

AD 742834

*Technical Report 3*

## A SCENARIO FOR A MANAGERIAL SUPPORT COMPLEX — OPERATIONAL STANDARDS

*By:* MARSHALL C. PEASE ABRAHAM WAKSMAN

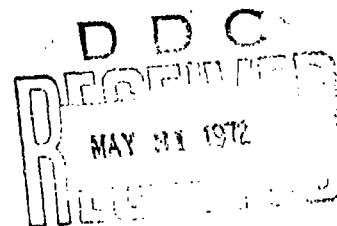
*Prepared for:*

OFFICE OF NAVAL RESEARCH  
DEPARTMENT OF THE NAVY  
ARLINGTON, VIRGINIA 22217

Contract Monitor: MARVIN DENICOFF, PROGRAM DIRECTOR  
INFORMATION SYSTEMS BRANCH

CONTRACT N00014-71-C-0210

Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of  
Commerce for sale to the general public.



**STANFORD RESEARCH INSTITUTE**  
Menlo Park, California 94025 • U.S.A.

Reproduced by  
**NATIONAL TECHNICAL  
INFORMATION SERVICE**  
Springfield, Va. 22151

67

## DOCUMENT CONTROL DATA - R &amp; D

Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified

1. ORIGINATING ACTIVITY (Corporate author) Stanford Research Institute Menlo Park, California 94025		2a. REPORT SECURITY CLASSIFICATION Unclassified	
		2b. GROUP N/A	
3. REPORT TITLE A SCENARIO FOR A MANAGERIAL SUPPORT COMPLEX--OPERATIONAL STANDARDS			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Technical Report 3			
5. AUTHOR(S) (First name, middle initial, last name) Marshall C. Pease and Abraham Waksman			
6. REPORT DATE May 1972		7a. TOTAL NO. OF PAGES 66	7b. NO. OF REFS 0
8a. CONTRACT OR GRANT NO. N00014-71-C-0210		8a. ORIGINATOR'S REPORT NUMBER(S) Technical Report 3 SRI Project 1031	
b. PROJECT NO.		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
c.			
d.			
10. DISTRIBUTION STATEMENT Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce for sale to the general public.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY Office of Naval Research Department of the Navy Arlington, Virginia 22217	
13. ABSTRACT <p>An interactive program complex that can be used in a scenario for managerial studies of operational standards is described. The scenario is based on aircraft maintenance and flight data obtained from the Naval 3-M (Maintenance and Material Management) System. The scenario simulates the data selection and processing that a manager needs when he is seeking to establish standards relating operational use of the aircraft under certain circumstances to maintenance actions.</p> <p>The design of the program complex has some interesting features. Emphasis has been given to the use of a dialogue mode of interaction, so that effective use of the system does not depend on programming expertise by the user. Also, there is a clear separation of the command input phase from the data processing phase, with only the appropriate computer facilities being used in each phase. The interface between these two subsystems illustrates an effective means of transfer between such subsystems.</p> <p>Another concept which is believed to be important is that of a "satellite" program. Properly speaking, a satellite program is not a program at all--it will not run by itself with any input. It acts, rather, to modify the main program to which it is a satellite. To facilitate the design of satellites, emphasis is given to the identification of "switch nodes" in the main program. A rigorous discipline is enforced that requires all actions that specifically accumulate, process, or output data be done only from these switch nodes. This discipline clarifies the functions of the several sections of the main program, and the interaction of those sections with the code</p>			

DD FORM 1473

NOV 68 (PAGE 1)

S/N 0101-807-6801

Unclassified

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Scenario						
Management      Management Scenario						
Decision Process						
Management System						
Program Complex						
Interactive Process						
Standards						
Operational Standards						
Satellite programs						
Overlayed programs						

Item 13. ABSTRACT (continued)

added by the satellites. It is effective in enhancing the quality of "open-endedness," making it relatively easy to add new satellites as the need arises.

Finally, there is discussed a list of possible extensions and modifications of the complex, ranging from those that only involve minor details to those that concern major questions of direction and structure. The relationships of these modifications to the overall objectives of the research program are considered.



STANFORD RESEARCH INSTITUTE  
Menlo Park, California 94025 · U.S.A.

*Technical Report 3*

*May 1972*

## A SCENARIO FOR A MANAGERIAL SUPPORT COMPLEX — OPERATIONAL STANDARDS

*By:* MARSHALL C. PEASE ABRAHAM WAKSMAN

*Prepared for:*

OFFICE OF NAVAL RESEARCH  
DEPARTMENT OF THE NAVY  
ARLINGTON, VIRGINIA 22217

Contract Monitor: MARVIN DENICOFF, PROGRAM DIRECTOR  
INFORMATION SYSTEMS BRANCH

CONTRACT N00014-71-C-0210

SRI Project 1031

Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce for sale to the general public.

*Approved by:*

DAVID R. BROWN, *Director*  
*Information Science Laboratory*

BONNAR COX, *Executive Director*  
*Information Science and Engineering Division*

Copy No. ....<sup>231</sup>

## ABSTRACT

An interactive program complex that can be used in a scenario for managerial studies of operational standards is described. The scenario is based on aircraft maintenance and flight data obtained from the Naval 3-M (Maintenance and Material Management) System. The scenario simulates the data selection and processing that a manager needs when he is seeking to establish standards relating operational use of the aircraft under certain circumstances to maintenance actions.

The design of the program complex has some interesting features. Emphasis has been given to the use of a dialogue mode of interaction, so that effective use of the system does not depend on programming expertise by the user. Also, there is a clear separation of the command input phase from the data processing phase, with only the appropriate computer facilities being used in each phase. The interface between these two subsystems illustrates an effective means of transfer between such subsystems.

Another concept which is believed to be important is that of a "satellite" program. Properly speaking, a satellite program is not a program at all--it will not run by itself with any input. It acts, rather, to modify the main program to which it is a satellite. To facilitate the design of satellites, emphasis is given to the identification of "switch nodes" in the main program. A rigorous discipline is enforced that requires all actions that specifically accumulate, process, or output data be done only from these switch nodes. This discipline clarifies the functions of the several sections of the main program, and the interaction of those sections with the code added by the satellites. It is effective in enhancing the quality of "open-endedness," making it relatively easy to add new satellites as the need arises.

Finally, there is discussed a list of possible extensions and modifications of the complex, ranging from those that only involve minor

details to those that concern major questions of direction and structure. The relationships of these modifications to the overall objectives of the research program are considered.

# CONTENTS

ABSTRACT .....	iii
LIST OF ILLUSTRATIONS .....	vi
I INTRODUCTION .....	1
A. General .....	1
B. Objectives .....	1
II THE SCENARIO .....	3
III GENERAL APPROACH .....	5
A. Introduction .....	5
B. /STANDARDS/ .....	7
C. /INTERROGATION/ .....	8
D. /SUMMARY/ .....	12
E. /SAVESUMM/ and /DATAENT/ .....	14
F. The Processing Complex .....	15
G. Data .....	17
H. Control Transfer .....	18
I. Code .....	18
IV OPEN-ENDEDNESS .....	19
V POSSIBLE EXTENSIONS AND MODIFICATIONS .....	23
A. Immediate Extensions .....	23
B. Language and Facilities .....	23
C. Structural Modifications .....	24
D. Outputs .....	24
E. Working Files .....	25
F. Interactive Output .....	25
G. Utility File .....	26
H. Anticipation .....	26
VI CONCLUSIONS .....	27
APPENDIX A--CONTROL INFORMATION .....	28
APPENDIX B--CODE FOR THE PROGRAM COMPLEX .....	32

DD FORM 1473



## ILLUSTRATIONS

1	Program Complex .....	6
2	Command Input .....	9
3	Summary Printout and Return to /INTERROGATION/ .....	13
4	Flow Chart for /EXECUTE/ .....	20

## I INTRODUCTION

### A. General

Stanford Research Institute is conducting a broad program of research into large file systems that will provide effective support for Naval Managers in the exercise of their responsibilities. The objective of this program is to determine design principles that will enable the construction of operational systems that will be much more effective than are any existing systems.

Technical Report 1, entitled "Large File Management Information Systems," issued September 1971, discusses in general terms the properties and capabilities that an effective management support system should have. It also specifies a number of areas within which research is needed. In that report, we recognized that the research efforts should include constructing a data system on which one or more scenarios can be executed. It was believed that this system should be based on real data, and that each scenario should illustrate some specific area of real managerial concern.

This report describes the data system that we have established based on maintenance and operational data on Naval aircraft obtained from the 3-M (Maintenance and Material Management) System. It is designed to execute scenarios that will illustrate the managerial activity of developing operational standards and monitoring conformance to them.

Subsequent sections describe the system at various levels of detail, discuss the significance of what has been accomplished, and discuss various possibilities for further development.

### B. Objectives

The objectives of the scenario are as follows:

- (1) To demonstrate the possibility of the type of facility considered, and to provide means through which potential

users of such a facility can assess its relevance to their needs.

- (2) To keep the general research program "grounded" in problems of managerial importance with actual data, and thereby to develop better understanding of the realities of the managerial process.
- (3) To identify the factors that are present in achieving multilevel operation and in obtaining stable performance with it. This will provide a necessary basis for considering adaptive behavior within any given level.
- (4) Similarly, to identify the factors present in multifunctional operation. Again, this provides a basis for studying adaptivity, since such divisions limit the scope of an adaptive process within a horizontal level.
- (5) To consider how to obtain programming "open-endedness" so that the range of use of a facility can be extended with minimal effort and confusion. This objective is closely related to the functional separation cited above, since functional divisions limit the extent of possible interference as new functions are added.
- (6) To provide a foundation for a more extensive system to serve as a vehicle for the general program of research on management support systems. This objective includes a number of areas for research such as the need for linguistic capabilities in interaction, model building and study, adaptive file organization and utilization, and others.

## II THE SCENARIO

This section describes the managerial problem that is involved, and the specific context in which it is considered.

The general area is that of developing operational standards on the basis of historical evidence and monitoring subsequent performance. The basic assumption is that the responsibility of the manager includes overseeing the maintenance and use of some type or types of complex equipment. The specific context considered is the naval manager who is directly responsible for maintenance and use of certain aircraft--specifically F4-J aircraft.

We assume the existence of dated records of maintenance actions and of the use of equipment of the type involved. In the specific context, we use a subset of the records for naval aircraft from the 3-M (Maintenance and Material Management) System. Specifically, we use the data reported on code 71, 76, and 79 cards reporting maintenance actions, flights, and monthly summary cards.

We assume that the manager is concerned with standards that relate operational use of the equipment to maintenance actions. For example, he may be concerned with the time between maintenance actions or some classes of actions. Alternatively, he may be concerned with expected operational time, or the number of occurrences of certain operational events. From the data base employed here, the manager might determine, for example, the total flight time that can be expected to accumulate before the recurrence of a certain class of maintenance actions. Similarly, he might be concerned with the number of landings, the number of arrested landings, or some other factor.

Broadly speaking, the process of establishing a standard from historical evidence starts from a hypothesis of the existence of some type of pattern in time. The user tests the hypothesis against the data (or some

part of it), adjusting the parameters of the hypothesis as appropriate. Discrepancies are investigated to determine whether they are atypical (and therefore ignorable) or indicate need for adjustment or revision of the hypothesis. This cycle may be repeated as many times as appropriate.

