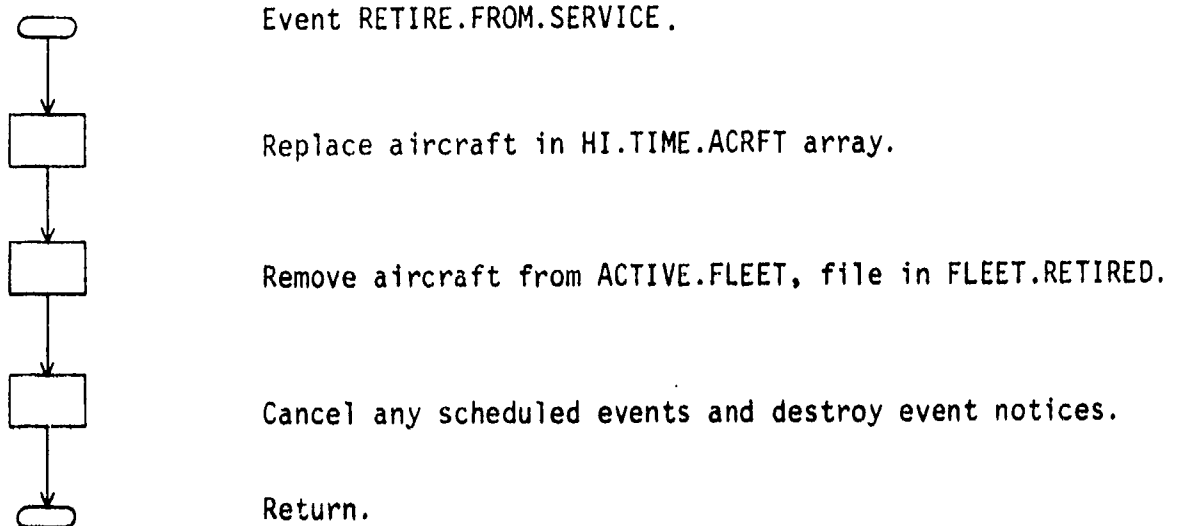
This event represents the retirement of an aircraft from active service. The aircraft is replaced in the HI.TIME.ACRFT array by that active aircraft not in the array which has the highest time in service. The aircraft being retired is removed from the set ACTIVE.FLEET and filed in the set FLEET.RETIRED. All remaining scheduled events for this aircraft are cancelled and the event notices destroyed. This event can only be scheduled in the event ENTER.SERVICE.

31.2 Local Variables

HOLD - This integer variable is used as an intermediate storage to hold aircraft identification numbers during the HI.TIME.ACRFT replacement.

IDRET - This integer variable is the identification number of the aircraft being retired from service.

31.3 Flow Chart



32. REPAIR

32.1 Description

This event represents the structural repair of an element. If there is a modification pending on the element, it is installed at this time. If the events FAILURE and REACH.FAILSAFE.LGT are scheduled, they are cancelled. It is assumed that all existing defects are repaired and that new times to defect occurrences are determined in the same manner as when the aircraft entered service. The size and number of all existing cracks are compared with the inspection interval decrease criteria. If these criteria are met, then the events which decrease inspection intervals and perform special fleet-wide inspections are scheduled. All defects that were scheduled but had not occurred by this time are not affected by this event and will occur as originally scheduled. This event can be scheduled in events D.LEVEL.INSPECTION, EXAMINE, and IMMEDIATE.FLEET.INSPECTION.

32.2 Local Variables

AAFL - This real variable is the actual average fatigue life of the element design.

CL - This real variable is the calculated fatigue crack length.

HOURS.TC.CORROSION - This real variable is the time in flight hours to corrosion initiation.

MAX.CRK - This real variable is the maximum crack length in the element.

RST - This real variable is the remaining service time of the aircraft.

STR.RED - This real variable is the element strength reduction because of all existing cracks.

TA1 - This real variable is the simulation time of the first crack initiation.

TA3 - This real variable is the simulation time of the third crack initiation.

CCL - This real variable is the element critical crack length.

FIRST.LIFE - This real variable is the time in flight hours to first crack initiation.

IDREP - This integer variable is the aircraft identification number.

POT.CRK - This real variable is the sum of the maximum crack length and the length that the crack will grow during the shortest internal inspection interval.

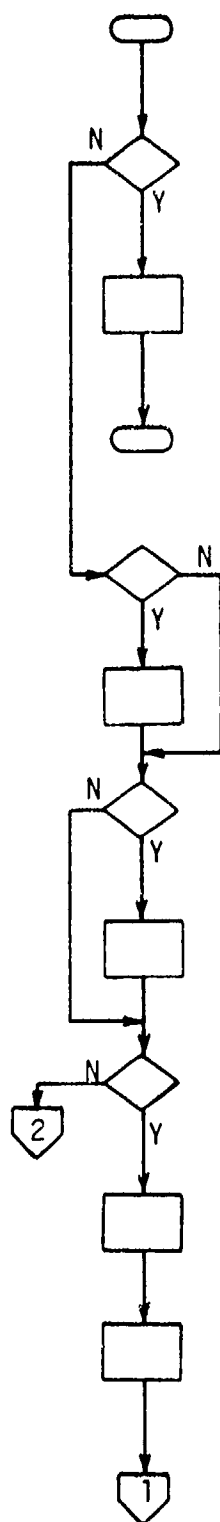
SECOND.LIFE - This real variable is the time in flight hours to the second crack initiation.

TAC - This real variable is the simulation time of the corrosion initiation.

TA2 - This real variable is the simulation time of the second crack initiation.

THIRD.LIFE - This real variable is the time in flight hours to the third crack initiation.

32.3 Flow Chart



Event REPAIR.

Is modification pending on this aircraft?

Install modification.

Return.

Is event FAILURE scheduled?

Cancel event FAILURE.

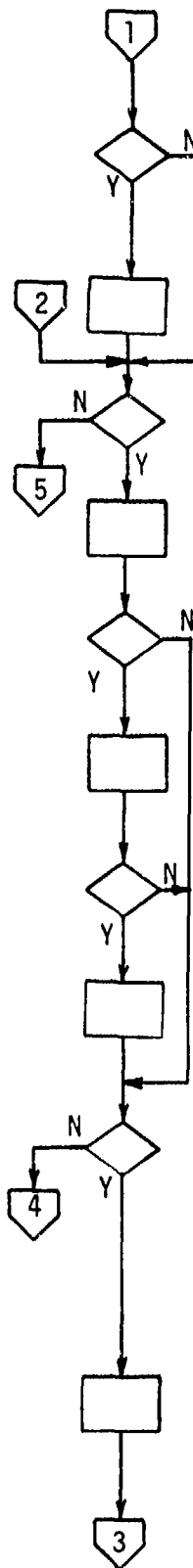
Is event REACH.FAIL.SAFE.LGT scheduled?

Cancel event REACH.FAIL.SAFE.LGT.

Does corrosion exist?

Set corrosion factor equal to 1.0.

Determine new time to corrosion initiation.



Is time to corrosion initiation less than remaining service life of aircraft?

Schedule corrosion initiation.

Does first crack exist?

Calculate crack length.

Does second crack exist?

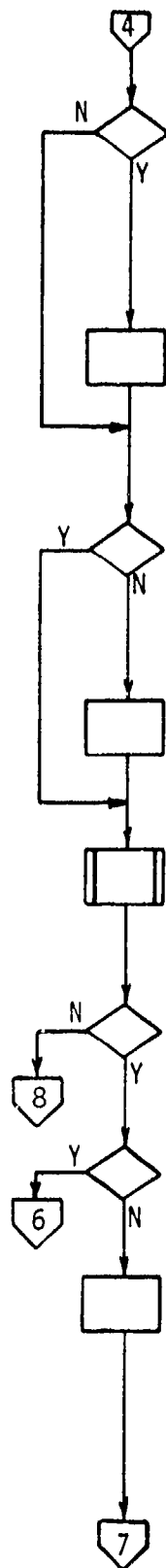
Calculate crack length.

Does third crack exist?

Calculate crack length.

Is sum of crack lengths plus crack growth rate times inspection interval greater than one-half the fail-safe crack length?

Increase inspection frequency and perform a special fleet wide inspection.



Is sum of crack lengths found in entire fleet greater than one-fifth of fleet size times fail-safe crack length?

Increase inspection frequency and perform a special fleet wide inspection.

Is there a modification pending somewhere in the fleet?

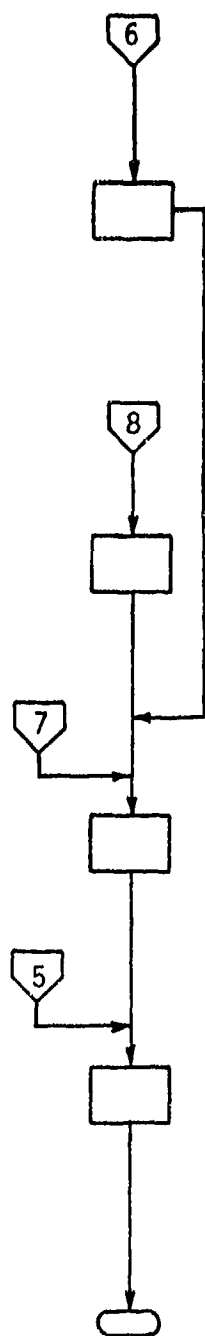
Schedule event DECISION.ON.MOD.

Call routine FATIGUE.LIFE.SCATTER.

Does second crack exist?

Does third crack exist?

Let time to first crack initiation be set to time to third crack initiation and times to second and third crack initiations be set to shortest times from routine FATIGUE.LIFE.SCATTER.



Let times to all three crack initiations be taken from FATIGUE.LIFE.SCATTER.

Set time to first crack to time to second crack. Set time to second crack to time to third crack. Set time to third crack from FATIGUE.LIFE.SCATTER.

Reschedule crack initiations for those cracks with times to crack initiation less than remaining service life of aircraft.

Cancel scheduled inspections below overhaul level.

Return.

33. T. INSPECTION: INCREASE

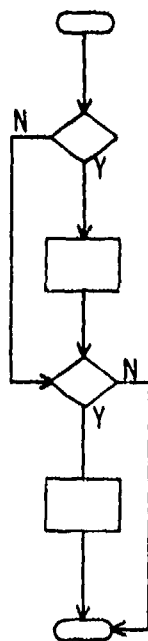
33.1 Description

This event represents an inspection frequency increase for a particular aircraft pending a structural modification because of a fatigue test failure. The factor by which the inspection frequency for both the close internal and the close external inspection is increased depends on the element being considered and is an input parameter. This event can only be scheduled in event T.IMPLEMENT.MOD.

33.2 Local Variables

IDTI - This integer variable is the identification number of the aircraft under consideration.

33.3 Flow Chart



Event T.INSPECTION.INCREASE.

Is lowest internal level or lowest external level of inspection equal to C-level?

Increase C-level inspection frequency.

Is lowest internal level or lowest external level of inspection equal to D-level?

Increase D-level inspection frequency.

Return.

34. INCREASE.INSPECTION.FREQUENCY

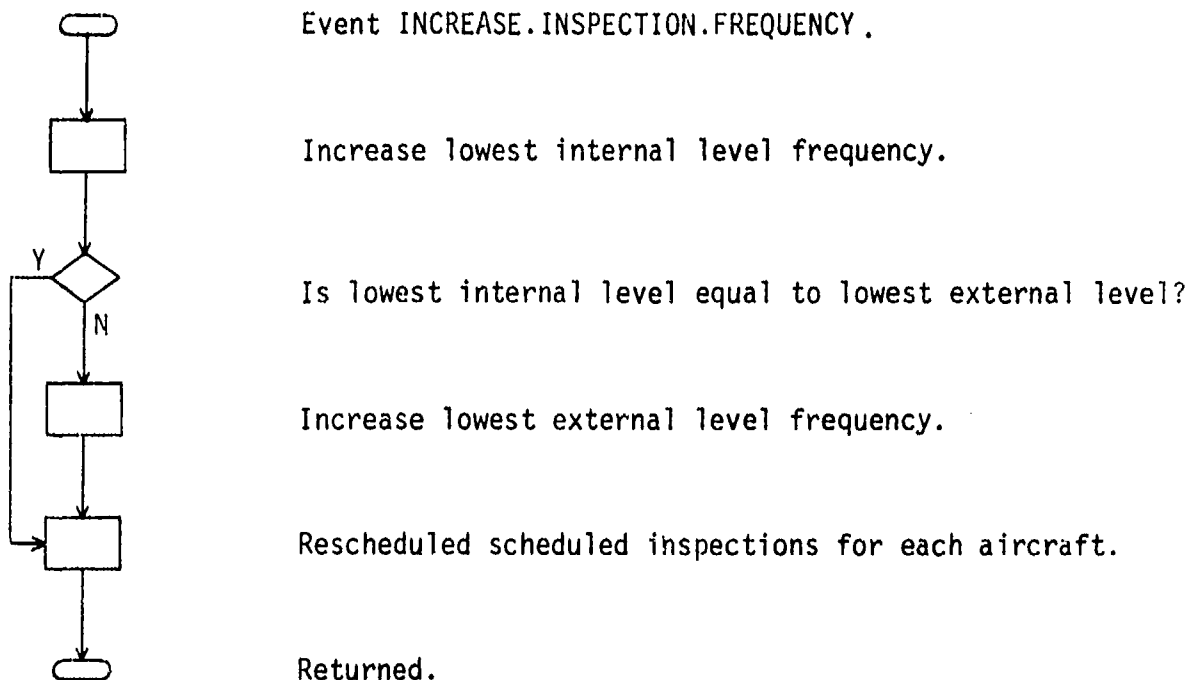
34.1 Description

This event represents a fleet-wide increase in the frequencies of the lowest level internal and external inspections. These may or may not be the same levels. However, the A-level and B-level frequencies are never changed. All scheduled inspections are cancelled and rescheduled to reflect the frequency increase. This event can be scheduled in events REPAIR and IMMEDIATE.FLEET.INSPECTION.

34.2 Local Variables

DIFF - This real variable is the difference in flight hours between the old and new lowest internal level inspection intervals.

34.3 Flow Chart



35. IMMEDIATE.FLEET.INSPECTION

35.1 Description

This event represents an immediate fleet-wide inspection caused by finding a defect considered too hazardous to depend on scheduled inspections for detection of additional defects. Existing crack lengths and corrosion areas are calculated along with the associated probabilities of detection. As in the scheduled inspections, these probabilities are compared with a random number to determine whether or not the defect is detected. This event is always preceded by the event INCREASE.INSPECTION.FREQUENCY. Defects found during this inspection can cause an additional increase in the frequency of normally scheduled inspections. This event can only be scheduled in event REPAIR.

35.2 Local Variables

AREA - This real variable is the area in sq. inches of an existing corrosion defect.

CL - This real variable is the length in inches of an existing fatigue crack.

M1 - This real variable is the slow crack growth rate for a particular aircraft.

POT.CRK - This real variable is the sum of the maximum crack length detected on a particular aircraft plus the product of the average crack growth rate and the current lowest internal level inspection interval.

SR - This real variable is the sum of the lengths of all cracks detected on a particular aircraft.

TAC - This real variable is the time of initiation of an existing corrosion defect.

TA2 - This real variable is the time of initiation of an existing second fatigue crack.

CCL - This real variable is the critical crack length of the element under consideration.

FOUND - This real variable serves as a switch which is set equal to two when corrosion is detected and set equal to one when a crack is detected.

MAX.CRK - This real variable is the maximum crack length detected on a particular aircraft.

M2 - This real variable is the fast crack growth rate for a particular aircraft.

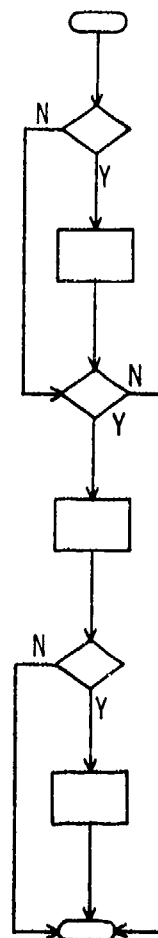
POT.STR.RED - This real variable is the sum of the detected crack lengths on a particular aircraft plus the product of the crack growth rate and the current lowest internal level inspection interval.

STR.RED - This real variable is equal to SR.

TA1 - This real variable is the time of initiation of an existing first fatigue crack.

TA3 - This real variable is the time of initiation of an existing third fatigue crack.

35.3 Flow Chart



Event IMMEDIATE.FLEET.INSPECTION.

Does corrosion exist?

Calculate corrosion area and test for detection.

Do cracks exist?

Calculate crack lengths and test for detection.

Do detected cracks meet criteria for inspection frequency increase?

Schedule event INCREASE.INSPECTION.FREQUENCY to occur immediately.

Return.

36. DECISION.ON.MOD

36.1 Description

This event makes the decision on whether or not to develop a structural modification because of service experience. The decision to develop a modification is made by comparing the cost per flight hour of the modification with the repair cost per flight hour plus the increased inspection cost per flight hour. The modification cost per flight hour is found by dividing the total fleet modification cost by the remaining service life of the fleet. The repair cost per flight hour is found by dividing the total fleet repair costs since the last modification by the fleet flight time since the last modification. The increased inspection cost per flight hour is found by dividing the projected increased inspection costs by the remaining service life of the fleet. A modification is justified when

$$MCPH < RCPH + ICPH$$

where MCPH = modification cost per flight hour
 RCPH = repair cost per flight hour
 ICPH = increased inspection cost per flight hour

This event can be scheduled only in event REPAIR.

36.2 Local Variables

ACCUMULATED.HRS - This real variable is the total fleet time since the last modification.

MD.COST - This real variable is the cost of installing a modification on a single aircraft. The costs for additional modifications can differ from those for the initial modifications.

NFTS - This real variable is the total production time of all aircraft entering service after the second production rate goes into affect.

POST.MOD.HRS - This real variable is the total fleet service time remaining after the modification.

TOOLING - This real variable is the tooling cost in the development of a modification. The costs for additional modifications can differ from those for the initial modifications.

ICPH - This real variable is the increased inspection cost per flight hour.

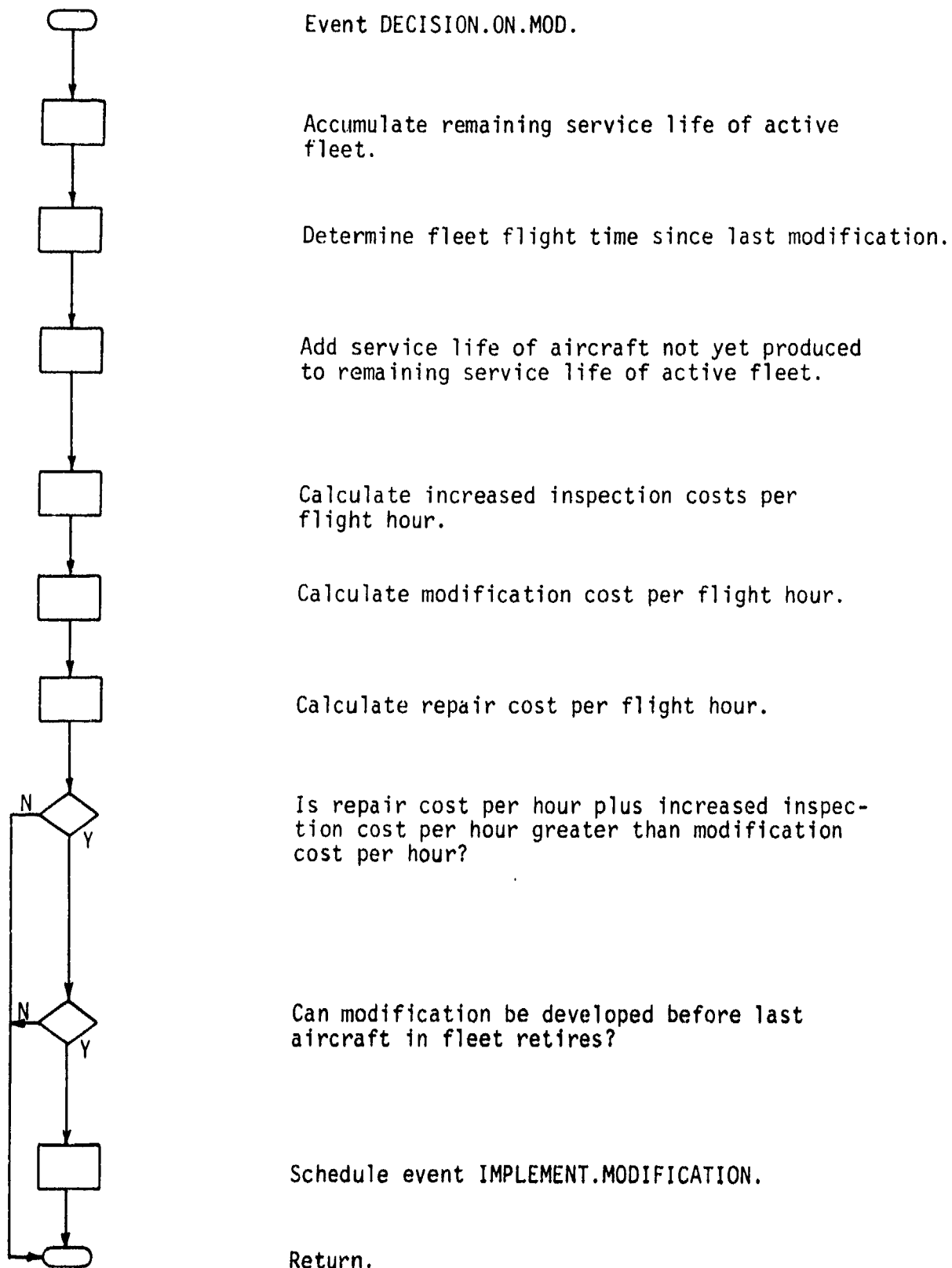
MCPH - This real variable is the modification cost per flight hour.

MRFH - This real variable is the service time remaining on a particular aircraft after its modification.

NPDL - This real variable is the number of aircraft which have entered service.

RCPH - This real variable is the repair cost per flight hour of the fleet.

36.3 Flow Chart



37. IMPLEMENT.MODIFICATION

37.1 Description

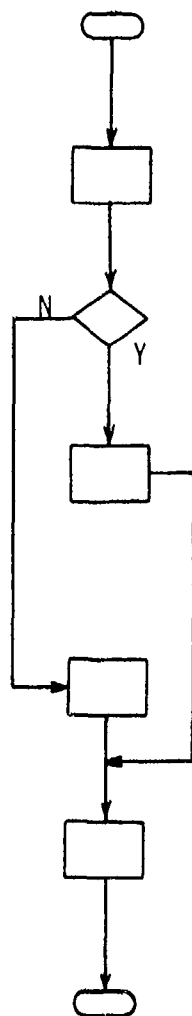
This event represents the development of a modification because of service experience. If the modification is to be fatigue tested, the actual average fatigue life is set equal to the original predicted life of the element design. Otherwise, the actual average fatigue life is determined by calling event REAL.LIFE. Elements of the alpha array SMOD.PENDING are set equal to "YES" to indicate which aircraft have service modifications pending. These modifications will be installed at the next D-level inspection or defect repair. Intervals for all levels of inspection are set to their initial values. This event can be scheduled only in event DECISION.ON.MOD.

37.2 Local Variables

NSIG - This real variable is the standard deviation of the ratio distribution passed to routine REAL.LIFE.

NMU - This real variable is the mean of the ratio distribution passed to routine REAL.LIFE.

37.3 Flow Chart



Event IMPLEMENT.MODIFICATION.

Set inspection intervals to initial values.

Is modification tested?

Set actual average fatigue life equal to predicted average fatigue life.

Call routine REAL.LIFE.

Set alpha array SMOD.PENDING.

Return.

38. DISPLAY.OUTPUT

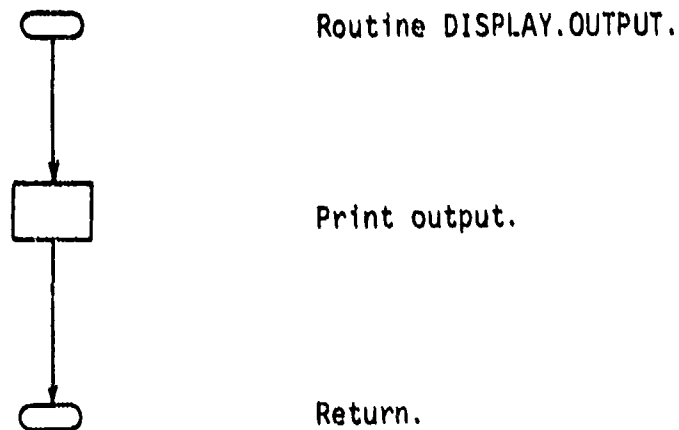
38.1 Description

This routine prints the standard output for each element. It is called from the MAIN program immediately after the completion of each element simulation. This output is suppressed if the long list option is in effect.

38.2 Local Variables

All the local variables are used to temporarily hold output values.

38.3 Flow Chart



39. SUMMARY

39.1 Description

This routine prints the standard output summary for each element type. The output of routine DISPLAY.OUTPUT for all the elements in an element type is contained in this output summary. This routine is called from the MAIN program each time a new element type is read in and at the end of the program run. This output is suppressed if the long list option is in effect.

39.2 Local Variables

All the local variables are used to temporarily hold output values.

39.3 Flow Chart



Routine SUMMARY.

Print output.

Return.

APPENDIX B
INITIAL PROGRAM SOURCE LISTING

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDO

[illegible]

LINE CAGE SIMSCRIPT II.5 RELEASE RF 09/01/76 PAGE 2

```

54 **
55 ** INTEGER ARRAYS
56 **
57 DEFINE AISCN, A2SN, A3SN, AF, AIRPLANE, AAL, AHL, ACL, ADL, AC, AYII, ACIO,
58 AICN, AICN, SACIO, GICCN, GICCN, MICTF, ACRFT, APIU, SAPIU, STIM,
59 SSIIA, FLTH, SFLTH, ARESL, QAC, AIP, AZE, ARE
60 AS INTEGER, 1-DIMENSIONAL ARRAYS
61 DEFINE TLID AS AN INTEGER, 2-DIMENSIONAL ARRAY
62 **
63 ** ALPHA ARRAYS
64 **
65 DEFINE WOFEL, FLPMET, 1-CH-EXISTS, 2-CH-EXISTS, 3-CH-EXISTS, CO-EXISTS,
66 SP-SCN, SSYAN, SFLW, AIL, FSM, IEL, IEP, IES,
67 TON-DEFINING, SNO-PENDING, INSP-SCN, 1-INT, 2-INT, 3-INT, C-INT
68 AS ALPHA, 1-DIMENSIONAL ARRAYS
69 **
70 DEFINE ELIN AS AN ALPHA, 2-DIMENSIONAL ARRAY
71 **
72 TEMPORARY ENTITIES
73 **
74 EVERY AIRCRAFT HAS AN ENTRY-TIME, A TAIL-ID
75 AND MAY BELONG TO AN ACTIVE-FLIGHT, A FLIGHT-RETIRED AND A CRASHED-FLIGHT
76 DEFINE TAIL-ID AS AN INTEGER VARIABLE
77 **
78 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
79 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
80 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
81 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
82 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
83 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
84 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
85 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
86 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
87 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
88 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
89 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
90 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
91 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
92 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
93 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
94 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
95 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
96 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
97 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
98 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
99 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
100 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
101 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
102 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
103 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
104 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
105 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO
106 TALLY MICRO AS THE MINIMUM, MICRO AS THE MAXIMUM, MICRO AS THE MEAN OF MICRO

```

EVENT NOTICES INCLUDE ENTER-SERVICE, IMPLEMENTATION, IMPLEMENT-MODIFICATION,

09/01/76 PAGE 3

LINE CACI SIMSCRIPT II.5 RELEASE AF

```

107  "
108  INCREASE.INSPECTION.FREQUENCY, IMMEDIATE.FLEET.INSPECTION,
109  DECISION.ON.WND
110  EVERY IN-SERVICE.DAMAGE HAS AN IDSDM
111  DEFINE IDSDM AS AN INTEGER VARIABLE
112  EVERY CONSTRUCTION HAS AN IDCN
113  DEFINE IDCN AS AN INTEGER VARIABLE
114  EVERY 1-STRENGTH.REDUCTION HAS AN ID1SR
115  DEFINE ID1SR AS AN INTEGER VARIABLE
116  EVERY 2-STRENGTH.REDUCTION HAS AN ID2SR
117  DEFINE ID2SR AS AN INTEGER VARIABLE
118  EVERY 3-STRENGTH.REDUCTION HAS AN ID3SR
119  DEFINE ID3SR AS AN INTEGER VARIABLE
120  EVERY 1-ITE HAS AN ID1E
121  DEFINE ID1E AS AN INTEGER VARIABLE
122  EVERY 2-ITE HAS AN ID2E
123  DEFINE ID2E AS AN INTEGER VARIABLE
124  EVERY 3-ITE HAS AN ID3E
125  DEFINE ID3E AS AN INTEGER VARIABLE
126  EVERY A-LEVEL.INSPECTION HAS AN IDA
127  DEFINE IDA AS AN INTEGER VARIABLE
128  EVERY M-LEVEL.INSPECTION HAS AN IDM
129  DEFINE IDM AS AN INTEGER VARIABLE
130  EVERY C-LEVEL.INSPECTION HAS AN IDC
131  DEFINE IDC AS AN INTEGER VARIABLE
132  EVERY D-LEVEL.INSPECTION HAS AN IDD
133  DEFINE IDD AS AN INTEGER VARIABLE
134  EVERY I-INSPECTION.INCREASE HAS AN IIDI
135  DEFINE IIDI AS AN INTEGER VARIABLE
136  EVERY REPAIR HAS AN IDREP
137  DEFINE IDREP AS AN INTEGER VARIABLE
138  EVERY FAILURE HAS AN IDFA
139  DEFINE IDFA AS AN INTEGER VARIABLE
140  EVERY WEACH.FAIL.SAFF.LGT HAS AN IDWFL
141  DEFINE IDWFL AS AN INTEGER VARIABLE
142  EVERY RETIRE.FROM.SERVICE HAS AN IDRET
143  DEFINE IDRET AS AN INTEGER VARIABLE
144  "
145  PRIORITY ORDER IS REPAIR, INCREASE.INSPECTION.FREQUENCY,
146  IMMEDIATE.FLEET.INSPECTION, D.LEVEL.INSPECTION, C.LEVEL.INSPECTION,
147  M.LEVEL.INSPECTION
148  "
149  OFFICE PODD AS A REAL FUNCTION
150  DEFINE RATE AS A REAL FUNCTION
151  "
152  END