On the other hand, the monitoring of a standard entails the testing of new data against the standard to identify exceptional instances. As a subsequent step, the identified instances may be investigated to determine whether they require individual action, or whether they suggest the need for revision of the standard.

It is assumed that the manager-user does not have expertise in programming. Furthermore, insofar as possible, it is desirable to minimize the demand for any prior knowledge of the system or of how to make use of it. On the other hand, we also recognize the need for allowing flexibility of use by a user who may have programming expertise and who may make use of the "open-endedness" that has been a major functional goal of the scenario development, as will be discussed later.

In the particular context of the scenario, the entire range of capabilities suggested here has not yet been implemented. The operating goal has been to demonstrate the general type of capability required, rather than to exhaust the possibilities of this type of application. Since one functional goal of the scenario has also been to achieve programmatic "open-endedness", it does provide a foundation on which a complete system for the analysis of standards could be built.

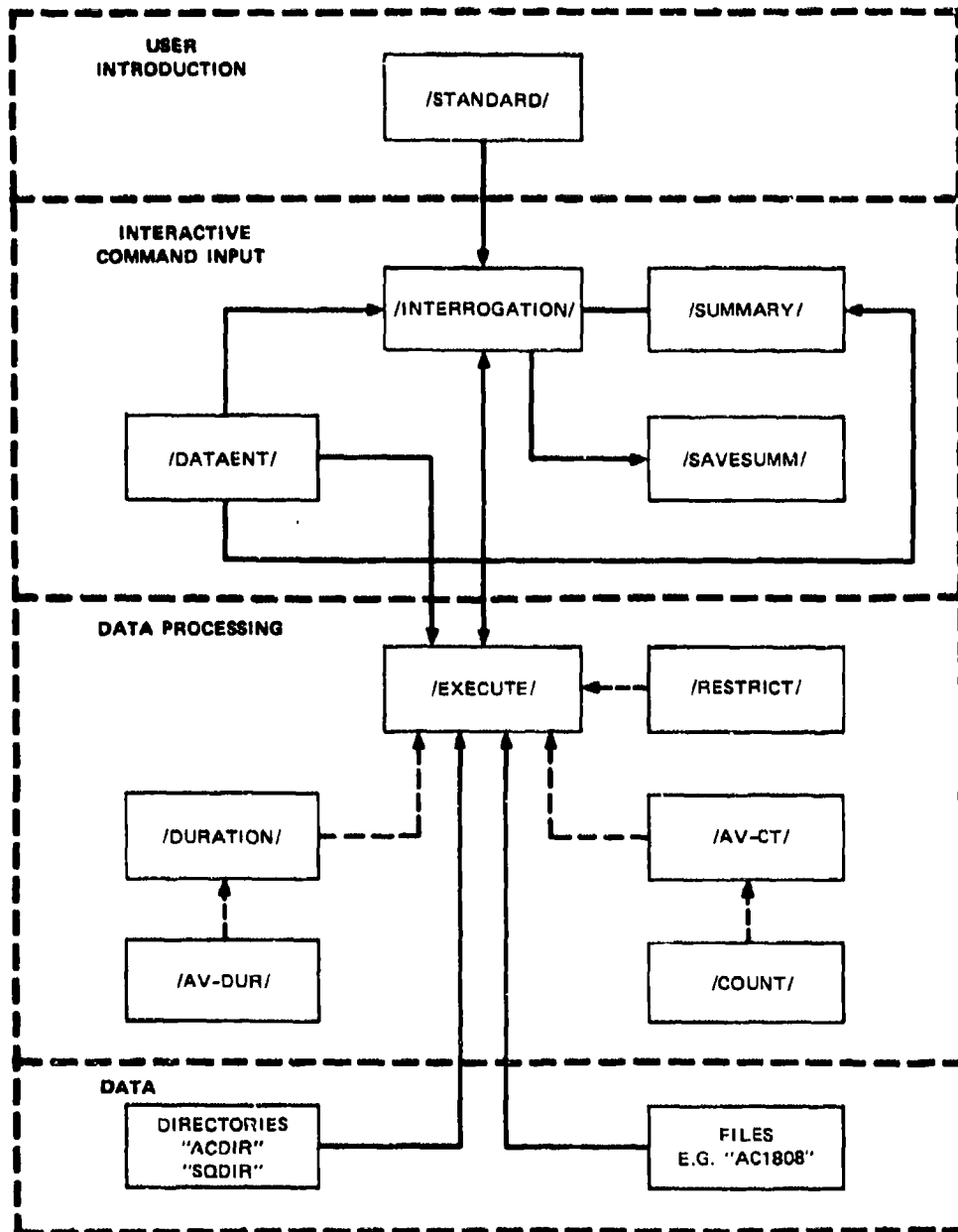
### III GENERAL APPROACH

#### A. Introduction

This scenario has been developed on the basis of the following viewpoint:

- It is an interactive system, i.e., it presumes that the user's requirements for information will be modified as the system responds and that the user will want to enter his modified requirements immediately. We believe this to be very important during the time when the manager is first developing his concepts, and the absence of this quality is a serious lack in most existing managerial information systems.
- We are not here addressing problems of data entry or validation. We acknowledge the importance of these problems, but have not sought to include them in the scenario.
- We are not addressing problems of communicating with other systems or data files. Such problems may be considered later as we expand the range of the scenario.
- We give primary attention to the highest level user who is likely to be concerned with the subject of the scenario-- in this case, specific operational standards. This user has the most subtle and difficult problem. We believe that a system that can serve him effectively is likely to be useful also to a lower-level manager whose responsibilities are of a more detailed nature, while the converse is not likely to be true.

The general configuration of the system as it currently exists is outlined in Figure 1, which exhibits the components of what we call the "program complex."



SA-1031-22

FIGURE 1 Program complex

Three levels of programs are distinguished, plus a fourth level that includes the actual data in the various files and file directories. The top level is labeled "User Introduction." This level provides the user with a general introduction to the system if he needs it. It tells him what can be done with the system and the mechanics that he needs to know to use it. This level can be bypassed if the user is already familiar with the system.

The next level is labeled "Interactive Command Input." The basic function here is to develop in detail what information the user wants. This is not simply a matter of entering the command. It includes provision for modifying the current command so that the user can conveniently execute a linked series of interrogations. It also includes provision for displaying a concise summary of the current command, and for saving the current command when desired. Finally, it includes means by which a previously saved command can be reentered.

The third level actually executes the command, processing the data retrieved from the data files as required.

These levels, and the components that make them up are discussed in detail in the following subsections.

#### B. /STANDARDS/

Use of the complex is initiated by calling /STANDARDS/. This offers the user the option of having printed out a fairly detailed description of how he can use the complex. However, he may bypass this printout if he desires.

It would be desirable also--although not yet done--to include here the option of calling for a more technical description of the complex including such matters as what data are available and what options can be employed. It might be desirable to provide two or more such descriptions



at successively greater depth of technicality that can be called up or not by the user. There would be no difficulty in providing any desired options here.

/STANDARDS/ also sets some of the initializations required--e.g., dimensionality of the control vector of standard formats.

### C. /INTERROGATION/

Upon completion, /STANDARDS/ links to /INTERROGATION/. The link command that is available in the programming language used (Super-Basic) clears the current program while retaining data and variable assignments, loads the called program, and starts execution. These programs are, therefore, handled as "coprograms," as are /SUMMARY/ and /EXECUTE/ (to be discussed shortly). In Figure 1, this kind of transfer is indicated by a solid arrow.

/INTERROGATION/ first asks if the user wants to use a previously saved command. If so, the program links to /DATAENT/, as will be discussed.\* If the user does not want a previously saved command, /INTERROGATION/ then conducts a dialogue with him to establish his command.

A typical sequence is shown in Figure 2, the exact sequence of questions depends on the responses given. The user-generated responses are underlined in the figure.

We have chosen the multiple-choice form of input to avoid either requiring the user to know a set of precise formatting conventions, or

---

\* The system can also be entered directly through /DATAENT/, eliminating these preliminary steps. However, inclusion of this option here allows the user to switch to a saved command at appropriate times during a sequence of processing without quitting and reentering the system.

\*\*\*\*FLIGHT STANDARDS\*\*\*\*

IF YOU NEED DETAILED INSTRUCTIONS, TYPE '1' FOLLOWED  
BY THE RETURN KEY. ELSE TYPE '2' AND RETURN. 2

\*\*\*COMMAND INPUT\*\*\*

DO YOU WANT TO ENTER :

1. A NEW COMMAND.
2. A PREVIOUSLY SAVED COMMAND? 1

(A) WHAT TYPE OF AIRCRAFT DO YOU WANT TO CONSIDER? AFPH

(B) DO YOU WANT TO CONSIDER

1. ALL A/C OF TYPE AFPH.
2. CERTAIN A/C OF TYPE AFPH? 2  
HOW MANY A/C DO YOU WANT TO LIST? 3  
LIST THE A/C DESIRED:

? 153808,155903,155799

(C) DO YOU WANT TO CONSIDER

1. ALL SQUADRONS.
2. CERTAIN SQUADRONS? 2  
HOW MANY SQUADRONS? 2  
LIST THE SQUADRONS DESIRED:

? AC7,PC1

(D) WHAT TIME PERIOD DO YOU WANT?

TYPE THE JULIAN DATES, SEPERATED BY COMMAS. ? 0240,1020

- DO YOU WANT THE FINAL EVENTS TO BE:
1. ENDED ON/BEFORE THE FINAL DATE.
  2. STARTED ON/BEFORE THE FINAL DATE? 1

(E) HOW ARE EVENTS TO BE DEFINED?

1. BY ANY WUC.
2. BY A PARTICULAR SET OF WUC CODES.
3. BY ANY WUC EXCEPT A PARTICULAR SET? 3  
HOW MANY CODES DO YOU WANT TO LIST? 2  
LIST THE CODES TO BE EXCLUDED:

? 74240,30000

(F) DO YOU WANT TO LIMIT ATTENTION TO EVENTS MEETING SOME CONDITION?

1. YES.
2. NO? 1  
DOES THE CONDITION APPLY TO:  
1. THE NUMBER OF FLIGHTS IN THE EVENT.  
2. THE NUMBER OF LANDINGS IN THE EVENT.  
3. THE FLIGHT TIME IN THE EVENT? 2  
DO YOU WANT:  
1. ALL LANDINGS.  
2. ONLY CERTAIN LANDING CODES CONSIDERED.  
3. ALL EXCEPT CERTAIN LANDING CODES? 3  
HOW MANY LANDING CODES? 2  
LIST THE LANDING CODES INVOLVED:

? E,F

WHAT IS THE RESTRICTION?

THE OPERATORS =, <, >, <>, <=>, >=> MAY BE USED.  
THE FORM OF THE INPUT SHOULD BE, E.G., X>3.25.  
IF TIME IS INVOLVED, USE DECIMAL HOURS.

SA-1031-23

FIGURE 2 Command input

? X=2

(G) WHAT DATA IS TO BE OUTPUTTED?