```

09/01/76 PAGE 4

```

LINE  CACT SIMSCRIPT II.5  RELEASE AF
1  MAIN
2  **
3  ** SET RANDOM NUMBER SEEDS
4  **
5  LET SEED.V(5) = SEED.V(4)
6  LET SEED.V(6) = SEED.V(5)
7  **
8  ** RESERVE AIRCRAFT ARRAYS
9  **
10 RESERVE HI.TIME.ACFT(*), WIND(*) AS 10
11 RESERVE SC(*), SD(*) AS 30
12 RESERVE ACID(*), FLTHR(*) AS 10
13 RESERVE APID(*), STIM(*) AS 20
14 RESERVE SACID(*), SFLTHR(*) AS 100
15 RESERVE SAPID(*), SSTIM(*) AS 200
16 RESERVE WDEL(*) AS 2 AND FLEMET(*) AS 4
17 RESERVE ANCH(*) AS 100, CARCD(*) AS 4
18 **
19 ** READ AIRCRAFT INPUT
20 **
21 READ MODEL(1), MODEL(2) AS 2 & 4
22 READ SIZE.UP.FLEET, USAGE.LIFE
23 READ BEGIN.PRODUCTION.TIME, 2.PRODUCTION.TIME, PNCMS
24 READ START.TEST, TEST.ACCEL.FACT, C.GRINTH.DATE
25 READ C7, C24, C20, MU.W, SIG.W, A, F
26 READ IANCH(1), IANCH(2), IANCH(3), IANCH(4)
27 READ CANCH(1), CARCD(2), CARCD(3), CARCD(4)
28 READ SAMPLING, LINC.LIST, PCCI
29 IF LONG.LIST = "YES"
30 READ NCF
31 RESERVE FLID(*,*) AS 4 BY NCF AND ANAC(*) AS NCF
32 RESERVE FLID(*,*) AS NCF BY *
33 FOR J = 1 TO NCF
34 DO
35 START NEW CARD
36 READ FLID(1,1), FLID(2,1), FLID(3,1), FLID(4,1) AS 4 & 4
37 READ ANAC(1)
38 RESERVE FLID(1,*) AS ANAC(1)
39 ALSO FOR J = 1 TO ANAC(1)
40 DO
41 READ FLID(1,1)
42 LOOP
43 ALWAYS
44 ** RESERVE FLEMET ARRAYS
45 **
46 RESERVE NICH(*), NICH(*), NICH(*), NICH(*), NICH(*) AS 4
47 RESERVE APPLA.EC(*), TEND.PAVING(*), SHIP.PAVING(*) AS SIZE.UP.FLEET
48 RESERVE C.INTERVAL(*), AIL(*), APSEL(*), ESP(*) AS SIZE.UP.FLEET
49 RESERVE D.INTERVAL(*), OCCUR.OCC(*), CREFP.TIME(*) AS SIZE.UP.FLEET
50 RESERVE COMF.TIME(*), LCP.EXISTS(*), PCW.EXISTS(*) AS SIZE.UP.FLEET
51 RESERVE 3.CP.EXISTS(*), CP.EXISTS(*), INSP.SCH(*) AS SIZE.UP.FLEET
52 RESERVE AIS(*), ASP(*), ASK(*) AS 4, A(*) AS 4, A(*) AS 4, A(*) AS 4
53 RESERVE AIS(*), ASP(*), ASK(*) AS 4, A(*) AS 4, A(*) AS 4, A(*) AS 4

```

ALL INFORMATION CONTAINED HEREIN IS UNCLASSIFIED

```

54 RESERVE AAL(a), ARL(a), ACL(a), ADL(a) AS SIZE_OF_FLEET
55 RESERVE IINT(a), ZINT(a), ZINT(a), CINT(a) AS SIZE_OF_FLEET
56 RESERVE W(a), W(a), LAST, SD(a) AS SIZE_OF_FLEET
57 RESERVE W(a), C(a), C(a) AS SIZE_OF_FLEET
58 RESERVE IF(a), IF(a), IF(a), IF(a) AS SIZE_OF_FLEET
59 RESERVE AIF, AIF, AIF, AIF AS SIZE_OF_FLEET
60
61 ** READ ELEMENT INPUT
62
63 ** READ' ADD 1 TO COUNT, ELEMENT
64
65 READ ELEMENT(1), ELEMENT(2), ELEMENT(3), ELEMENT(4) AS A A
66 IF ELEMENT(1) = "END"
67 GO TO END
68 OTHERWISE
69
70 READ ELEMENT(1), FAILURE, ACTUAL_AVG_FAT_LIFE, W1, W2, W3, W4, W5, W6, W7, W8, W9, W10, W11, W12, W13, W14, W15, W16, W17, W18, W19, W20, W21, W22, W23, W24, W25, W26, W27, W28, W29, W30, W31, W32, W33, W34, W35, W36, W37, W38, W39, W40, W41, W42, W43, W44, W45, W46, W47, W48, W49, W50, W51, W52, W53, W54, W55, W56, W57, W58, W59, W60, W61, W62, W63, W64, W65, W66, W67, W68, W69, W70, W71, W72, W73, W74, W75, W76, W77, W78, W79, W80, W81, W82, W83, W84, W85, W86, W87, W88, W89, W90, W91, W92, W93, W94, W95, W96, W97, W98, W99, W100, W101, W102, W103, W104, W105, W106, W107, W108, W109, W110, W111, W112, W113, W114, W115, W116, W117, W118, W119, W120, W121, W122, W123, W124, W125, W126, W127, W128, W129, W130, W131, W132, W133, W134, W135, W136, W137, W138, W139, W140, W141, W142, W143, W144, W145, W146, W147, W148, W149, W150, W151, W152, W153, W154, W155, W156, W157, W158, W159, W160, W161, W162, W163, W164, W165, W166, W167, W168, W169, W170, W171, W172, W173, W174, W175, W176, W177, W178, W179, W180, W181, W182, W183, W184, W185, W186, W187, W188, W189, W190, W191, W192, W193, W194, W195, W196, W197, W198, W199, W200, W201, W202, W203, W204, W205, W206, W207, W208, W209, W210, W211, W212, W213, W214, W215, W216, W217, W218, W219, W220, W221, W222, W223, W224, W225, W226, W227, W228, W229, W230, W231, W232, W233, W234, W235, W236, W237, W238, W239, W240, W241, W242, W243, W244, W245, W246, W247, W248, W249, W250, W251, W252, W253, W254, W255, W256, W257, W258, W259, W260, W261, W262, W263, W264, W265, W266, W267, W268, W269, W270, W271, W272, W273, W274, W275, W276, W277, W278, W279, W280, W281, W282, W283, W284, W285, W286, W287, W288, W289, W290, W291, W292, W293, W294, W295, W296, W297, W298, W299, W300, W301, W302, W303, W304, W305, W306, W307, W308, W309, W310, W311, W312, W313, W314, W315, W316, W317, W318, W319, W320, W321, W322, W323, W324, W325, W326, W327, W328, W329, W330, W331, W332, W333, W334, W335, W336, W337, W338, W339, W340, W341, W342, W343, W344, W345, W346, W347, W348, W349, W350, W351, W352, W353, W354, W355, W356, W357, W358, W359, W360, W361, W362, W363, W364, W365, W366, W367, W368, W369, W370, W371, W372, W373, W374, W375, W376, W377, W378, W379, W380, W381, W382, W383, W384, W385, W386, W387, W388, W389, W390, W391, W392, W393, W394, W395, W396, W397, W398, W399, W400, W401, W402, W403, W404, W405, W406, W407, W408, W409, W410, W411, W412, W413, W414, W415, W416, W417, W418, W419, W420, W421, W422, W423, W424, W425, W426, W427, W428, W429, W430, W431, W432, W433, W434, W435, W436, W437, W438, W439, W440, W441, W442, W443, W444, W445, W446, W447, W448, W449, W450, W451, W452, W453, W454, W455, W456, W457, W458, W459, W460, W461, W462, W463, W464, W465, W466, W467, W468, W469, W470, W471, W472, W473, W474, W475, W476, W477, W478, W479, W480, W481, W482, W483, W484, W485, W486, W487, W488, W489, W490, W491, W492, W493, W494, W495, W496, W497, W498, W499, W500, W501, W502, W503, W504, W505, W506, W507, W508, W509, W510, W511, W512, W513, W514, W515, W516, W517, W518, W519, W520, W521, W522, W523, W524, W525, W526, W527, W528, W529, W530, W531, W532, W533, W534, W535, W536, W537, W538, W539, W540, W541, W542, W543, W544, W545, W546, W547, W548, W549, W550, W551, W552, W553, W554, W555, W556, W557, W558, W559, W560, W561, W562, W563, W564, W565, W566, W567, W568, W569, W570, W571, W572, W573, W574, W575, W576, W577, W578, W579, W580, W581, W582, W583, W584, W585, W586, W587, W588, W589, W590, W591, W592, W593, W594, W595, W596, W597, W598, W599, W600, W601, W602, W603, W604, W605, W606, W607, W608, W609, W610, W611, W612, W613, W614, W615, W616, W617, W618, W619, W620, W621, W622, W623, W624, W625, W626, W627, W628, W629, W630, W631, W632, W633, W634, W635, W636, W637, W638, W639, W640, W641, W642, W643, W644, W645, W646, W647, W648, W649, W650, W651, W652, W653, W654, W655, W656, W657, W658, W659, W660, W661, W662, W663, W664, W665, W666, W667, W668, W669, W670, W671, W672, W673, W674, W675, W676, W677, W678, W679, W680, W681, W682, W683, W684, W685, W686, W687, W688, W689, W690, W691, W692, W693, W694, W695, W696, W697, W698, W699, W700, W701, W702, W703, W704, W705, W706, W707, W708, W709, W710, W711, W712, W713, W714, W715, W716, W717, W718, W719, W720, W721, W722, W723, W724, W725, W726, W727, W728, W729, W730, W731, W732, W733, W734, W735, W736, W737, W738, W739, W740, W741, W742, W743, W744, W745, W746, W747, W748, W749, W750, W751, W752, W753, W754, W755, W756, W757, W758, W759, W760, W761, W762, W763, W764, W765, W766, W767, W768, W769, W770, W771, W772, W773, W774, W775, W776, W777, W778, W779, W780, W781, W782, W783, W784, W785, W786, W787, W788, W789, W790, W791, W792, W793, W
```

20/01/76 PAGE 0

LINE CACT SIMSCRIPT II.5 REFLESE AF

```

107 REGARDLESS
108 LET IAPFL = ACTUAL.AVG.FAT.LIFE
109 LET APTS = (SIZE_OF_FLEET + 1 - TRUNC_F(PWMC/PRODUCTION.TIME)) *
110 2.PRODUCTION.TIME
111 LET SAIL = START.TIME + IAPFL/IFST.ACC.FACT * LEAD.TIME
112 IF SAIL LT NEXT.PRODUCTION + PWMC * HTS * USAGE.LIFE
113 IF PREDICTED.LIFE LT 2.0 * USAGE.LIFE AND IAPFL LT PREDICTED.LIFE UM
114 PREDICTED.LIFE CE 2.0 * USAGE.LIFE AND IAPFL LT 2.0 * USAGE.LIFE
115 SCHEDULE A 1 TO ELEMENT WHEN AT SAIL
116 LET TFS.FAILURE = "YES"
117 REGARDLESS
118 REGARDLESS
119 IF TIME = "YES"
120 START NEW PAGE
121 WRITE MODEL(1), "MODEL(1) AS S 50, "AIRCRAFT TYPE: ", P & Q
122 SKIP 2 OUTPUT LINE WITH LINE AS FOLLOWS
123 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
124 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
125 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
126 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
127 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
128 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
129 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
130 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
131 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
132 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
133 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
134 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
135 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
136 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
137 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
138 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
139 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
140 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
141 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
142 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
143 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
144 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
145 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
146 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
147 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
148 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
149 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
150 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS
151 PRINT 1 DOUBLE LINE WITH LINE AS FOLLOWS

```

09/01/76 PAGE 7

```

LINE CACI SIMSCRIPT II.5 RELEASE AF
152 REMOVE AIRPLANE(TAIL.ID) FROM FLEET-RETIRE
153 DESTROY THE AIRCRAFT CALLED AIRPLANE(TAIL.ID)
154 LOOP
155 **
156 GO READ
157 'EOU' IF LONGLIST = 'NO'
158 CALL SUMMARY
159 ALWAYS
160 START NEW PAGE
161 PRINT LINE AS FOLLOWS
END OF SIMULATION
162 STOP
163 END

```

LOCAL VARIABLES IN THIS ROUTINE

I.1	INTEGER	WORD 3	I.2	INTEGER	WORD 8
I.3	INTEGER	WORD 5	J.1	INTEGER	WORD 6
K.1	INTEGER	WORD 4	K.2	INTEGER	WORD 9
L.1	INTEGER	WORD 10	M.1	INTEGER	WORD 11
L.11	INTEGER	WORD 14	L.15	INTEGER	WORD 19
L.15	INTEGER	WORD 20	L.17	INTEGER	WORD 21
L.22	INTEGER	WORD 22	L.24	INTEGER	WORD 24
L.9	INTEGER	WORD 17	N.1	INTEGER	WORD 7
N.1	INTEGER	WORD 13	N.1	INTEGER	WORD 1
N.15	INTEGER	WORD 15			

09/01/76 PAGE 8

LINE CACI SIMSCRIPT II.5 RELEASE NF

```

1 ROUTINE INITIALIZATION
2
3 ** CHANGE INSPECTION LEVEL CODE TO NUMERIC VALUE
4
5 IF INT.LVL.INSPE = "A" LET ITRNL = 1 JUMP AHEAD
6 OTHERWISE
7 IF INT.LVL.INSPE = "H" LET ITRNL = 2 JUMP AHEAD
8 OTHERWISE
9 IF INT.LVL.INSPE = "C" LET ITRNL = 3 JUMP AHEAD
10 OTHERWISE
11 LET ITRNL = 4
12
13 **
14 IF EXT.LVL.INSPE = "A" LET EXT.INSPE.LEVEL = 1 JUMP AHEAD
15 OTHERWISE
16 IF EXT.LVL.INSPE = "H" LET EXT.INSPE.LEVEL = 2 JUMP AHEAD
17 OTHERWISE
18 IF EXT.LVL.INSPE = "C" LET EXT.INSPE.LEVEL = 3 JUMP AHEAD
19 OTHERWISE
20 LET EXT.INSPE.LEVEL = 4
21
22 **
23 ** CHANGE COMBUSTION RESISTANCE FACTOR TO MULTIPLYING FACTOR
24
25 IF CRF = 1 LET CRNF = 1.50 JUMP AHEAD
26 OTHERWISE
27 IF CRF = 2 LET CRNF = 1.25 JUMP AHEAD
28 OTHERWISE
29 IF CRF = 3 LET CRNF = 1.00 JUMP AHEAD
30 OTHERWISE
31 LET CRNF = 0.75
32
33 **
34 FOR I = 1 TO 4
35 LET AMCO(I) = 1/AMCO(I)
36 LET OICM(I) = 0
37 LET OICN(I) = 0
38
39 **
40 FOR I = 1 TO 10
41 LET MTIME-ACFT(I) = 1
42 LET WROD(I) = 0.0
43
44 **
45 LET FICK = 0
46 LET LMTA = 10
47 LET NICH = 1
48 LET NICH = 0
49 LET USOW = 0
50 LET USOW = 0
51 LET USOW = 0
52 LET USOW = 0
53 LET USOW = 0

```

09/01/76 PAGE 9

LINE CACI SINCRIPT II.5 RELEASE AF

```

54 LET MSFL = 0
55 LET NRFS = 0
56 LET WMD = 0
57 LET YSMD = 0
58 LET NSIC = 0
59 LET 1.NUM.OF.RETIME = 0
60 LET 2.NUM.OF.CRASH = 0
61 LET ICA = 0
62 LET TIME.V = 0.0
63 LET AIRFRAME.TIME = 0.0
64 LET CUST.OF.REPAIRS = 0.0
65 LET TMSL = TARCH(1)
66 LET OTMSL = TARCH(4)
67 LET LIL = ITRM
68 LET LFL = EXT.INSPELVEL
69 IF LFL = 1 OR LFL = 2
70 IF LIL = 1 OR LIL = 2
71 LET LIL = 3
72 ALWAYS
73 LET LEL = 3
74 REGARDLESS
75 LET T.FREQ.CMG = 1.0 - 1.FREQ.CMG
76 LET S.FREQ.CMG = 1.0 - S.FREQ.CMG
77 LET PRE.OFCREASE = 1.0 + PRE.OFCREASE
78 LET LTM = "NO"
79 LET TES.FAILURE = "NO"
80 LET MOD.TESTED = "NO"
81 LET OFC.OF.WO.SCH = "NO"
82 LET IMP.WO.SCH = "NO"
83 LET PREVIOUSLY.WO.TESTED = "NO"
84
85 FOR I = 1 TO SIZE.OF.FLEFT
86 DO
87 LET IE1(I) = "NO"
88 LET IE2(I) = "NO"
89 LET IE3(I) = "NO"
90 LET CGRI(I) = 1.0
91 LET OCCUR.WO(I) = 0.0
92 LET AIL(I) = "NO"
93 LET FSM(I) = "NO"
94 LET INSP.SCH(I) = "NO"
95 LET WND.PENDING(I) = "NO"
96 LET C.INTERVAL(I) = TARCH(3)
97 LET D.INTERVAL(I) = TARCH(4)
98 LET SWND.PENDING(I) = "NO"
99 LET 1.CW.EXISTS(I) = "NO"
100 LET 2.CW.EXISTS(I) = "NO"
101 LET 3.CW.EXISTS(I) = "NO"
102 LET CW.EXISTS(I) = "NO"
103 LET SD.SCH(I) = "NO"
104 LET SD.SCH(I) = "NO"
105 LOOP
106
107 ** RESET TALLY COUNTERS

```

09/01/76 PAGE 10

LIME CACT SIMSCRIPT II.5 RELEASE AF

107 " RESET TOTALS OF ICRT
108 RESET TOTALS OF ICRT
109 RESET TOTALS OF ISOT
110 RESET TOTALS OF ACRKL
111 RESET TOTALS OF WCRKL
112 RESET TOTALS OF CCRKL
113 RESET TOTALS OF DCRKL
114 RESET TOTALS OF SCRKL
115 RESET TOTALS OF ACA
116 RESET TOTALS OF PCA
117 RESET TOTALS OF CCA
118 RESET TOTALS OF DCA
119 RESET TOTALS OF SCA
120 RETURN
121 END
122

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDQ

09/01/76 PAGE 11

LINE CACI SINGAPORE 11.5 RELEASE 46

1 RUNTIME SUM, INITIALIZE
FOR I = 1 TO 4

2 DO

3 LET GCHC(1) = 0

4 LET GCHC(2) = 0

5 LET GCHC(3) = 0

6 LET GCHC(4) = 0

7 LET SASC = 0

8 LET SASC = 0

9 LET SASC = 0

10 LET SASC = 0

11 LET SASC = 0

12 LET GCHC = 0

13 LET GCHC = 0

14 LET GCHC = 0

15 LET GCHC = 0

16 LET GCHC = 0

17 LET GCHC = 0

18 LET GCHC = 0

19 RESET TOTALS OF GCHC

20 RESET TOTALS OF GCHC

21 RESET TOTALS OF GCHC

22 RESET TOTALS OF GCHC

23 RESET TOTALS OF GCHC

24 RESET TOTALS OF GCHC

25 RESET TOTALS OF GCHC

26 RESET TOTALS OF GCHC

27 RESET TOTALS OF GCHC

28 RESET TOTALS OF GCHC

29 RESET TOTALS OF GCHC

30 RESET TOTALS OF GCHC

31 RESET TOTALS OF GCHC

32 RESET TOTALS OF GCHC

33 RESET TOTALS OF GCHC

34 RESET TOTALS OF GCHC

35 RESET TOTALS OF GCHC

36 RESET TOTALS OF GCHC

00/21/76 PAGE 12

```

LINE CACI SCRIPT II.5 RELEASE 41
1 ROUTINE REAL.LIFE(MEAN,STD.DEV,POL) YIELDING REL
2 " "
3 " " POL = PREDICTED AVERAGE FATIGUE LIFE OF ELEMENT REL
4 " " REL = ACTUAL AVERAGE FATIGUE LIFE OF ELEMENT DESIGN
5 " "
6 " " RATIO = RATIO OF ACTUAL LIFE / PREDICTED DESIGN
7 " " IF RATIO > 10, RATIO IS SET TO 10
8 " " IF RATIO < .1, RATIO IS SET TO .1
9 " " DISTRIBUTION OF RATIOS IS SET TO .1
10 " " MEAN = MEAN OF RATIOS IS REPRESENTED AS LOG NORMAL
11 " " STD.DEV = STANDARD DEVIATION OF RATIOS
12 " "
13 LET RATIO = LOG.NORMAL.F(MEAN,STD.DEV,.1)
14 IF RATIO > 10.0
15 IF RATIO < 0.1
16 LET REL = RATIO * POL
17 RETURN
END

```

MEANLESS
MEGAPRESS

LOCAL VARIABLES OF THIS ROUTINE

NAME	TYPE	VALUE
MEAN	REAL	1
RATIO	REAL	0
STD.DEV	REAL	1
POL	REAL	1
REL	REAL	1
RND	REAL	1

09/01/76 PAGE 13

LINE CACI SIMSCRIPT II.5 RELEASE 8F

```

1  EVENT ENTER SERVICE
2  LET ILOC = ILOC + 1
3  LET ID = ILOC
4  CREATE AN AIRCRAFT CALLED AIRPLANE(ID)
5  LET TAIL.ID(AIRPLANE(ID)) = 10
6  LET ENTRY.TIME(AIRPLANE(ID)) = TIME.V
7  FILE AIRPLANE(ID) IN ACTIVE.FLEET
8  CALL FATIGUE.LIFE(SCATTER(ACTUAL.AVG.FAT.LIFE,A) YIELDING FIRST.LIFE,
9  SECOND.LIFE AND THIRD.LIFE
10 LET STD.FAST = M2.MEAN + .15
11 LET STD.SLOW = M1.MEAN + .15
12 LET MSR(ID) = RATE(M1.MEAN,STD.FAST,A)
13 LET MSR(ID) = RATE(M2.MEAN,STD.SLOW,A)
14 CALL PREDICT.CORROSION YIELDING HOURS.10.CORROSION
15 CALL PREDICT.SERVICE.DAMAGE YIELDING HOURS.TU.SERVICE.DAMAGE
16 IF HOURS.TU.CORROSION(ID) AT TIME.V + HOURS.10.CORROSION
17 SCHEDULE A CORROSION(ID) AT TIME.V + HOURS.10.CORROSION
18 LET AC(ID) = CORROSION
19 LET COMEP.TIME(ID) = TIME.V
20 LET CO.EXISTS(ID) = "MS"
21 REGARDLESS
22 **
23 IF HOURS.TU.SERVICE.DAMAGE LT USAGE.LIFE
24 SCHEDULE AN TU.SERVICE.DAMAGE(ID) AT TIME.V + HOURS.TU.SERVICE.DAMAGE
25 LET LAST.SD(ID) = TIME.V
26 LET SD.SCH(ID) = "YES"
27 REGARDLESS
28 IF RANDOM.F(7) LE MIN.TM.OBJECT.PROBABILITY
29 LET AN = RANDOM.F(3)
30 LET DEFECT.LIFE = (-4042.6 + LOG.E.F(MR))**.0132
31 ADD 1 TO SDID
32 IF DEFECT.LIFE LT FIRST.LIFE
33 LET FIRST.LIFE = DEFECT.LIFE
34 REGARDLESS
35 REGARDLESS
36 IF LTWO = "YES"
37 FOR I = 1 TO MAX(LDZ)
38 DO
39 IF ID = TLID(LDZ,I)
40 LET LIST = 1.0
41 SKIP 1 INPUT LINE
42 PRINT 1 LINE WITH ID, TIME.V AS FOLLOWS
43 A/C NO. *** ENTERS SERVICE ***** HOURS FROM START OF SIMULATION
44 IF DEFECT.LIFE GT 0.0
45 PRINT 1 LINE AS FOLLOWS
46 PRODUCTION DEFECT PRESENT
47 ALWAYS
48 SKIP 1 INPUT LINE
49 PRINT 3 LINES WITH FIRST.LIFE, SECOND.LIFE, THIRD.LIFE AS FOLLOWS
50 1ST CRACK INITIATION PROJECTED AT ***** FLIGHT HOURS
51 2ND CRACK INITIATION PROJECTED AT ***** FLIGHT HOURS
52 3RD CRACK INITIATION PROJECTED AT ***** FLIGHT HOURS

```

LINE CACT SIMSCRIPT II.5 RELEASE 8F 09/01/76 PAGE 12

```

40 PRINT 2 LINES WITH MSR(ID), MPR(ID) AS FOLLOWS
    SLOW CRACK GROWTH RATE = .000000 INCHES/MIN
    FAST CRACK GROWTH RATE = .000000 INCHES/HOUR
50 LEAVE
51 ELSE
52 LOOP
53 ALWAYS
54
55 IF FIRST.LIFE LT USAGE.LIFE OR CO.EXISTS(ID)="NS" OR SD.SCH(ID)="YES"
56 SCHEDULE A 1.STRENGTH.REDUCTION(ID) AT TIME.V + FIRST.LIFE
57 LET AISR(ID) = 1.STRENGTH.REDUCTION
58 LET CREP.TIME(ID) = TIME.V
59 LET 1.CR.EXISTS(ID) = "NS"
60 IF SECOND.LIFE LT USAGE.LIFE OR CO.EXISTS(ID)="NS" OR SD.SCH(ID)="YES"
61 SCHEDULE A 2.STRENGTH.REDUCTION(ID) AT TIME.V + SECOND.LIFE
62 LET ARSR(ID) = 2.STRENGTH.REDUCTION
63 LET 2.CR.EXISTS(ID) = "NS"
64 IF THIRD.LIFE LT USAGE.LIFE OR CO.EXISTS(ID)="NS" OR SD.SCH(ID)="YES"
65 SCHEDULE A 3.STRENGTH.REDUCTION(ID) AT TIME.V + THIRD.LIFE
66 LET AISR(ID) = 3.STRENGTH.REDUCTION
67 LET 3.CR.EXISTS(ID) = "NS"
68 REGARDLESS
69 REGARDLESS
70 REGARDLESS
71 SCHEDULE A 0.LEVEL.INSPECTION(ID) AT TIME.V + 0.INTERVAL(ID)
72 LET ADL(ID) = 0.LEVEL.INSPECTION
73 SCHEDULE A 4.TIME.PROV.SERVICE(ID) AT TIME.V + USAGE.LIFE
74 IF INCK EQ SIZE.OF.FLSET
75 RETURN
76 OTHERWISE
77 IF TIME.V = BEGIN.PRODUCTION OF PRCHG
78 SCHEDULE AN ENTER.SERVICE AT TIME.V + 2.PRODUCTION.TIME
79 RETURN
80 OTHERWISE
81 SCHEDULE AN ENTER.SERVICE AT TIME.V + PRODUCTION.TIME
82 RETURN
83 END

```

LOCAL VARIABLES OF THIS ROUTINE

DEFECT.LIF	DOUBLE	WORD 17	FIRST.LIFE	DOUBLE	WORD 1
MOOMS.TO.C	DOUBLE	WORD 11	1.1	INTEGER	WORD 25
I.2	INTEGER	WORD 26	1.3	INTEGER	WORD 27
J.1	INTEGER	WORD 28	K.1	INTEGER	WORD 30
K.2	INTEGER	WORD 31	K.3	INTEGER	WORD 32
K.4	INTEGER	WORD 33	L.4	INTEGER	WORD 31
L.6	INTEGER	WORD 34	LIST	DOUBLE	WORD 19
K.1	INTEGER	WORD 29	UMS.TO.SP	DOUBLE	WORD 13
K.1	DOUBLE	WORD 23	AN	DOUBLE	WORD 15
SECOND.LIF	DOUBLE	WORD 3	STD.FAST	DOUBLE	WORD 7
STD.SLOW	DOUBLE	WORD 9	THIRD.LIFE	DOUBLE	WORD 5

09/01/76 PAGE 15

LINE CACI SIMSCRIPT II.5 RELEASE 8F

```

1  RUNTIME FATIGUE.LIFE,SCATTER(MEL,N) YIELDING FIRST.LIFE, SECOND.LIFE AND
2  THIRD.LIFE
3  **
4  ** GENERATES FLEAFIT FATIGUE LIVES REFLECTING BASIC FATIGUE SCATTER AND LOAD
5  ** ENVIRONMENT VARIATION
6  **
7  ** DISTRIBUTION OF FATIGUE LIVES IS REPRESENTED AS TRIU PARAMETER *FINULL
8  ** ALPHA = SHAPE PARAMETER
9  ** BETA = SCALE PARAMETER (CHARACTERISTIC VALUE)
10 ** N = RANDOM NUMBER STREAM
11 **
12  DEFINE N AS AN INTEGER VARIABLE
13  DEFINE LIFE AS A REAL, 1-DIMENSIONAL ARRAY
14  RESERVE LIFE(*) AS 3
15  LET ALPHA = 2.78
16  LET BETA = 4617.80018
17  LET M4 = MAXIMUM.F.(2)
18  FOR I = 1 TO 3, DO
19    LET LIFE(I) = BETA * (LOG.E.(1/M4))**(1/ALPHA)
20    LET M4 = UNIFORM.F.(0.0,M4,1)
21  LOOP
22  LET FIRST.LIFE = LIFE(1)
23  LET SECOND.LIFE = LIFE(2)
24  LET THIRD.LIFE = LIFE(3)
25  RELEASE LIFE(*)
26  RETURN
27  END

```

LOCAL VARIABLES OF THIS ROUTINE

ALPHA	DOUBLE	WORD 13	BETA	DOUBLE	WORD 15
FIRST.LIFE	DOUBLE	WORD 5	LIFE	REAL	WORD 11
N	INTEGER	WORD 3	M4	DOUBLE	WORD 1
M4	DOUBLE	WORD 17	SECOND.LIFE	DOUBLE	WORD 7
THIRD.LIFE	DOUBLE	WORD 9			

09/01/76 PAGE 16

LINE CACI SIMSCRIPT II.5 RELEASE AF

```

1 ROUTINE INSTALL MODIFICATION
2
3 ** THIS ROUTINE REPRESENTS THE INSTALLATION OF A STRUCTURAL MODIFICATION CAUSED
4 BY A FATIGUE TEST FAILURE OR BY A SERVICE DEFECT FOUND IN THE ELEMENT
5
6 LET (SV) = VSND + 1
7 LET REEAMODIFIED = REENMODIFIED + 1
8 LET C.INTERVAL(ID) = LARCD(3)
9 LET D.INTERVAL(ID) = LARCD(4)
10 LET (CCURMOD(ID) = TIME.V
11
12 ** CANCEL SCHEDULED INSPECTIONS
13
14 IF INSP.SCH(ID) = "YES"
15 CALL CANCEL.SCHEDULED.INSPECTIONS
16 REGARDLESS
17
18 IF FSM(ID) = "YES"
19 LET FSM(ID) = "NO"
20 LET FAILURE = AF(ID)
21 CANCEL THE FAILURE
22 DESTROY THE FAILURE
23 ALWAYS
24 CALL FATIGUE.LIFE.SCATTER(ACTUAL.AVG.FAT.LIFE,A) YIELDING FIRST.LIFE.
25 SECOND.LIFE AND THIRD.LIFE
26 LET STD.SLOX = M1.MEAN + .15
27 LET STD.FAST = M2.MEAN + .15
28 LET WSR(ID) = RATE(M1.MEAN,STD.SLOX,S)
29 LET WFR(ID) = RATE(M2.MEAN,STD.FAST,S)
30 LET WST = USAGE.LIFE - TIME.V + FATUR.LIFE (DISPLAYE(ID))
31
32 ** CANCEL PREVIOUSLY SCHEDULED CRACK AND CORROSION INITIATIONS, RESCHEDULE
33 THESE EVENTS IF THEY OCCUR WITHIN THE REMAINING SERVICE LIFE (RST) OF THE
34 AIRCRAFT
35
36 IF CO.EXISTS(ID) NE "NN"
37 LET CORROSION = AC(ID)
38 IF CO.EXISTS(ID) = "AS"
39 CANCEL THE CORROSION
40 ALWAYS
41 DESTROY THE CORROSION
42 LET CO.EXISTS(ID) = "NN"
43 LET CORN(ID) = 1.0
44 REGARDLESS
45 CALL PREDICT CORROSION YIELDING MUORS.ID,CORROSION
46 IF MUORS.ID,CORROSION LT WST
47 SCHEDULE A CORROSION AT TIME.V + MUORS.ID,CORROSION
48 LET AC(ID) = CORROSION
49 LET CORN.TIME(ID) = TIME.V
50 LET CO.EXISTS(ID) = "AS"
51 REGARDLESS
52
53 ** IF WANDOW.F(7) LE HIGHM.DEFECT.PRIORABILITY

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDQ

09/01/76 PAGE 17

```

LINE  CACI SIMSCRIPT II.5  RELEASE AF
54      LET UPD = OPD + 1
55      LET RN = RANDOM.F(3)
56      LET DEFECT.LIFE = (-4042.6 * LOG.E.F(RN)) + 1.0132
57      IF DEFECT.LIFE LT FIRST.LIFE
58      LET FIRST.LIFE = DEFECT.LIFE
59      REGARDLESS
60      REGARDLESS
61      "
62      IF 1.CR.EXISTS(ID) NE "NN"
63      LET 1.STRENGTH.REDUCTION = AISR(ID)
64      IF 1.CR.EXISTS(ID) = "NS"
65      CANCEL THE 1.STRENGTH.REDUCTION
66      REGARDLESS
67      DESTROY THE 1.STRENGTH.REDUCTION
68      LET 1.CR.EXISTS(ID) = "NN"
69      IF 2.CR.EXISTS(ID) NE "NN"
70      LET 2.STRENGTH.REDUCTION = APSW(ID)
71      IF 2.CR.EXISTS(ID) = "NS"
72      CANCEL THE 2.STRENGTH.REDUCTION
73      REGARDLESS
74      DESTROY THE 2.STRENGTH.REDUCTION
75      LET 2.CR.EXISTS(ID) = "NN"
76      IF 3.CR.EXISTS(ID) NE "NN"
77      LET 3.STRENGTH.REDUCTION = ASSW(ID)
78      IF 3.CR.EXISTS(ID) = "NS"
79      CANCEL THE 3.STRENGTH.REDUCTION
80      REGARDLESS
81      DESTROY THE 3.STRENGTH.REDUCTION
82      LET 3.CR.EXISTS(ID) = "NN"
83      REGARDLESS
84      REGARDLESS
85      IF IE1(ID) = "YES"
86      LET 1.ITE = AIE(ID)
87      LET IE1(ID) = "NN"
88      CANCEL THE 1.ITE
89      DESTROY THE 1.ITE
90      IF IE2(ID) = "YES"
91      LET 2.ITE = APE(ID)
92      LET IE2(ID) = "NN"
93      CANCEL THE 2.ITE
94      DESTROY THE 2.ITE
95      IF IE3(ID) = "YES"
96      LET 3.ITE = AYE(ID)
97      LET IE3(ID) = "NN"
98      CANCEL THE 3.ITE
99      DESTROY THE 3.ITE
100     ALWAYS
101     ALWAYS
102     ALWAYS
103     ALWAYS
104     IF AIL(ID) = "YES"
105     LET REACH.FAIL.SAFE.LGT = AMFSL(ID)
106     CANCEL THE REACH.FAIL.SAFE.LGT

```

09/01/76 PAGE 1A

LINE CACI SIMSCRIPT II.5 RELEASE RF

```

107 DESTROY THE REACH.FAIL.SAFE.LGT
108 LET AIL(ID) = "NO"
109 REGARDLESS
110 ..
111 IF FIRST.LIFE LT RST OR CO.EXISTS(ID) = "NS" OR SD.SCH(ID) = "YES"
112 SCHEDULE A 1.STRENGTH.REDUCTION(ID) AT TIME.V + FIRST.LIFE
113 LET AISR(ID) = 1.STRENGTH.REDUCTION
114 LET CREP.TIME(ID) = TIME.V
115 LET 1.CR.EXISTS(ID) = "NS"
116 IF SECOND.LIFE LT RST OR CO.EXISTS(ID) = "NS" OR SD.SCH(ID) = "YES"
117 SCHEDULE A 2.STRENGTH.REDUCTION(ID) AT TIME.V + SECOND.LIFE
118 LET APSR(ID) = 2.STRENGTH.REDUCTION
119 LET 2.CR.EXISTS(ID) = "NS"
120 IF THIRD.LIFE LT RST OR CO.EXISTS(ID) = "NS" OR SD.SCH(ID) = "YES"
121 SCHEDULE A 3.STRENGTH.REDUCTION(ID) AT TIME.V + THIRD.LIFE
122 LET AISR(ID) = 3.STRENGTH.REDUCTION
123 LET 3.CR.EXISTS(ID) = "NS"
124 REGARDLESS
125 REGARDLESS
126 REGARDLESS
127 ..
128 IF LTIM = "YES"
129 FOR I = 1 TO NMAC(LDX)
130 DO
131 IF IO = TLID(LOX,I)
132 SKIP 1 OUTPUT LINE
133 PRINT 6 LINES WITH ID, TIME.V-ENTRY.TIME(AIRPLANE(ID)), FIRST.LIFE,
134 SECOND.LIFE, THIRD.LIFE, MSR(ID), MFR(ID) AS FOLLOWS
      MODIFICATION INSTALLED ON A/C NO. *** AT ***** FLIGHT HOURS
      1ST CRACK INITIATION PROJECTED AT ***** FLIGHT HOURS
      2ND CRACK INITIATION PROJECTED AT ***** FLIGHT HOURS
      3RD CRACK INITIATION PROJECTED AT ***** FLIGHT HOURS
      SLOW CRACK GROWTH RATE = ***** INCHES/HOUR
      FAST CRACK GROWTH RATE = ***** INCHES/HOUR
135 LEAVE
136 ELSE
137 LOOP
138 ALWAYS
139 RETURN
140 END

```

LOCAL VARIABLES OF THIS ROUTINE

OBJECT.LIF	DOUBLE	WORD 17	FIRST.LIFE	DOUBLE	WORD 1
MOURS.TO.C	DOUBLE	WORD 13	I.1	INTEGER	WORD 23
I.2	INTEGER	WORD 24	I.3	INTEGER	WORD 25
J.1	INTEGER	WORD 26	K.1	INTEGER	WORD 28
K.2	INTEGER	WORD 29	K.3	INTEGER	WORD 30
K.4	INTEGER	WORD 31	L.4	INTEGER	WORD 19
N.1	INTEGER	WORD 27	R.1	DOUBLE	WORD 21
RN	DOUBLE	WORD 15	RST	DOUBLE	WORD 11
SECOND.LIF	DOUBLE	WORD 3	STD.FAST	DOUBLE	WORD 9

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

09/01/76 PAGE 19

LINE CACT SINSRIPT 11.5 RELEASE RF
STD.SLOW DOUBLE WORD 7 THIND.LIFE DOUBLE WORD 5

09/01/76 PAGE 20

LINE CACT SYMSCRIPT II.5 RELEASE RF

```

1 EVENT IN.SERVICE.DAMAGE(IDSNM)
2 **
3 ** REPRESENTS THE OCCURRENCE OF A SERVICE DAMAGE DEFECT; RESULTS IN IMMEDIATE
4 ** INITIATION OF NEXT SCHEDULED CRACK
5 **
6 **
7 **
8 **
9 **
10 **
11 **
12 **
13 **
14 **
15 **
16 **
17 **
18 **
19 **
20 **
21 **
22 **
23 **
24 **
25 **
26 **
27 **
28 **
29 **
30 **
31 **
32 **
33 **
34 **
35 **
36 **
37 **
38 **
39 **
40 **
41 **
42 **
43 **
44 **
45 **
46 **
47 **
48 **
49 **
50 **
51 **

```

DEFER INSDX AS AN INTEGER VARIABLE
 LET ID = INSDM
 IF TIME = "YES"
 FOR I = 1 TO MAXC(LOX)
 ON
 IF ID = ALLC(LOX, I)
 LET LIST = 1.0
 SKIP 1 INPUT LINE
 P-T-T P LINES WITH ID, TIME, V-ENTRY, TIME(AIRPLANE(ID)) AS FOLLOWS
 A/C NO. *** EXPERIENCES SERVICE DAMAGE AT ***** FLIGHT HOURS
 CAUSES IMMEDIATE INITIATION OF NEXT SCHEDULED CRACK
 LEAVE
 FLSP
 LOOP
 ALWAYS
 LET INSDM = INSDM + 1
 LET GOSDM = GOSDM + 1
 LET ISDT = TIME.V - LAST.SD(ID)
 LET LISD = TIME.V - LAST.SD(ID)
 CALL PREDICT.SERVICE.DAMAGE.YIELDING(DMS, IO.SERVICE.DAMAGE
 LET PST = USAGE.LIFE - TIME.V + ENTRY.TIME(AIRPLANE(ID))
 LET SO.SCH(ID) = "NO"
 IF DMS.IO.SERVICE.DAMAGE (T.MSI
 SCHEDULE AN IN.SERVICE.DAMAGE(ID) AT TIME.V + (DMS.IO.SERVICE.DAMAGE
 LET LAST.SD(ID) = TIME.V
 LET SO.SCH(ID) = "YES"
 REGARDLESS
 ** TEST FOR SCHEDULED FIRST CRACK INITIATION
 **
 LET 1.STRENGTH.REDUCTION = AIS*(ID)
 LET 2.STRENGTH.REDUCTION = APS*(ID)
 LET 3.STRENGTH.REDUCTION = AIS*(ID)
 IF 1.CR.EXISTS(ID) = "NS"
 CANCEL THE 1.STRENGTH.REDUCTION
 RESCHEDULE THE 1.STRENGTH.REDUCTION(ID) ***
 IF SO.SCH(ID) = "NO" AND TIME.A(2.STRENGTH.REDUCTION) GE MSI + TIME.V
 AND CO.EXISTS(ID) GE "NS"
 CANCEL THE 2.STRENGTH.REDUCTION
 OFSTWY THE 2.STRENGTH.REDUCTION
 CANCEL THE 3.STRENGTH.REDUCTION
 DESTROY THE 3.STRENGTH.REDUCTION
 LET 2.CR.EXISTS(ID) = "NO"
 LET 3.CR.EXISTS(ID) = "NO"
 ALWAYS
 RETURN
 OTHERWISE

09/01/70 PAGE 21

LOCAL VARIABLES AT THIS ROUTINE					
I.1	INTERM	WORD 9	I.2	INTERM	WORD 10
I.3	INTERF	WORD 11	LOSDW	INTERF	WORD 11
J.1	INTERL	WORD 12	K.1	INTERM	WORD 12
K.2	INTERW	WORD 15	L.1	INTERF	WORD 15
K.3	INTERW	WORD 17	L.2	INTERF	WORD 15
LIST	NUMBL	WORD 3	N.1	INTERM	WORD 13
NUM	NUMBL	WORD 19	N.1	NUMBL	WORD 7
NUM	NUMBL	WORD 21			

LINE CACI SIMSCRIPT II.5 RELEASE NF 09/01/76 PAGE 22

```

1 EVENT T.IMPLEMENT.MOD
2 **
3 ** REPRESENTS DEVELOPMENT OF MODIFICATION BECAUSE OF FATIGUE TEST FAILURE
4 **
5     LET AMO = AMO + 1
6     LET FOCK = IOCK
7     LET REPAIR.MODIFIED = 0
8     LET TO.REPAIR.MODIFIED = IOCK
9     LET COST.OF.REPAIRS = 0.0
10    LET AIRFRAME.TIME = 0.0
11    LET ACTUAL.AVG.FAT.LIFE = PREDICTED.LIFE
12    IF MOD.TESTED = "NO"
13    LET AMO = AMO + ((1.0-MO.R)*.15)
14    LET NSIG = SIG.R + .45
15    CALL RECALC.LIFE(NMU,NSIG,PREDICTED.LIFE) YIELDING ACTUAL.AVG.FAT.LIFE
16    REGARDLESS
17    FOR EVERY AIRCRAFT IN ACTIVE.FLEET
18    DO
19        LET ID = TAIL.ID
20        IF IAFL LT USAGE.LIFE
21        LET TMOD.PEYOTING(ID) = "YES"
22        IF ENTRY.TIME(AIRPLANE(ID)) + C7+IAFL GT TIME.Y
23        SCHEDULE A T.TSPECTION.INCREASE(ID) AT ENTRY.TIME(AIRPLANE(ID)) + C7 +
24        IAFL
25        LET ATII(ID) = T.INSPECTION.INCREASE
26        ALWAYS
27        REGARDLESS
28        LOOP
29        RETURN
30    END

```

LOCAL VARIABLES OF THIS ROUTINE

INTEGER	AMO	5	NMU		DOUBLE	WORD	1
DOUBLE	AMO	4					

L.9
NSIG

LINE CACT SIMSCRIPT II.5 RELEASE PF 09/01/76 PAGE 23

```

1 ROUTINE PREDICT SERVICE DAMAGE YIELDING DMS.TU.SERVICE.DAMAGE
2 **
3 ** GENERATES TIME TO SERVICE DAMAGE OCCURRENCE FOR A GIVEN AIRCRAFT FROM A
4 ** CONSTANT SERVICE DAMAGE OCCURRENCE RATE
5 **
6 ** IF SPM.OCCURRENCE RATE LE 0.0
7   LET DMS.TU.SERVICE.DAMAGE = 2.0 * (USARF.LIFE
8   RETURN
9 ELSE
10  LET DO = MAXIMUM(2)
11  LET DMS.TU.SERVICE.DAMAGE = -LOG.F.F.(DO) / SPM.OCCURRENCE.RATE
12  RETURN
13

```

LOCAL VARIABLES OF THIS ROUTINE

DMS.TU.SERVICE.DAMAGE 1 DO 20 DMS.TU.SERVICE.DAMAGE 5

LINE CACT SIMSCRIPT II.5 RELEASE OF 09/01/76 PAGE 24

```

1 ROUTINE PREICT.CORROSION YIELDING HOURS.TO.CORROSION
2 **
3 ** GENERATES TIME TO CORROSION INITIATION FROM ELEMENT HAZARD RATE WHICH IS
4 ** APPROXIMATED BY TWO CONSTANT OCCURRENCE RATES
5 **
6 ** 1.CCM.OCCURRENCE.RATE = FIRST CONSTANT OCCURRENCE RATE
7 ** 2.CCM.OCCURRENCE.RATE = SECOND CONSTANT OCCURRENCE RATE
8 ** CCM.RATE.CHANGE = TIME OF AIRFRAME AT WHICH SECOND RATE IS USED
9 **
10 LET HN = MAXIMUM.I(4)
11 LET CACT = CCM.RATE.CHANGE - TIME.V * ENTRY.TIME(AIRPLANE(10))
12 IF CACT LE 0.0
13 IF 2.CCM.OCCURRENCE.RATE LE 0.0
14 LET MINUS.I(1).CORROSION = 2.0 * USAGE.LIFE
15 RETURN
16 ELSE
17 LET HOURS.TO.CORROSION = LOG.F.F(HN) / (-2.CCM.OCCURRENCE.RATE)
18 RETURN
19 OTHERWISE
20 **
21 IF HN GE MAX.F(-CACT * 1.CCM.OCCURRENCE.RATE)
22 IF 1.CCM.OCCURRENCE.RATE LE 0.0
23 LET MINUS.I(1).CORROSION = 2.0 * USAGE.LIFE
24 RETURN
25 ELSE
26 LET MINUS.I(1).CORROSION = LOG.F.F(HN) / (-1.CCM.OCCURRENCE.RATE)
27 RETURN
28 OTHERWISE
29 IF 2.CCM.OCCURRENCE.RATE LE 0.0
30 LET MINUS.I(1).CORROSION = 2.0 * USAGE.LIFE
31 RETURN
32 ELSE
33 LET LN = (2.CCM.OCCURRENCE.RATE - 1.CCM.OCCURRENCE.RATE) * CACT
34 LET HOURS.TO.CORROSION = LOG.E.F(HN/EXP.F(LN)) / (-2.CCM.OCCURRENCE.RATE)
35 RETURN
36 END

```

LOCAL VARIABLES OF THIS ROUTINE

NAME	TYPE	VALUE	INITIAL	FINAL
CACT	DOUBLE	5	MINUS.I(1).C	MINUS.I(1).C
LN	DOUBLE	7	MINUS.I(1).C	MINUS.I(1).C

LINE CACT SIMSCRIPT II.5 RELEASE AF 09/01/76 PAGE 25

```

1  EVENT COMBUSTION(LOC) SAVING THE EVENT NOTICE
2  ..
3  ..
4  ..
5  ..
6  DEFINE ILOC AS AN INTEGER VARIABLE
7  LET ILOC = ILOC
8  LET RST = JSACT.LIFE - TIME.V + ENTRY.TIME(AIRPLANE(ID))
9  LET VCMT = TIME.V - CTIME.TIME(ID)
10 LET RICHM = TIME.V - CTIME.TIME(ID)
11 LET RICHM = ILOC + 1
12 LET RICHM = RICHM + 1
13 LET CTIME(ID) = "NO"
14 IF RICHM.F(10) LF CTIME
15 LET CTIME(ID) = "YES"
16 IF CTIME(ID) = "YES"
17 LET CTIME(ID) = "YES"
18 IF INSP.SCH(10) = "NO"
19 CALL INSPECTION.SCHEDULE(2)
20 IF CTIME(ID) = "YES"
21 LET CTIME.MULPLYING.FACTOR = 1.0 - C20
22 IF RICHM.F(10) LF INCRATED.IN.SIMPSS.COM
23 LET CTIME.MULPLYING.FACTOR = 1.0 - C20
24 IF CTIME(ID) = "YES"
25 LET CTIME(ID) = 2.0 - CTIME.MULPLYING.FACTOR
26 IF CTIME(ID) = "YES"
27 FOR I = 1 TO CTIME(LOC)
28 DO
29 IF ILOC = ILOC(LOC)
30 LET LIST = 1.0
31 SET ILOC(LOC) = ILOC
32 SET ILOC(LOC) = ILOC
33 SET ILOC(LOC) = ILOC
34 SET ILOC(LOC) = ILOC
35 SET ILOC(LOC) = ILOC
36 SET ILOC(LOC) = ILOC
37 SET ILOC(LOC) = ILOC
38 SET ILOC(LOC) = ILOC
39 SET ILOC(LOC) = ILOC
40 SET ILOC(LOC) = ILOC
41 SET ILOC(LOC) = ILOC
42 SET ILOC(LOC) = ILOC
43 SET ILOC(LOC) = ILOC
44 SET ILOC(LOC) = ILOC
45 SET ILOC(LOC) = ILOC
46 SET ILOC(LOC) = ILOC
47 SET ILOC(LOC) = ILOC
48 SET ILOC(LOC) = ILOC
49 SET ILOC(LOC) = ILOC
50 SET ILOC(LOC) = ILOC

```

09/01/70 PAGE 26

```

LINE CACI S1500001 11.5 RELEASE RF
51 LET 2.ITE = ADE(ID)
52 LET TRI = (TIME.A(2.ITE)-TIME.V) / CGN1(ID)
53 CANCEL THE 2.ITE
54 IF TRI LT RST
55 RESCHEDULE THE 2.ITE AT TIME.V + TRI
56 JUMP AHEAD
57 ELSE
58 DESTROY THE 2.ITE
59 LET I2(ID) = "NO"
60 WHERE
61 IF I2(ID) = "YES"
62 LET 3.ITE = ARE(ID)
63 LET TRI = (TIME.A(3.ITE)-TIME.V) / CGN1(ID)
64 CANCEL THE 3.ITE
65 IF TRI LT RST
66 RESCHEDULE THE 3.ITE AT TIME.V + TRI
67 JUMP AHEAD
68 ELSE
69 DESTROY THE 3.ITE
70 LET I3(ID) = "NO"
71 WHERE
72 IF I3(ID) = "YES"
73 ALWAYS
74 ALWAYS
75 ALWAYS
76 IF ALL(ID) = "YES"
77 LET REACH.FAIL.SAFE.LGT = ANFS1(ID)
78 LET TRI = (TIME.A(REACH.FAIL.SAFE.LGT) - TIME.V) / CGN1(ID)
79 CANCEL THE REACH.FAIL.SAFE.LGT
80 IF TRI LT RST
81 RESCHEDULE THE REACH.FAIL.SAFE.LGT
82 JUMP AHEAD
83 ELSE
84 DESTROY THE REACH.FAIL.SAFE.LGT
85 LET ALL(ID) = "NO"
86 WHERE
87 RECAPLESS
88 IF FSH(ID) = "YES"
89 LET FAILURE = AF(ID)
90 LET .PTA = TIME.A(FAILURE) - TIME.V
91 PRINT 1 LINE WITH TIME.V+PTA*CONV.MULTIPLYING.FAC10W-
92 ENTRY, TIME(AMPLAMF(10)) AS FOLLOWS
93 ALWAYS ELEMENT FAILURE PROJECTED AT ***** FLIGHT QUINS
94 CANCEL THE FAILURE
95 IF NOT LT RST
96 RESCHEDULE THE FAILURE(ID) AT TIME.V + PTA / CGN1(ID)
97 JUMP AHEAD
98 ELSE
99 DESTROY THE FAILURE
100 LET FSH(ID) = "NO"
101 WHERE
102 RECAPLESS

```

09/01/76 PAGE 27

LINE CACT SIMSCHIPI II.5 RELEASE AF

```

103 **
104 ** TEST FOR SCHEDULED FIRST CRACK INITIATION
105 **
106 IF 1.CC.EXISTS(10) = "NS"
107 LET 1.STRENGTH.REDUCTION = AISC(10)
108 LET REMAINING.LIFE = TIME.A(1.STRENGTH.REDUCTION) - TIME.V
109 LET REDUCED.REMAINING.LIFE = REMAINING.LIFE * (CDM.MULTIPLYING.FACTOR
110 IF LIST = 1.0
111 PRINT 1 LINE WITH TIME.V+REDUCED.REMAINING.LIFE+ENTRY.TIME(AIMPLANE(10))
112 AS FOLLOWS
113 1ST CRACK INITIATION PROJECTED AT ***** FLIGHT HOURS
114 ALWAYS
115 CANCEL THE 1.STRENGTH.REDUCTION
116 DESTROY THE 1.STRENGTH.REDUCTION
117 LET 1.CC.EXISTS(10) = "NS"
118 IF REDUCED.REMAINING.LIFE LT 1ST OR SD.SCH(10) = "YES"
119 SCHEDULE A 1.STRENGTH.REDUCTION(10) AT TIME.V + REDUCED.REMAINING.LIFE
120 LET AISC(10) = 1.STRENGTH.REDUCTION
121 LET 1.CC.EXISTS(10) = "NS"
122 ALWAYS
123 MEGAWLESS
124 **
125 ** TEST FOR SCHEDULED SECOND CRACK INITIATION
126 **
127 IF 2.CC.EXISTS(10) = "NS"
128 LET 2.STRENGTH.REDUCTION = A2SR(10)
129 LET REMAINING.LIFE = TIME.A(2.STRENGTH.REDUCTION) - TIME.V
130 LET REDUCED.REMAINING.LIFE = REMAINING.LIFE * (CDM.MULTIPLYING.FACTOR
131 IF LIST = 1.0
132 PRINT 1 LINE WITH TIME.V+REDUCED.REMAINING.LIFE+ENTRY.TIME(AIMPLANE(10))
133 AS FOLLOWS
134 2ND CRACK INITIATION PROJECTED AT ***** FLIGHT HOURS
135 ALWAYS
136 CANCEL THE 2.STRENGTH.REDUCTION
137 DESTROY THE 2.STRENGTH.REDUCTION
138 LET 2.CC.EXISTS(10) = "NS"
139 IF REDUCED.REMAINING.LIFE LT 1ST OR SD.SCH(10) = "YES"
140 SCHEDULE A 2.STRENGTH.REDUCTION(10) AT TIME.V + REDUCED.REMAINING.LIFE
141 LET A2SR(10) = 2.STRENGTH.REDUCTION
142 LET 2.CC.EXISTS(10) = "NS"
143 ALWAYS
144 MEGAWLESS
145 **
146 ** TEST FOR SCHEDULED THIRD CRACK INITIATION
147 **
148 IF 3.CC.EXISTS(10) = "NS"
149 LET 3.STRENGTH.REDUCTION = A3SR(10)
150 LET REMAINING.LIFE = TIME.A(3.STRENGTH.REDUCTION) - TIME.V
151 LET REDUCED.REMAINING.LIFE = REMAINING.LIFE * (CDM.MULTIPLYING.FACTOR
152 IF LIST = 1.0
153 PRINT 1 LINE WITH TIME.V+REDUCED.REMAINING.LIFE+ENTRY.TIME(AIMPLANE(10))
154 AS FOLLOWS
155 3RD CRACK INITIATION PROJECTED AT ***** FLIGHT HOURS

```

09/01/76 PAGE 28

```

LINE CACTI STNSCRIPT II.5 RELEASE HF
153 ALWAYS
154 CANCEL THE 1,STRENGTH,REDUCTION
155 DESTRUY THE 1,STRENGTH,REDUCTION
156 LET 3,C4,EXIST(10) = "NO"
157 IF REDUCED,REMAINING,LIFE = "NO"
158 SCHEMULE 1,1,STRENGTH,REDUCTION,LI 4ST IN ST,SC(10) = "YES"
159 LET 4,SS-(10) = 3,STRENGTH,REDUCTION,LI 4ST IN ST,SC(10) = "YES"
160 LET 1,C4,EXIST(10) = "NO"
161 ALWAYS
162 RETURN
163 END

```

LOCAL VARIABLES OF THIS ROUTINE

CONV. MULTIP	DOUBLE	WORD	5	1.1	INTEGER	WORD	15
I.2	INTEGER	WORD	14	I.3	INTEGER	WORD	15
I.3	INTEGER	WORD	1	J.1	INTEGER	WORD	16
K.1	INTEGER	WORD	14	K.2	INTEGER	WORD	19
K.3	INTEGER	WORD	20	L.4	INTEGER	WORD	21
L.4	INTEGER	WORD	9	L.6	INTEGER	WORD	22
LIST	DOUBLE	WORD	7	N.1	INTEGER	WORD	17
MTM	DOUBLE	WORD	25	N.1	DOUBLE	WORD	11
REDUCED,WF	DOUBLE	WORD	29	REMAINING,WF	DOUBLE	WORD	27
4ST	DOUBLE	WORD	3	REMAINING,WF	DOUBLE	WORD	23

09/01/76 PAGE 29

LINE CACI SYNOPSIS II.5 RELEASE AF

```

1 ROUTINE RATE(A,S,N)
2
3 ** GENERATES ELEMENT CRACK PROPAGATION RATES REFLECTING VARIATION IN MATERIAL
4 ** PROPERTIES AND LOAD ENVIRONMENT VARIATION
5
6 ** DISTRIBUTION OF CRACK PROPAGATION RATES IS REPRESENTED AS NORMAL
7 ** MEAN PROPAGATION RATE
8 ** S = STANDARD DEVIATION
9 ** N = RANDOM NUMBER STREAM
10
11 ** METHOD BASED ON APPROXIMATIONS IN C. HASTINGS, APPROXIMATIONS FOR DIGITAL
12 ** COMPUTERS
13
14 ** IF PROCEDURE YIELDS NEGATIVE RATE, RATE IS SET EQUAL TO ZERO MINUS FOUR
15 ** STANDARD DEVIATIONS
16
17 ** DEFINE Z AS AN INTEGER VARIABLE
18 LET Z = 1.0 - RANDOM.F(N)
19 LET Z = Z * N
20 IF Z < 0.5
21 LET Z = 1.0 - Z
22 RANDOMLESS
23 LET A = SINT.F((LOG.F.F(1.0/Z)))
24 LET G1 = 2.515517 * .0028688 * .0103286 * Z
25 LET G2 = 1.0 + 1.432788 * .140286 * Z + .001308 * Z**3
26 LET Z = A - G1/G2
27 IF Z < 0.5
28 LET Z = -Z
29 RANDOMLESS
30 LET Z = Z * S + A
31 IF Z < 0
32 LET Z = -Z * (A * S)
33 RANDOMLESS
34 RETURN A * N /
35

```

LOCAL VARIABLES OF THIS ROUTINE

NAME	TYPE	VALUE	ADDRESS
DOUBLE	WORD	13	G2
DOUBLE	WORD	1	N
INTEGER	WORD	5	S
DOUBLE	WORD	7	A
DOUBLE	WORD	3	A
DOUBLE	WORD	17	A

09/01/76 PAGE 30

```

LINE CACI SIMSCRIPT IL-5 RELEASE AF
1 EVENT 1- STRENGTH REDUCTION (IDISH) SAVING THE EVENT NOTICE
2 **
3 ** REPRESENTS FIRST CRACK INITIATION
4 **
5 OFFICE IDISH AS AN INTEGER VARIABLE
6 LET ID = IDISH
7 LET IAN = EVERY.TIME(AIRPLANE(ID)) * USAGE.LIFE
8 IF LTIM = "YES"
9   FOR I = 1 TO MAXCLOCX
10    DO
11     IF ID = LIDCLOCX(I)
12      LET LIST = 1.0
13      SKIP 1 OUTPUT LINE
14      PRINT 1 LINE WITH ID, TIME-V-EMPTY.TIME(AIRPLANE(ID)) AS FOLLOWS
15      A/C NO. *** EXPERIENCES 1ST CRACK INITIATION AT ***** HOURS
16      LEAVE
17     ELSE
18      ALWAYS
19      LET ICMT = TIME.V - CARRP.TIME(ID)
20      LET SICM = TIME.V - CARRP.TIME(ID)
21      LET OICM = TICM + 1
22      LET GICM = GICM + 1
23      LET LTIM(ID) = "NO"
24      IF MAXCLOCX(ID) LE 1.0
25       LET LTIM(ID) = "YES"
26       IF LIST = 1.0
27        PRINT 1 LINE AS FOLLOWS
28        CRACK INITIATES INTERNALLY
29        ALWAYS
30        LET I = (CRIT.CMK.LGT*PCCCL)/(MSW(ID)*CMT(ID))
31        IF PCCCL GT 1.0
32         LET I = CRIT.CMK.LGT/(MSR(ID)*CMT(ID)) + (CRIT.CMK.LGT*(PCCCL-1.0))/
33          (PCCCL)*CMT(ID)
34        ALWAYS
35        IF TIME.V + 1 LT IAN OR CULXISTS(ID) = "NS"
36         SCHEDULE A 1-IF(ID) AT TIME.V + I
37         LET AIC(ID) = 1.1*IF
38         LET IF1(ID) = "YES"
39         ALWAYS
40         REGARDLESS
41         LET 1-CA.EATSTS(ID) = "YES"
42         IF INSP.SCM(ID) = "NO"
43          CALL INSPECTION.SCHEDULEW(I)
44         REGARDLESS
45 ** PREDICT LIFE 1- EST. ONE FROM FIRST CRACK INITIATION
46 **
47 LET S0 = 2.75
48 LET SP = 1.0
49 LET KKH(ID) = MAXCLOCX(I)
50 LET LG = LOG.E.F(MW(ID))
51 LET GM1 = MSR(ID) * CMT(I)

```

09/01/76 PAGE 51

LINE CACTI SUBSCRIPT II.5 RELEASE AF