1. A LIST OF EVENTS.
  2. A LIST OF COUNTS
  3. A LIST OF ACCUMULATED DURATIONS.
  4. A LIST OF AVERAGE COUNTS.
  5. A LIST OF AVERAGE DURATIONS? 5
- WHAT AVERAGE DO YOU WANT?
1. THE AV. FLT DUR. PER EVENT FOR ALL EVENTS.
  2. THE AV. FLT DUR. PER EVENT FOR EACH A/C.
  3. THE AV. FLT DUR. PER EVENT FOR EACH SQUADRON.
  4. THE AV. FLT DUR. PER EVENT FOR EACH A/C IN EACH SQ.
  5. THE AV. DURATION OF EVENTS FOR EACH A/C.
  6. THE AV. DURATION OF EVENTS FOR EACH SQUADRON.
  7. THE AV. DURATION OF EVENTS FOR EACH A/C IN EACH SQ.
  8. THE AV. DURATION OF EVENTS FOR ALL EVENTS? 4

DO YOU WANT TO INCLUDE:

1. ALL FLIGHT CODES.
  2. ONLY CERTAIN FLIGHT CODES.
  3. ALL EXCEPT CERTAIN FLIGHT CODES? 3
- HOW MANY CODES? 4

LIST THE FLIGHT CODES INVOLVED:

? 4A0,3F0,3A3,3A6

DO YOU WANT TO:

1. EXECUTE.
2. ENTER A NEW COMMAND.
3. MODIFY THE PRESENT COMMAND.
4. SEE A SUMMARY OF THE PRESENT COMMAND.
5. SAVE THE PRESENT COMMAND.
6. END? 4

SA-1031-23a

FIGURE 2 Command input (Concluded)

having to use extensive syntactic and semantic analysis on a free form response. (A small amount of syntactic analysis is required if a restriction is entered in Section F.)

Some of the sections are trapped. Section A, type of aircraft, will accept only the response "AFPH." Any other response causes the program to print out that it has no data on the type specified, and that it only has AFPH; it then asks for a new input. Similarly, Section B will accept only 153808, 155903, and 155799, which are the aircraft for which files have been established. Section C will accept only AC1, AC7, and PC1, the codes for the organizations with which the aircraft listed were associated during the time span of the data. (Note that we could equally have used the more usual squadron designations. We used these codes simply because they were available to us.) Finally the dates in Section D are trapped to ensure that they are four-digit numbers, that they are in proper order, and that they are in the range 0213 to 1201, inclusive. In each case, a trapped response causes a printout of the acceptability conditions, and then asks for a new input.

In Section E the user defines what he wants for an "event." As presently permitted, an event is defined as extending from the completion of one of the designated maintenance actions (as indicated by the WUC code) to the start of the next for a given aircraft, providing something interesting happened in between--i.e. a flight or a nondesignated maintenance action. The maintenance actions of interest can be designated either by listing the desired actions, or by listing the ones that are not to be considered.

The options involved in Section E probably should be extended to include the category number (Cat. No.). No difficulty would be encountered in doing so.

Section F asks if there is some particular condition that should be applied to events. For example, the user might want to restrict attention only to events that include at least one carrier landing. Or he might be interested in events that are at least 24 hours long.

Section F allows the user to test directly the validity of a hypothesized standard, or to have listed for him exceptions to a given standard. Thus, it is critical to the basic concept of the scenario.

It should be noted that the restriction is quite independent of the data output specified in Section G. For example, the restriction can apply to the number of landings while the output concerns total flight time or event duration.

Most of Section F is concerned with identifying what is to be the restriction variable. The final question allows the user to enter his actual condition as an algebraic expression.

The final section, G, then asks the user to specify what output he desires. He has the option of specifying a list of events, counting occurrences, accumulating durations, or calculating average counts or durations. Depending on his response, quite different trees of questions lead to identification of exactly what is wanted.

Following completion of this input operation, the user is asked what he wants to do next. If he calls for a summary, the program links to /SUMMARY/. If he calls for execution, it links to /EXECUTE/. If he wishes to save his current command, it links to /SAVESUMM/. Otherwise, it either ends or loops appropriately within /INTERROGATION/.

D. /SUMMARY/

If the user calls for a summary, the program links to /SUMMARY/, which prints out the current command in a concise form. Figure 3 shows the printout for the input of Figure 2.

**\*\*SUMMARY\*\***

- A. TYPE = AFPH
  - B. AIRCRAFT NUMBERED:
    - 153808
    - 155903
    - 155799
  - C. SQUADRONS NUMBER:
    - AC7
    - PC1
  - D. TIME INTERVAL:
    - STARTING DATE = 0240
    - TERMINAL DATE = 1020
    - ONLY EVENTS ENDING ON/BEFORE TERMINAL DATE INCLUDED.
  - E. AN 'EVENT' IS DEFINED BY:
    - ANY WUC CODE EXCEPT THE FOLLOWING:
      - 74240
      - 30000
  - F. THE DATA IS RESTRICTED BY: X>=2
    - WHERE X IS THE NUMBER OF LANDINGS IN AN EVENT.
- THE LANDINGS APPLICABLE HERE INCLUDE ALL EXCEPT THE FOLLOWING CODES:  
E, F.
- G. THE OUTPUT IS TO BE:
    - THE AVERAGE FLIGHT DURATION PER EVENT FOR ALL EVENTS.
    - ALL FLIGHT CODES EXCEPT THE FOLLOWING:
      - 4A0, 3F0, 3A3, 3A6.
- \*\*END OF SUMMARY\*\***

ARE YOU READY TO EXECUTE?

- 1. YES.
- 2. NO

? 2

DO YOU WANT TO:

- 1. EXECUTE.
- 2. ENTER A NEW COMMAND.
- 3. MODIFY THE PRESENT COMMAND.
- 4. SEE A SUMMARY OF THE PRESENT COMMAND.
- 5. SAVE THE PRESENT COMMAND.
- 6. END? 3

WHICH SECTION DO YOU WANT TO CHANGE?  
TYPE THE APPROPRIATE LETTER-- ONE ONLY? G

SB-1031-24

FIGURE 3 Summary printout and return to /INTERROGATION/

At the completion of this printout, the user is asked if he is ready to execute the command. If so, the program links to /EXECUTE/. If not, it links back to /INTERROGATION/, passing immediately to the section that offers the principal user options, as illustrated at the bottom of Figure 3.

Figure 3 also illustrates, at the bottom, what happens in /INTERROGATION/ when the user decides to modify the current command. He is then asked which section he wants to modify. He is then returned to the indicated part of the command input. This cycle can be repeated as often as desired.

Upon returning to /INTERROGATION/ after either execution or /SUMMARY/, if the user opts for entering a new command, the first question is that shown at the top of Figure 2 asking if he wants a previously saved command. He might wish to do so, for example, if this returns him to his starting point, or if some other saved command is a more convenient one to modify into what he wants to do next.

E. /SAVESUMM/ and DATAENT/

There are also two small additional programs in the interactive level. The first is /SAVESUMM/ (for "save summary") to which we can link from /INTERROGATION/. This program sets up a data file that saves the data and variable assignments of his current command in a file named by the user. It is trapped against the user calling for the name of any of the programs in the complex. If he wishes, the user may return to /INTERROGATION/ after saving the current command.

The other program is /DATAENT/ (for "data enter"). It can be called initially or from /INTERROGATION/. It asks for the name of a data file previously established under /SAVESUMM/ and reinitializes with the saved command. The user is then given the choice of linking to /INTERROGATION/ to modify the command, to /SUMMARY/ to see it, or to /EXECUTE/ to execute it.

#### F. The Processing Complex

So far the complex has been concerned with the interrogation input. No files have been opened, no data processed, no computations performed. The only computer facilities that have been claimed have been those necessary for the dialogue. Once the interrogation has been entered and execution called for, control links to /EXECUTE/ and the system shifts to its processing mode. It will stay in this mode until execution is complete, when it will return to /INTERROGATION/ jumping to the main control question.

/EXECUTE/ has associated with it a number of quasi-programs that we call its "satellites." These are not programs or subprograms in any normal sense of these terms. They will not operate by themselves with any input. They are intended, instead, to overlay /EXECUTE/ and to modify its operation.

The basic unit of search is the "event." In general, this is defined as the period between the completion of a designated maintenance action on an aircraft to the start of another, providing this period includes some significant occurrence--either some undesignated maintenance action or one or more flights. The maintenance actions that determine an event are specified by the user as part of his command input. He can further modify what events are to be considered by imposing a restriction on the occurrences within them.

The determination of events is basic to all allowed commands. The satellite programs, therefore, do not affect the recognition of events (except for /RESTRICT/, as discussed below) but modify the data output or accumulation, and the format of the output. It is the commonality of the process of locating events that makes the device of satellite programs a useful one here.



/EXECUTE/ manipulates the data files and directories and determines the events. If not modified by a satellite, its output lists the starting and ending dates and times of all qualifying events. It checks the aircraft and squadron numbers against the command to determine what data files should be opened. It runs through a file until the starting date has been passed. It checks the WUC code against the command to determine the limits of an event. Finally, it checks either the beginning or end date of an event, as commanded, against the terminal date of the command.

This procedure is a common requirement for all commands. What is done during the process, including what data are collected during it and when and what data are printed out is modified by the input command through the device of the satellite programs.

The instruction that brings in a satellite is an indirect LOAD command. The effect of this command is to retain all data and variable assignments, to retain also the original program, and to load the called program. Where the original program and the called one have instructions with the same number, only the second one remains, the first is overwritten.

Normally, after a load command, control reverts to the terminal in the system we are using. To avoid this, each of the satellite files ends with the direct command "START," which initiates execution without destroying the data (which "RUN" would do).

This satellite linkage is indicated in Figure 1 by a dotted arrow.

/RESTRICT/ is loaded as a satellite if the command input calls for a restriction in Section F. Then, as an event is searched, it accumulates the data relevant to the restriction. When the end of an event is found, it checks these accumulated data against the condition, and either accepts or rejects the event.

/COUNT/ is loaded as a satellite if, in Section G, top level, the output is to be either counts or average counts. In the latter case /AV-CT/ is loaded as a satellite to /COUNT/.

/DURATION/ is loaded as a satellite if the output is to be either accumulated durations, or average durations. In the latter case, /AV-DUR/ is loaded as a secondary satellite.

#### G. Data

The data consists of a set of files and two directories: "ACDIR" is a directory organized by aircraft, and SQDIR is organized by squadron. ACDIR, for example, has a main listing of the aircraft, the earliest file for each, and a link to the next directory entry for that aircraft; hence, it is a list-organized file of file names.

Each file covers one aircraft while it is in one squadron. The name is the code for the squadron and the last three digits of the aircraft number--e.g., "AC1808" for 155808 while in AC1.

The entries in a file are arranged chronologically by date and time with some conventions to resolve ambiguities. Thus, the data from codes 71 and 76 cards have been interleaved--the code 79 card images have not so far been used.\* This permits us to search a file sequentially to determine an event, and to accumulate as we go the data for the restriction condition, if any, and for outputting.

Parenthetically, the files do not contain the direct card images. The data have been reformatted to conserve space.

---

\*The designation of data cards as codes 71, 76, and 79 is in accord with established Naval standards. The code 71 cards describe readiness status. The code 76 cards contain flight data. The code 79 cards are monthly summary cards for each aircraft.