```

52 LET GR2 = MER(ID) + CCR1(ID)
53 LET I1 = CRIT.CRM.LGT / GR1
54 LET T2 = T1 + (FSAF.LGT - CRIT.CRM.LGT) / GR2
55 LET M1 = ((SU-SF)*GR1) / FSAF.LGT
56 LET R2 = ((SU-SF)*GR2) / FSAF.LGT
57 LET R3 = GR2 / (LCMT.IN.FAILURE - FSAF.LGT)
58 ..
59 IF TIME.V + T2 LT TAN IN CO.EXISTS(ID) = "S"
60 SCHEDULE A REACH.FAIL.SAF.LGT(ID) AT TIME.V + T2
61 LET AFRSL(ID) = GRACH.FAIL.SAF.LGT
62 LET AIL(ID) = "YES"
63 REGARDLESS
64 LET K1 = A + EXP.F(SU * R)
65 LET K2 = A * R1
66 LET TTF = -LOG.E.F((-R2/K1)*LG + 1.0)/R2
67 IF TTF LE T1
68 JUMP AHEAD
69 ELSE
70 LET S1 = SU - R1 * T1
71 LET K3 = M * R2
72 LET LK5 = LOG.F.F(A) + (M * S1 + R * R2 * T1)
73 LET K4 = (A*EXP.F(R*SI))/K3
74 LET K5 = (K1/R2)*(EXP.F(-R*SI)/T1)-1.0
75 LET TTF = (LK5 - LOG.F.F(K4*K5)) - LOG.E.F((K4*K5-K3))/K4
76 IF TTF LE T2
77 JUMP AHEAD
78 ELSE
79 LET K10 = M*R3
80 LET ARC = -R*SF*MER3*AT2
81 IF ARC LT -175.0
82 LET ANG = -175.0
83 ALWAYS
84 LET K11 = K10/(A*EXP.F(ARC))
85 LET K12 = A*EXP.F(R*SI)/K10
86 LET K13 = EXP.F(MER2*AT1 - MER2*T2) - 1.0
87 LET TTF = -LOG.F.F(K11*(LG*K12-(K4*K13)-0.0))/K10
88 HERE
89 IF LIST = 1.0
90 PRINT 1 LINE WITH TIME.V+TTF-FINITY.TTF(ZIMPLANE(ID)) AS FOLLWS
    ELEMENT FAILURE PROJECTION AT 00000 ELEMENT MQUIMS
    ALWAYS
91 IF TIME.V + TTF LT TAN IN CO.EXISTS(ID) = "S"
92 SCHEDULE A FAILURE(ID) AT TIME.V + TTF
93 LET AF(ID) = FAILURE
94 LET FSM(ID) = "YES"
95 ALWAYS
96 RFTIM.
97 END
98

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDQ

09/01/76 PAGE 32

LINE CACI SIMSCRIPT II.5 RELEASE 6F

LOCAL VARIABLES OF THIS ROUTINE

ARG	WORD 41	GM1	WORD 29
GR2	DOUBLE	I.1	WORD 11
I.2	INTEGER	I.3	WORD 13
IDISH	INTEGER	J.1	WORD 14
K.1	INTEGER	K.2	WORD 17
K.3	INTEGER	K.4	WORD 19
K1	DOUBLE	K10	WORD 59
K11	DOUBLE	K12	WORD 65
K13	DOUBLE	K2	WORD 45
K4	DOUBLE	K4	WORD 55
K9	DOUBLE	L.4	WORD 7
LG	DOUBLE	LGR5	WORD 53
LIST	DOUBLE	N.1	WORD 15
M.1	DOUBLE	R1	WORD 37
M2	DOUBLE	R5	WORD 41
SF	DOUBLE	SU	WORD 23
S1	DOUBLE	T	WORD 21
TAR	DOUBLE	TIF	WORD 47
T1	DOUBLE	T2	WORD 35

09/01/76 PAGE 53

LINE CACI SIMSCRIPT ILS RELEASE AF

1 EVENT P-STRENGTH-REDUCTION(102SM) SAVING THE EVENT NOTICE

2 ** REPRESENTS SECOND CRACK INITIATION

3 **

4 **

5 **

6 **

7 **

8 **

9 **

10 **

11 **

12 **

13 **

14 **

15 **

16 **

17 **

18 **

19 **

20 **

21 **

22 **

23 **

24 **

25 **

26 **

27 **

28 **

29 **

30 **

31 **

32 **

33 **

34 **

35 **

36 **

37 **

38 **

39 **

40 **

41 **

42 **

43 **

44 **

45 **

46 **

47 **

48 **

49 **

50 **

51 **

09/01/76 PAGE 34

LINE CACT SIMSCRIPT II.5 RELEASE AF

```

52 LET CL = CCL + ((CL-CCL)/MSR(ID))*MSR(ID)
53 JUMP AHEAD
54 OTHERWISE
55 JUMP AHEAD
56 OTHERWISE
57 LET DL = (TAC-TAI) * MSR(ID)
58 IF CCL LT DL
59 LET CL = CCL + ((DL-CCL)/MSR(ID))*MSR(ID) + (TIME,V-TAC)*MSR(ID)*CGRI(ID)
60 JUMP AHEAD
61 OTHERWISE
62 LET CL = DL + (TIME,V-TAC)*MSR(ID)*CGRI(ID)
63 IF CCL LT CL
64 LET CL = CCL + ((CL-CCL)/MSR(ID))*MSR(ID)
65 REGARDLESS
66 JUMP AHEAD
67 OTHERWISE
68 LET CL = (TIME,V-TAI) * MSR(ID)
69 IF CL GT CCL
70 LET CL = CCL + ((CL-CCL)/MSR(ID))*MSR(ID)
71 REGARDLESS
72 HERE
73 LET SU = 2.75 - (1.75 * CL) / FSAF.LGT
74 IF CL GT FSAF.LGT
75 LET SU = 1.0 - (CL-FSAF.LGT)/(LGHT.TN-FAILURE-FSAF.LGT)
76 REGARDLESS
77 ** PREDICT TIME TO FAILURE FROM SECOND CHECK INITIATION
78 **
79 **
80 LET T1 = 0.0
81 IF CL LT CCL
82 LET T1 = (CCL-CCL) / (MSR(ID) * CGRI(ID))
83 LET GR1 = 2.0 * MSR(ID) * CGRI(ID)
84 IF TAI LE TIME,V + TCL
85 LET GR2 = (MSR(ID) * MSR(ID)) * CGRI(ID)
86 JUMP AHEAD
87 OTHERWISE
88 LET X(1) = 0.0
89 LET Y(1) = 0.0
90 LET X(2) = TCL-T1
91 LET Y(2) = CCL + X(2) * MSR(ID) * CGRI(ID) - T1 * MSR(ID) * CGRI(ID)
92 LET X(3) = TAI - TIME,V - T1
93 LET Y(3) = Y(2) + (2.0 * MSR(ID) * CGRI(ID))
94 LET N = 3
95 JUMP AHEAD
96 REGARDLESS
97 IF TAI LE TIME,V + TCL
98 LET GR1 = (MSR(ID) * MSR(ID)) * CGRI(ID)
99 LET GR2 = GR1
100 JUMP AHEAD
101 OTHERWISE
102 LET X(1) = 0.0
103 LET Y(1) = 0.0
104 LET X(2) = TCL

```

09/01/76 PAGE 35

LINE CACI SIMS-INT 11.5 RELEASE NF

```

105 LET V(1) = CCL + TCLAMER(ID)*CGM1(ID)
106 LET V(2) = TAD + TIME.V
107 LET V(3) = V(2) + (V(3)-TCL) + 2.0 * WFR(ID) + CGM1(ID)
108 LET W = 5
109 WEND
110 IF W GE 5
111 LET T(1) = 100.0
112 LET T = 2 * T
113 W = 1
114 LET W(1) = 1.0
115 LET W(2) = 1.0
116 LET W(3) = 1.0
117 W = 1
118 LET W = W + W(1) + W(2)
119 LET W = W + W(3) + W(1) + V(1)
120 LET W = W + W(1) + V(1)
121 LET W = W + W(1) + W(1) + W(1) + 2
122 LET W = W + W(1) + W(1)
123 LET W = W + W(1)
124 LET W(2) = (SWARMY - W(1)) / (SWARMY - W(2))
125 WEND
126 LET T2 = 11 + (FSAF.LGT - (SP10T1))/GR2
127 IF CL LF CCL
128 LET W1 = 622
129 LET T2 = (FSAF.LGT-CL)/GR2
130 WEND
131 LET SF = 1.0
132 LET W1 = ((W1-SF)*GR1) / FSAF.LGT
133 LET W2 = ((W1-SF)*GR2) / FSAF.LGT
134 LET W3 = GR2 / (LG-T10.FAULTS - FSAF.LGT)
135 IF CL LF FSAF.LGT
136 LET W1 = W1 / (LG-T10.FAULTS - FSAF.LGT)
137 LET W2 = W2 / (LG-T10.FAULTS - FSAF.LGT)
138 LET W3 = W3
139 LET T2 = 1.0
140 WEND
141 IF (TIME.V + 12 LT TAD OR CLEAFSES(ID) = "YES") AND T2 GT 0.0
142 IF ALL(ID) = "YES"
143 LET TAD-FAIL-SAF.LGT = AMESL(ID)
144 CANCEL THE PERFORM-FAIL-SAF.LGT
145 RESCHEDULE THE TAD-FAIL-SAF.LGT(ID) AT TIME.V + T2
146 JUMP 100
147 ELSE
148 RESCHEDULE A TAD-FAIL-SAF.LGT(ID) AT TIME.V + T2
149 LET AMESL(ID) = TAD-FAIL-SAF.LGT
150 LET ALL(ID) = "YES"
151 WEND
152 WEND
153 LET LG = CCL*E.F(CM-ID)
154 LET W1 = 1 + FID.FISJ + W
155 LET W2 = W + W1
156 LET TFC = -LG*E.F((C2/W1)*LG + 1.0)/W2
157 IF TFC LT 1

```

09/01/76 PAGE 36

LINE CACT SIMSCRIPT II.5 RELEASE RF.

```

158 JUMP AHEAD
159 ELSE
160 LET S1 = SU - R1 * T1
161 LET KQ = R * R2
162 LET LKMS = LOG.E.F(A) + (H * S1 + R * R2 * T1)
163 LET KA = (A*EXP.F(M*SI))/KQ
164 LET KQ = (K1/K2)*(EXP.F(-M*W1*T1)-1.0)
165 LET TTF = (LKMS - LOG.E.F(M*SI))/KQ
166 IF TTF LE T2
167 JUMP AHEAD
168 ELSE
169 LET K10 = M*Q5
170 LET AMG = M*SF+M*Q1+T2
171 IF AMG LT -175.0
172 LET AMG = -175.0
173 ALWAYS
174 LET K11 = K10/(A*EXP.F(AMG))
175 LET K12 = A*EXP.F(M*SI)/K10
176 LET K13 = EXP.F(M*Q2+T1 - M*Q2+T2) - 1.0
177 LET TTF = -LOG.F.F(K11*(LG*K12-(K*Q13)-Q9))/K10
178 MEME
179 IF LIST = 1.0
180 PRINT 1 LINE WITH TIME,V,TTF-FATRY.TIME(AIRPLANE(ID)) AS FOLLOWS
      ELEMENT FAILURE PROJECTED AT ***** FLIGHT HOURS
      ALWAYS
181 IF TIME.V + TTF LT IAR OR CO.FXISTS(ID) = "NS"
182 IF FSM(ID) = "YES"
183 LET FAILURE = AF(ID)
184 CANCEL THE FAILURE
185 RESCHEDULE THE FAILURE(ID) AT TIME.V + TTF
186 JUMP AHEAD
187 ELSE
188 SCHEDULE A FAILURE(ID) AT TIME.V + TTF
189 LET AF(ID) = FAILURE
190 LET FSM(ID) = "YES"
191 MEME
192 ALWAYS
193 RELEASE X(*), Y(*), A(*)
194 RETURN
195 END

```

LOCAL VARIABLES OF THIS ROUTINE

ARG	DOUBLE	WORD	RQ	CCL	DOUBLE	WORD	27
CL	DOUBLE	WORD	35	PL	DOUBLE	WORD	37
CS1	DOUBLE	WORD	45	Q12	DOUBLE	WORD	45
I-1	INTEGER	WORD	15	I-2	INTEGER	WORD	16
I-3	INTEGER	WORD	17	ID2SR	INTEGER	WORD	1
J-1	INTEGER	WORD	14	K-1	INTEGER	WORD	26
K-2	INTEGER	WORD	21	K-3	INTEGER	WORD	22
K-4	INTEGER	WORD	23	K1	DOUBLE	WORD	71
K10	DOUBLE	WORD	87	K11	DOUBLE	WORD	91

09/01/76 PAGE 37

LINE	CACI	SIMSCHLOT	II.5	RELEASE	AF
K12	DDUMLE	WORD 93	K12		
K2	DDUMLE	WORD 73	K4		
K4	DDUMLE	WORD 43	K9		
L.4	INTEGEN	WORD 11	L6		
L.4	DDUMLE	WORD 41	LIST		
L.4	DDUMLE	WORD 47	K.1		
K.1	DDUMLE	WORD 15	K1		
K2	DDUMLE	WORD 44	K1		
SF	DDUMLE	WORD 61	S2		
S1	DDUMLE	WORD 30	S1		
T	DDUMLE	WORD 24	TAC		
TAM	DDUMLE	WORD 7	TAT		
TCL	DDUMLE	WORD 51	TIF		
T1	DDUMLE	WORD 41	T2		
K.1	WEAL	WORD 5	K1		
K15	DDUMLE	WORD 55	K15		
K1	DDUMLE	WORD 53	K1		
K1	WEAL	WORD 4	K1		

DDUMLE	WORD 95
DDUMLE	WORD 79
DDUMLE	WORD 85
DDUMLE	WORD 69
DDUMLE	WORD 9
INTEGEN	WORD 19
DDUMLE	WORD 63
DDUMLE	WORD 67
DDUMLE	WORD 57
DDUMLE	WORD 77
DDUMLE	WORD 55
DDUMLE	WORD 29
DDUMLE	WORD 75
DDUMLE	WORD 59
DDUMLE	WORD 89
DDUMLE	WORD 51
WEAL	WORD 3

09/01/76 PAGE 34

LINE CACT SIMSCRIPT II.5 RELEASE 8F

1 EVENT 3-STRENGTH-REDUCTION(IDSR) SAVING THE EVENT NOTICE

2 ** REPRESENTS THIRD CRACK INITIATION

3 **

4 DEFINE IDSR AS AN INTEGER VARIABLE

5 DEFINE K, Y, X AS REAL, 1-DIMENSIONAL ARRAYS

6 RESERVE X(5), Y(5), W(5) AS 5

7 LET ID = IDSR

8 LET TAP = ENTRY.TIME(AIRPLANE(ID)) + USAGE.LIFE

9 IF LTIM = "YES"

10 FOR I = 1 TO WAC(LD*)

11 DO

12 IF ID = TLID(LD*,I)

13 LET LIST = 1.0

14 SKIP 1 OUTPUT LINE

15 PRINT 1 LINE WITH ID, TIME, V-ENTRY.TIME(AIRPLANE(ID)) AS FOLLOWS

16 A/C %). *** EXPERIENCES 3RD CRACK INITIATION AT ***** HOURS

17 LEAVE

18 ELSE

19 LOOP

20 ALWAYS

21 LET 3.INT(ID) = "NO"

22 IF RANDM.F(10) LE 1.0WU-

23 LET 3.INT(ID) = "YES"

24 IF LIST = 1.0

25 PRINT 1 LINE AS FOLLOWS

26 CRACK INITIATES INTERNALLY

27 ALWAYS

28 LET T = (CRIT.CRK.LGT*PCLL)/(USR(ID)*CGW(ID))

29 IF PCLL GT 1.0

30 LET T = CRIT.CRK.LGT/(WSW(ID)*CGW(ID)) + (CRIT.CRK.LGT*(PCLL-1.0))/

31 (WCR(ID)*CGW(ID))

32 ALWAYS

33 IF TIME.V + T LT TAW ON CO-EXISTS(ID) = "NS"

34 SCHEDULE A 3.ITF(ID) AT TIME.V + T

35 LET ASE(ID) = 3.ITF

36 LET TES(ID) = "YES"

37 ALWAYS

38 IF RANDM.F(5) = "YES"

39 LET 1-STRENGTH-REDUCTION = AISR(ID)

40 LET 2-STRENGTH-REDUCTION = ARSR(ID)

41 LET CEL = CRIT.CRK.LGT

42 LET TCL = CEL/WSW(ID)*CGW(ID)

43 LET TAI = TIME.A(1-STRENGTH-REDUCTION)

44 LET TAP = TIME.A(2-STRENGTH-REDUCTION)

45 ** INTERVIEW PERSONAL STRENGTH REDUCTION ** (CAUSE OF FIRST AND SECOND CRACKS)

46 **

47 **

48 IF CO-EXISTS(ID) = "YES"

49 LET CORRUSION = AC(ID)

50 LET FAC = TIME.A(CORRUSION)

51 IF TAC LT TAI

09/01/76 PAGE 39

```

LINE CACI SIMSCRIPT 11.5 RELEASE MF
52 LET ICL = (TIME.V-TA1) * MSW(ID) * CGM(ID)
53 LET PCL = (TIME.V-TA2) * MSW(ID) * CGM(ID)
54 IF ICL GT CCL
55 LET ICL = CCL + ((ICL-CCL)/MSW(ID))*MSW(ID)
56 *FLAGDLESS
57 IF PCL GT CCL
58 LET PCL = CCL + ((PCL-CCL)/MSW(ID))*MSW(ID)
59 *FLAGDLESS
60 LET CL = ICL + PCL
61 JUMP AMEAD
62 *TIMEBASE
63 IF TAC LT TA2
64 LET NL = (TAC-TA1) * ASW(ID)
65 LET ICL = CL + (TIME.V-TAC)*MSW(ID)*CGM(ID)
66 LET PCL = (TIME.V-TA2) * ASW(ID) * CGM(ID)
67 IF CCL LT NL
68 LET PCL = (TIME.V-TA2) * ASW(ID) * CGM(ID)
69 *FLAGDLESS
70 IF PCL GT CCL
71 LET PCL = CCL + ((PCL-CCL)/MSW(ID))*MSW(ID)
72 *FLAGDLESS
73 *FLAGDLESS
74 IF TAC LT TA1
75 LET ICL = (TIME.V-TA1) * MSW(ID) * CGM(ID)
76 IF ICL GT CCL
77 LET ICL = CCL + ((ICL-CCL)/MSW(ID))*MSW(ID)
78 *FLAGDLESS
79 *FLAGDLESS
80 *TIMEBASE
81 IF TAC LT TA1
82 LET ICL = (TIME.V-TA1) * MSW(ID)
83 IF ICL GT CCL
84 LET ICL = CCL + ((ICL-CCL)/MSW(ID))*MSW(ID)
85 *FLAGDLESS
86 LET PCL = (TIME.V-TA2) * MSW(ID)
87 IF PCL GT CCL
88 LET PCL = CCL + ((PCL-CCL)/MSW(ID))*MSW(ID)
89 *FLAGDLESS
90 *FLAGDLESS
91 LET CL = ICL + PCL
92 LET SW = 2.75 * (CL * CL) / FSAF.LGT
93 IF CL GT FSAF.LGT
94 LET SW = 1.75 * (CL-FSAF.LGT)/(LGT-FSAF.LGT)
95 *FLAGDLESS
96
97 ** PREDICT TIME TO FAILURE FROM INTER CHACK INITIATION.
98
99 LET T1 = 1.0
100 IF ICL LT CCL
101 LET T1 = (CCL-ICL) / (MSW(ID)*CGM(ID))
102 LET T2 = 1.0 * MSW(ID) * CGM(ID)
103 LET T2 = (CCL-PCL) / (MSW(ID)*CGM(ID))
104 IF T1 LT TIME.V + T2

```

09/31/75 PAGE 08

```

LINE  CACT ST-SC-1PT II.5  RELEASE AF
105      LET GR2 = (WER(10) + 2.0*MSW(10)) * CGW(10)
106      JUMP AMEAD
107      OTHERWISE
108          LET X(1) = 0.0
109          LET Y(1) = 0.0
110          LET X(2) = T2-T1
111          LET Y(2) = X(2) * (WER(10) + 2.0*MSW(10)) * CGW(10)
112          IF TAR LE TIME.V + TCL
113              LET X(3) = TAR - TIME.V - T1
114              LET Y(3) = Y(2) + (X(3)-X(2)) * (2.0*WER(10) + MSW(10)) * CGW(10)
115              LET N = 3
116              JUMP AMEAD
117              OTHERWISE
118              LET X(3) = TCL - T1
119              LET N = X(3) - T2 + T1
120              LET Y(3) = (X(3)*X(3))*WER(10)*CGW(10) + (T2-T1+X(3))*MSW(10)*CGW(10)
121              LET X(4) = TAR - TIME.V - T1
122              LET Y(4) = (X(4)-X(3)) * 3.0 * WER(10) * CGW(10)
123              LET N = 4
124              JUMP AMEAD
125              OTHERWISE
126              LET X(1) = 0.0
127              LET Y(1) = 0.0
128              IF TCL LT TCL
129                  LET T2 = (CCL-2CL)/(MSR(10)*CGW(10))
130                  IF TAR LE TIME.V + T2
131                      LET GR2 = (2.0 * MSR(10) + WER(10)) * CGW(10)
132                      LET GR1 = GR2
133                      JUMP AMEAD
134                      OTHERWISE
135                      IF TAR LE TIME.V + TCL
136                          LET X(2) = T2
137                          LET Y(2) = T2 * (2.0 * MSW(10) + WER(10)) * CGW(10)
138                          LET X(3) = TAR - TIME.V
139                          LET Y(3) = Y(2) + (X(3)-X(2)) * (2.0 * WER(10) + MSW(10)) * CGW(10)
140                          LET N = 3
141                          JUMP AMEAD
142                          OTHERWISE
143                          LET X(2) = T2
144                          LET Y(2) = T2 * (2.0 * MSW(10) + WER(10)) * CGW(10)
145                          LET X(3) = TCL
146                          LET Y(3) = Y(2) + (TCL-T2) * (2.0 * WER(10) + MSW(10)) * CGW(10)
147                          LET X(4) = TAR - TIME.V
148                          LET Y(4) = Y(3) + (X(4)-TCL) * (3.0 * WER(10) + CGW(10))
149                          LET N = 4
150                          JUMP AMEAD
151                          OTHERWISE
152                          IF TAR LE TIME.V + TCL
153                              LET GR2 = (2.0 * WER(10) + MSW(10)) * CGW(10)
154                              LET GR1 = GR2
155                              JUMP AMEAD
156                              OTHERWISE
157                              LET X(2) = TCL

```

09/01/76 PAGE 41

LINE CACI SIMSCRIPT II.5 RELEASE RF

```

154 LET V(2) = TOL * (2.0 * WFR(ID) + MSR(ID)) * CGMI(ID)
155 LET X(3) = TAR - TIME.V
156 LET V(3) = V2 + (X(3)-X(2)) * (3.0 * WFR(ID)) * CGMI(ID)
157 LET V1 = V
158 NAME
159 IF N.GE 3
160 LET A(1) = 100.0
161 FOR I = 1 TO N
162 DO
163 LET A(I) = 1.0
164 CONTINUE
165 FOR I = 1 TO N
166 DO
167 LET A(I) = 1.0
168 CONTINUE
169 FOR I = 1 TO N
170 DO
171 LET A(I) = A(I) * A(I)
172 LET V1 = V1 + A(I) * V(I)
173 LET V1 = V1 + A(I) * V(I)
174 LET X3 = WFR * A(I) * X(1) * A(I)
175 LET SPN = SPN + A(I)
176 CONTINUE
177 LET GR2 = (SPN * WFR - WFR * X) / (SPN * WFR - WFR * 2)
178 REGARDLESS
179 LET T2 = T1 + (FSAF.LGT - CL) / GR2
180 IF TOL.GE CCL
181 LET GR1 = GR2
182 LET T2 = (FSAF.LGT - CL) / GR2
183 REGARDLESS
184 LET SF = 1.0
185 LET M1 = ((N1-SF)*GR1) / FSAF.LGT
186 LET M2 = ((N1-SF)*GR2) / FSAF.LGT
187 LET R1 = GR2 / (LGHT.TC.FAILURE - FSAF.LGT)
188 LET CL GE FSAF.LGT
189 LET M1 = GR1 / (LGHT.TC.FAILURE - FSAF.LGT)
190 LET R2 = GR2 / (LGHT.TC.FAILURE - FSAF.LGT)
191 LET R3 = R2
192 LET T2 = 0.0
193 REGARDLESS
194 IF (TIME.V * T2 LT TAW OR CUL EXISTS(ID) = "NS") AND T2 GT 0.0
195 IF AIL(ID) = "YES"
196 LET REACH.FAIL.SAFE.LGT = AMSL(ID)
197 CANCEL THE REACH.FAIL.SAFE.LGT
198 RESCHEDULE THE REACH.FAIL.SAFE.LGT(IP) AT TIME.V + T2
199 JUMP AHEAD
200 ELSE
201 SCHEDULE A REACH.FAIL.SAFE.LGT(ID) AT TIME.V + T2
202 LET AMSL(ID) = REACH.FAIL.SAFE.LGT
203 LET AIL(ID) = "YES"
204 NAME
205 REGARDLESS
206 LET M1 = A * EXP.F(SU * M)
207 LET M2 = M * R1
208 LET LG = LG.E.F(XNM(ID))
209 LET TTF = -LG.E.F((K2/K1)*LG + 1.0)/M2
210 IF TTF LT T1

```

09/01/76 PAGE 42

LINE CACI SCRIPT II.5 PLEASE MF

```

211 JUMP AHEAD
212 ELSE
213 LET S1 = SU - R1 + T1
214 LET K4 = R + K2
215 LET LK4 = LOG.E.F(A) + (K + S) + K + K2 + T1
216 LET K4 = (A*EXP.F(M*SI))/K4
217 LET K9 = (K1/K2)*(EXP.F(M*SI+1))-1.0
218 LET ITF = (LK4 - LOG.F.F(A*MS.F(K4)) - LOG.E.F(A*MS.F(K4)))/K4
219 IF ITF LT T2
220 JUMP AHEAD
221 ELSE
222 LET K10 = M*J5
223 LET ARG = M*SE+M*J3+T2
224 IF ARG LT -175.0
225 LET ARG = -175.0
226 ALWAYS
227 LET K11 = K10/(A*EXP.F(ARG))
228 LET K12 = A*EXP.F(M*SI)/K10
229 LET K13 = EXP.F(M*J2+T1) - M*J2+T2 - 1.0
230 LET ITF = -LOG.E.F(K11*(LOG.F(K12*(M*J3+K9)))/K10)
231 MEME
232 IF LIST = 1.0
233 PRINT 1 LINE WITH TIME,V,TIME,F,TIME(ATWPLANE(ID)) AS FOLLOWS
      FLIGHT FAILURE PROJECTED AT ***** FLIGHT HOURS
234 ALWAYS
235 IF TIME,V + ITF LT T4 (M COEFFICIENTS) = "NS"
236 IF FSH(ID) = "YES"
237 LET FAILURE = AF(ID)
238 CANCEL THE FAILURE
239 RESCHEDULE THE FAILURE(ID) AT TIME,V + ITF
240 JUMP AHEAD
241 ELSE
242 RESCHEDULE A FAILURE(ID) AT TIME,V + ITF
243 LET AF(ID) = FAILURE
244 LET FSH(ID) = "YES"
245 MEME
246 ALWAYS
247 RELEASE X(*), Y(*), Z(*)
248 RETURN
249 END

```

LOCAL VARIABLES OF THIS MODULE

ARG	DOUBLE	WORD 99	CCL	DOUBLE	WORD 27
CL	DOUBLE	WORD 41	DL	DOUBLE	WORD 45
GM1	DOUBLE	WORD 30	GM2	DOUBLE	WORD 53
I.1	INTEGER	WORD 15	I.2	INTEGER	WORD 16
I.3	INTEGER	WORD 17	I.4	INTEGER	WORD 1
J.1	INTEGER	WORD 14	K.1	INTEGER	WORD 26
K.2	INTEGER	WORD 21	K.3	INTEGER	WORD 22
K.4	INTEGER	WORD 23	K1	DOUBLE	WORD 79
K10	DOUBLE	WORD 97	K11	DOUBLE	WORD 101

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDQ

09/01/74 PAGE 43

LINE	CACI	SIMSCRIPT	II.5	RELEASE	AF
K12	DOUBLE	WORD103		K13	
K2	DOUBLE	WORD A1		K4	
KA	DOUBLE	WORD 93		K9	
L.4	INTEGER	WORD 11		LG	
LKNS	DOUBLE	WORD 91		LIST	
N	DOUBLE	WORD 55		R.1	
R.1	DOUBLE	WORD 13		R1	
R2	DOUBLE	WORD 75		R5	
SF	DOUBLE	WORD 71		SPN	
SU	DOUBLE	WORD 45		SI	
T	DOUBLE	WORD 25		TAC	
TAR	DOUBLE	WORD 7		TAL	
TA2	DOUBLE	WORD 33		TCL	
TTF	DOUBLE	WORD A5		T1	
T2	DOUBLE	WORD 51		N	
AA	DOUBLE	WORD 61		AS	
AY	DOUBLE	WORD 63		AT	
A	REAL	WORD 3		KK	
V	REAL	WORD 4		V2	
ICL	DOUBLE	WORD 37		ZCL	

DOUBLE	WORD 105
DOUBLE	WORD 89
DOUBLE	WORD 95
DOUBLE	WORD A3
DOUBLE	WORD 9
DOUBLE	WORD 19
INTEGER	WORD 73
DOUBLE	WORD 77
DOUBLE	WORD 69
DOUBLE	WORD 87
DOUBLE	WORD 35
DOUBLE	WORD 31
DOUBLE	WORD 29
DOUBLE	WORD 87
DOUBLE	WORD 5
REAL	WORD 67
DOUBLE	WORD 85
DOUBLE	WORD 57
DOUBLE	WORD 59
DOUBLE	WORD 39