It happens that all of the files except PC1799 are sufficiently small so that they could be completely loaded during search. PC1799 is too large for the system being used--it exhausts the available core memory. It has seemed desirable, however, for the purposes of our research on large files, to act as if all the files were too large. Hence all of the files are paged in as blocks of 50 entries at a time.

#### H. Control Transfer

One key element in any program complex is the way control is passed from one program to another. The principal vehicle used for this is a 16-component string vector labeled D(I). In addition, there may be other numeric values (N2 to N6, inclusive) and the string arrays B, C, M, F, and G depending on the choices made in /INTERROGATION/. The significances of these parameters are detailed in Appendix A.

#### I. Code

The actual code for the complex, as it exists at the present time, is given in Appendix B. Although it is a running program, not all possible commands have been tested and some difficulties may remain uncorrected as yet.

The code, as written, does of course depend on the specific language. It is believed, however, that it could be rewritten fairly easily into any language, or any version, having equivalent basic capabilities. The principles that have been used--such as the use of satellite programs--should be broadly applicable and should greatly facilitate such a translation should it be desired.

#### IV OPEN-ENDEDNESS

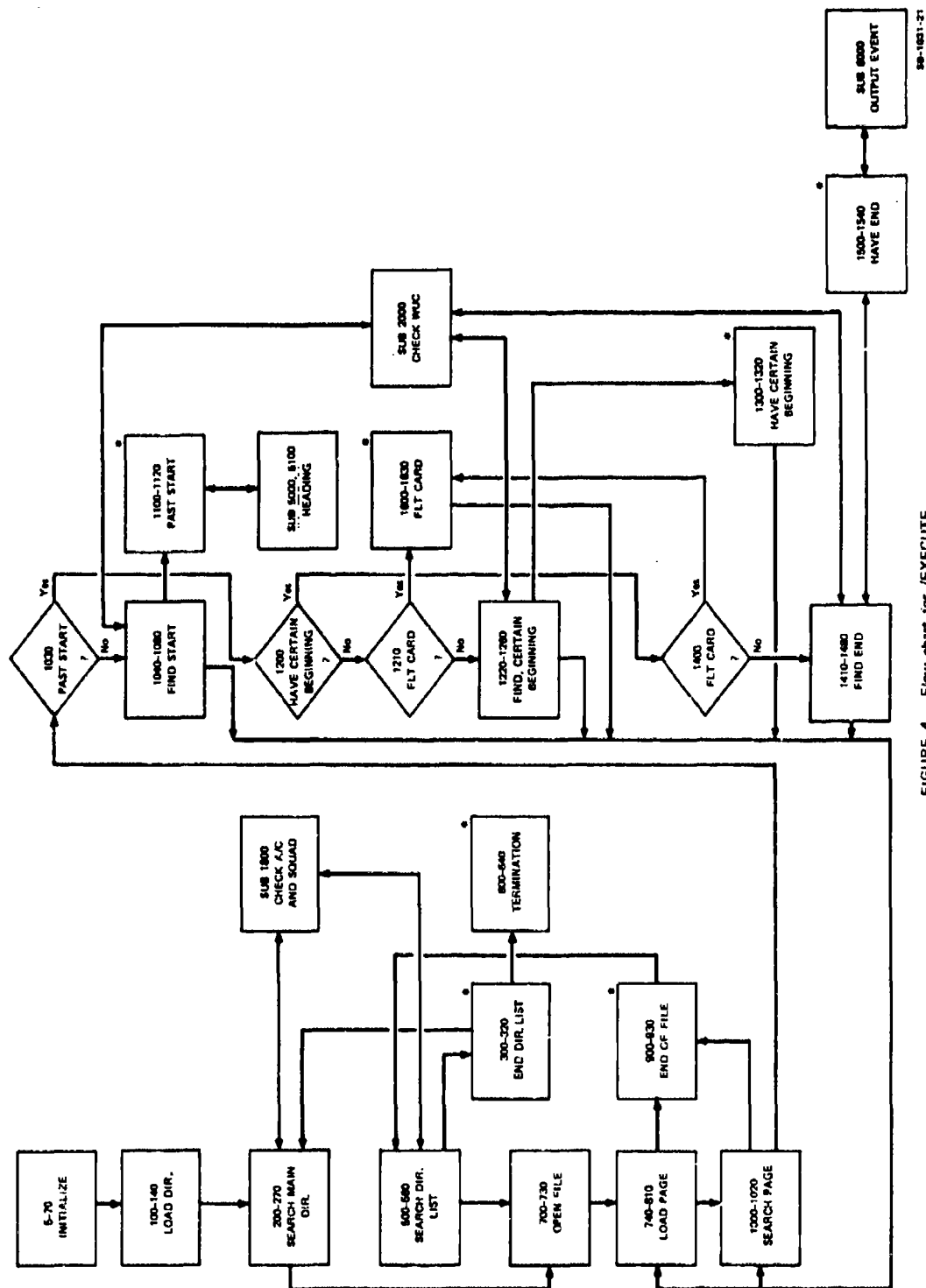
The flow chart for /EXECUTE/ is given in Figure 4. This is of interest since it provides the key to open-ended quality that we believe we have achieved.

The key to the flow chart is in the starred blocks. In /EXECUTE/ these are generally quite trivial bits of code. However, they are what we call "switch points." They are the points of attachment for the various satellite programs. They are passed through at the critical times in the basic search procedure, as follows:

##### Lines

300-320	Entered when a directory list is exhausted, indicating completion of the search on a particular aircraft or squadron.
600-640	Entered on completion of execution.
900-930	Entered on the end of a file, indicating completion of the search for an aircraft in a given squadron, or vice versa.
1100-1120	Entered when past the starting date of the user's command--the start of data collection.
1300-1320	Entered at a known beginning of an event.
1500-1540	Entered at the end of an event. (Note, not entered if end does not satisfy terminal date specified in the user's command.)
1600-1630	Entered when file entry is a flight card.

In addition, a satellite program will generally have to modify the initialization block (lines 5-70).



**FIGURE 4** Flow chart for /EXECUTE

The clear identification of the switch points makes it easy to write satellite programs that will work with the main program. For example, /EXECUTE/ does nothing with the 1600 block except, in 1600 itself, test if past initial date ( $S2 = 0$  otherwise) and, if so, to confirm a possible beginning of an event (by changing  $S2$  from 2 to 1 if it is not already so). However, a restriction on the event will concern the number of flights or landings, or the duration of flights in the event. Hence, /RESTRICT/ takes off from 1610 to accumulate data in the restriction variable,  $X$ . Similarly, /COUNT/ and /DURATION/ take off from 1620 if they may call for data output from the 76 cards. (They may not if the user is calling for either counting or accumulating the duration of the events themselves.)

Similarly, /RESTRICT/ takes off from 1500 to determine if the event is acceptable or not. If not, it comes back to 1530, bypassing any activity at 1510 or 1520. /EXECUTE/ itself branches at 1510 to output the event. /COUNT/ and /DURATION/ both overlay 1510 to kill the printout required by /EXECUTE/ itself. They use 1510 to accumulate data when appropriate, and 1520 to print output when appropriate, or to transfer data to an accumulation when this is required. Since /RESTRICT/ returns to 1530 if the event is unacceptable, it bypasses 1510 and 1520, and the data output and accumulation are performed only over those events that pass the restriction screen.

If we visualize the flow chart of /EXECUTE/ as a directed graph, then the overlay of a satellite expands the graph by attaching other arcs and nodes, with all attachments being at the switch points of the original graph.

Another way of looking at this is to say that we have constructed a kind of higher-level language. Each block of code between switch nodes may be regarded as defining a higher-level function or command, which could be given a name. The isolation and identification of the switch nodes in effect permits us to write a satellite first in this higher-level

implicit language, and then detailing the needed functions or commands that are special to that satellite.

Our experience has been that this approach greatly simplifies constructing satellite programs--and even satellites of satellites. It is this quality that leads us to consider that we have made the system open-ended to a useful and practical degree.

## V POSSIBLE EXTENSIONS AND MODIFICATIONS

The program complex, as it currently exists, could be extended or adapted in a number of directions, of which the following list is representative.

### A. Immediate Extensions (Not Involving Structural or Conceptual Changes)

It would be a relatively simple matter to add the use of the "Cat. No." for defining events. We could also include the option of counting catapult take-offs, or accumulating NOR hours in an event. Other data could also be collected if they proved to be of interest. For example, we could make the system detect events--defined by some class of maintenance actions--that contained some other disjoint class of maintenance actions. Or we could determine the time between events, and so accumulate the duration (or average duration) of the event-defining class of maintenance actions. These and similar additions to the available options could be added with relatively minor effort.

### B. Language and Facilities

We have given some consideration to the question of what other programming languages might be used. The only difficulty we see with FORTRAN is that it requires that all arrays be dimensioned before execution starts. This would slightly limit the flexibility of the command input. For example, we currently permit the user to list any number of aircraft. In FORTRAN, a maximum number would have to be specified. There are ways around this limitation, but they are awkward and inefficient. Otherwise, FORTRAN would be quite satisfactory.

ALGOL and ALGOL-type languages should be quite feasible. PL1 should offer no difficulties. Probably LISP, and perhaps SNOBOL could be used. Similar languages are possible candidates.

For the satellite concept, we need the LINK and LOAD commands or their equivalents. These are mainly functions of the operating system, rather than the language itself.



So far, we have designed the complex to operate interactively, generally in an interpretive mode. With a much larger set of files, it would probably be desirable to have the option of compiling the program as it is finally put together after loading the required satellites, and to execute it in a batch mode type of operation. There are no apparent difficulties in doing this.

We have provided only for printout on a teletype or other printing device. The use of a graphic display device would be a convenient facility, particularly for some of the additions discussed below. There would be no particular difficulty in implementing the present outputs on a display unit.

#### C. Structural Modification

A minor structural change that would further facilitate open-endedness would be to limit the transfer-of-control vector D to its first 12 components. The additional information contained in D(13) to D(16) would then be held in a subsidiary array. This would limit D to "top-level" choices in /INTERROGATION/ and open the way to an indefinitely deep treeing of possible commands for information output. The modifications required would be straightforward.

#### D. Outputs

From the viewpoint of a potential user, it would be desirable to include means for obtaining simultaneous multiple outputs--for example, both accumulated flight time and a count of landings in an event. This has not been done, primarily to keep the command input reasonably compact. In addition, it would complicate the formatting of the output, although this could be handled. As the complex is written, we have made no provision for the simultaneous loading of /COUNT/ and /DURATION/ or of other combinations of satellite programs. However, there is no reason it could not be done.

#### E. Working Files

As a possible means of handling D, above, and other important output options, it might be desirable to output in two stages, first to a working file and then to the actual printout or display. The working file would accumulate all potentially relevant data, however relevancy is defined, for all qualifying events. Actual output would then include only a small subset of this data, or appropriate summarizations of the data such as accumulated durations or counts or averages.

The use of a working file would permit other types of output, such as the display of histograms showing the distribution of (for example) flight durations or landing counts. We could then adaptively adjust cutoff values, such as determining what limiting duration of events accounts for, say, 98 percent of all events. This possibility has obvious significance for the final adjustment of a proposed standard.

A further advantage of a working file would be that it would facilitate looking in detail at exceptional events. If the user has tentatively established a standard that includes, say, 98 percent of all events, he would probably then want to look in detail at the remaining 2 percent. At present this can only be done by a complete rerun using the standard as a restriction to screen out the nonexceptional events.