00/01/70 PAGE 42

```

1 LINE CACI NEWSMPT II.5 RELEASED
2
3   EVENT LATEC(DIE)
4     DEFINE INTR AS AN INTEGER VARIABLE
5     LET ID = IDIE
6     LET LATEC(DIE) = "NO"
7     IF (TIME = "YES")
8       FOR I = 1 TO WAC(LATE)
9         IF I = 1 THEN WAC(LATE)
10        WAC(LATE) = WAC(LATE) + 1
11        WAC(LATE) = WAC(LATE) + 1
12        WAC(LATE) = WAC(LATE) + 1
13        WAC(LATE) = WAC(LATE) + 1
14        WAC(LATE) = WAC(LATE) + 1
15        WAC(LATE) = WAC(LATE) + 1
16        WAC(LATE) = WAC(LATE) + 1
17        WAC(LATE) = WAC(LATE) + 1
18        WAC(LATE) = WAC(LATE) + 1
19        WAC(LATE) = WAC(LATE) + 1
20        WAC(LATE) = WAC(LATE) + 1
21        WAC(LATE) = WAC(LATE) + 1
22        WAC(LATE) = WAC(LATE) + 1
23        WAC(LATE) = WAC(LATE) + 1
24        WAC(LATE) = WAC(LATE) + 1
25        WAC(LATE) = WAC(LATE) + 1
26        WAC(LATE) = WAC(LATE) + 1
27        WAC(LATE) = WAC(LATE) + 1
28        WAC(LATE) = WAC(LATE) + 1
29        WAC(LATE) = WAC(LATE) + 1
30        WAC(LATE) = WAC(LATE) + 1
31        WAC(LATE) = WAC(LATE) + 1
32        WAC(LATE) = WAC(LATE) + 1
33        WAC(LATE) = WAC(LATE) + 1
34        WAC(LATE) = WAC(LATE) + 1
35        WAC(LATE) = WAC(LATE) + 1
36        WAC(LATE) = WAC(LATE) + 1
37        WAC(LATE) = WAC(LATE) + 1
38        WAC(LATE) = WAC(LATE) + 1
39        WAC(LATE) = WAC(LATE) + 1
40        WAC(LATE) = WAC(LATE) + 1
41        WAC(LATE) = WAC(LATE) + 1
42        WAC(LATE) = WAC(LATE) + 1
43        WAC(LATE) = WAC(LATE) + 1
44        WAC(LATE) = WAC(LATE) + 1
45        WAC(LATE) = WAC(LATE) + 1
46        WAC(LATE) = WAC(LATE) + 1
47        WAC(LATE) = WAC(LATE) + 1
48        WAC(LATE) = WAC(LATE) + 1
49        WAC(LATE) = WAC(LATE) + 1
50        WAC(LATE) = WAC(LATE) + 1
51        WAC(LATE) = WAC(LATE) + 1
52        WAC(LATE) = WAC(LATE) + 1
53        WAC(LATE) = WAC(LATE) + 1
54        WAC(LATE) = WAC(LATE) + 1
55        WAC(LATE) = WAC(LATE) + 1
56        WAC(LATE) = WAC(LATE) + 1
57        WAC(LATE) = WAC(LATE) + 1
58        WAC(LATE) = WAC(LATE) + 1
59        WAC(LATE) = WAC(LATE) + 1
60        WAC(LATE) = WAC(LATE) + 1
61        WAC(LATE) = WAC(LATE) + 1
62        WAC(LATE) = WAC(LATE) + 1
63        WAC(LATE) = WAC(LATE) + 1
64        WAC(LATE) = WAC(LATE) + 1
65        WAC(LATE) = WAC(LATE) + 1
66        WAC(LATE) = WAC(LATE) + 1
67        WAC(LATE) = WAC(LATE) + 1
68        WAC(LATE) = WAC(LATE) + 1
69        WAC(LATE) = WAC(LATE) + 1
70        WAC(LATE) = WAC(LATE) + 1
71        WAC(LATE) = WAC(LATE) + 1
72        WAC(LATE) = WAC(LATE) + 1
73        WAC(LATE) = WAC(LATE) + 1
74        WAC(LATE) = WAC(LATE) + 1
75        WAC(LATE) = WAC(LATE) + 1
76        WAC(LATE) = WAC(LATE) + 1
77        WAC(LATE) = WAC(LATE) + 1
78        WAC(LATE) = WAC(LATE) + 1
79        WAC(LATE) = WAC(LATE) + 1
80        WAC(LATE) = WAC(LATE) + 1
81        WAC(LATE) = WAC(LATE) + 1
82        WAC(LATE) = WAC(LATE) + 1
83        WAC(LATE) = WAC(LATE) + 1
84        WAC(LATE) = WAC(LATE) + 1
85        WAC(LATE) = WAC(LATE) + 1
86        WAC(LATE) = WAC(LATE) + 1
87        WAC(LATE) = WAC(LATE) + 1
88        WAC(LATE) = WAC(LATE) + 1
89        WAC(LATE) = WAC(LATE) + 1
90        WAC(LATE) = WAC(LATE) + 1
91        WAC(LATE) = WAC(LATE) + 1
92        WAC(LATE) = WAC(LATE) + 1
93        WAC(LATE) = WAC(LATE) + 1
94        WAC(LATE) = WAC(LATE) + 1
95        WAC(LATE) = WAC(LATE) + 1
96        WAC(LATE) = WAC(LATE) + 1
97        WAC(LATE) = WAC(LATE) + 1
98        WAC(LATE) = WAC(LATE) + 1
99        WAC(LATE) = WAC(LATE) + 1
100       WAC(LATE) = WAC(LATE) + 1

```

LOCAL VARIABLES OF THIS SECTION

INTEGEN	WORD 7	1,2	INTEGEN	WORD 4
INTEGEN	WORD 8	1,2	INTEGEN	WORD 5
INTEGEN	WORD 9	1,2	INTEGEN	WORD 6
INTEGEN	WORD 10	1,2	INTEGEN	WORD 7
INTEGEN	WORD 11	1,2	INTEGEN	WORD 8
INTEGEN	WORD 12	1,2	INTEGEN	WORD 9
INTEGEN	WORD 13	1,2	INTEGEN	WORD 10
INTEGEN	WORD 14	1,2	INTEGEN	WORD 11
INTEGEN	WORD 15	1,2	INTEGEN	WORD 12
INTEGEN	WORD 16	1,2	INTEGEN	WORD 13
INTEGEN	WORD 17	1,2	INTEGEN	WORD 14
INTEGEN	WORD 18	1,2	INTEGEN	WORD 15
INTEGEN	WORD 19	1,2	INTEGEN	WORD 16
INTEGEN	WORD 20	1,2	INTEGEN	WORD 17
INTEGEN	WORD 21	1,2	INTEGEN	WORD 18
INTEGEN	WORD 22	1,2	INTEGEN	WORD 19
INTEGEN	WORD 23	1,2	INTEGEN	WORD 20
INTEGEN	WORD 24	1,2	INTEGEN	WORD 21
INTEGEN	WORD 25	1,2	INTEGEN	WORD 22
INTEGEN	WORD 26	1,2	INTEGEN	WORD 23
INTEGEN	WORD 27	1,2	INTEGEN	WORD 24
INTEGEN	WORD 28	1,2	INTEGEN	WORD 25
INTEGEN	WORD 29	1,2	INTEGEN	WORD 26
INTEGEN	WORD 30	1,2	INTEGEN	WORD 27
INTEGEN	WORD 31	1,2	INTEGEN	WORD 28
INTEGEN	WORD 32	1,2	INTEGEN	WORD 29
INTEGEN	WORD 33	1,2	INTEGEN	WORD 30
INTEGEN	WORD 34	1,2	INTEGEN	WORD 31
INTEGEN	WORD 35	1,2	INTEGEN	WORD 32
INTEGEN	WORD 36	1,2	INTEGEN	WORD 33
INTEGEN	WORD 37	1,2	INTEGEN	WORD 34
INTEGEN	WORD 38	1,2	INTEGEN	WORD 35
INTEGEN	WORD 39	1,2	INTEGEN	WORD 36
INTEGEN	WORD 40	1,2	INTEGEN	WORD 37
INTEGEN	WORD 41	1,2	INTEGEN	WORD 38
INTEGEN	WORD 42	1,2	INTEGEN	WORD 39
INTEGEN	WORD 43	1,2	INTEGEN	WORD 40
INTEGEN	WORD 44	1,2	INTEGEN	WORD 41
INTEGEN	WORD 45	1,2	INTEGEN	WORD 42
INTEGEN	WORD 46	1,2	INTEGEN	WORD 43
INTEGEN	WORD 47	1,2	INTEGEN	WORD 44
INTEGEN	WORD 48	1,2	INTEGEN	WORD 45
INTEGEN	WORD 49	1,2	INTEGEN	WORD 46
INTEGEN	WORD 50	1,2	INTEGEN	WORD 47
INTEGEN	WORD 51	1,2	INTEGEN	WORD 48
INTEGEN	WORD 52	1,2	INTEGEN	WORD 49
INTEGEN	WORD 53	1,2	INTEGEN	WORD 50
INTEGEN	WORD 54	1,2	INTEGEN	WORD 51
INTEGEN	WORD 55	1,2	INTEGEN	WORD 52
INTEGEN	WORD 56	1,2	INTEGEN	WORD 53
INTEGEN	WORD 57	1,2	INTEGEN	WORD 54
INTEGEN	WORD 58	1,2	INTEGEN	WORD 55
INTEGEN	WORD 59	1,2	INTEGEN	WORD 56
INTEGEN	WORD 60	1,2	INTEGEN	WORD 57
INTEGEN	WORD 61	1,2	INTEGEN	WORD 58
INTEGEN	WORD 62	1,2	INTEGEN	WORD 59
INTEGEN	WORD 63	1,2	INTEGEN	WORD 60
INTEGEN	WORD 64	1,2	INTEGEN	WORD 61
INTEGEN	WORD 65	1,2	INTEGEN	WORD 62
INTEGEN	WORD 66	1,2	INTEGEN	WORD 63
INTEGEN	WORD 67	1,2	INTEGEN	WORD 64
INTEGEN	WORD 68	1,2	INTEGEN	WORD 65
INTEGEN	WORD 69	1,2	INTEGEN	WORD 66
INTEGEN	WORD 70	1,2	INTEGEN	WORD 67
INTEGEN	WORD 71	1,2	INTEGEN	WORD 68
INTEGEN	WORD 72	1,2	INTEGEN	WORD 69
INTEGEN	WORD 73	1,2	INTEGEN	WORD 70
INTEGEN	WORD 74	1,2	INTEGEN	WORD 71
INTEGEN	WORD 75	1,2	INTEGEN	WORD 72
INTEGEN	WORD 76	1,2	INTEGEN	WORD 73
INTEGEN	WORD 77	1,2	INTEGEN	WORD 74
INTEGEN	WORD 78	1,2	INTEGEN	WORD 75
INTEGEN	WORD 79	1,2	INTEGEN	WORD 76
INTEGEN	WORD 80	1,2	INTEGEN	WORD 77
INTEGEN	WORD 81	1,2	INTEGEN	WORD 78
INTEGEN	WORD 82	1,2	INTEGEN	WORD 79
INTEGEN	WORD 83	1,2	INTEGEN	WORD 80
INTEGEN	WORD 84	1,2	INTEGEN	WORD 81
INTEGEN	WORD 85	1,2	INTEGEN	WORD 82
INTEGEN	WORD 86	1,2	INTEGEN	WORD 83
INTEGEN	WORD 87	1,2	INTEGEN	WORD 84
INTEGEN	WORD 88	1,2	INTEGEN	WORD 85
INTEGEN	WORD 89	1,2	INTEGEN	WORD 86
INTEGEN	WORD 90	1,2	INTEGEN	WORD 87
INTEGEN	WORD 91	1,2	INTEGEN	WORD 88
INTEGEN	WORD 92	1,2	INTEGEN	WORD 89
INTEGEN	WORD 93	1,2	INTEGEN	WORD 90
INTEGEN	WORD 94	1,2	INTEGEN	WORD 91
INTEGEN	WORD 95	1,2	INTEGEN	WORD 92
INTEGEN	WORD 96	1,2	INTEGEN	WORD 93
INTEGEN	WORD 97	1,2	INTEGEN	WORD 94
INTEGEN	WORD 98	1,2	INTEGEN	WORD 95
INTEGEN	WORD 99	1,2	INTEGEN	WORD 96
INTEGEN	WORD 100	1,2	INTEGEN	WORD 97

```

1  EVENT 2,DIFF(1024)
2  SETFIVE LOG2 AS AN I-10000
3
4  LET I0 = 10000
5
6  LET P1-1(10) = "Q1"
7
8  LET I-1(10) = "Q1"
9
10 LET I-1 = "YES"
11
12 FOR I = 1 TO 10000(1000)
13   DO
14     IF I0 = 10000(1000)
15       SETP 1 0000000000
16       PRINT 1 0000000000 LINE 110
17       SETFIVE(10000(1000))
18       Z/C 0000000000 WAS 1000000000
19     LEAVE
20   ELSE
21     LEAVE
22   ALWAYS
23   END
24

```

29. Title: S47MVA 7531:7

1.1	1.1534	4.70	7	1.2	1.1514	4.70	5
1.3	1.1564	4.70	9	1.2	1.1544	4.70	12
1.1	1.1564	4.70	10	1.1	1.1574	4.70	13
1.2	1.1534	4.70	13	1.2	1.1564	4.70	14
1.1	1.1564	4.70	15	1.2	1.1534	4.70	15
1.1	1.1534	4.70	11	1.1	1.1564	4.70	16

09/01/76 PAGE 46

```

LINE  CACI SIMSCMPI 11.5  RELEASE AF
1  EVENT 3.ITE(103E)
2  DEFINE 103E AS AN INTEGER VARIABLE
3  LET 10 = 103E
4  LET 3.ITE(10) = "NO"
5  LET 103(10) = "NO"
6  IF 100 = "YES"
7  DO 1 = 1 17 NOAC(10X)
8  DO
9  IF 10 = 110(10X,1)
10  SKIP 1 OUTPUT LINE
11  PRINT 1 DOUBLE LINE
12  ENW.TIME(AIRPLANF(10)) AS PULLDAS
13  A/C NO. *** WAS INTERNAL THIRD CRACK BECAME EXTERNAL AT *** INCHES AND ***** FLIGHT MOUNHS
14  LEAVE
15  ELSE
16  LOOP
17  ALWAYS
18  RETURN
19  END

```

LOCAL VARIABLES OF THIS ROUTINE

INTEGER	WORD 7	1.2	INTEGER	WORD 8
INTEGER	WORD 9	104E	INTEGER	WORD 1
INTEGER	WORD 10	K.1	INTEGER	WORD 12
INTEGER	WORD 13	K.3	INTEGER	WORD 14
INTEGER	WORD 15	L.4	INTEGER	WORD 3
INTEGER	WORD 17	K.1	DOUBLE	WORD 5

1.1
1.3
J.1
K.2
K.4
N.1

09/01/76 PAGE 47

```

LINE CACT SIMSCRIPT II.5 RELEASE MP
1 ROUTINE INSPECTION,SCHEDULE(C)
2 **
3 ** SCHEDULES INSPECTIONS REL. INTERVAL LEVEL ON A GIVEN AIRCRAFT SUCH THAT THE
4 ** AIRCRAFT IS NOT INSPECTED UNTIL IT REACHES MINIMUM DEFECTABLE LGT
5 **
6 OFFICE = AS AN INTERIM VARIABLE
7 LET A1 = ASH(10) * C(10)
8 LET C1 = C.GMRTIM,DATE * C(10)
9 IF FRT,INSP,LEVEL LE 1
10 LET TML = TIME.V + .551/V
11 IF N = 2
12 LET TML = TIME.V + .006/C1
13 ALWAYS
14 LET S,INSP,AT = ENTRY.TIME(AIRPLANE(10)) + 1400(1)
15 * TIME.C.F((TIME-ENTRY.TIME(AIRPLANE(10)))/1400(1)) + 1.0)
16 SCHEDULE AN A,LEVEL,INSPECTION(10) AT S,INSP,AT
17 LET AAL(10) = A,LEVEL,INSPECTION
18 *IF,AND,LESS
19 **
20 IF FRT,INSP,LEVEL LE 2
21 LET TML = TIME.V + .410/M1
22 IF N = 2
23 LET TML = TIME.V + .655/C1
24 ALWAYS
25 LET S,INSP,AT = ENTRY.TIME(AIRPLANE(10)) + 1400(2)
26 * TIME.C.F((TIME-ENTRY.TIME(AIRPLANE(10)))/1400(2)) + 1.0)
27 SCHEDULE A A,LEVEL,INSPECTION(10) AT S,INSP,AT
28 LET AAL(10) = A,LEVEL,INSPECTION
29 *IF,AND,LESS
30 **
31 IF FRT,INSP,LEVEL LE 3
32 LET TML = TIME.V + .200/M1
33 IF N = 2
34 LET TML = TIME.V + .563/C1
35 ALWAYS
36 LET S,INSP,AT = ENTRY.TIME(AIRPLANE(10)) + C,INTERVAL(10)
37 * TIME.C.F((TIME-ENTRY.TIME(AIRPLANE(10)))/C,INTERVAL(10)) + 1.0)
38 SCHEDULE A C,LEVEL,INSPECTION(10) AT S,INSP,AT
39 LET AAL(10) = C,LEVEL,INSPECTION
40 *IF,AND,LESS
41 **
42 LET INSP,SCH(10) = "YES"
43 RETURN
44 END

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDG

LINE CACJ SIMSCRIPT II.5 RELEASE AF
LOCAL VARIABLES OF THIS ROUTINE
09/01/76 PAGE 04
C1
N
TTL
DOUBLE WORD 5
INTERSECT WORD 1
DOUBLE WORD 7
S-INSPE.A1
DOUBLE
WORD 3
WORD 9

09/01/76 PAGE 49

LINE CACI SCRIPT II.5 REFUSE #4

```

1  EVAL  ALEVEL,INSECT(10A)
2  DEFINE 10A IS AN INTERIOR VARIABLE
3  LET 10 = 10A
4  GET FIRST.COST = A.4444444444
5  LET COST = 1.0
6  **
7  **  CALL EXACT(10.275,512.999,275.512,551) YIELDING 10.00
8  **
9  **  SOME OF A.ALEVEL,1 SECTIONS(10) AT 118.0 + 1400(1)
10  LET 20(1) = A.ALEVEL,INSECT(10)
11  **
12  **

```

LOCAL VARIABLES OF THIS ROUTINE

LINE	NAME	TYPE	VALUE
10	20(1)	10A	1.0000000000

09/01/76 PAGE 50

```

LINE CACI SIMSCRIPT 11.5 RELEASE AF
1 EVENT A.LEVEL.INSPECTION(ID)
2 DEFINE IDN AS AN INTEGER VARIABLE
3 LET ID = IDN
4
5 IF EXT.INSPECTION.LEVEL LE 1
6 LET A.LEVEL.INSPECTION = AAL(ID)
7 CANCEL THE A.LEVEL.INSPECTION
8 RESCHEDULE THE A.LEVEL.INSPECTION(ID) AT TIME.V + IACCD(1)
9 LET AAL(ID) = A.LEVEL.INSPECTION
10 ALWAYS
11 LET FIXIT.COST = A.REPAIR.COST
12 LET ASWP = 1.0
13
14 CALL EXAMINE(2,.002,.513,.035,.002,.513,.010) YIELDING FUUND
15
16 SCHEDULE A A.LEVEL.INSPECTION(ID) AT TIME.V + IACCD(2)
17 LET AAL(ID) = A.LEVEL.INSPECTION
18 RETURN
19 END

```

LOCAL VARIABLES OF THIS MODULE

FUUND	NUMBER	TYPE	INTEGER	WORD
	1			

09/01/70 2565 15

```

000000
LINE      CALL INSTRUCTIONS, ILO, REFERENCE #
1  10001  CALL CALFDELINSTRUCTION(LOC)
2  10002  OPENE INO AS A PROFILE VARIANT
3  10003  CFI IN = LOC
4  10004  "
5  10005  IF CFI IN.EQUAL TO 2
6  10006  LET CALFDELINSTRUCTION = 100010
7  10007  CALL THE CALFDELINSTRUCTION
8  10008  RESUME AT THE CALFDELINSTRUCTION AT 100010 + 1000100
9  10009  "
10 10010  LET CALFDELINSTRUCTION = 1
11 10011  LET CALFDELINSTRUCTION = 100010
12 10012  CALL THE CALFDELINSTRUCTION
13 10013  RESUME AT THE CALFDELINSTRUCTION AT 100010 + 1000100
14 10014  LET CAL(10) = CALFDELINSTRUCTION
15 10015  "
16 10016  "
17 10017  "
18 10018  "
19 10019  "
20 10020  "
21 10021  CALL THE CALFDELINSTRUCTION(10) AT 100010 + 1000100
22 10022  "
23 10023  "
24 10024  "
25 10025  "
26 10026  "
27 10027  "
28 10028  "
29 10029  "
30 10030  "
31 10031  "
32 10032  "
33 10033  "
34 10034  "
35 10035  "
36 10036  "
37 10037  "
38 10038  "
39 10039  "
40 10040  "
41 10041  "
42 10042  "
43 10043  "
44 10044  "
45 10045  "
46 10046  "
47 10047  "
48 10048  "
49 10049  "
50 10050  "
51 10051  "
52 10052  "
53 10053  "
54 10054  "
55 10055  "
56 10056  "
57 10057  "
58 10058  "
59 10059  "
60 10060  "
61 10061  "
62 10062  "
63 10063  "
64 10064  "
65 10065  "
66 10066  "
67 10067  "
68 10068  "
69 10069  "
70 10070  "
71 10071  "
72 10072  "
73 10073  "
74 10074  "
75 10075  "
76 10076  "
77 10077  "
78 10078  "
79 10079  "
80 10080  "
81 10081  "
82 10082  "
83 10083  "
84 10084  "
85 10085  "
86 10086  "
87 10087  "
88 10088  "
89 10089  "
90 10090  "
91 10091  "
92 10092  "
93 10093  "
94 10094  "
95 10095  "
96 10096  "
97 10097  "
98 10098  "
99 10099  "
100 10100  "

```

4-11-41 #1547-1548 7-20-7

— 10 —

09/01/70 PAGE 52

LINE CACI SCRIPT II.5 RELEASE

```

1  LET I=LEVEL,INSPECTION(100)
2  SET I TO AN INTEGER VARIABLE
3  SET I TO AN INTEGER VARIABLE
4  LET I=100
5  IF I=100 THEN
6    LET I=100
7    DO
8      IF I=100 THEN
9        SET I TO AN INTEGER VARIABLE
10       SET I TO AN INTEGER VARIABLE
11       LET I=100
12       IF I=100 THEN
13         LET I=100
14         ALWAYS
15         IF I=100 THEN
16           LET I=100
17           LET I=100
18           LET I=100
19           LET I=100
20           LET I=100
21           LET I=100
22           LET I=100
23           LET I=100
24           LET I=100
25           LET I=100
26           LET I=100
27           LET I=100
28           LET I=100
29           LET I=100
30           LET I=100
31           LET I=100
32           LET I=100
33           LET I=100
34           LET I=100
35           LET I=100
36           LET I=100
37           LET I=100
38           LET I=100
39           LET I=100
40           LET I=100
41           LET I=100
42           LET I=100
43           LET I=100
44           LET I=100
45           LET I=100
46           LET I=100
47           LET I=100
48           LET I=100
49           LET I=100
50           LET I=100
51           LET I=100
52           LET I=100

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

09/01/76 PAF: 53

[illegible]

-173-

[illegible]

09/01/76 PAGE 55

LIST CACI SIMULATOR ILS RELEASE AF

1 ROUTINE FLOWCHART (CASI, CACI, CACI, CACI) YIELDING FLOWCHART

2 OFFICE AS AN INTERCOM VARIATION

3 OFFICE / AS AN ALPHA VARIATION

4 IF LIST = "YES"

5 END IF = 1 (1) NOAC(LIST)

6

7 IF LIST = "YES"

8 LET LIST = 1

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

09/01/76 PAGE 56

LINE CACI SIMSCRIPT II.5 RELEASE HF

```

53 LET TAT = TIME-ACI-STRENGTH-REDUCTION
54 LET CL = (TIME-V-TAT) * 1 + CGRI(10)
55 IF TAC LT TAT
56 LET CL = (TAC-TAT)*1 + (TIME-V-TAC)*1 + CGRI(10)
57 IF CL LT CCL
58 LET CL = CCL + ((CCL-CCL)/1)*1
59 IF CL LT CCL
60 LET CL = CCL + ((CCL-CCL)/1)*1
61 IF CL LT CCL
62 IF CL LT CCL
63 LET CL = CCL + ((CCL-CCL)/1)*1
64 LET CL = CCL + ((CCL-CCL)/1)*1
65 LET CL = CCL + ((CCL-CCL)/1)*1
66 LET CL = CCL + ((CCL-CCL)/1)*1
67 LET CL = CCL + ((CCL-CCL)/1)*1
68 LET CL = CCL + ((CCL-CCL)/1)*1
69 LET CL = CCL + ((CCL-CCL)/1)*1
70 LET CL = CCL + ((CCL-CCL)/1)*1
71 LET CL = CCL + ((CCL-CCL)/1)*1
72 LET CL = CCL + ((CCL-CCL)/1)*1
73 LET CL = CCL + ((CCL-CCL)/1)*1
74 LET CL = CCL + ((CCL-CCL)/1)*1
75 LET CL = CCL + ((CCL-CCL)/1)*1
76 LET CL = CCL + ((CCL-CCL)/1)*1
77 LET CL = CCL + ((CCL-CCL)/1)*1
78 LET CL = CCL + ((CCL-CCL)/1)*1
79 LET CL = CCL + ((CCL-CCL)/1)*1
80 LET CL = CCL + ((CCL-CCL)/1)*1
81 LET CL = CCL + ((CCL-CCL)/1)*1
82 LET CL = CCL + ((CCL-CCL)/1)*1
83 LET CL = CCL + ((CCL-CCL)/1)*1
84 LET CL = CCL + ((CCL-CCL)/1)*1
85 LET CL = CCL + ((CCL-CCL)/1)*1
86 LET CL = CCL + ((CCL-CCL)/1)*1
87 LET CL = CCL + ((CCL-CCL)/1)*1
88 LET CL = CCL + ((CCL-CCL)/1)*1
89 LET CL = CCL + ((CCL-CCL)/1)*1
90 LET CL = CCL + ((CCL-CCL)/1)*1
91 LET CL = CCL + ((CCL-CCL)/1)*1
92 LET CL = CCL + ((CCL-CCL)/1)*1
93 LET CL = CCL + ((CCL-CCL)/1)*1
94 LET CL = CCL + ((CCL-CCL)/1)*1
95 LET CL = CCL + ((CCL-CCL)/1)*1
96 LET CL = CCL + ((CCL-CCL)/1)*1
97 LET CL = CCL + ((CCL-CCL)/1)*1
98 LET CL = CCL + ((CCL-CCL)/1)*1
99 LET CL = CCL + ((CCL-CCL)/1)*1
100 LET CL = CCL + ((CCL-CCL)/1)*1
101 LET CL = CCL + ((CCL-CCL)/1)*1
102 LET CL = CCL + ((CCL-CCL)/1)*1
103 LET CL = CCL + ((CCL-CCL)/1)*1
104 LET CL = CCL + ((CCL-CCL)/1)*1

```

66/08/76 WDAF 51

```

04/01/76 PAGE 57

LINE      CACT SIM-COMPUT II.5  WFLPASE 46
105      IF N = 1
106          LET ACMAL = CL
107          LET SACOM = CL
108          LET Z = "C"
109          JUMP AMEAL
110      IF N = 2
111          LET ACMAL = CL
112          LET SACOM = CL
113          LET Z = "A"
114          JUMP AMEAL
115      IF N = 3
116          LET ACMAL = CL
117          LET SACOM = CL
118          LET Z = "M"
119          JUMP AMEAL
120      IF N = 4
121          LET ACMAL = CL
122          LET SACOM = CL
123          LET Z = "C"
124          JUMP AMEAL
125      IF N = 5
126          LET ACMAL = CL
127          LET SACOM = CL
128          LET Z = "A"
129          JUMP AMEAL
130      IF N = 6
131          LET ACMAL = CL
132          LET SACOM = CL
133          LET Z = "M"
134          JUMP AMEAL
135      IF N = 7
136          LET ACMAL = CL
137          LET SACOM = CL
138          LET Z = "C"
139          JUMP AMEAL
140      IF N = 8
141          LET ACMAL = CL
142          LET SACOM = CL
143          LET Z = "A"
144          JUMP AMEAL
145      IF N = 9
146          LET ACMAL = CL
147          LET SACOM = CL
148          LET Z = "M"
149          JUMP AMEAL
150      IF N = 10
151          LET ACMAL = CL
152          LET SACOM = CL
153          LET Z = "C"
154          JUMP AMEAL
155      IF N = 11
156          LET ACMAL = CL
157          LET SACOM = CL
158          LET Z = "A"
159          JUMP AMEAL
160      IF N = 12
161          LET ACMAL = CL
162          LET SACOM = CL
163          LET Z = "M"
164          JUMP AMEAL
165      IF N = 13
166          LET ACMAL = CL
167          LET SACOM = CL
168          LET Z = "C"
169          JUMP AMEAL
170      IF N = 14
171          LET ACMAL = CL
172          LET SACOM = CL
173          LET Z = "A"
174          JUMP AMEAL
175      IF N = 15
176          LET ACMAL = CL
177          LET SACOM = CL
178          LET Z = "M"
179          JUMP AMEAL
180      IF N = 16
181          LET ACMAL = CL
182          LET SACOM = CL
183          LET Z = "C"
184          JUMP AMEAL
185      IF N = 17
186          LET ACMAL = CL
187          LET SACOM = CL
188          LET Z = "A"
189          JUMP AMEAL
190      IF N = 18
191          LET ACMAL = CL
192          LET SACOM = CL
193          LET Z = "M"
194          JUMP AMEAL
195      IF N = 19
196          LET ACMAL = CL
197          LET SACOM = CL
198          LET Z = "C"
199          JUMP AMEAL
200      IF N = 20
201          LET ACMAL = CL
202          LET SACOM = CL
203          LET Z = "A"
204          JUMP AMEAL
205      IF N = 21
206          LET ACMAL = CL
207          LET SACOM = CL
208          LET Z = "M"
209          JUMP AMEAL
210      IF N = 22
211          LET ACMAL = CL
212          LET SACOM = CL
213          LET Z = "C"
214          JUMP AMEAL
215      IF N = 23
216          LET ACMAL = CL
217          LET SACOM = CL
218          LET Z = "A"
219          JUMP AMEAL
220      IF N = 24
221          LET ACMAL = CL
222          LET SACOM = CL
223          LET Z = "M"
224          JUMP AMEAL
225      IF N = 25
226          LET ACMAL = CL
227          LET SACOM = CL
228          LET Z = "C"
229          JUMP AMEAL
230      IF N = 26
231          LET ACMAL = CL
232          LET SACOM = CL
233          LET Z = "A"
234          JUMP AMEAL
235      IF N = 27
236          LET ACMAL = CL
237          LET SACOM = CL
238          LET Z = "M"
239          JUMP AMEAL
240      IF N = 28
241          LET ACMAL = CL
242          LET SACOM = CL
243          LET Z = "C"
244          JUMP AMEAL
245      IF N = 29
246          LET ACMAL = CL
247          LET SACOM = CL
248          LET Z = "A"
249          JUMP AMEAL
250      IF N = 30
251          LET ACMAL = CL
252          LET SACOM = CL
253          LET Z = "M"
254          JUMP AMEAL
255      IF N = 31
256          LET ACMAL = CL
257          LET SACOM = CL
258          LET Z = "C"
259          JUMP AMEAL
260      IF N = 32
261          LET ACMAL = CL
262          LET SACOM = CL
263          LET Z = "A"
264          JUMP AMEAL
265      IF N = 33
266          LET ACMAL = CL
267          LET SACOM = CL
268          LET Z = "M"
269          JUMP AMEAL
270      IF N = 34
271          LET ACMAL = CL
272          LET SACOM = CL
273          LET Z = "C"
274          JUMP AMEAL
275      IF N = 35
276          LET ACMAL = CL
277          LET SACOM = CL
278          LET Z = "A"
279          JUMP AMEAL
280      IF N = 36
281          LET ACMAL = CL
282          LET SACOM = CL
283          LET Z = "M"
284          JUMP AMEAL
285      IF N = 37
286          LET ACMAL = CL
287          LET SACOM = CL
288          LET Z = "C"
289          JUMP AMEAL
290      IF N = 38
291          LET ACMAL = CL
292          LET SACOM = CL
293          LET Z = "A"
294          JUMP AMEAL
295      IF N = 39
296          LET ACMAL = CL
297          LET SACOM = CL
298          LET Z = "M"
299          JUMP AMEAL
300      IF N = 40
301          LET ACMAL = CL
302          LET SACOM = CL
303          LET Z = "C"
304          JUMP AMEAL
305      IF N = 41
306          LET ACMAL = CL
307          LET SACOM = CL
308          LET Z = "A"
309          JUMP AMEAL
310      IF N = 42
311          LET ACMAL = CL
312          LET SACOM = CL
313          LET Z = "M"
314          JUMP AMEAL
315      IF N = 43
316          LET ACMAL = CL
317          LET SACOM = CL
318          LET Z = "C"
319          JUMP AMEAL
320      IF N = 44
321          LET ACMAL = CL
322          LET SACOM = CL
323          LET Z = "A"
324          JUMP AMEAL
325      IF N = 45
326          LET ACMAL = CL
327          LET SACOM = CL
328          LET Z = "M"
329          JUMP AMEAL
330      IF N = 46
331          LET ACMAL = CL
332          LET SACOM = CL
333          LET Z = "C"
334          JUMP AMEAL
335      IF N = 47
336          LET ACMAL = CL
337          LET SACOM = CL
338          LET Z = "A"
339          JUMP AMEAL
340      IF N = 48
341          LET ACMAL = CL
342          LET SACOM = CL
343          LET Z = "M"
344          JUMP AMEAL
345      IF N = 49
346          LET ACMAL = CL
347          LET SACOM = CL
348          LET Z = "C"
349          JUMP AMEAL
350      IF N = 50
351          LET ACMAL = CL
352          LET SACOM = CL
353          LET Z = "A"
354          JUMP AMEAL
355      IF N = 51
356          LET ACMAL = CL
357          LET SACOM = CL
358          LET Z = "M"
359          JUMP AMEAL
360      IF N = 52
361          LET ACMAL = CL
362          LET SACOM = CL
363          LET Z = "C"
364          JUMP AMEAL
365      IF N = 53
366          LET ACMAL = CL
367          LET SACOM = CL
368          LET Z = "A"
369          JUMP AMEAL
370      IF N = 54
371          LET ACMAL = CL
372          LET SACOM = CL
373          LET Z = "M"
374          JUMP AMEAL
375      IF N = 55
376          LET ACMAL = CL
377          LET SACOM = CL
378          LET Z = "C"
379          JUMP AMEAL
380      IF N = 56
381          LET ACMAL = CL
382          LET SACOM = CL
383          LET Z = "A"
384          JUMP AMEAL
385      IF N = 57
386          LET ACMAL = CL
387          LET SACOM = CL
388          LET Z = "M"
389          JUMP AMEAL
390      IF N = 58
391          LET ACMAL = CL
392          LET SACOM = CL
393          LET Z = "C"
394          JUMP AMEAL
395      IF N = 59
396          LET ACMAL = CL
397          LET SACOM = CL
398          LET Z = "A"
399          JUMP AMEAL
400      IF N = 60
401          LET ACMAL = CL
402          LET SACOM = CL
403          LET Z = "M"
404          JUMP AMEAL
405      IF N = 61
406          LET ACMAL = CL
407          LET SACOM = CL
408          LET Z = "C"
409          JUMP AMEAL
410      IF N = 62
411          LET ACMAL = CL
412          LET SACOM = CL
413          LET Z = "A"
414          JUMP AMEAL
415      IF N = 63
416          LET ACMAL = CL
417          LET SACOM = CL
418          LET Z = "M"
419          JUMP AMEAL
420      IF N = 64
421          LET ACMAL = CL
422          LET SACOM = CL
423          LET Z = "C"
424          JUMP AMEAL
425      IF N = 65
426          LET ACMAL = CL
427          LET SACOM = CL
428          LET Z = "A"
429          JUMP AMEAL
430      IF N = 66
431          LET ACMAL = CL
432          LET SACOM = CL
433          LET Z = "M"
434          JUMP AMEAL
435      IF N = 67
436          LET ACMAL = CL
437          LET SACOM = CL
438          LET Z = "C"
439          JUMP AMEAL
440      IF N = 68
441          LET ACMAL = CL
442          LET SACOM = CL
443          LET Z = "A"
444          JUMP AMEAL
445      IF N = 69
446          LET ACMAL = CL
447          LET SACOM = CL
448          LET Z = "M"
449          JUMP AMEAL
450      IF N = 70
451          LET ACMAL = CL
452          LET SACOM = CL
453          LET Z = "C"
454          JUMP AMEAL
455      IF N = 71
456          LET ACMAL = CL
457          LET SACOM = CL
458          LET Z = "A"
459          JUMP AMEAL
460      IF N = 72
461          LET ACMAL = CL
462          LET SACOM = CL
463          LET Z = "M"
464          JUMP AMEAL
465      IF N = 73
466          LET ACMAL = CL
467          LET SACOM = CL
468          LET Z = "C"
469          JUMP AMEAL
470      IF N = 74
471          LET ACMAL = CL
472          LET SACOM = CL
473          LET Z = "A"
474          JUMP AMEAL
475      IF N = 75
476          LET ACMAL = CL
477          LET SACOM = CL
478          LET Z = "M"
479          JUMP AMEAL
480      IF N = 76
481          LET ACMAL = CL
482          LET SACOM = CL
483          LET Z = "C"
484          JUMP AMEAL
485      IF N = 77
486          LET ACMAL = CL
487          LET SACOM = CL
488          LET Z = "A"
489          JUMP AMEAL
490      IF N = 78
491          LET ACMAL = CL
492          LET SACOM = CL
493          LET Z = "M"
494          JUMP AMEAL
495      IF N = 79
49
```