#### F. Interactive Output

Given the working file discussed above, we could, in the general outline of Figure 1, split the interaction level into two functions, one specifying interactively what the working file should be, the other interactively specifying the output. If appropriate, the processing level could also be split, with separate programs constructing the working file and manipulating it for output.

If this were done, it would probably be worthwhile to retain the present system as an option, with the establishment of one, or possibly

more, working files as another option. That is, the top level of G, output, in Figure 1 would include the option of deferring specification of the output. The system would then execute the search procedure, subject to the rest of the input command, but transferring data to the working file rather than printing it out. Upon completion, control would be transferred to a separate program from which the user could interactively call for a succession of different outputs.

#### G. Utility File

It is a small step from the concept of a working file as outlined above to an on-going "utility file," which is kept up to date as new data are entered into the system. Such a file would come into existence with the tentative or permanent adoption of a standard, and would become the principal instrument for the on-going monitoring of operations versus that standard.

#### H. Anticipation

Given the capability of establishing utility files, there is need for a decision process that will manage the utility files. Initially, this would probably be under the sole control of the user. However, there is also the possibility of the system itself entering into this process. This possibility would become important as the number of users increases, and it becomes more likely that different users may set up similar utility files or that an independent query could be more efficiently handled in some existing utility file.

At this point we can only talk about this area of extension in rather vague terms, but the possibilities of it are being considered.

## VI CONCLUSIONS

We have constructed a scenario based on data on aircraft utilization and maintenance that illustrates a number of interesting capabilities and properties. Among these capabilities and properties the following may be noted:

- The system is interactive without requiring programming expertise by the user, or any knowledge of formatting conventions or the like.
- The system is responsive to a wide range of user commands.
- The program exhibits useful characteristics of "open-endedness." It is relatively simple to write new satellite programs that will collect and print out information that is not already available.
- The system exhibits a clear separation of the dialogue and execution modes.
- The system also exhibits a clear division into different functional modes, obtained through the calling of different satellite programs.
- These qualities are put into the context of a practical area of managerial concern using actual data.

We have also given a rather extensive list of possible modifications and extensions that could be made to the scenario. This list illustrates the inherent capabilities of the approach used. It also provides a "shopping list," which is currently under study to determine what should be done next to advance the research goals of the project.

Appendix A  
CONTROL INFORMATION

Appendix A  
Control Information

- D(1): A. Type-- AFPH only, trapped.
- D(2): B. A/C-- 1 if all  
2 if listed in B(N2)--trapped.
- D(3): C. Squadrons-- 1 if all  
2 if listed in C(N3)-- trapped.
- D(4): D. Start date-- not less than 0213
- D(5): End date-- not greater than 1201
- D(6): How end date is to be treated:  
1 if events must end on/before end date,  
2 if events need only start on/before end date.
- D(7): E. Defn. of event in terms of WUC:  
1 if any WUC  
2 if only listed WUC  
3 if any but listed WUC.  
If 2 or 3, list in M(N4)
- D(8): F. Restriction on event?  
1 if there is one,  
2 if not.
- D(9): Restriction applies to:  
1: # flts in event,  
2: # lndgs in event,  
3: accum. flt time in event.
- D(10): 1: All flt/lndg codes,  
2: Listed flt/lndg codes-- list as F(N5)
- D(11): The restriction itself, as a string.
- D(12): G. Output  
1: List of events,  
2: List of counts,  
3: List of accum. durations,  
4: Average counts,  
5: Average accum. durations.

If D(12) = 1-- list of events:

D(16) = 1 if list by A/C,  
2 if list by squad.

---

If D(12) = 2-- list of counts:

D(13) = 1 if count of events,  
2 if count of flights,  
3 if count of landings.

D(14) = 1 if all flt/lndg codes, (D(13) not 1)  
2 if applicable codes listed,  
3 if all but listed codes.

If applicable, list as G(N6)

D(16) = 1 if list count for each event,  
2 if list count for each A/C,  
3 if list count for each squad.,  
4 if give total count for all events.

---

If D(12) = 3-- list of durations:

D(13) = 1 if durations of events,  
2 if flight time in events.

D(14) = 1 if all flight codes, (D(13) not 1)  
2 if listed codes only  
3 if all but listed codes.

If applicable, list as G(N6)

D(16) = 1 if list accum. durations for each event,  
2 if list accum. durations for each A/C,  
3 if list accum. durations for each squad.,  
4 if list accum. duration for all events.

---

If D(12) = 4-- list of average counts:

D(15) = 1 if av. # of flts/lndgs per event, av. over all events,  
2 if av. # flts/lndgs per event for each A/C,  
3 if av. # flts/lndgs per event for each squad.,  
4 if av. # flts/lndgs per event for each A/C in each squad.,  
5 if av. # events/flts/lndgs per A/C for each squad.,

D(13) = 1 if # of flights,  
2 if # of landings,  
3 if # of events (if appropriate.)

D(14) = 1 if all flt/lndg codes  
2 if listed codes  
3 if all but listed codes

If appropriate, list as G(N6)

If D(12) = 5-- list of average accumulated durations:

D(15) = 1 if av. flt dur. per event for all events,  
2 if av. flt dur. per event for each A/C,  
3 if av. flt dur. per event for each squad.,  
4 if av. flt dur. per event for each A/C in each squad.,  
5 if av. dur. of events for each A/C,  
6 if av. dur. of events for each squad.,  
7 if av. dur. of events for each A/C in each squad.,  
8 if av. dur. of events over all events.

D(14) = 1 if all flight codes, (if applicable)  
2 if listed codes,  
3 if all but listed codes.

If appropriate, list as G(N6)



Appendix B  
CODE FOR THE PROGRAM COMPLEX

# Appendix B

## CODE FOR PROGRAM COMPLEX

1. /STANDARDS/

```

10 PRINT"                ****FLIGHT STANDARDS****"
20 PRINT
30 PRINT
40 PRINT " IF YOU NEED DETAILED INSTRUCTIONS, TYPE '1' FOLLOWED"
50 PRINT "BY THE RETURN KEY. ELSE TYPE '2' AND RETURN."
60 INPUT H
70 IF H=2 THEN 360
80 PRINT
90 PRINT " THIS IS A COMMAND PROGRAM TO FACILITATE THE DEVELOPMENT OF,"
100 PRINT "AND USE OF, STANDARDS OF FLIGHT AVAILABILITY FOR NAVAL
AIRCRAFT."
110 PRINT
120 PRINT " YOU WILL BE ASKED TO INDICATE THE TYPE OF AIRCRAFT TO BE
CONSIDERED,"
130 PRINT "THE PARTICULAR AIRCRAFT (A/C) AND SQUADRONS OF INTEREST,"
140 PRINT "THE WUC CODES OF INTEREST, THE INCLUSIVE DATES, AND THE DATA"
150 PRINT "BEING SOUGHT. YOU WILL BE ABLE, IF YOU WISH, TO PUT CONDITION
160 PRINT "ON APPROPRIATE PARTS OF YOUR SPECIFICATION."
170 PRINT
180 PRINT " IN THE FOLLOWING, AN 'EVENT' COVERS ONE AIRCRAFT FROM"
190 PRINT "THE TIME OF COMPLETION OF ONE OF THE DESIGNATED WUC CODES"
200 PRINT "TO THE START OF THE NEXT."
210 PRINT
220 PRINT " WHEN OPTIONS ARE OFFERED YOU, INDICATE YOUR CHOICE BY TYPING
230 PRINT "THE NUMBER OR LETTER PRECEEDING THE SELECTED OPTION,"
240 PRINT "AND THE RETURN KEY."
250 PRINT
260 PRINT " WHEN A NUMBER IS ASKED FOR, TYPE THE NUMBER YOU WANT"
270 PRINT "FOLLOWED BY THE RETURN KEY."
280 PRINT
290 PRINT " WHERE SEVERAL CODES ARE REQUIRED, TYPE THEM SEPERATED"
300 PRINT "BY COMMAS; THEN END BY THE RETURN KEY. IF YOU HAVE"
310 PRINT "NOT ENTERRED ENOUGH, THE SYSTEM WILL TYPE A QUESTION MARK"
320 PRINT "AND WAIT FOR MORE. IF YOU HAVE ENTERRED TOO MANY, THE"
330 PRINT "SYSTEM WILL IGNORE THE EXTRA ONES. IF YOU MAKE A MISTAKE,"
340 PRINT "CONTINUE. YOU WILL HAVE A CHANCE TO CORRECT IT BEFORE"
350 PRINT "CALLING FOR EXECUTION OF THE COMMAND."

360 PRINT
370 PRINT
380 PRINT"                **COMMAND INPUT**"
390 PRINT
400 VAR=ZERO
410 STRING D(16),U
420 TO=0
430 LINK /INTERROGATION/

```

2. /INTERROGATION/

```

10 AS="A"
20 BS="  "
30 CS="    "
40 DS="      "
50 ES="        "
60 FS="          "
70 IF T0>0 THEN 400
80 PRINT
90 GOSUB 1200
100 PRINT
110 GOSUB 3000
120 PRINT
130 GOSUB 3100
140 PRINT
150 GOSUB 3200
160 PRINT
170 GOSUB 3300
180 PRINT
190 GOSUB 3400
200 PRINT
210 GOSUB 3500
220 PRINT
230 GOSUB 3700
240 PRINT

400 PRINT "DO YOU WANT TO:"
410 PRINT IN IMAGE BS:"1. EXECUTE."
420 PRINT IN IMAGE BS:"2. ENTER A NEW COMMAND."
430 PRINT IN IMAGE BS:"3. MODIFY THE PRESENT COMMAND."
440 PRINT IN IMAGE BS:"4. SEE A SUMMARY OF THE PRESENT COMMAND."
450 PRINT IN IMAGE BS:"5. SAVE THE PRESENT COMMAND."
455 PRINT IN IMAGE BS:"6. END":
460 INPUT X
470 PRINT
480 IF X=1 THEN 1000 ELSE IF X=2 THEN 80 ELSE IF X=3
    THEN 600 ELSE IF X=4 THEN 1100 ELSE IF X=5 THEN LINK
    /SAVESUMM/ ELSE END
600 PRINT "WHICH SECTION DO YOU WANT TO CHANGE?"
610 PRINT "TYPE THE APPROPRIATE LETTER-- ONE ONLY":
620 INPUT U
630 PRINT
640 IF U="A" THEN GOSUB 3000 ELSE IF U="B" THEN GOSUB 3100 ELSE
    IF U="C" THEN GOSUB 3200 ELSE IF U="D" THEN GOSUB 3300 ELSE
    IF U="E" THEN GOSUB 3400 ELSE IF U="F" THEN GOSUB 3500 ELSE
    GOSUB 3700
650 PRINT
660 PRINT "DOES THAT COMPLETE THE CHANGES?"
670 PRINT IN IMAGE BS:"1. YES."
680 PRINT IN IMAGE BS:"2. NO":
690 INPUT U
700 PRINT
710 IF U="1" THEN 400
720 GO TO 600