00/01/76 PAGE 5A

```

LINE 151  CALL STOPC-INT 11.5  RELEASE #4
152
153  LET DCMAL = CL
154  LET GCMAL = CL
155  LET Z = "0"
156
157  IF LIST = 1.0
158  THEN
159    PRINT "NO-OUT LIST"
160    GOTO 1
161  ELSE
162    CALLS
163    CURVE OF LENGTH 00.00 INCHES DETECTED DURING OUTLET INSPECTION OF ARC WELDER AT 000000 FLIGHT MILES
164    ALWAYS
165    LET DCMAL = 1
166    GOTO 1
167  ELSE
168    ALWAYS
169    ALWAYS
170    ALWAYS
171    ALWAYS
172  ..
173  ..
174  ..
175  IF DCMAL = 1
176  THEN
177    STOPC-INT 11.5  RELEASE #4
178    GOTO 1
179  ELSE
180    ALWAYS
181  END

```

177

LOCAL VARIABLES OF THIS ROUTINE

NAME	TYPE	VALUE	ADDRESS
AREA	REAL	1.0	0000 20
CL	REAL	1.0	0000 21
DCMAL	REAL	1.0	0000 22
GCMAL	REAL	1.0	0000 23
LIST	REAL	1.0	0000 24
Z	REAL	1.0	0000 25
DCMAL	REAL	1.0	0000 26
GCMAL	REAL	1.0	0000 27
LIST	REAL	1.0	0000 28
Z	REAL	1.0	0000 29
DCMAL	REAL	1.0	0000 30
GCMAL	REAL	1.0	0000 31
LIST	REAL	1.0	0000 32
Z	REAL	1.0	0000 33
DCMAL	REAL	1.0	0000 34
GCMAL	REAL	1.0	0000 35
LIST	REAL	1.0	0000 36
Z	REAL	1.0	0000 37
DCMAL	REAL	1.0	0000 38
GCMAL	REAL	1.0	0000 39
LIST	REAL	1.0	0000 40
Z	REAL	1.0	0000 41
DCMAL	REAL	1.0	0000 42
GCMAL	REAL	1.0	0000 43
LIST	REAL	1.0	0000 44
Z	REAL	1.0	0000 45
DCMAL	REAL	1.0	0000 46
GCMAL	REAL	1.0	0000 47
LIST	REAL	1.0	0000 48
Z	REAL	1.0	0000 49
DCMAL	REAL	1.0	0000 50
GCMAL	REAL	1.0	0000 51
LIST	REAL	1.0	0000 52
Z	REAL	1.0	0000 53
DCMAL	REAL	1.0	0000 54
GCMAL	REAL	1.0	0000 55
LIST	REAL	1.0	0000 56
Z	REAL	1.0	0000 57
DCMAL	REAL	1.0	0000 58
GCMAL	REAL	1.0	0000 59
LIST	REAL	1.0	0000 60
Z	REAL	1.0	0000 61
DCMAL	REAL	1.0	0000 62
GCMAL	REAL	1.0	0000 63
LIST	REAL	1.0	0000 64
Z	REAL	1.0	0000 65
DCMAL	REAL	1.0	0000 66
GCMAL	REAL	1.0	0000 67
LIST	REAL	1.0	0000 68
Z	REAL	1.0	0000 69
DCMAL	REAL	1.0	0000 70
GCMAL	REAL	1.0	0000 71
LIST	REAL	1.0	0000 72
Z	REAL	1.0	0000 73
DCMAL	REAL	1.0	0000 74
GCMAL	REAL	1.0	0000 75
LIST	REAL	1.0	0000 76
Z	REAL	1.0	0000 77
DCMAL	REAL	1.0	0000 78
GCMAL	REAL	1.0	0000 79
LIST	REAL	1.0	0000 80
Z	REAL	1.0	0000 81
DCMAL	REAL	1.0	0000 82
GCMAL	REAL	1.0	0000 83
LIST	REAL	1.0	0000 84
Z	REAL	1.0	0000 85
DCMAL	REAL	1.0	0000 86
GCMAL	REAL	1.0	0000 87
LIST	REAL	1.0	0000 88
Z	REAL	1.0	0000 89
DCMAL	REAL	1.0	0000 90
GCMAL	REAL	1.0	0000 91
LIST	REAL	1.0	0000 92
Z	REAL	1.0	0000 93
DCMAL	REAL	1.0	0000 94
GCMAL	REAL	1.0	0000 95
LIST	REAL	1.0	0000 96
Z	REAL	1.0	0000 97
DCMAL	REAL	1.0	0000 98
GCMAL	REAL	1.0	0000 99
LIST	REAL	1.0	0000 100
Z	REAL	1.0	0000 101
DCMAL	REAL	1.0	0000 102
GCMAL	REAL	1.0	0000 103
LIST	REAL	1.0	0000 104
Z	REAL	1.0	0000 105
DCMAL	REAL	1.0	0000 106
GCMAL	REAL	1.0	0000 107
LIST	REAL	1.0	0000 108
Z	REAL	1.0	0000 109
DCMAL	REAL	1.0	0000 110
GCMAL	REAL	1.0	0000 111
LIST	REAL	1.0	0000 112
Z	REAL	1.0	0000 113
DCMAL	REAL	1.0	0000 114
GCMAL	REAL	1.0	0000 115
LIST	REAL	1.0	0000 116
Z	REAL	1.0	0000 117
DCMAL	REAL	1.0	0000 118
GCMAL	REAL	1.0	0000 119
LIST	REAL	1.0	0000 120
Z	REAL	1.0	0000 121
DCMAL	REAL	1.0	0000 122
GCMAL	REAL	1.0	0000 123
LIST	REAL	1.0	0000 124
Z	REAL	1.0	0000 125
DCMAL	REAL	1.0	0000 126
GCMAL	REAL	1.0	0000 127
LIST	REAL	1.0	0000 128
Z	REAL	1.0	0000 129
DCMAL	REAL	1.0	0000 130
GCMAL	REAL	1.0	0000 131
LIST	REAL	1.0	0000 132
Z	REAL	1.0	0000 133
DCMAL	REAL	1.0	0000 134
GCMAL	REAL	1.0	0000 135
LIST	REAL	1.0	0000 136
Z	REAL	1.0	0000 137
DCMAL	REAL	1.0	0000 138
GCMAL	REAL	1.0	0000 139
LIST	REAL	1.0	0000 140
Z	REAL	1.0	0000 141
DCMAL	REAL	1.0	0000 142
GCMAL	REAL	1.0	0000 143
LIST	REAL	1.0	0000 144
Z	REAL	1.0	0000 145
DCMAL	REAL	1.0	0000 146
GCMAL	REAL	1.0	0000 147
LIST	REAL	1.0	0000 148
Z	REAL	1.0	0000 149
DCMAL	REAL	1.0	0000 150
GCMAL	REAL	1.0	0000 151
LIST	REAL	1.0	0000 152
Z	REAL	1.0	0000 153
DCMAL	REAL	1.0	0000 154
GCMAL	REAL	1.0	0000 155
LIST	REAL	1.0	0000 156
Z	REAL	1.0	0000 157
DCMAL	REAL	1.0	0000 158
GCMAL	REAL	1.0	0000 159
LIST	REAL	1.0	0000 160
Z	REAL	1.0	0000 161
DCMAL	REAL	1.0	0000 162
GCMAL	REAL	1.0	0000 163
LIST	REAL	1.0	0000 164
Z	REAL	1.0	0000 165
DCMAL	REAL	1.0	0000 166
GCMAL	REAL	1.0	0000 167
LIST	REAL	1.0	0000 168
Z	REAL	1.0	0000 169
DCMAL	REAL	1.0	0000 170
GCMAL	REAL	1.0	0000 171
LIST	REAL	1.0	0000 172
Z	REAL	1.0	0000 173
DCMAL	REAL	1.0	0000 174
GCMAL	REAL	1.0	0000 175
LIST	REAL	1.0	0000 176
Z	REAL	1.0	0000 177
DCMAL	REAL	1.0	0000 178
GCMAL	REAL	1.0	0000 179
LIST	REAL	1.0	0000 180
Z	REAL	1.0	0000 181
DCMAL	REAL	1.0	0000 182
GCMAL	REAL	1.0	0000 183
LIST	REAL	1.0	0000 184
Z	REAL	1.0	0000 185
DCMAL	REAL	1.0	0000 186
GCMAL	REAL	1.0	0000 187
LIST	REAL	1.0	0000 188
Z	REAL	1.0	0000 189
DCMAL	REAL	1.0	0000 190
GCMAL	REAL	1.0	0000 191
LIST	REAL	1.0	0000 192
Z	REAL	1.0	0000 193
DCMAL	REAL	1.0	0000 194
GCMAL	REAL	1.0	0000 195
LIST	REAL	1.0	0000 196
Z	REAL	1.0	0000 197
DCMAL	REAL	1.0	0000 198
GCMAL	REAL	1.0	0000 199
LIST	REAL	1.0	0000 200
Z	REAL	1.0	0000 201
DCMAL	REAL	1.0	0000 202
GCMAL	REAL	1.0	0000 203
LIST	REAL	1.0	0000 204
Z	REAL	1.0	0000 205
DCMAL	REAL	1.0	0000 206
GCMAL	REAL	1.0	0000 207
LIST	REAL	1.0	0000 208
Z	REAL	1.0	0000 209
DCMAL	REAL	1.0	0000 210
GCMAL	REAL	1.0	0000 211
LIST	REAL	1.0	0000 212
Z	REAL	1.0	0000 213
DCMAL	REAL	1.0	0000 214
GCMAL	REAL	1.0	0000 215
LIST	REAL	1.0	0000 216
Z	REAL	1.0	0000 217
DCMAL	REAL	1.0	0000 218
GCMAL	REAL	1.0	0000 219
LIST	REAL	1.0	0000 220
Z	REAL	1.0	0000 221
DCMAL	REAL	1.0	0000 222
GCMAL	REAL	1.0	0000 223
LIST	REAL	1.0	0000 224
Z	REAL	1.0	0000 225
DCMAL	REAL	1.0	0000 226
GCMAL	REAL	1.0	0000 227
LIST	REAL	1.0	0000 228
Z	REAL	1.0	0000 229
DCMAL	REAL	1.0	0000 230
GCMAL	REAL	1.0	0000 231
LIST	REAL	1.0	0000 232
Z	REAL	1.0	0000 233
DCMAL	REAL	1.0	0000 234
GCMAL	REAL	1.0	0000 235
LIST	REAL	1.0	0000 236
Z	REAL	1.0	0000 237
DCMAL	REAL	1.0	0000 238
GCMAL	REAL	1.0	0000 239
LIST	REAL	1.0	0000 240
Z	REAL	1.0	0000 241
DCMAL	REAL	1.0	0000 242
GCMAL	REAL	1.0	0000 243
LIST	REAL	1.0	0000 244
Z	REAL	1.0	0000 245
DCMAL	REAL	1.0	0000 246
GCMAL	REAL	1.0	0000 247
LIST	REAL	1.0	0000 248
Z	REAL	1.0	0000 249
DCMAL	REAL	1.0	0000 250
GCMAL	REAL	1.0	0000 251
LIST	REAL	1.0	0000 252
Z	REAL	1.0	0000 253
DCMAL	REAL	1.0	0000 254
GCMAL	REAL	1.0	0000 255
LIST	REAL	1.0	0000 256
Z	REAL	1.0	0000 257
DCMAL	REAL	1.0	0000 258
GCMAL	REAL	1.0	0000 259
LIST	REAL	1.0	0000 260
Z	REAL	1.0	0000 261
DCMAL	REAL	1.0	0000 262
GCMAL	REAL	1.0	0000 263
LIST	REAL	1.0	0000 264
Z	REAL	1.0	0000 265
DCMAL	REAL	1.0	0000 266
GCMAL	REAL	1.0	0000 267
LIST	REAL	1.0	0000 268
Z	REAL	1.0	0000 269
DCMAL	REAL	1.0	0000 270
GCMAL	REAL	1.0	0000 271
LIST	REAL	1.0	0000 272
Z	REAL	1.0	0000 273
DCMAL	REAL	1.0	0000 274
GCMAL	REAL	1.0	0000 275
LIST	REAL	1.0	0000 276
Z	REAL	1.0	0000 277
DCMAL	REAL	1.0	0000 278
GCMAL	REAL	1.0	0000 279
LIST	REAL	1.0	0000 280
Z	REAL	1.0	0000 281
DCMAL	REAL	1.0	0000 282
GCMAL	REAL	1.0	0000 283
LIST	REAL	1.0	0000 284
Z	REAL	1.0	0000 285
DCMAL	REAL	1.0	0000 286
GCMAL	REAL	1.0	0000 287
LIST	REAL	1.0	0000 288
Z	REAL	1.0	0000 289
DCMAL			

09/01/76 PAGE 50

LINE CASE SIMULATION ILLS RELEASE 44

1. MONITORING MODIFICATIONS

2. COMPUTES PROBABILITY OF DETECTING A SERVICE DEFECT OF A GIVEN SIZE
3. COMPUTES PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL
4. BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL
5. BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL
6. BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL
7. BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL
8. BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL
9. BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL
10. BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL
11. BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL
12. BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL

BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL

LOCAL VARIATIONS OF THIS METHOD

BASES ON PROBABILITY OF DETECTING A DEFECT AT A GIVEN INSPECTION LEVEL

-179-

-179-

-180-

1977 JAN 15 10 11 AM '77

```

1  EVENT REACHES FAILSAFE.LOG(IOWFS)
2
3  DEFINE IOWFS AS AN INTEGER VARIABLE
4
5  LET IOWFS = IOWFS
6
7  IF IOWFS = "YES"
8
9      IOWFS = IOWFS + 1
10
11  ELSE
12
13      ALWAYS
14
15      IF IOWFS LT 20
16
17          LET IOWFS = IOWFS + 1
18
19          LET STIME(IOWFS) = IOWFS
20
21          LET STIME(IOWFS) = TIME.Y - ELEV.Y.TIME(ADIPLANE(IOWFS))
22
23          IF STIME(IOWFS) = IOWFS
24
25              RETURN
26
27          LET AIL(IOWFS) = "yes"
28
29      END

```

LIC2L vA4TΔ4LES .IF 1-15 444:1115

Case	Age	Sex	Occupation	Duration of Illness	Site of Lesion	Microscopic Findings	Diagnosis
1.1	45	M	Teacher	10 years	Brain	Granuloma with eosinophils	Granuloma
1.2	55	F	Homemaker	5 years	Brain	Granuloma with eosinophils	Granuloma
1.3	60	M	Engineer	15 years	Brain	Granuloma with eosinophils	Granuloma
1.4	50	F	Homemaker	10 years	Brain	Granuloma with eosinophils	Granuloma
1.5	40	M	Teacher	10 years	Brain	Granuloma with eosinophils	Granuloma
1.6	55	F	Homemaker	10 years	Brain	Granuloma with eosinophils	Granuloma
1.7	60	M	Engineer	15 years	Brain	Granuloma with eosinophils	Granuloma
1.8	50	F	Homemaker	10 years	Brain	Granuloma with eosinophils	Granuloma
1.9	40	M	Teacher	10 years	Brain	Granuloma with eosinophils	Granuloma
1.10	55	F	Homemaker	10 years	Brain	Granuloma with eosinophils	Granuloma
1.11	60	M	Engineer	15 years	Brain	Granuloma with eosinophils	Granuloma
1.12	50	F	Homemaker	10 years	Brain	Granuloma with eosinophils	Granuloma
1.13	40	M	Teacher	10 years	Brain	Granuloma with eosinophils	Granuloma
1.14	55	F	Homemaker	10 years	Brain	Granuloma with eosinophils	Granuloma
1.15	60	M	Engineer	15 years	Brain	Granuloma with eosinophils	Granuloma
1.16	50	F	Homemaker	10 years	Brain	Granuloma with eosinophils	Granuloma
1.17	40	M	Teacher	10 years	Brain	Granuloma with eosinophils	Granuloma
1.18	55	F	Homemaker	10 years	Brain	Granuloma with eosinophils	Granuloma
1.19	60	M	Engineer	15 years	Brain	Granuloma with eosinophils	Granuloma
1.20	50	F	Homemaker	10 years	Brain	Granuloma with eosinophils	Granuloma

09/01/76 PAGE 62

LINE CACI SYNSCRIPT 11.5 RELEASE RF

```

1 EVENT FAILURE(10FA)
2 DEFINE 10FA AS AN INTEGER VARIABLE
3 DEFINE 10FA AS AN INTEGER VARIABLE
4 LET 10 = 10FA
5 IF 10 = "YES"
6   FOR I = 1 TO 10000(10X)
7     DO
8       IF 10 = 10000(10X)
9         SKIP 1 LINE WITH LINE
10        PRINT 1 LINE WITH LINE
11        A/C NO. AND EXPERIENCES (LEFT) FAILURE AT ***** FLIGHT HOURS
12        LET 1 = 10000(10X)
13        LET 1 = 10000(10X)
14        LET 1 = 10000(10X)
15        LET 1 = 10000(10X)
16        LET 1 = 10000(10X)
17        LET 1 = 10000(10X)
18        LET 1 = 10000(10X)
19        LET 1 = 10000(10X)
20        LET 1 = 10000(10X)
21        LET 1 = 10000(10X)
22        LET 1 = 10000(10X)
23        LET 1 = 10000(10X)
24        LET 1 = 10000(10X)
25        LET 1 = 10000(10X)
26        LET 1 = 10000(10X)
27        LET 1 = 10000(10X)
28        LET 1 = 10000(10X)
29        LET 1 = 10000(10X)
30        LET 1 = 10000(10X)
31        LET 1 = 10000(10X)
32        LET 1 = 10000(10X)
33        LET 1 = 10000(10X)
34        LET 1 = 10000(10X)
35        LET 1 = 10000(10X)
36        LET 1 = 10000(10X)
37        LET 1 = 10000(10X)
38        LET 1 = 10000(10X)
39        LET 1 = 10000(10X)
40        LET 1 = 10000(10X)
41        LET 1 = 10000(10X)
42        LET 1 = 10000(10X)
43        LET 1 = 10000(10X)
44        LET 1 = 10000(10X)
45        LET 1 = 10000(10X)
46        LET 1 = 10000(10X)
47        LET 1 = 10000(10X)
48        LET 1 = 10000(10X)
49        LET 1 = 10000(10X)
50        LET 1 = 10000(10X)
51        LET 1 = 10000(10X)
52        LET 1 = 10000(10X)

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDG

09/01/76 PAGE 03

LINE CACT SIMSCRIPT ILS RELEASE OF

SIZE OF CRACK LENGTHS AT FAILURE = *** INCHES
RESIDUAL STRENGTH AT FAILURE = ** ULTIMATE

LEAVE

ELSE

LOOP

ALWAYS

SCHEMATIC AN INCREASE INSPECTION FREQUENCY AND

SCHEMATIC AN IMMEDIATE FUEL INSPECTION RUN

LET FSH(10) = 0.00

IF SPSL(1) = 0

LET SPSL = SPSL + 1

LET A(10)(USPL) = 10

LET FLW(USPL) = TIME.V - ENTRY.TIME(AIRPLANE(10))

IF FLW(USPL) = 0

IF SPSL(1) = 100

LET SPSL = SPSL + 1

LET SACT(USPL) = 10

LET SFLW(USPL) = TIME.V - ENTRY.TIME(AIRPLANE(10))

LET SFLW(USPL) = FLW(USPL)

IF FLW(USPL) = 0

FOR I = 1 TO 10

DO

IF MI.TIME.ACFT(I) = 10

ADD 1 TO LMTA

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

IF LMTA(1) = 10

09/01/76 PAGE 64

```

104 CACI SIMSCRIPT II.5 WELFARE AF
105 REGARDLESS
106 REMOVE AIRPLANE(10) FROM ACTIVE.FLEET
107 FILE AIRPLANE(10) IN CRASHED.FLEET
108 ADD 1 TO 2.NUM.OF.CRASH
109
110 ** CANCEL SCHEDULED EVENTS AND DESTROY EVENT NOTICES
111 **
112 IF 1E(10) = "YES"
113 LET 1.LIFE = 1E(10)
114 CANCEL THE 1.LIFE
115 DESTROY THE 1.LIFE
116 IF 1E(10) = "YES"
117 LET 2.LIFE = 2E(10)
118 CANCEL THE 2.LIFE
119 DESTROY THE 2.LIFE
120 IF 1E(10) = "YES"
121 LET 3.LIFE = 3E(10)
122 CANCEL THE 3.LIFE
123 DESTROY THE 3.LIFE
124 ALWAYS
125
126 IF 1E(10) = "YES"
127 LET COMBUSTION = 1E(10)
128 IF COMBUSTION = "YES"
129 CANCEL THE COMBUSTION
130 ALWAYS
131 DESTROY THE COMBUSTION
132 LET COMBUSTION = "NO"
133
134 IF 1E(10) = "YES"
135 LET 1.STRENGTH.REDUCTION = 1E(10)
136 IF 1E(10) = "YES"
137 CANCEL THE 1.STRENGTH.REDUCTION
138 ALWAYS
139 DESTROY THE 1.STRENGTH.REDUCTION
140 LET 1E(10) = "NO"
141
142 IF 2E(10) = "YES"
143 LET 2.STRENGTH.REDUCTION = 2E(10)
144 IF 2E(10) = "YES"
145 CANCEL THE 2.STRENGTH.REDUCTION
146 ALWAYS
147 DESTROY THE 2.STRENGTH.REDUCTION
148 LET 2E(10) = "NO"
149
150 IF 3E(10) = "YES"
151 LET 3.STRENGTH.REDUCTION = 3E(10)
152 IF 3E(10) = "YES"
153 CANCEL THE 3.STRENGTH.REDUCTION
154 ALWAYS
155 DESTROY THE 3.STRENGTH.REDUCTION
156 LET 3E(10) = "NO"
157
158 REGARDLESS
159

```

09/01/76 PAGE 85

LINE CACI ST-SCH-INT T1.5 RELEASE AT

```

157 *END*
158 IF ALL(0) = "YES"
159 LET WEACH.FAIL.SAF.LGT = AMFSL(0)
160 CANCEL THE WEACH.FAIL.SAF.LGT
161 DESTROY THE WEACH.FAIL.SAF.LGT
162 IF ALL(0) = "NO"
163 *END*
164 LET D.LEVEL.INSPECTION = AMFSL(0)
165 CANCEL THE D.LEVEL.INSPECTION
166 DESTROY THE D.LEVEL.INSPECTION
167 IF D.LEVEL.INSPECTION = "YES"
168 CALL CANCEL.SCHEDULE.INSPECTIONS
169 *END*
170 **
171 FOR EACH WEACH.FAIL.SAF.LGT IN EV.SCHEDULE.INSPECTIONS
172 DO
173 IF TIME = 10 CANCEL THE WEACH.FAIL.SAF.LGT
174 DESTROY THE WEACH.FAIL.SAF.LGT
175 LEAVE
176 ELSE
177 FOR EACH T1.SERVICE.DAMAGE IN EV.SCHEDULE.INSPECTIONS
178 DO
179 IF TIME = 10 CANCEL THE T1.SERVICE.DAMAGE
180 DESTROY THE T1.SERVICE.DAMAGE
181 LEAVE
182 ELSE
183 FOR EACH T1.INSPECTION.INCREASE IN EV.SCHEDULE.INSPECTIONS
184 DO
185 IF TIME = 10 CANCEL THE T1.INSPECTION.INCREASE
186 DESTROY THE T1.INSPECTION.INCREASE
187 LEAVE
188 ELSE
189 RETURN
190 *END*

```

LOCAL VARIABLES OF THIS ROUTINE

COL	WORD 19	CL	WORD 33
COL	WORD 3	CL	WORD 7
COL	WORD 8	CL	WORD 9
COL	WORD 1	CL	WORD 10
COL	WORD 12	CL	WORD 15
COL	WORD 14	CL	WORD 18
COL	WORD 17	CL	WORD 21
COL	WORD 20	CL	WORD 24
COL	WORD 23	CL	WORD 26
COL	WORD 25	CL	WORD 29
COL	WORD 28	CL	WORD 32
COL	WORD 31	CL	WORD 35

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

09/01/76 PAGE 06

```

LINE CALL SIMSCRIPT 11.5 RELEASE 40
1 EVENT RETURNED FROM SERVICE (1000)
2 IF 1000 = 1 THEN
3   IF 1000 = 1 THEN
4     IF 1000 = 1 THEN
5       IF 1000 = 1 THEN
6         IF 1000 = 1 THEN
7           IF 1000 = 1 THEN
8             IF 1000 = 1 THEN
9               IF 1000 = 1 THEN
10                IF 1000 = 1 THEN
11                 IF 1000 = 1 THEN
12                  IF 1000 = 1 THEN
13                   IF 1000 = 1 THEN
14                    IF 1000 = 1 THEN
15                     IF 1000 = 1 THEN
16                      IF 1000 = 1 THEN
17                       IF 1000 = 1 THEN
18                        IF 1000 = 1 THEN
19                         IF 1000 = 1 THEN
20                          IF 1000 = 1 THEN
21                           IF 1000 = 1 THEN
22                            IF 1000 = 1 THEN
23                             IF 1000 = 1 THEN
24                              IF 1000 = 1 THEN
25                               IF 1000 = 1 THEN
26                                IF 1000 = 1 THEN
27                                 IF 1000 = 1 THEN
28                                  IF 1000 = 1 THEN
29                                   IF 1000 = 1 THEN
30                                    IF 1000 = 1 THEN
31                                     IF 1000 = 1 THEN
32                                      IF 1000 = 1 THEN
33                                       IF 1000 = 1 THEN
34                                        IF 1000 = 1 THEN
35                                         IF 1000 = 1 THEN
36                                          IF 1000 = 1 THEN
37                                           IF 1000 = 1 THEN
38                                            IF 1000 = 1 THEN
39                                             IF 1000 = 1 THEN
40                                              IF 1000 = 1 THEN
41                                               IF 1000 = 1 THEN
42                                                IF 1000 = 1 THEN
43                                                 IF 1000 = 1 THEN
44                                                  IF 1000 = 1 THEN
45                                                   IF 1000 = 1 THEN
46                                                    IF 1000 = 1 THEN
47                                                     IF 1000 = 1 THEN
48                                                      IF 1000 = 1 THEN
49                                                       IF 1000 = 1 THEN
50                                                        IF 1000 = 1 THEN
51                                                         IF 1000 = 1 THEN
52                                                          IF 1000 = 1 THEN

```

- 186 -

09/01/76 PAGE 68

LINE CACI SIMSCRIPT II.5 RELEASE AF

```

106 MEGAROLESS
107 LET D.LEVEL.INSPECTION = ADL(ID)
108 CANCEL THE D.LEVEL.INSPECTION
109 DESTROY THE D.LEVEL.INSPECTION
110 IF ATL(ID) = "YES"
111 LET REACH.FAIL.SAFE.LGT = AWFSL(ID)
112 CANCEL THE REACH.FAIL.SAFE.LGT
113 DESTROY THE REACH.FAIL.SAFE.LGT
114 LET ALL(ID) = "NO"
115 REGRADLESS
116 IF FSW(ID) = "YES"
117 LET FAILURE = AF(ID)
118 CANCEL THE FAILURE
119 DESTROY THE FAILURE
120 LET FSW(ID) = "NO"
121 ALWAYS
122 "
123 RETURN
124 END

```

LOCAL VARIABLES OF THIS ROUTINE

ADL	ADL	3	I.1	INTEGER	WORD 7
AF	AF	4	I.3	INTEGER	WORD 9
ATL	ATL	1	J.1	INTEGER	WORD 10
AWFSL	AWFSL	12	K.2	INTEGER	WORD 15
AWFSL	AWFSL	18	K.3	INTEGER	WORD 15
AWFSL	AWFSL	10	L.2	INTEGER	WORD 9
AWFSL	AWFSL	11	M.1	DOUBLE	WORD 5

09/01/76 PAGE 69

```

L14 CALL STASCP111.5 RELEASE AF
1 EVENT REMAIN(IMP)
2 DEFINE TIMEP AS AN INTEGER VARIABLE
3 LET ID = IDREP
4 IF TIMEPENDING(ID) = "YES" OR SHUD.PENDING(ID) = "YES"
5 IF TIMEPENDING(ID) = "YES"
6 CALL INSTALL.MIFICATION
7 LET TIMEPENDING(ID) = "NO"
8 IF TIMEPENDING(ID) = "NO"
9 LET TIMEPENDING(ID) = "NO"
10 LET TIMEPENDING(ID) = "NO"
11 LET TIMEPENDING(ID) = "NO"
12 LET TIMEPENDING(ID) = "NO"
13 LET TIMEPENDING(ID) = "NO"
14 LET TIMEPENDING(ID) = "NO"
15 LET TIMEPENDING(ID) = "NO"
16 LET TIMEPENDING(ID) = "NO"
17 LET TIMEPENDING(ID) = "NO"
18 LET TIMEPENDING(ID) = "NO"
19 LET TIMEPENDING(ID) = "NO"
20 LET TIMEPENDING(ID) = "NO"
21 LET TIMEPENDING(ID) = "NO"
22 LET TIMEPENDING(ID) = "NO"
23 LET TIMEPENDING(ID) = "NO"
24 LET TIMEPENDING(ID) = "NO"
25 LET TIMEPENDING(ID) = "NO"
26 LET TIMEPENDING(ID) = "NO"
27 LET TIMEPENDING(ID) = "NO"
28 LET TIMEPENDING(ID) = "NO"
29 LET TIMEPENDING(ID) = "NO"
30 LET TIMEPENDING(ID) = "NO"
31 LET TIMEPENDING(ID) = "NO"
32 LET TIMEPENDING(ID) = "NO"
33 LET TIMEPENDING(ID) = "NO"
34 LET TIMEPENDING(ID) = "NO"
35 LET TIMEPENDING(ID) = "NO"
36 LET TIMEPENDING(ID) = "NO"
37 LET TIMEPENDING(ID) = "NO"
38 LET TIMEPENDING(ID) = "NO"
39 LET TIMEPENDING(ID) = "NO"
40 LET TIMEPENDING(ID) = "NO"
41 LET TIMEPENDING(ID) = "NO"
42 LET TIMEPENDING(ID) = "NO"
43 LET TIMEPENDING(ID) = "NO"
44 LET TIMEPENDING(ID) = "NO"
45 LET TIMEPENDING(ID) = "NO"
46 LET TIMEPENDING(ID) = "NO"
47 LET TIMEPENDING(ID) = "NO"
48 LET TIMEPENDING(ID) = "NO"
49 LET TIMEPENDING(ID) = "NO"
50 LET TIMEPENDING(ID) = "NO"
51 LET TIMEPENDING(ID) = "NO"
52 LET TIMEPENDING(ID) = "NO"
53 LET TIMEPENDING(ID) = "NO"

```

09/01/76 PAGE 70

LINE CACI SIMSCRIPT II.5 HELFASE AF

```

52 LET CL = (TIME.V-TA1) * MSR(ID) * CGM1(ID)
53 IF TAC GT TA1
54 LET CL = (TAC-TA1)*MSR(ID) + (TIME.V-TAC)*MSR(ID)*CGM1(ID)
55 MEGAROLESS
56 IF CCL LT CL
57 LET CL = CCL + ((CCL-CCL)/MSR(ID))*MSR(ID)
58 MEGAROLESS
59 LET MAX.CRK = CL
60 LET SIM.WED = CL
61 IF 2.CM.FA1S1S(ID) = "YES"
62 LET 2.SIM.WED = 2.SIM.WED + 2.SIM.WED
63 LET TA2 = TIME.A1(2.SIM.WED).REDUCTION
64 LET CL = (TIME.V-TA2) * MSR(ID) * CGM1(ID)
65 IF TAC GT TA2
66 LET CL = (TAC-TA2)*MSR(ID) + (TIME.V-TAC)*MSR(ID)*CGM1(ID)
67 MEGAROLESS
68 IF CCL LT CL
69 LET CL = CCL + ((CCL-CCL)/MSR(ID))*MSR(ID)
70 MEGAROLESS
71 IF MAX.CRK LT CL
72 LET MAX.CRK = CL
73 MEGAROLESS
74 ADD CL TO STR.WED
75 IF 3.CM.EXISTS(ID) = "YES"
76 LET 3.SIM.WED = 3.SIM.WED + 3.SIM.WED
77 LET TA3 = TIME.A1(3.SIM.WED).REDUCTION
78 LET CL = (TIME.V-TA3) * MSR(ID) * CGM1(ID)
79 IF TAC GT TA3
80 LET CL = (TAC-TA3)*MSR(ID) + (TIME.V-TAC)*MSR(ID)*CGM1(ID)
81 MEGAROLESS
82 IF CCL LT CL
83 LET CL = CCL + ((CCL-CCL)/MSR(ID))*MSR(ID)
84 MEGAROLESS
85 IF MAX.CRK LT CL
86 LET MAX.CRK = CL
87 MEGAROLESS
88 ADD CL TO STR.WED
89 MEGAROLESS
90 LET PUFF1.SIM.WED = PUFF1.SIM.WED + STR.WED
91 LET PUFF1.CRK = MAX.CRK + 1.0 * MAX.CRK * ANCL(LIL)
92 IF PUFF1.CRK GT CCL
93 LET PUFF1.CRK = CCL + ((PUFF1.CRK-CCL)/N1.MEAN) * V2.MEAN
94 MEGAROLESS
95 IF MAX.CRK GT CCL
96 LET PUFF1.CRK = MAX.CRK + V2.MEAN * ANCL(LIL)
97 MEGAROLESS
98 IF 2.1.CM + SIM.WED = MAX.CRK GT PUFF1.CRK
99 SOMEONE AT FAC-BASE INSPECTION, PUFF1.CRK
100 SOMEONE AT FAC-BASE INSPECTION, PUFF1.CRK
101 LET CRK.FA1S1S = TIME.V
102 JUMP AMED
103 MEGAROLESS
104
105
106

```

09/01/76 PAGE 71

LT-4 CACI STASCH-1 11.5 RELEASE RE

```

107 IF FLEET,STR,RED GT (PSAF,LCI,5,0) * (INDM-1,NUM,OF,RETIME-2,NUM,OF,CRASH)
108 SCHEDULE AN INCREASE,INSPECTION,FREQUENCY NDA
109 SCHEDULE AN IMMEDIATE,FLEET,INSPECTION NDA
110 LET TIME,OF,RETIME = TIME,V
111 REMARK,LESS
112 IF-
113 **
114 LET COST,OF,REPAIRS = CUST,OF,REPAIRS + FLXIT,COST
115 IF TIME,OF,RETIME = "N" AND DEC,OF,RETIME,OF,SCH = "NO"
116 IF IS,FAILURE = "YES" AND TIME,V LT START,TEST + ACTUAL,AVG,PAT,LIFE /
117 TEST,ACCEL,FACT OR IS,FAILURE = "YES" AND TIME,V GT START,TEST +
118 ACTUAL,AVG,PAT,LIFE / TEST,ACCEL,FACT + LPAQ,TIME OR IS,FAILURE = "NO"
119 IF TIME,OF,RETIME = "YES" AND TIME,V LT START,TEST + ACTUAL,AVG,PAT,LIFE /
120 TEST,ACCEL,FACT OR IS,FAILURE = "YES" AND TIME,V GT START,TEST +
121 ACTUAL,AVG,PAT,LIFE / TEST,ACCEL,FACT + LPAQ,TIME OR IS,FAILURE = "NO"
122 SCHEDULE A REVISION,OF,RETIME NDA
123 LET RE,OF,RETIME,OF,SCH = "YES"
124 REMARK,LESS
125 REMARK,LESS
126 REMARK,LESS
127 **
128 LET AAFL = ACTUAL,AVG,PAT,LIFE
129 IF I LE FOUR
130 LET AAFL = 1AAFL
131 REMARK,LESS
132 CALL PAT,OF,LIFE,SCATTER(AAFL,5) YIELDING FIRST,LIFE, SECOND,LIFE AND
133 THIRD,LIFE
134 LET 1,STRENGTH,REDUCTION = ASK(10)
135 LET 2,STRENGTH,REDUCTION = ASK(10)
136 LET 3,STRENGTH,REDUCTION = ASK(10)
137 IF 2,STRENGTH,REDUCTION = "YES"
138 DESTROY THE 2,STRENGTH,REDUCTION
139 IF 3,STRENGTH,REDUCTION = "YES"
140 DESTROY THE 3,STRENGTH,REDUCTION
141 JUMP AHEAD
142 **
143 IF 3,STRENGTH,REDUCTION = "YES"
144 IF TIME,AC3,STRENGTH,REDUCTION - TIME,V LT FIRST,LIFE
145 LET THIRD,LIFE = SECOND,LIFE
146 LET SECOND,LIFE = FIRST,LIFE
147 LET FIRST,LIFE = TIME,AC3,STRENGTH,REDUCTION - TIME,V
148 CANCEL THE 3,STRENGTH,REDUCTION
149 DESTROY THE 3,STRENGTH,REDUCTION
150 JUMP AHEAD
151 **
152 IF TIME,AC3,STRENGTH,REDUCTION - TIME,V LT SECOND,LIFE
153 LET THIRD,LIFE = SECOND,LIFE
154 LET SECOND,LIFE = TIME,AC3,STRENGTH,REDUCTION - TIME,V
155 CANCEL THE 3,STRENGTH,REDUCTION
156 DESTROY THE 3,STRENGTH,REDUCTION
157 JUMP AHEAD
158 **
159 LET THIRD,LIFE = TIME,AC3,STRENGTH,REDUCTION - TIME,V
160 CANCEL THE 3,STRENGTH,REDUCTION

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDG

09/01/76 PAGE 72

LINE CACI SCRIPT II.5 RELEASE 8F

```

160 DESTROY THE 3.STRENGTH.REDUCTION
161 JUMP AHEAD
162 OTHERWISE
163 LET THIRD.LIFE = USAGE.LIFE
164 JUMP AHEAD
165 OTHERWISE
166 IF 2.CR.EXISTS(10) = "NS"
167 IF 3.CR.EXISTS(10) = "NS"
168 IF TIME.A(3.STRENGTH.REDUCTION) - TIME.V LT FIRST.LIFE
169 LET THIRD.LIFE = FIRST.LIFE
170 LET SECOND.LIFE = TIME.A(3.STRENGTH.REDUCTION) - TIME.V
171 LET FIRST.LIFE = TIME.A(2.STRENGTH.REDUCTION) - TIME.V
172 CANCEL THE 2.STRENGTH.REDUCTION
173 DESTROY THE 2.STRENGTH.REDUCTION
174 CANCEL THE 3.STRENGTH.REDUCTION
175 DESTROY THE 3.STRENGTH.REDUCTION
176 JUMP AHEAD
177 OTHERWISE
178 IF TIME.A(2.STRENGTH.REDUCTION) - TIME.V LT FIRST.LIFE
179 LET THIRD.LIFE = TIME.A(3.STRENGTH.REDUCTION) - TIME.V
180 LET SECOND.LIFE = FIRST.LIFE
181 LET FIRST.LIFE = TIME.A(2.STRENGTH.REDUCTION) - TIME.V
182 CANCEL THE 2.STRENGTH.REDUCTION
183 DESTROY THE 2.STRENGTH.REDUCTION
184 CANCEL THE 3.STRENGTH.REDUCTION
185 DESTROY THE 3.STRENGTH.REDUCTION
186 JUMP AHEAD
187 OTHERWISE
188 LET SECOND.LIFE = TIME.A(2.STRENGTH.REDUCTION)
189 LET THIRD.LIFE = TIME.A(3.STRENGTH.REDUCTION)
190 CANCEL THE 2.STRENGTH.REDUCTION
191 DESTROY THE 2.STRENGTH.REDUCTION
192 CANCEL THE 3.STRENGTH.REDUCTION
193 DESTROY THE 3.STRENGTH.REDUCTION
194 JUMP AHEAD
195 OTHERWISE
196 IF TIME.A(2.STRENGTH.REDUCTION) - TIME.V LT FIRST.LIFE
197 LET THIRD.LIFE = USAGE.LIFE
198 LET SECOND.LIFE = FIRST.LIFE
199 LET FIRST.LIFE = TIME.A(2.STRENGTH.REDUCTION)
200 CANCEL THE 2.STRENGTH.REDUCTION
201 DESTROY THE 2.STRENGTH.REDUCTION
202 JUMP AHEAD
203 OTHERWISE
204 LET SECOND.LIFE = TIME.A(2.STRENGTH.REDUCTION)
205 JUMP AHEAD
206 OTHERWISE
207 LET SECOND.LIFE = USAGE.LIFE
208 LET THIRD.LIFE = USAGE.LIFE
209 HERE
210 LET 1.CR.EXISTS(10) = "NN"
211 LET 2.CR.EXISTS(10) = "NN"
212 LET 3.CR.EXISTS(10) = "NN"

```

09/01/76 PAGE 73

LINE (ACI) SCRIPT IL-5 RELEASE AF

```

213 IF FIRSTLIFE LT 1ST OR COLLEXTS(10) = "NS" OR SO.SCH(10) = "YES"
214   SCHEDULE A 1-STEP GTH-REDUCTION(10) AT TIME.V + FIRSTLIFE
215   LET A1SCH(10) = 1-STEP GTH-REDUCTION
216   LET 1-COLLEXTS(10) = "NS"
217 IF SECONDALIFE LT 1ST OR COLLEXTS(10) = "NS" OR SO.SCH(10) = "YES"
218   SCHEDULE A 2-STEP GTH-REDUCTION(10) AT TIME.V + SECONDALIFE
219   LET A2SCH(10) = 2-STEP GTH-REDUCTION
220   LET 2-COLLEXTS(10) = "NS"
221 IF THIRDLIFE LT 1ST OR COLLEXTS(10) = "NS" OR SO.SCH(10) = "YES"
222   SCHEDULE A 3-STEP GTH-REDUCTION(10) AT TIME.V + THIRDLIFE
223   LET A3SCH(10) = 3-STEP GTH-REDUCTION
224   LET 3-COLLEXTS(10) = "NS"
225   REMARKLESS
226   REMARKLESS
227   REMARKLESS
228   REMARKLESS
229   IF 1P(10) = "YES"
230     LET 1-LITE = A1E(10)
231     LET 1P(10) = "NO"
232   CANCEL THE 1-LITE
233   DESTROY THE 1-LITE
234   IF 1P(10) = "YES"
235     LET 2-LITE = A2E(10)
236     LET 1P(10) = "NO"
237   CANCEL THE 2-LITE
238   DESTROY THE 2-LITE
239   IF 1P(10) = "YES"
240     LET 3-LITE = A3E(10)
241     LET 1P(10) = "NO"
242   CANCEL THE 3-LITE
243   DESTROY THE 3-LITE
244   ALWAYS
245   ALWAYS
246   ALWAYS
247   IF LIME = "YES"
248     FOR I = 1 TO 100 (LIME)
249       IF I = 100 (LIME)
250         SKIP 1 OUTPUT LINE
251         PRINT A LIME WITH ID, TIME.V-6000, TIME(10-PLANE(10)), FIRSTLIFE,
252           SECONDALIFE, THIRDLIFE AS FOLLOWS
253           AVE 100, 100 WAS ALL DEFECTS REPAIRED AT ***** FLIGHT HOURS
254           1ST CHECK INITIATION, PROJECTED AT ***** FLIGHT HOURS
255           2ND CHECK INITIATION, PROJECTED AT ***** FLIGHT HOURS
256           3RD CHECK INITIATION, PROJECTED AT ***** FLIGHT HOURS
257       LEAVE
258     FLSH
259     LIMP
260     ALWAYS
261   ** CANCEL SCHEDULED INSPECTIONS ON THIS AIRPORT
262   ** CALL CANCEL-SCHEM-RED-INSPECTIONS

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

09/01/76 PAGE 74

LIVE CACI SCRIPT II.5 RELEASE 44

202 RETURN
203 END

LOCAL VARIABLES IN THIS ROUTINE

AAFL	DOUBLE	WORD 25	CCL	DOUBLE	WORD 9
CL	DOUBLE	WORD 13	FIRST.LIFE	DOUBLE	WORD 27
WORDS.TU.C	DOUBLE	WORD 7	I.1	INTEGER	WORD 37
I.2	INTEGER	WORD 39	I.3	INTEGER	WORD 39
INFP	INTEGER	WORD 1	I.1	INTEGER	WORD 40
K.1	INTEGER	WORD 42	K.2	INTEGER	WORD 43
K.3	INTEGER	WORD 44	K.4	INTEGER	WORD 45
L.4	INTEGER	WORD 35	AAFL.CK	DOUBLE	WORD 15
M.1	INTEGER	WORD 41	OTI.CK	DOUBLE	WORD 23
SPC IND.LIF	DOUBLE	WORD 29	AST	DOUBLE	WORD 17
TAC	DOUBLE	WORD 5	ST.460	DOUBLE	WORD 11
T82	DOUBLE	WORD 19	T81	DOUBLE	WORD 21
THIRD.LIFE	DOUBLE	WORD 31	T83	DOUBLE	WORD 21

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

09/01/76 PAGE 75

```

11 5 CACTI SYNOPSIS II.5 RELEASE ME
1  EVALUATE INCREMENTAL INCREASE (IOTI)
2  DEFINE IOTI AS AN INTEGER VARIABLE
3  LET IOTI = IOTI
4  IF IOTI = 0 THEN LET IOTI = 1
5  LET IOTI = IOTI + 1
6  LET IOTI = IOTI + 1
7  LET IOTI = IOTI + 1
8  LET IOTI = IOTI + 1
9  LET IOTI = IOTI + 1
10 LET IOTI = IOTI + 1
11 LET IOTI = IOTI + 1
12 LET IOTI = IOTI + 1
13 LET IOTI = IOTI + 1
14 LET IOTI = IOTI + 1
15 LET IOTI = IOTI + 1
16 LET IOTI = IOTI + 1
17 LET IOTI = IOTI + 1
18 LET IOTI = IOTI + 1
19 LET IOTI = IOTI + 1
20 LET IOTI = IOTI + 1
21 LET IOTI = IOTI + 1
22 LET IOTI = IOTI + 1
23 LET IOTI = IOTI + 1
24 LET IOTI = IOTI + 1
25 LET IOTI = IOTI + 1
26 LET IOTI = IOTI + 1
27 LET IOTI = IOTI + 1
28 LET IOTI = IOTI + 1
29 LET IOTI = IOTI + 1
30 LET IOTI = IOTI + 1
31 LET IOTI = IOTI + 1
32 LET IOTI = IOTI + 1
33 LET IOTI = IOTI + 1
34 LET IOTI = IOTI + 1
35 LET IOTI = IOTI + 1
36 LET IOTI = IOTI + 1
37 LET IOTI = IOTI + 1
38 LET IOTI = IOTI + 1
39 LET IOTI = IOTI + 1
40 LET IOTI = IOTI + 1
41 LET IOTI = IOTI + 1
42 LET IOTI = IOTI + 1
43 LET IOTI = IOTI + 1
44 LET IOTI = IOTI + 1
45 LET IOTI = IOTI + 1
46 LET IOTI = IOTI + 1
47 LET IOTI = IOTI + 1
48 LET IOTI = IOTI + 1
49 LET IOTI = IOTI + 1
50 LET IOTI = IOTI + 1
51 LET IOTI = IOTI + 1
52 LET IOTI = IOTI + 1
53 LET IOTI = IOTI + 1
54 LET IOTI = IOTI + 1
55 LET IOTI = IOTI + 1
56 LET IOTI = IOTI + 1
57 LET IOTI = IOTI + 1
58 LET IOTI = IOTI + 1
59 LET IOTI = IOTI + 1
60 LET IOTI = IOTI + 1
61 LET IOTI = IOTI + 1
62 LET IOTI = IOTI + 1
63 LET IOTI = IOTI + 1
64 LET IOTI = IOTI + 1
65 LET IOTI = IOTI + 1
66 LET IOTI = IOTI + 1
67 LET IOTI = IOTI + 1
68 LET IOTI = IOTI + 1
69 LET IOTI = IOTI + 1
70 LET IOTI = IOTI + 1
71 LET IOTI = IOTI + 1
72 LET IOTI = IOTI + 1
73 LET IOTI = IOTI + 1
74 LET IOTI = IOTI + 1
75 LET IOTI = IOTI + 1
76 LET IOTI = IOTI + 1
77 LET IOTI = IOTI + 1
78 LET IOTI = IOTI + 1
79 LET IOTI = IOTI + 1
80 LET IOTI = IOTI + 1
81 LET IOTI = IOTI + 1
82 LET IOTI = IOTI + 1
83 LET IOTI = IOTI + 1
84 LET IOTI = IOTI + 1
85 LET IOTI = IOTI + 1
86 LET IOTI = IOTI + 1
87 LET IOTI = IOTI + 1
88 LET IOTI = IOTI + 1
89 LET IOTI = IOTI + 1
90 LET IOTI = IOTI + 1
91 LET IOTI = IOTI + 1
92 LET IOTI = IOTI + 1
93 LET IOTI = IOTI + 1
94 LET IOTI = IOTI + 1
95 LET IOTI = IOTI + 1
96 LET IOTI = IOTI + 1
97 LET IOTI = IOTI + 1
98 LET IOTI = IOTI + 1
99 LET IOTI = IOTI + 1
100 LET IOTI = IOTI + 1

```

LOCAL VARIABLES IN THIS ROUTINE

IOTI = 0

IOTI

09/01/76 PAGE 76

LIVE CACI SCRIPT II.5 RELEASE AF

```

1  EVFMT INCREASE,INSPECTION,FREQUENCY
2  **
3  ** INCREASES THE FREQUENCY OF THE LOWEST LEVEL INTERNAL AND EXTERNAL INSPECTIONS
4  ** THE SP MAY BE MAY NOT BE THE SAME LEVEL OF INSPECTION
5  ** THE A-LEVEL AND N-LEVEL FREQUENCIES ARE NEVER CHANGED
6  **
7  LET FILE1,STR,RED = 0.0
8  IF ARCD(LIL)*S.FREQ.CMG GE IARCD(LIL-1) OR ARCD(LEL)*S.FREQ.CMG GE IARCD(LEL-1)
9  IF ARCD(LIL)*S.FREQ.CMG GE IARCD(LIL-1)
10 LET DIFF = ARCD(LIL) - ARCD(LIL-1) * S.FREQ.CMG
11 LET ARCD(LIL) = ARCD(LIL) + S.FREQ.CMG
12 ALWAYS
13 IF LIL NE LEL
14 IF ARCD(LEL)*S.FREQ.CMG GE IARCD(LEL-1)
15 LET ARCD(LEL) = ARCD(LEL-1) + S.FREQ.CMG
16 ALWAYS
17 IF LTIMO = "YES"
18 SETP 1,INITIAL LINE
19 5 LINES WITH ARCD(3), ARCD(4) AS FOLLOWS
20 INSPECTION INTERVAL DECREASE IMPLEMENTED
    C-LEVEL INTERVAL WITH ***** HOURS
    D-LEVEL INTERVAL WITH ***** MINUTES
    ALWAYS
21 LET CINSL = ARCD(3)
22 LET DINSL = ARCD(4)
23 LET NINSL = ARCD(4)
24 LET NINSL = NINSL + 1
25 IF NINSL GT 14
26 JUMP AHEAD
27 OTHERWISE
28 LET SC(NINSL) = ARCD(3)
29 LET SC(NINSL) = ARCD(4)
30 MEME
31 FROM EVENT AIRCRAFT IN ACTIVE,FILE1
32 ON
33 LET IN = TAIL.IN
34 LET C.INTERVAL(IN) = ARCD(3)
35 LET D.INTERVAL(IN) = ARCD(4)
36 IF LIL = 4
37 LET N.LEVEL.INSPECTION = ARCD(IN)
38 CANCEL THE N.LEVEL.INSPECTION
39 IF TIME-ARCD.LEVEL.INSPECTION - TIME.V LT DIFF
40 RESCHEDULE THE D.LEVEL.INSPECTION(IN) AND
41 JUMP AHEAD
42 OTHERWISE
43 RESCHEDULE THE D.LEVEL.INSPECTION(IN) AT TIME,ARCD.LEVEL.INSPECTION - DIFF
44 JUMP AHEAD
45 OTHERWISE
46 IF INSP,SC(IN) = "YES"
47 LET C.LEVEL.INSPECTION = ARCD(IN)
48 CANCEL THE C.LEVEL.INSPECTION
49 IF TIME,ARCD.LEVEL.INSPECTION - TIME.V LT DIFF
50

```

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

92/01/76
part
77

[illegible][illegible]

09/01/76 PAGE 7A

```

LINE CACI SCHEDULED INSPECTION RELEASE AF
1 EVENT IMMEDIATE FLEET INSPECTION
2 **
3 ** THIS EVENT REPRESENTS AN IMMEDIATE FLEET WIDE INSPECTION CAUSED BY FINDING A
4 ** DEFECT CONSIDERED TO BE/ADVERSE TO DEPEND ON SCHEDULED INSPECTIONS
5 **
6 ** THIS EVENT IS ALWAYS ENTERED BY THE FIRST INCREASE INSPECTION FREQUENCY
7 **
8 IF LTMS = "YES"
9   SETD 1 ENTER IT LINE
10   PRINT 1 LINE AS FOLLOWS
11   FLEET WIDE SPECIAL INSPECTION PERFORMED
12   ALWAYS
13   LET FCL = FCL + 1
14   LET MSIC = MSIC + 1
15   LET FLEET = FLEET + 1
16   LET FLEET = FLEET + 1
17   LET LIST = 0.0
18   LET I = 1
19   IF LTMS = "YES"
20     LET I = 1
21     LET I = 1
22   IF I = 1
23     LET LIST = 1.0
24     LET I = 1
25     LET I = 1
26     LET I = 1
27     LET I = 1
28     LET I = 1
29     LET I = 1
30     LET I = 1
31     LET I = 1
32     LET I = 1
33     LET I = 1
34     LET I = 1
35     LET I = 1
36     LET I = 1
37     LET I = 1
38     LET I = 1
39     LET I = 1
40     LET I = 1
41     LET I = 1
42     LET I = 1
43     LET I = 1
44     LET I = 1
45     LET I = 1
46     LET I = 1
47     LET I = 1
48     LET I = 1
49     LET I = 1
50     LET I = 1
51     LET I = 1
52     LET I = 1
53     LET I = 1
54     LET I = 1
55     LET I = 1
56     LET I = 1
57     LET I = 1
58     LET I = 1
59     LET I = 1
60     LET I = 1
61     LET I = 1
62     LET I = 1
63     LET I = 1
64     LET I = 1
65     LET I = 1
66     LET I = 1
67     LET I = 1
68     LET I = 1
69     LET I = 1
70     LET I = 1
71     LET I = 1
72     LET I = 1
73     LET I = 1
74     LET I = 1
75     LET I = 1
76     LET I = 1
77     LET I = 1
78     LET I = 1
79     LET I = 1
80     LET I = 1
81     LET I = 1
82     LET I = 1
83     LET I = 1
84     LET I = 1
85     LET I = 1
86     LET I = 1
87     LET I = 1
88     LET I = 1
89     LET I = 1
90     LET I = 1
91     LET I = 1
92     LET I = 1
93     LET I = 1
94     LET I = 1
95     LET I = 1
96     LET I = 1
97     LET I = 1
98     LET I = 1
99     LET I = 1
100    LET I = 1

```

***** FLIGHT NUMS

09/01/76 PAGE 79

LINE CACI SIMSCRIPT ILS RELEASE AF

```

52 LET SIMSCRIPT_RELEASE = AISC(10)
53 LET TAT = TIME-ARC-SIMSCRIPT_RELEASE(10)
54 LET CL = (TIME-V-TAT) * 10 * (CGL(10))
55 IF TAT GT 100
56 LET CL = (TIME-V-TAT) * 10 * (CGL(10))
57 +CGL(10)
58 IF CL GT 100
59 LET CL = CL * (CGL(10)/100)
60 IF CL GT 100
61 LET CL = 100
62 LET TAT = 100
63 LET TAT = 100
64 LET TAT = 100
65 LET TAT = 100
66 LET TAT = 100
67 LET TAT = 100
68 LET TAT = 100
69 LET TAT = 100
70 LET TAT = 100
71 LET TAT = 100
72 LET TAT = 100
73 LET TAT = 100
74 LET TAT = 100
75 LET TAT = 100
76 LET TAT = 100
77 LET TAT = 100
78 LET TAT = 100
79 LET TAT = 100
80 LET TAT = 100
81 LET TAT = 100
82 LET TAT = 100
83 LET TAT = 100
84 LET TAT = 100
85 LET TAT = 100
86 LET TAT = 100
87 LET TAT = 100
88 LET TAT = 100
89 LET TAT = 100
90 LET TAT = 100
91 LET TAT = 100
92 LET TAT = 100
93 LET TAT = 100
94 LET TAT = 100
95 LET TAT = 100
96 LET TAT = 100
97 LET TAT = 100
98 LET TAT = 100
99 LET TAT = 100
100 LET TAT = 100

```

10/01/75 PAGE 20

LIN CACI SUGGESTION 11.5 RELEASE AF