```

```

1000 PRINT
1010 TO=1
1020 LINK /EXECUTE/

1100 PRINT
1110 TO=1
1120 LINK /SUMMARY/

1200 PRINT "DO YOU WANT TO ENTER : "
1210 PRINT IN IMAGE BS:"1. A NEW COMMAND."
1220 PRINT IN IMAGE BS:"2. A PREVIOUSLY SAVED COMMAND":
1230 INPUT U
1240 IF U="2" THEN LINK /DATAENT/ ELSE RETURN

3000 PRINT IN IMAGE AS:
      "(A) WHAT TYPE OF AIRCRAFT DO YOU WANT TO CONSIDER":
3010 INPUT D(1)
3020 IF D(1)="AFPH" THEN RETURN
3030 PRINT "DATA ON TYPE "D(1)":" NOT AVAILABLE. ONLY HAVE AFPH."
3040 PRINT IN IMAGE BS:"TYPE":
3050 GO TO 3010

3100 PRINT IN IMAGE AS:"(B) DO YOU WANT TO CONSIDER"
3110 PRINT IN IMAGE BS:"1. ALL A/C OF TYPE "+D(1)+"."
3112 PRINT IN IMAGE BS:"2. CERTAIN A/C OF TYPE "+D(1):
3120 INPUT D(2)
3125 IF D(2)="1" THEN RETURN
3130 PRINT IN IMAGE CS:"HOW MANY A/C DO YOU WANT TO LIST":
3135 INPUT N2
3136 IF N2>3 THEN 3190
3140 STRING B(N2)
3145 PRINT IN IMAGE CS:"LIST THE A/C DESIRED:"
3150 MAT INPUT B
3155 L1="153808",L2="155903",L3="155799"
3165 FOR I2=1 TO N2
3170 IF B(I2)=L1 THEN 3180
3171 IF B(I2)=L2 THEN 3180
3172 IF B(I2)=L3 THEN 3180
3174 PRINT "AVAILABLE DATA INCLUDES A/C ("L1", "L2", "L3:")"
3175 GO TO 3145
3180 NEXT I2
3185 RETURN
3190 PRINT "DATA INCLUDES ONLY 3 A/C."
3195 GO TO 3130

3200 PRINT IN IMAGE AS:"(C) DO YOU WANT TO CONSIDER"
3210 PRINT IN IMAGE BS:"1. ALL SQUADRONS."
3212 PRINT IN IMAGE BS:"2. CERTAIN SQUADRONS":
3220 INPUT D(3)
3225 IF D(3)="1" THEN RETURN
3230 PRINT IN IMAGE CS:"HOW MANY SQUADRONS":
3235 INPUT N3
3236 IF N3>3 THEN 3290
3240 STRING C(N3)
3245 PRINT IN IMAGE CS:"LIST THE SQUADRONS DESIRED:"
3250 MAT INPUT C

```

```

3255 L1="AC1",L2="AC7",L3="PC1"
3265 FOR I2=1 TO N3
3270 IF C(I2)=L1 THEN 3280
3271 IF C(I2)=L2 THEN 3280
3272 IF C(I2)=L3 THEN 3280
3274 PRINT "AVAILABLE DATA INCLUDES SQUADRONS ("I1:", "I2:", "I3:")"
3275 GO TO 3245
3280 NEXT I2
3285 RETURN
3290 PRINT "DATA INCLUDES ONLY 3 SQUADRONS."
3295 GO TO 3230

3300 PRINT IN IMAGE A3:"(D) WHAT TIME PERIOD DO YOU WANT?"
3301 PRINT IN IMAGE A3:
      "TYPE THE JULIAN DATES, SEPERATED BY COMMAS.":
3305 INPUT D(4),D(5)
3310 IF LENGTH(D(4))<>4 THEN 3380
3311 IF LENGTH(D(5))<>4 THEN 3380
3315 IF VAL(D(4))>VAL(D(5)) THEN 3350
3320 IF VAL(D(4))<213 THEN 3340
3321 IF VAL(D(5))>1201 THEN 3340
3325 PRINT IN IMAGE B3:
      "DO YOU WANT THE FINAL EVENTS TO BE:"
3326 PRINT IN IMAGE C3:"1. ENDED ON/BEFORE THE FINAL DATE."
3327 PRINT IN IMAGE C3:"2. STARTED ON/BEFORE THE FINAL DATE":
3330 INPUT D(6)
3335 RETURN
3340 PRINT "AVAILABLE DATA IS FROM 0213 TO 1201, INCLUSIVE."
3345 PRINT "WHAT TIME PERIOD":
3346 GO TO 3305
3350 PRINT "THE STARTING DATE IS AFTER THE BEGINNING ONE."
3351 GO TO 3345
3380 PRINT
      "THESE ARE NOT JULIAN DATES. THEY SHOULD BE 4-DIGIT NUMBERS."
3381 PRINT
      "THE FIRST DIGIT IS THE LAST DIGIT OF THE YEAR. THE OTHER"
3382 PRINT
      "DIGITS ARE THE DAY NUMBER IN THE YEAR."
3390 GO TO 3345

3400 PRINT IN IMAGE A3:"(E) HOW ARE EVENTS TO BE DEFINED?"
3401 PRINT IN IMAGE B3:"1. BY ANY WUC."
3402 PRINT IN IMAGE B3:"2. BY A PARTICULAR SET OF WUC CODES."
3403 PRINT IN IMAGE B3:"3. BY ANY WUC EXCEPT A PARTICULAR SET":
3410 INPUT D(7)
3420 IF D(7)="1" THEN RETURN ELSE IF D(7)="2" THEN A3="INCLUDED:"
      ELSE A3="EXCLUDED:"
3430 PRINT IN IMAGE C3:"HOW MANY CODES DO YOU WANT TO LIST":
3435 INPUT N4
3440 STRING M(N4)
3450 PRINT IN IMAGE C3:"LIST THE CODES TO BE "+A3
3460 MAT INPUT M
3462 FOR I=1 TO N4
3464 IF LENGTH(M(I))<>5 THEN 3480
3466 NEXT I

```

```

3470 RETURN
3480 PRINT "WUC CODES ARE 5-DIGIT STRINGS."
3490 GO TO 3440

3500 PRINT IN IMAGE A$;
      "(F) DO YOU WANT TO LIMIT ATTENTION TO EVENTS MEETING SOME CONDITION?"
3501 PRINT IN IMAGE B$;"1. YES."
3502 PRINT IN IMAGE B$;"2. NO";
3503 INPUT D(8)
3504 IF D(8)="2" THEN RETURN
3510 PRINT IN IMAGE B$;"DOES THE CONDITION APPLY TO:"
3511 PRINT IN IMAGE C$;"1. THE NUMBER OF FLIGHTS IN THE EVENT."
3512 PRINT IN IMAGE C$;"2. THE NUMBER OF LANDINGS IN THE EVENT."
3513 PRINT IN IMAGE C$;"3. THE FLIGHT TIME IN THE EVENT";
3514 INPUT D(9)
3515 IF D(9)="2" THEN A3="LANDING" ELSE A3="FLIGHT"
3516 PRINT IN IMAGE C$;"DO YOU WANT:"
3517 PRINT IN IMAGE D$;"1. ALL "+A3+"S."
3518 PRINT IN IMAGE D$;"2. ONLY CERTAIN "+A3+" CODES CONSIDERED."
3519 PRINT IN IMAGE D$;"3. ALL EXCEPT CERTAIN "+A3+" CODES";
3520 INPUT D(10)
3521 IF D(10)="1" THEN 3580
3522 PRINT IN IMAGE D$;"HOW MANY "+A3+" CODES";
3523 INPUT N5
3524 STRING F(N5)
3525 PRINT IN IMAGE D$;"LIST THE "+A3+" CODES INVOLVED:"
3526 MAT INPUT F
3527 PRINT IN IMAGE D$;"WHAT IS THE RESTRICTION?"
3528 PRINT IN IMAGE D$;"THE OPERATORS =, <, >, <=>, <=>, >=> MAY BE USED."
3529 PRINT IN IMAGE D$;"THE FORM OF THE INPUT SHOULD BE, E.G., X>3.25."
3530 PRINT IN IMAGE D$;"IF TIME IS INVOLVED, USE DECIMAL HOURS."
3531 INPUT D(11)
3532 RETURN

3700 PRINT IN IMAGE A$;"(G) WHAT DATA IS TO BE OUTPUTTED?"
3701 PRINT IN IMAGE B$;"1. A LIST OF EVENTS."
3702 PRINT IN IMAGE B$;"2. A LIST OF COUNTS"
3703 PRINT IN IMAGE B$;"3. A LIST OF ACCUMULATED DURATIONS."
3704 PRINT IN IMAGE B$;"4. A LIST OF AVERAGE COUNTS."
3705 PRINT IN IMAGE B$;"5. A LIST OF AVERAGE DURATIONS";
3706 INPUT D(12)
3707 IF D(12)="1" THEN GOSUB 4000 ELSE IF D(12)="2" THEN 4100 ELSE
      IF D(12)="3" THEN 4200 ELSE IF D(12)="4" THEN 4300 ELSE 4500
3708 RETURN

4000 PRINT IN IMAGE B$;"DO YOU WANT EVENTS GROUPED FIRST BY:"
4001 PRINT IN IMAGE C$;"1. A/C."
4002 PRINT IN IMAGE C$;"2. SQUADRON";
4003 INPUT D(13)
4004 RETURN

4100 GOSUB 4650
4101 IF D(13)="1" THEN 4130
4102 GOSUB 4700
4103 GOSUB 4900
4104 RETURN

```

```

4200 GOSUB 4600
4210 IF D(13)="1" THEN 4230
4220 GOSUB 4700
4230 GOSUB 5000
4240 RETURN

```

```

4300 PRINT IN IMAGE B$:"WHAT AVERAGE DO YOU WANT?"
4310 PRINT IN IMAGE C$:
      "1. THE AV. NO. OF FLTS/LDGS PER EVENT FOR ALL EVENTS."
4320 PRINT IN IMAGE C$:
      "2. THE AV. NO. OF FLTS/LDGS PER EVENT FOR EACH A/C."
4330 PRINT IN IMAGE C$:
      "3. THE AV. NO. OF FLTS/LDGS PER EVENT FOR EACH SQUADRON."
4340 PRINT IN IMAGE C$:
      "4. THE AV. NO. OF FLTS/LDGS PER EVENT FOR EACH A/C IN EACH SQ."
4350 PRINT IN IMAGE C$:
      "5. THE AV. NO. OF EVENTS/FLTS/LDGS PER A/C FOR EACH SQ.":
4360 INPUT D(15)
4370 PRINT IN IMAGE C$: "IN THE ABOVE, DO YOU WANT:"
4380 PRINT IN IMAGE D$:"1. FLIGHTS."
4390 IF D(15)="5" THEN 4420
4400 PRINT IN IMAGE D$:"2. LANDINGS":
4410 G0 T0 4440
4420 PRINT IN IMAGE D$:"2. LANDINGS."
4430 PRINT IN IMAGE D$:"3. EVENTS":
4440 INPUT D(13)
4450 IF D(13)="1" THEN A4="FLIGHT" ELSE IF D(13)="2" THEN
      A4="LANDING" ELSE RETURN
4460 G0 T0 4700

```

```

4500 PRINT IN IMAGE B$:"WHAT AVERAGE DO YOU WANT?"
4505 PRINT IN IMAGE C$:
      "1. THE AV. FLT DUR. PER EVENT FOR ALL EVENTS."
4510 PRINT IN IMAGE C$:
      "2. THE AV. FLT DUR. PER EVENT FOR EACH A/C."
4515 PRINT IN IMAGE C$:
      "3. THE AV. FLT DUR. PER EVENT FOR EACH SQUADRON."
4520 PRINT IN IMAGE C$:
      "4. THE AV. FLT DUR. PER EVENT FOR EACH A/C IN EACH SQ."
4525 PRINT IN IMAGE C$:
      "5. THE AV. DURATION OF EVENTS FOR EACH A/C."
4530 PRINT IN IMAGE C$:
      "6. THE AV. DURATION OF EVENTS FOR EACH SQUADRON."
4535 PRINT IN IMAGE C$:
      "7. THE AV. DURATION OF EVENTS FOR EACH A/C IN EACH SQ."
4540 PRINT IN IMAGE C$:
      "8. THE AV. DURATION OF EVENTS FOR ALL EVENTS":
4545 INPUT D(15)
4550 A4="FLIGHT"
4555 IF VAL(D(15))<5 THEN 4700 ELSE RETURN

```

```

4600 PRINT IN IMAGE B$:"DO YOU WANT:"
4605 PRINT IN IMAGE C$:"1. THE DURATION OF EVENTS."
4610 PRINT IN IMAGE C$:"2. FLIGHT TIME IN THE EVENT":
4615 INPUT D(13)
4620 A4="FLIGHT"
4625 RETURN

```

```

4650 PRINT IN IMAGE B3:"DO YOU WANT COUNTS OF:"
4655 PRINT IN IMAGE C3:"1. EVENTS."
4660 PRINT IN IMAGE C3:"2. FLIGHTS."
4665 PRINT IN IMAGE C3:"3. LANDINGS:"
4670 INPUT D(13)
4680 IF D(13)="2" THEN A4="FLIGHT" ELSE A4="LANDING"
4690 RETURN

```

```

4700 PRINT IN IMAGE C3:"DO YOU WANT TO INCLUDE:"
4705 PRINT IN IMAGE D3:"1. ALL "+A4+" CODES."
4710 PRINT IN IMAGE D3:"2. ONLY CERTAIN "+A4+" CODES."
4715 PRINT IN IMAGE D3:"3. ALL EXCEPT CERTAIN "+A4+" CODES:"
4720 INPUT D(14)
4725 IF D(14)="1" THEN RETURN
4730 PRINT IN IMAGE D3:"HOW MANY CODES:"
4735 INPUT N6
4740 STRING G(N6)
4745 PRINT IN IMAGE D3:"LIST THE "+A4+" CODES INVOLVED:"
4750 MAT INPUT G
4755 RETURN

```

```

4900 PRINT IN IMAGE D3:"SHOULD THE COUNTS BE MADE OVER:"
4910 PRINT IN IMAGE E3:"1. EACH EVENT."
4920 PRINT IN IMAGE E3:"2. THE HISTORY OF EACH A/C."
4930 PRINT IN IMAGE E3:"3. THE HISTORY OF EACH SQUADRON."
4940 PRINT IN IMAGE E3:"4. ALL OCCURENCES:"
4950 INPUT D(16)
4960 RETURN

```

```

5000 PRINT IN IMAGE D3:"SHOULD THE DURATIONS BE ACCUMULATED OVER:"
5010 GO TO 4910

```



3. /SUMMARY/

```

10 VAR=ZERO
20 PRINT"                                     **SUMMARY**"
30 GOSUB 2000
40 GOSUB 2100
50 GOSUB 2200
60 GOSUB 2300
70 GOSUB 2400
80 PRINT"                                     **END OF SUMMARY**"
90 PRINT

100 PRINT "ARE YOU READY TO EXECUTE?"
120 PRINT IN IMAGE BS:"1. YES."
130 PRINT IN IMAGE BS:"2. NO:"
135 PRINT
140 INPUT X
150 TO=2
160 IF X=1 THEN LINK /EXECUTE/ ELSE LINK /INTERROGATION/

2000 PRINT "A. TYPE = ":D(1)

2005 PRINT "B. ALL AIRCRAFT OF TYPE." IF D(2)="1"
2010 IF D(2)="1" THEN 2050
2020 PRINT "B. AIRCRAFT NUMBERED:" IF D(2)="2"
2025 PRINT IN IMAGE BS:B(1) FOR I=1 TO N2

2050 PRINT "C. ALL SQUADRONS." IF D(3)="1"
2055 IF D(3)="1" THEN RETURN
2060 PRINT "C. SQUADRONS NUMBER:" IF D(3)="2"
2065 PRINT IN IMAGE BS:C(1) FOR I=1 TO N3
2070 RETURN

2100 PRINT"D. TIME INTERVAL:"
2110 PRINT IN IMAGE BS:"STARTING DATE = "+D(4)
2120 PRINT IN IMAGE BS:"TERMINAL DATE = "+D(5)
2130 PRINT IN IMAGE CS:"ONLY EVENTS ENDING ON/BEFORE TERMINAL
    DATE INCLUDED." IF D(6)="1"
2131 PRINT IN IMAGE CS:"EVENTS STARTED ON/BEFORE TERMINAL
    DATE INCLUDED." IF D(6)="2"
2140 RETURN

2200 PRINT "E. AN 'EVENT' IS DEFINED BY:"
2210 PRINT IN IMAGE BS:"ANY WUC" IF D(7)="1"
2211 PRINT IN IMAGE BS:"THE FOLLOWING WUC CODES:" IF D(7)="2"
2212 PRINT IN IMAGE BS:"ANY WUC CODE EXCEPT THE FOLLOWING:"
    IF D(7)="3"
2220 IF D(7)="1" THEN RETURN
2230 PRINT IN IMAGE CS:M(1) FOR I=1 TO N4
2240 RETURN

2300 PRINT "F. NO RESTRICTION ON APPLICABLE DATA." IF D(8)="2"
2310 IF D(8)="2" THEN RETURN
2320 PRINT "E. THE DATA IS RESTRICTED BY:"D(11)

```

```

2330 IF D(9)="1" THEN B1=" THE NUMBER OF FLIGHTS IN AN EVENT."
      ELSE IF D(9)="2" THEN B1=" THE NUMBER OF LANDINGS IN AN EVENT."
      ELSE B1=" THE FLIGHT TIME IN THE EVENT."
2340 PRINT IN IMAGE B3:"WHERE X IS"+B1
2350 IF D(9)="2" THEN C1="LANDINGS" ELSE C1="FLIGHTS"
2360 IF D(10)="1" THEN E1=" ALL CODES." ELSE IF D(10)="2" THEN
      E1=" THE FOLLOWING CODES:"
      ELSE E1=" ALL EXCEPT THE FOLLOWING CODES:"
2370 PRINT IN IMAGE C3:"THE "+C1+" APPLICABLE HERE INCLUDE"+E1
2380 F1=F(1)
2391 IF N5=1 THEN 2385
2392 F1=F1+", "+F(1) FOR I=2 TO N5
2395 PRINT IN IMAGE D3:F1+"."
2390 RETURN

2400 PRINT "3. THE OUTPUT IS TO BE:"
2410 IF D(12)="1" THEN GOSUB 2500 ELSE IF D(12)="2" THEN GOSUB
      2600 ELSE IF D(12)="3" THEN GOSUB 2700 ELSE IF D(12)="4"
      THEN GOSUB 2800 ELSE GOSUB 2900
2420 RETURN

2500 IF D(16)="2" THEN 2550
2510 PRINT IN IMAGE B3:"EVENTS LISTED BY A/C."
2520 RETURN

2550 PRINT IN IMAGE B3:"EVENTS LISTED BY SQUADRON."
2560 RETURN

2600 IF D(13)="1" THEN A4="EVENT" ELSE IF D(13)="2" THEN A4="FLIGHT"
      ELSE A4="LANDING"
2610 PRINT IN IMAGE B3:"COUNTS OF "+A4+"S."
2620 IF D(13)="1" THEN 2650 ELSE IF D(13)="2" THEN GOSUB 3000 ELSE
      GOSUB 3100
2650 IF D(16)="1" THEN A4="EACH EVENT." ELSE IF D(16)="2" THEN
      A4="EACH A/C." ELSE IF D(16)="3" THEN A4="EACH SQUADRON." ELSE
      A4="ALL OCCURENCES."
2660 PRINT IN IMAGE C3:"THE COUNTS ARE TO BE ACCUMULATED OVER "+A4
2670 RETURN

2700 A2="THE ACCUMULATED "
2705 IF D(13)="2" THEN B2="FLIGHT TIME" ELSE B2=
      "DURATION OF THE EVENTS"
2710 IF D(16)="1" THEN C2="EACH EVENT." ELSE IF D(16)="2" THEN
      C2="EACH A/C." ELSE IF D(16)="3" THEN C2="EACH SQUADRON."
      ELSE C2="ALL OCCURENCES."
2720 PRINT IN IMAGE B3:A2+B2+" TOTALLED OVER "+C2
2730 IF D(13)="2" THEN GOSUB 3000
2740 RETURN

2800 IF D(13)="1" THEN B2="FLIGHTS " ELSE IF D(13)="2" THEN
      B2="LANDINGS " ELSE B2="EVENTS "
2810 IF D(15)="1" THEN PRINT
      " THE AV. NO. OF "B2;" PER EVENT FOR ALL EVENTS."
2820 IF D(15)="2" THEN PRINT
      " THE AV. NO. OF "B2;" PER EVENT FOR EACH A/C."