```

103 **
104 **
105 **
106 **
107 **
108 **
109 **
110 **
111 **
112 **
113 **
114 **
115 **
116 **
117 **
118 **
119 **
120 **
121 **
122 **
123 **
124 **
125 **
126 **
127 **
128 **
129 **
130 **
131 **
132 **
133 **
134 **
135 **
136 **
137 **
138 **
139 **
140 **
141 **
142 **
143 **
144 **
145 **
146 **
147 **
148 **
149 **
150 **
151 **
152 **
153 **
154 **
155 **
156 **
157 **
158 **
159 **
160 **
161 **
162 **
163 **
164 **
165 **
166 **
167 **
168 **
169 **
170 **
171 **
172 **
173 **
174 **
175 **
176 **
177 **
178 **
179 **
180 **
181 **
182 **
183 **
184 **
185 **
186 **
187 **
188 **
189 **
190 **
191 **
192 **
193 **
194 **
195 **
196 **
197 **
198 **
199 **
200 **
201 **
202 **
203 **
204 **
205 **
206 **
207 **
208 **
209 **
210 **
211 **
212 **
213 **
214 **
215 **
216 **
217 **
218 **
219 **
220 **
221 **
222 **
223 **
224 **
225 **
226 **
227 **
228 **
229 **
230 **
231 **
232 **
233 **
234 **
235 **
236 **
237 **
238 **
239 **
240 **
241 **
242 **
243 **
244 **
245 **
246 **
247 **
248 **
249 **
250 **
251 **
252 **
253 **
254 **
255 **
256 **
257 **
258 **
259 **
260 **
261 **
262 **
263 **
264 **
265 **
266 **
267 **
268 **
269 **
270 **
271 **
272 **
273 **
274 **
275 **
276 **
277 **
278 **
279 **
280 **
281 **
282 **
283 **
284 **
285 **
286 **
287 **
288 **
289 **
290 **
291 **
292 **
293 **
294 **
295 **
296 **
297 **
298 **
299 **
300 **
301 **
302 **
303 **
304 **
305 **
306 **
307 **
308 **
309 **
310 **
311 **
312 **
313 **
314 **
315 **
316 **
317 **
318 **
319 **
320 **
321 **
322 **
323 **
324 **
325 **
326 **
327 **
328 **
329 **
330 **
331 **
332 **
333 **
334 **
335 **
336 **
337 **
338 **
339 **
340 **
341 **
342 **
343 **
344 **
345 **
346 **
347 **
348 **
349 **
350 **
351 **
352 **
353 **
354 **
355 **
356 **
357 **
358 **
359 **
360 **
361 **
362 **
363 **
364 **
365 **
366 **
367 **
368 **
369 **
370 **
371 **
372 **
373 **
374 **
375 **
376 **
377 **
378 **
379 **
380 **
381 **
382 **
383 **
384 **
385 **
386 **
387 **
388 **
389 **
390 **
391 **
392 **
393 **
394 **
395 **
396 **
397 **
398 **
399 **
400 **
401 **
402 **
403 **
404 **
405 **
406 **
407 **
408 **
409 **
410 **
411 **
412 **
413 **
414 **
415 **
416 **
417 **
418 **
419 **
420 **
421 **
422 **
423 **
424 **
425 **
426 **
427 **
428 **
429 **
430 **
431 **
432 **
433 **
434 **
435 **
436 **
437 **
438 **
439 **
440 **
441 **
442 **
443 **
444 **
445 **
446 **
447 **
448 **
449 **
450 **
451 **
452 **
453 **
454 **
455 **
456 **
457 **
458 **
459 **
460 **
461 **
462 **
463 **
464 **
465 **
466 **
467 **
468 **
469 **
470 **
471 **
472 **
473 **
474 **
475 **
476 **
477 **
478 **
479 **
480 **
481 **
482 **
483 **
484 **
485 **
486 **
487 **
488 **
489 **
490 **
491 **
492 **
493 **
494 **
495 **
496 **
497 **
498 **
499 **
500 **
501 **
502 **
503 **
504 **
505 **
506 **
507 **
508 **
509 **
510 **
511 **
512 **
513 **
514 **
515 **
516 **
517 **
518 **
519 **
520 **
521 **
522 **
523 **
524 **
525 **
526 **
527 **
528 **
529 **
530 **
531 **
532 **
533 **
534 **
535 **
536 **
537 **
538 **
539 **
540 **
541 **
542 **
543 **
544 **
545 **
546 **
547 **
548 **
549 **
550 **
551 **
552 **
553 **
554 **
555 **
556 **
557 **
558 **
559 **
560 **
561 **
562 **
563 **
564 **
565 **
566 **
567 **
568 **
569 **
570 **
571 **
572 **
573 **
574 **
575 **
576 **
577 **
578 **
579 **
580 **
581 **
582 **
583 **
584 **
585 **
586 **
587 **
588 **
589 **
590 **
591 **
592 **
593 **
594 **
595 **
596 **
597 **
598 **
599 **
600 **
601 **
602 **
603 **
604 **
605 **
606 **
607 **
608 **
609 **
610 **
611 **
612 **
613 **
614 **
615 **
616 **
617 **
618 **
619 **
620 **
621 **
622 **
623 **
624 **
625 **
626 **
627 **
628 **
629 **
630 **
631 **
632 **
633 **
634 **
635 **
636 **
637 **
638 **
639 **
640 **
641 **
642 **
643 **
644 **
645 **
646 **
647 **
648 **
649 **
650 **
651 **
652 **
653 **
654 **
655 **
656 **
657 **
658 **
659 **
660 **
661 **
662 **
663 **
664 **
665 **
666 **
667 **
668 **
669 **
670 **
671 **
672 **
673 **
674 **
675 **
676 **
677 **
678 **
679 **
680 **
681 **
682 **
683 **
684 **
685 **
686 **
687 **
688 **
689 **
690 **
691 **
692 **
693 **
694 **
695 **
696 **
697 **
698 **
699 **
700 **
701 **
702 **
703 **
704 **
705 **
706 **
707 **
708 **
709 **
710 **
711 **
712 **
713 **
714 **
715 **
716 **
717 **
718 **
719 **
720 **
721 **
722 **
723 **
724 **
725 **
726 **
727 **
728 **
729 **
730 **
731 **
732 **
733 **
734 **
735 **
736 **
737 **
738 **
739 **
740 **
741 **
742 **
743 **
744 **
745 **
746 **
747 **
748 **
749 **
750 **
751 **
752 **
753 **
754 **
755 **
756 **
757 **
758 **
759 **
760 **
761 **
762 **
763 **
764 **
765 **
766 **
767 **
768 **
769 **
770 **
771 **
772 **
773 **
774 **
775 **
776 **
777 **
778 **
779 **
780 **
781 **
782 **
783 **
784 **
785 **
786 **
787 **
788 **
789 **
790 **
791 **
792 **
793 **
794 **
795 **
796 **
797 **
798 **
799 **
800 **
801 **
802 **
803 **
804 **
805 **
806 **
807 **
808 **
809 **
810 **
811 **
812 **
813 **
814 **
815 **
816 **
817 **
818 **
819 **
820 **
821 **
822 **
823 **
824 **
825 **
826 **
827 **
828 **
829 **
830 **
831 **
832 **
833 **
834 **
835 **
836 **
837 **
838 **
839 **
840 **
841 **
842 **
843 **
844 **
845 **
846 **
847 **
848 **
849 **
850 **
851 **
852 **
853 **
854 **
855 **
856 **
857 **
858 **
859 **
860 **
861 **
862 **
863 **
864 **
865 **
866 **
867 **
868 **
869 **
870 **
871 **
872 **
873 **
874 **
875 **
876 **
877 **
878 **
879 **
880 **
881 **
882 **
883 **
884 **
885 **
886 **
887 **
888 **
889 **
890 **
891 **
892 **
893 **
894 **
895 **
896 **
897 **
898 **
899 **
900 **
901 **
902 **
903 **
904 **
905 **
906 **
907 **
908 **
909 **
910 **
911 **
912 **
913 **
914 **
915 **
916 **
917 **
918 **
919 **
920 **
921 **
922 **
923 **
924 **
925 **
926 **
927 **
928 **
929 **
930 **
931 **
932 **
933 **
934 **
935 **
936 **
937 **
938 **
939 **
940 **
941 **
942 **
943 **
944 **
945 **
946 **
947 **
948 **
949 **
950 **
951 **
952 **
953 **
954 **
955 **
956 **
957 **
958 **
959 **
960 **
961 **
962 **
963 **
964 **
965 **
966 **
967 **
968 **
969 **
970 **
971 **
972 **
973 **
974 **
975 **
976 **
977 **
978 **
979 **
980 **
981 **
982 **
983 **
984 **
985 **
986 **
987 **
988 **
989 **
990 **
991 **
992 **
993 **
994 **
995 **
996 **
997 **
998 **
999 **
1000 **

```

09/01/76 PAGE 41

LINE CACI SIMSCRIPT II.5 RELEASE AF

```

155 LET POT.CRK = CCL + ((POT.CRK-CCL)/M1.MEAN) * M2.MEAN
156 REGARDLESS
157 IF MAX.CRK GT CCL
158 LET POT.CRK = MAX.CRK + 42.4444 * ARCO(L10)
159 REGARDLESS
160 IF POT.STA.REO LT POT.CRK + STW.REO - MAX.CRK
161 LET POT.STA.REO = POT.CRK + STW.REO - MAX.CRK
162 REGARDLESS
163 REGARDLESS
164 REGARDLESS
165 LOOP
166 **
167 ** EFFECTS FOUND DURING THIS INSPECTION CAN CAUSE AN ADDITIONAL INCREASE IN THE
168 ** FREQUENCY OF NORMALLY SCHEDULED INSPECTIONS
169 **
170 IF POT.STA.REO GT FRF.LST/2.0
171 SCHEDULE AN INCREASE IN INSPECTION FREQUENCY FOR
172 REGARDLESS
173 **
174 RETURN
175 END

```

LOCAL VARIABLES OF THIS ROUTINE

AREA	DATA	TYPE	VALUE	ADDRESS
CCL	0.000000	DOUBLE	0.000000	0000 15
L1	0.000000	DOUBLE	0.000000	0000 25
L2	0.000000	DOUBLE	0.000000	0000 35
L3	0.000000	DOUBLE	0.000000	0000 45
L4	0.000000	DOUBLE	0.000000	0000 55
L5	0.000000	DOUBLE	0.000000	0000 65
L6	0.000000	DOUBLE	0.000000	0000 75
L7	0.000000	DOUBLE	0.000000	0000 85
L8	0.000000	DOUBLE	0.000000	0000 95
L9	0.000000	DOUBLE	0.000000	0001 05
L10	0.000000	DOUBLE	0.000000	0001 15
L11	0.000000	DOUBLE	0.000000	0001 25
L12	0.000000	DOUBLE	0.000000	0001 35
L13	0.000000	DOUBLE	0.000000	0001 45
L14	0.000000	DOUBLE	0.000000	0001 55
L15	0.000000	DOUBLE	0.000000	0001 65
L16	0.000000	DOUBLE	0.000000	0001 75
L17	0.000000	DOUBLE	0.000000	0001 85
L18	0.000000	DOUBLE	0.000000	0001 95
L19	0.000000	DOUBLE	0.000000	0002 05
L20	0.000000	DOUBLE	0.000000	0002 15
L21	0.000000	DOUBLE	0.000000	0002 25
L22	0.000000	DOUBLE	0.000000	0002 35
L23	0.000000	DOUBLE	0.000000	0002 45
L24	0.000000	DOUBLE	0.000000	0002 55
L25	0.000000	DOUBLE	0.000000	0002 65
L26	0.000000	DOUBLE	0.000000	0002 75
L27	0.000000	DOUBLE	0.000000	0002 85
L28	0.000000	DOUBLE	0.000000	0002 95
L29	0.000000	DOUBLE	0.000000	0003 05
L30	0.000000	DOUBLE	0.000000	0003 15
L31	0.000000	DOUBLE	0.000000	0003 25
L32	0.000000	DOUBLE	0.000000	0003 35
L33	0.000000	DOUBLE	0.000000	0003 45
L34	0.000000	DOUBLE	0.000000	0003 55
L35	0.000000	DOUBLE	0.000000	0003 65
L36	0.000000	DOUBLE	0.000000	0003 75
L37	0.000000	DOUBLE	0.000000	0003 85
L38	0.000000	DOUBLE	0.000000	0003 95
L39	0.000000	DOUBLE	0.000000	0004 05
L40	0.000000	DOUBLE	0.000000	0004 15
L41	0.000000	DOUBLE	0.000000	0004 25
L42	0.000000	DOUBLE	0.000000	0004 35
L43	0.000000	DOUBLE	0.000000	0004 45
L44	0.000000	DOUBLE	0.000000	0004 55
L45	0.000000	DOUBLE	0.000000	0004 65
L46	0.000000	DOUBLE	0.000000	0004 75
L47	0.000000	DOUBLE	0.000000	0004 85
L48	0.000000	DOUBLE	0.000000	0004 95
L49	0.000000	DOUBLE	0.000000	0005 05
L50	0.000000	DOUBLE	0.000000	0005 15
L51	0.000000	DOUBLE	0.000000	0005 25
L52	0.000000	DOUBLE	0.000000	0005 35
L53	0.000000	DOUBLE	0.000000	0005 45
L54	0.000000	DOUBLE	0.000000	0005 55
L55	0.000000	DOUBLE	0.000000	0005 65
L56	0.000000	DOUBLE	0.000000	0005 75
L57	0.000000	DOUBLE	0.000000	0005 85
L58	0.000000	DOUBLE	0.000000	0005 95
L59	0.000000	DOUBLE	0.000000	0006 05
L60	0.000000	DOUBLE	0.000000	0006 15
L61	0.000000	DOUBLE	0.000000	0006 25
L62	0.000000	DOUBLE	0.000000	0006 35
L63	0.000000	DOUBLE	0.000000	0006 45
L64	0.000000	DOUBLE	0.000000	0006 55
L65	0.000000	DOUBLE	0.000000	0006 65
L66	0.000000	DOUBLE	0.000000	0006 75
L67	0.000000	DOUBLE	0.000000	0006 85
L68	0.000000	DOUBLE	0.000000	0006 95
L69	0.000000	DOUBLE	0.000000	0007 05
L70	0.000000	DOUBLE	0.000000	0007 15
L71	0.000000	DOUBLE	0.000000	0007 25
L72	0.000000	DOUBLE	0.000000	0007 35
L73	0.000000	DOUBLE	0.000000	0007 45
L74	0.000000	DOUBLE	0.000000	0007 55
L75	0.000000	DOUBLE	0.000000	0007 65
L76	0.000000	DOUBLE	0.000000	0007 75
L77	0.000000	DOUBLE	0.000000	0007 85
L78	0.000000	DOUBLE	0.000000	0007 95
L79	0.000000	DOUBLE	0.000000	0008 05
L80	0.000000	DOUBLE	0.000000	0008 15
L81	0.000000	DOUBLE	0.000000	0008 25
L82	0.000000	DOUBLE	0.000000	0008 35
L83	0.000000	DOUBLE	0.000000	0008 45
L84	0.000000	DOUBLE	0.000000	0008 55
L85	0.000000	DOUBLE	0.000000	0008 65
L86	0.000000	DOUBLE	0.000000	0008 75
L87	0.000000	DOUBLE	0.000000	0008 85
L88	0.000000	DOUBLE	0.000000	0008 95
L89	0.000000	DOUBLE	0.000000	0009 05
L90	0.000000	DOUBLE	0.000000	0009 15
L91	0.000000	DOUBLE	0.000000	0009 25
L92	0.000000	DOUBLE	0.000000	0009 35
L93	0.000000	DOUBLE	0.000000	0009 45
L94	0.000000	DOUBLE	0.000000	0009 55
L95	0.000000	DOUBLE	0.000000	0009 65
L96	0.000000	DOUBLE	0.000000	0009 75
L97	0.000000	DOUBLE	0.000000	0009 85
L98	0.000000	DOUBLE	0.000000	0009 95
L99	0.000000	DOUBLE	0.000000	0010 05
L100	0.000000	DOUBLE	0.000000	0010 15

09/01/74 PAGE 82

LINE CACI SCHEDULE II.5 RELEASE RF

```

1  EVENT DECISION, ON, MOD
2  **
3  ** DECISION IS MADE TO IMPLEMENT A STRUCTURAL MODIFICATION WHENEVER
4  ** MCPH(MPAIR COST PER HOUR) + ICPH(INSPECTION COST PER HOUR)
5  ** IS GREATER THAN MCPH(MODIFICATION COST PER HOUR)
6  **
7  DEFINE MODL AS AN INTEGER VARIABLE
8  LET DECISION, ON, MOD = "NO"
9  LET TAILING = 1ST, TAILING
10 LET MODL COST = 1ST, MODL COST
11 IF POPULOUSLY MODIFIED = "YES"
12 LET TAILING = AD, TAILING
13 LET MODL COST = AD, MODL COST
14 MEGARULESS
15 FOR EVERY AIRCRAFT IN ACTIVE, FLEET
16 DO
17 LET MPM = ENTRY, TIME + USAGE, LIFE - TIME, V - LEAD, TIME - AHCD(LIL)/2
18 IF MPM GT 0.0
19 LET POST, MOD, MPM = POST, MOD, MPM + MPM
20 MEGARULESS
21 **
22 IF TMO, PENDING(TAIL, ID) = "NO" AND SMO, PENDING(TAIL, ID) = "NO"
23 IF ACCUM, MOD(TAIL, ID) GT 0.0
24 LET ACCUMULATED, MPM = ACCUMULATED, MPM + TIME, V - DECUM, MOD(TAIL, ID)
25 JUMP AMPL
26 OTHERWISE
27 LET ACCUMULATED, MPM = ACCUMULATED, MPM + TIME, V - ENTRY, TIME
28 MPM
29 MEGARULESS
30 LOOP
31 LET ACCUMULATED, MPM = ACCUMULATED, MPM + DILEMMA, TIME
32 LET MODL = TRUNC(LEAD, TIME/PRODUCTION, TIME)
33 IF MODL GT SIZE, OF, FLEET - 1000
34 LET MODL = SIZE, OF, FLEET - 1000
35 MEGARULESS
36 LET POST, MOD, MPM = POST, MOD, MPM + (LEAD, TIME - LEAD, TIME + AHCD(LIL)/2)
37 + (SIZE, OF, FLEET - 1000 - MODL) * USAGE, LIFE
38 LET TCM = CARCD(LIL) * (1/AHCD(LIL) - 1/TERCD(LIL)) + CARCD(LFL) *
39 (1/AHCD(LFL) - 1/TERCD(LFL))
40 IF LIL TO LFL
41 LET TCM = CARCD(LFL) * (1/AHCD(LFL) - 1/TERCD(LFL))
42 MEGARULESS
43 IF POST, MOD, MPM = 0.0
44 RETURN
45 OTHERWISE
46 LET MCPH = (MOD, COST * (100 - MODL) - 1, MOD, OF, ENTRY, 2, MOD, OF, CRASH) + TAILING,
47 POST, MOD, MPM
48 LET ACM = COST, OF, REPAIRS / ACCUMULATED, MPM
49 IF ACM + TCM GT MCPH
50 LET MPM = (SIZE, OF, FLEET + 1 - TCM, OF, PRODUCTION, TIME) *
51 PRODUCTION, TIME
52 IF TIME, V + LEAD, TIME LT MPM, PRODUCTION, MPM + MCPH + FIS + USAGE, LIFE
53 SCHEDULE AC, MODL, MODIFICATION AT TIME, V + LEAD, TIME

```

09/01/76 PAGE 43

LINE CACT STMSCHMT 11.5 RELEASE NF

```

54 ALWAYS
55 LET TWO_ND_SCH = "YES"
56 LET OVERLAPSLY_MODIFIED = "YES"
57 IF TEST_FAILURE = "YES" AND TIME = 0 START TEST + ACTUAL_AVG_FAIL_LIFE /
58 TEST_ACCEPTANCE
59 LET TEST_FAILURE = "NO"
60 CANCEL THE IMPLEMENTATION
61 RESTART THE IMPLEMENTATION
62 RELEASE
63 RELEASE
64 END

```

LOCAL VARIABLES IF THIS ROUTINE

ACCUMULATE	DOUBLE	WORD 14	ICOM	DOUBLE	WORD 15
LOCAL	INTERM	WORD 7	ICOM	DOUBLE	WORD 17
SPC_CUST	DOUBLE	WORD 5	ICOM	DOUBLE	WORD 9
NFTS	DOUBLE	WORD 21	ICOM	DOUBLE	WORD 1
POST_WOOL	DOUBLE	WORD 11	ICOM	DOUBLE	WORD 19
TUNING	DOUBLE	WORD 3			

LINE CACI SYNOPSIS II.5 RELEASE OF 09/01/76 PAGE 24

```

1 EVENT IMPLEMENTATION MODIFICATION
2 **
3 ** REPRESENTS DEVELOPMENT OF MODIFICATION BECAUSE OF SERVICE EXPERIENCE
4 **
5 LET MOD = MOD + 1
6 LET MOD = 0
7 LET TIME_MODIFIED = 1000
8 LET REPAIR_MODIFIED = 0
9 LET IMPROVISED_SCH = "NO"
10 LET ARCD(LIL) = 1000(LIL)
11 LET ARCD(LIL) = 1000(LIL)
12
13 IF ARCD(LIL) < 0
14   JUMP ARCD
15 OTHERWISE
16 LET SCIN(MOD) = ARCD(LIL)
17 LET SCIN(MOD) = ARCD(LIL)
18 HERE
19 LET COST_OF_REPAIRS = 0.0
20 LET AIRFRAME_TIME = 0.0
21
22 IF MODTESTED = "NO"
23   LET MODTESTED = "NO"
24   LET MODTESTED = "NO"
25   LET MODTESTED = "NO"
26   LET MODTESTED = "NO"
27   LET MODTESTED = "NO"
28   LET MODTESTED = "NO"
29   LET MODTESTED = "NO"
30   LET MODTESTED = "NO"
31   LET MODTESTED = "NO"
32   LET MODTESTED = "NO"
33   LET MODTESTED = "NO"
34   LET MODTESTED = "NO"
35   LET MODTESTED = "NO"
36   LET MODTESTED = "NO"
37   LET MODTESTED = "NO"
38   LET MODTESTED = "NO"
39   LET MODTESTED = "NO"
40   LET MODTESTED = "NO"
41   LET MODTESTED = "NO"
42   LET MODTESTED = "NO"
43   LET MODTESTED = "NO"
44   LET MODTESTED = "NO"
45   LET MODTESTED = "NO"
46   LET MODTESTED = "NO"
47   LET MODTESTED = "NO"
48   LET MODTESTED = "NO"
49   LET MODTESTED = "NO"
50   LET MODTESTED = "NO"
51   LET MODTESTED = "NO"
52   LET MODTESTED = "NO"
53   LET MODTESTED = "NO"
54   LET MODTESTED = "NO"
55   LET MODTESTED = "NO"
56   LET MODTESTED = "NO"
57   LET MODTESTED = "NO"
58   LET MODTESTED = "NO"
59   LET MODTESTED = "NO"
60   LET MODTESTED = "NO"
61   LET MODTESTED = "NO"
62   LET MODTESTED = "NO"
63   LET MODTESTED = "NO"
64   LET MODTESTED = "NO"
65   LET MODTESTED = "NO"
66   LET MODTESTED = "NO"
67   LET MODTESTED = "NO"
68   LET MODTESTED = "NO"
69   LET MODTESTED = "NO"
70   LET MODTESTED = "NO"
71   LET MODTESTED = "NO"
72   LET MODTESTED = "NO"
73   LET MODTESTED = "NO"
74   LET MODTESTED = "NO"
75   LET MODTESTED = "NO"
76   LET MODTESTED = "NO"
77   LET MODTESTED = "NO"
78   LET MODTESTED = "NO"
79   LET MODTESTED = "NO"
80   LET MODTESTED = "NO"
81   LET MODTESTED = "NO"
82   LET MODTESTED = "NO"
83   LET MODTESTED = "NO"
84   LET MODTESTED = "NO"
85   LET MODTESTED = "NO"
86   LET MODTESTED = "NO"
87   LET MODTESTED = "NO"
88   LET MODTESTED = "NO"
89   LET MODTESTED = "NO"
90   LET MODTESTED = "NO"
91   LET MODTESTED = "NO"
92   LET MODTESTED = "NO"
93   LET MODTESTED = "NO"
94   LET MODTESTED = "NO"
95   LET MODTESTED = "NO"
96   LET MODTESTED = "NO"
97   LET MODTESTED = "NO"
98   LET MODTESTED = "NO"
99   LET MODTESTED = "NO"
100  LET MODTESTED = "NO"

```

LOCAL VARIABLES OF THIS ROUTINE

NAME	TYPE	VALUE
MOD	INTEGER	0
TIME_MODIFIED	INTEGER	1000
REPAIR_MODIFIED	INTEGER	0
IMPROVISED_SCH	STRING	"NO"
ARCD(LIL)	ARRAY	1000
COST_OF_REPAIRS	REAL	0.0
AIRFRAME_TIME	REAL	0.0
MODTESTED	STRING	"NO"

- 204 -

- 205 -

— — — — —
— — — — —

STILLING SMITH

Label	Integer	Word	Byte	Integer	Word	Byte
L.1	INTEGER	WORD	5	I.2	INTEGER	WORD
L.3	INTEGER	WORD	7	I.1	INTEGER	WORD
K.1	INTEGER	WORD	10	K.2	INTEGER	WORD
L.3	INTEGER	WORD	12	K.2	INTEGER	WORD
L.1	INTEGER	WORD	14	L.11	INTEGER	WORD
L.13	INTEGER	WORD	42	L.15	INTEGER	WORD
L.17	INTEGER	WORD	58	L.19	INTEGER	WORD
L.21	INTEGER	WORD	76	L.23	INTEGER	WORD
L.24	INTEGER	WORD	78	L.25	INTEGER	WORD
L.9	INTEGER	WORD	14	L.7	INTEGER	WORD
L.9	INTEGER	WORD	14	L.1	INTEGER	WORD
PNACA	DOUBLE	WORD	55	PNACA	DOUBLE	WORD
PNACA	DOUBLE	WORD	59	PNACA	DOUBLE	WORD
PNCCA	DOUBLE	WORD	63	PNCCA	DOUBLE	WORD
PNACA	DOUBLE	WORD	67	PNACA	DOUBLE	WORD
PNACA	DOUBLE	WORD	71	PNACA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	75	PNCCA	DOUBLE	WORD
PNCSO	DOUBLE	WORD	79	PNCCA	DOUBLE	WORD
PNCSO	DOUBLE	WORD	83	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	87	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	91	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	95	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	99	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	103	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	107	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	111	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	115	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	119	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	123	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	127	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	131	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	135	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	139	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	143	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	147	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	151	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	155	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	159	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	163	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	167	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	171	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	175	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	179	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	183	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	187	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	191	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	195	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	199	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	203	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	207	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	211	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	215	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	219	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	223	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	227	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	231	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	235	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	239	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	243	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	247	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	251	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	255	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	259	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	263	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	267	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	271	PNCCA	DOUBLE	WORD
PNCCO	DOUBLE	WORD	27			

LINE CACI SCRIPT 11.5 RELEASE AF

```

COUNTING SUMMARY
OFFICE RANGE AS AN INTEGER VARIABLE
START NEW PAGE
WRITE MODEL(1), MODEL(2) AS S 50, "AIRCRAFT TYPE: ", P A 0
SKIP 2 OUTPUT LINES
PRINT 1 DOUBLE LINE WITH SIZE, OF, FLEET, USAGE, LIFE AS FOLLOWS
      NUMBER OF AIRCRAFT IN FLEET: ****
AIRCRAFT SERVICE LIFE: ****

```

AIRCRAFT SERVICE LIFE: ***** MONTHS

1

POINT 1 DOUBLE TIME AS FOLLOWS

NUMBER AND TIME TO INITIATION OF AIRCRAFT DEFECTS

```

SKIP 1 OUTP=1 LINE
IF GOSUB GT 0      LET PSICK = SICK      IFT PLICK = LICK      ALWAYS
IF GOSUB GT 0      LET PSICK = SICK      IFT PLICK = LICK      ALWAYS
IF GOSUB GT 0      LET PSICK = SICK      IFT PLICK = LICK      ALWAYS
PRINT "DUP-LE LINES WITH GOSUB, GOSUB, GOSUB, GOSUB, PLICK, PSICK, PSICK, PSICK,
PLICK, PLICK, PLICK, PLICK, AICK, AICK, ALSO AS FOLLOWS

```

	FIRST CRACK	COMPRESSION	SERVICE DAMAGE	PRODUCTION DEFECTS
FREQUENCIES				
MIN (HPS)	*****	*****	*****	*****
MAX (HPS)	*****	*****	*****	*****
Avg (HPS)	*****	*****	*****	*****

NUMERICAL LENGTH OF CRACKS DETECTED AT EACH LEVEL OF INSPECTION:

[illegible]

OCURRENCES	A-LEVEL	M-LEVEL	C-LEVEL	D-LEVEL	SPELLEN
MIN(T ₂)	*****	*****	*****	*****	*****
MAX(IN)	*****	*****	*****	*****	*****
MAX(OUT)	*****	*****	*****	*****	*****
AUG(IE)	*****	*****	*****	*****	*****
SUM	*****	*****	*****	*****	*****

```

COP 1 OUTPUT LINES *****
SKIP 2 OUTPUT LINES *****
PRINT 1 DOUBLE LINE AS FOLLOWS *****
-JIMEN AND AREA OF COMMISSION DEFECTS DETECTED AT EACH LEVEL OF INSPECTION
COP 1 OUTPUT LINES *****

```

[illegible]

11

- 208 -

THIS PAGE IS BEST QUALITY PRACTICABLE
FROM COPY FURNISHED TO DDC

00/01/70 PAGE 90

LINE CACT STMSCHPT 11.5 RELEASE RF

LOCAL VARIABLES OF THIS ROUTINE

I-1	INTEGER	WORD 5	I-2	INTEGER	WORD 6
I-3	INTEGER	WORD 7	J-1	INTEGER	WORD 8
K-1	INTEGER	WORD 10	K-2	INTEGER	WORD 11
L-1	INTEGER	WORD 12	K-4	INTEGER	WORD 13
L-11	INTEGER	WORD 14	L-11	INTEGER	WORD 32
L-13	INTEGER	WORD 53	L-15	INTEGER	WORD 54
L-17	INTEGER	WORD 75	L-19	INTEGER	WORD 76
L-21	INTEGER	WORD 77	L-3	INTEGER	WORD 15
L-5	INTEGER	WORD 14	L-7	INTEGER	WORD 17
L-9	INTEGER	WORD 31	K-11	INTEGER	WORD 9
PLACA	DOUBLE	WORD 57	PLACW	DOUBLE	WORD 35
PLBCA	DOUBLE	WORD 61	PLCCW	DOUBLE	WORD 39
PLCCA	DOUBLE	WORD 65	PLCCW	DOUBLE	WORD 43
PLCCA	DOUBLE	WORD 69	PLCCW	DOUBLE	WORD 47
PLSCA	DOUBLE	WORD 73	PLSCW	DOUBLE	WORD 51
PLISD	DOUBLE	WORD 25	PLISW	DOUBLE	WORD 21
PLISD	DOUBLE	WORD 29	PSACA	DOUBLE	WORD 55
PSACB	DOUBLE	WORD 33	PSACA	DOUBLE	WORD 59
PSACB	DOUBLE	WORD 37	PSCCA	DOUBLE	WORD 63
PSCCR	DOUBLE	WORD 41	PSCCA	DOUBLE	WORD 67
PSCCR	DOUBLE	WORD 45	PSCCA	DOUBLE	WORD 71
PSCCR	DOUBLE	WORD 49	PSYCH	DOUBLE	WORD 23
PSICH	DOUBLE	WORD 19	PSYCH	DOUBLE	WORD 27
K-1	DOUBLE	WORD 3	WGE	INTEGER	WORD 1