```

```

2830 IF D(15)="3" THEN PRINT
    " THE AV. NO. OF "B2" PER EVENT FOR EACH SQUADRON."
2840 IF D(15)="4" THEN PRINT
    " THE AV. NO. OF "B2" PER EVENT FOR EACH A/C IN EACH SQUADRON"
2850 IF D(15)="5" THEN PRINT
    " THE AV. NO. OF "B2" PER A/C IN EACH SQUADRON."
2860 IF D(13)="1" THEN 3000 ELSE IF D(13)="2" THEN 3100 ELSE RETURN

2900 IF VAL(D(15))<5 THEN A2="FLIGHT DURATION PER EVENT" ELSE
    A2="DURATION OF EVENTS"
2910 IF D(15)="1" THEN B2="ALL EVENTS." ELSE IF D(15)="2" THEN
    B2="EACH A/C." ELSE IF D(15)="3" THEN B2="EACH SQUADRON" ELSE
    IF D(15)="4" THEN B2="EACH A/C IN EACH SQUADRON."
2920 IF D(15)="5" THEN B2="EACH A/C." ELSE IF D(15)="6" THEN B2=
    "EACH SQUADRON." ELSE IF D(15)="7" THEN B2="EACH A/C IN EACH SQ."
    ELSE B2="ALL EVENTS."
2930 PRINT" THE AVERAGE "A2" FOR "B2
2940 IF VAL(D(15))<5 THEN 3000 ELSE RETURN

3000 PRINT IN IMAGE C$ "ALL FLIGHT CODES INCLUDED." IF D(14)="1"
3010 PRINT IN IMAGE C$ "THE FOLLOWING FLIGHT CODES TO BE INCLUDED:"
    IF D(14)="2"
3015 PRINT IN IMAGE C$ "ALL FLIGHT CODES EXCEPT THE FOLLOWING:"
    IF D(14)="3"
3020 IF D(14)="1" THEN RETURN
3030 G2=G(1)
3040 IF N6=1 THEN 3060
3050 G2=G2+", "G(1) FOR I = 2 TO N6
3060 PRINT IN IMAGE D$G2+", "
3070 RETURN

3100 PRINT IN IMAGE C$ "ALL LANDING CODES INCLUDED." IF D(14)="1"
3110 PRINT IN IMAGE C$ "THE FOLLOWING LANDING CODES TO BE INCLUDED:"
    IF D(14)="2"
3115 PRINT IN IMAGE C$ "ALL LANDING CODES EXCEPT THE FOLLOWING:"
    IF D(14)="3"
3120 IF D(14)="1" THEN RETURN ELSE 3030

```

```

4.  /SAVESUMM/

10 STRING UI
50 PRINT "UNDER WHAT FILE NAME DO YOU WANT THE SUMMARY SAVED?":
60 INPUT U
70 FOR I=1 TO 11
75 READ UI
80 IF U=UI THEN 1000
85 NEXT I
90 DATA /STANDARDS//INTERROGATION//SUMMARY//EXECUTE//
  /RESTRICT//COUNT//AV-CT//DURATION//AV-DUR//SAVESUMM//
  /DATAENT/

100 OPEN U,OUTPUT,1
200 WRITE ON 1:N2,N3,N4,N5,N6,TO

300 FOR I= 1 TO 16
310 WRITE ON 1:D(I)
320 NEXT I

400 IF N2=0 THEN 500
410 FOR I=1 TO N2
420 WRITE ON 1:B(I)
430 NEXT I

500 IF N3=0 THEN 600
510 FOR I=1 TO N3
520 WRITE ON 1:C(I)
530 NEXT I

600 IF N4=0 THEN 700
610 FOR I=1 TO N4
620 WRITE ON 1:M(I)
630 NEXT I

700 IF N5=0 THEN 800
710 FOR I=1 TO N5
720 WRITE ON 1:F(I)
730 NEXT I

800 IF N6=0 THEN 900
810 FOR I=1 TO N6
820 WRITE ON 1:G(I)
830 NEXT I

900 CLOSE 1
950 PRINT
960 PRINT"THE DATA CAN NOW BE RE-ENTERED BY THE DIRECT COMMAND"
970 PRINT"'LINK /DATAENT/' WHEN YOU WILL BE ASKED FOR THE NAME."
975 PRINT
980 PRINT "DO YOU WANT TO END OFF NOW?"
985 PRINT IN IMAGE BS:"1. YES."
990 PRINT IN IMAGE BS:"2. NO":
991 INPUT X
992 TO=1
995 IF X=2 THEN LINK /INTERROGATION/ ELSE LINK /ADAPTATION/
1000 RESTORE

```

```
1010 PRINT
1020 PRINT"THE NAME YOU HAVE CHOSEN IS RESERVED."
1030 PRINT "THE LIST OF RESERVED NAMES IS:"
1040 FOR I=1 TO 11
1050 READ U1
1060 PRINT IN IMAGE CS:U1
1070 NEXT I
1080 PRINT
1090 PRINT"CHOOSE A NEW NAME:"
1100 RESTORE
1110 GO TO 60
```

5. /DATAENT/

```
10 STRING U
50 PRINT "UNDER WHAT FILE NAME IS YOUR COMMAND SAVED?"
60 INPUT U
100 VAR=ZERO
110 INTEGER N2,N3,N4,N5,N6,T0
120 OPEN U,INPUT,1
130 ON ENDFILE(1) GO TO 900
200 INPUT FROM 1:N2,N3,N4,N5,N6,T0
210 STRING D(16), B(N2+1),C(N3+1),M(N4+1),F(N5+1),G(N6+1)

300 FOR I=1 TO 16
310 INPUT FROM 1 IN FORM "R":D(I)
320 NEXT I

400 IF N2=0 THEN 500
410 FOR I=1 TO N2
420 INPUT FROM 1 IN FORM "R":B(I)
430 NEXT I

500 IF N3=0 THEN 600
510 FOR I=1 TO N3
520 INPUT FROM 1 IN FORM "R":C(I)
530 NEXT I

600 IF N4=0 THEN 700
610 FOR I=1 TO N4
620 INPUT FROM 1 IN FORM "R":M(I)
630 NEXT I

700 IF N5=0 THEN 800
710 FOR I=1 TO N5
720 INPUT FROM 1 IN FORM "R":F(I)
730 NEXT I

800 IF N6=0 THEN 900
810 FOR I=1 TO N6
820 INPUT FROM 1 IN FORM "R":G(I)
830 NEXT I

900 AS=""
910 BS=""
920 CS=""
930 DS=""
940 ES=""
950 FS=""
1000 CLOSE 1

2000 T0=1
2010 PRINT"DO YOU WANT TO:"
2020 PRINT IN IMAGE BS:"1. EXECUTE."
2030 PRINT IN IMAGE BS:"2. SEE A SUMMARY."
2040 PRINT IN IMAGE BS:"3. OTHER:"
2050 INPUT X
2060 IF X=1 THEN LINK /EXECUTE/ ELSE IF X=2 THEN LINK /SUMMARY/
    ELSE LINK /INTERROGATION/
```

6. /EXECUTE/

```

5 STRING U,UO,W(50)
10 STRING V,V0,V1,V2,V3
12 IF D(8)="1" THEN LOAD /RESTRICT/
15 TO=4      !***
20 IF D(12)="2" THEN LOAD /COUNT/ ELSE IF D(12)="3" THEN LOAD
   /DURATION/ ELSE IF D(12)="4" THEN LOAD /COUNT/ ELSE IF
   D(12)="5" THEN LOAD /DURATION/  !*** INDICATES SWITCH NODE.
25 D8=VAL(D(4)),D9=VAL(D(5))  !START AND END DATES.
30 IF D(16)="1" THEN A1="A/C" ELSE A1="SQUADRON"      !***
35 IF D(16)="1" THEN A2="SQUADRON" ELSE A2="A/C"      !***
40 IF D(16)="1" THEN A0="ACDIR" ELSE A0="S0DIR"      !***
45 IF D(16)="1" THEN S0=1 ELSE S0=0  !FLAG WHETHER TO LINK FILES OR NOT.
50 PRINT"
55 PRINT
60 PRINT "LIST OF QUALIFYING EVENTS."  !***
65 PRINT
70 PRINT"
100 OPEN A0,INPUT,1
110 INPUT FROM 1:NO,M0
120 STRING K(NO)
130 MAT INPUT FROM 1:K
140 CLOSE 1      !DIR. LOADED AS K

200 FOR IO=1 TO M0
210 S1=0,S2=0,S3=0,S4=0  !S1=WHETHER PAST INITIAL DATE OR NOT;
   S2=WHETHER HAVE CERTAIN (=1) OR POSSIBLE (=2) BEG. OF EVENT;
   S3=WHETHER HAVE END OF EVENT OR NOT;
   S4=WHETHER INITIAL OR SUBSEQUENT FILE.
220 UO=K(IO)  !INITIAL ENTRY IN DIR LIST
230 GOSUB 1800
240 IF T2=0 THEN 500  !GET NEXT FILE IN DIR LIST.
250 IF S0=0 THEN S2=0,S3=0
260 V=SUBSTR(UO,K1+1,6)  !FILE NAME
270 GO TO 700

300 !*** END OF DIR. LIST
310 NEXT IO
320 GO TO 600

500 JO=VAL(RIGHT(UO,1)),S4=1
510 IF JO=0 THEN 300
520 UO=K(JO)
530 GOSUB 1800
540 IF T2=0 THEN 500
550 V=SUBSTR(UO,K1+1,6)  !FILE NAME
560 GO TO 700

600 PRINT
610 PRINT "EXECUTION COMPLETE."
620 PRINT"
630 PRINT
640 LINK /INTERROGATION/

```

```

700 OPEN V,INPUT,1
710 INPUT FROM 1:XI
720 ON ENDFILE(1) GO TO 790
730 I1=0
740 I3=50
750 FOR I2=1 TO 50
760 INPUT FROM 1:H(I2)
770 NEXT I2
780 GO TO 1000
790 I3=I2-1
800 IF I3=0 THEN 900
810 GO TO 1000

900 IF S0=0 THEN S1=0,S2=0,S3=0
910 *** END OF FILE
920 CLOSE 1
930 GO TO 500

1000 FOR I2=1 TO I3  ISEARCH PAGE. PASS INITIAL DATE
1010 I1=I1+1
1020 IF I1>XI THEN 900
1030 IF S1=1 THEN 1200  IALREADY PAST
1040 IF LEFT(H(I2),1)="6" THEN 1080
1050 IF VAL(SUBSTR(H(I2),2,4))<D8 THEN 1080
1060 IF D(7)="1" THEN 1100 ELSE GOSUB 2000
1070 IF T1=1 THEN 1100
1080 NEXT I2
1090 GO TO 740

1100 S1=1      IPAST STARTING DATE
1110 IF S4=0 THEN 5000 ELSE 5100  *** HEADING
1120 GO TO 1240

1200 IF S2=1 THEN 1400  IBLOCK TO FIND BEG OF EVENT
1210 IF LEFT(H(I2),1)="6" THEN 1600
1220 IF D(7)="1" THEN 1240 ELSE GOSUB 2000
1230 IF T1=0 THEN 1260
1240 D1=SUBSTR(H(I2),2,4),E1=SUBSTR(H(I2),10,4),S2=2
1250 GO TO 1080
1260 IF S2=2 THEN S2=1 ELSE IF S2=0 THEN 1080

1300 *** BRANCH TO /RESTRICT/
1310 *** CERTAIN BEG. OF EVENT
1320 GO TO 1080

1400 IF LEFT(H(I2),1)="6" THEN IF S2=0 THEN 1080 ELSE 1600
      I BLOCK TO SEEK END OF EVENT.
1410 IF D(7)="1" THEN 1430 ELSE GOSUB 2000
1420 IF T1=0 THEN 1080  IEND NOT FOUND
1430 F1=SUBSTR(H(I2),2,4),G1=SUBSTR(H(I2),6,4),S3=1
1450 IF D(6)="1" THEN D2=F1 ELSE D2=D1
1460 IF VAL(D2)>D9 THEN 900  IT00 LATE
1465 GO TO 1500
1470 D1=SUBSTR(H(I2),2,4),E1=SUBSTR(H(I2),10,4),S2=2
1480 GO TO 1080

```



```

1500 I** BRANCH TO /RESTRICT/
1510 GO TO 6000 I** END OF EVENT.
1530 S3=0
1540 GO TO 1470

1600 IF S2=2 THEN S2=1 ELSE IF S2=0 THEN 1080
1610 I** BRANCH TO /RESTRICT/
1620 I** FLT CARD
1630 GO TO 1080

1800 T2=0 ISUB TO CHECK IF A/C AND SQ. IN DESIRED LISTS
1810 K0=INDEX(U0,"-"),K1=INDEX(U0,"/")
1820 V0=LEFT(U0,K0-1),V1=SUBSTR(U0,K0+1,K1-K0-1) INAMES OF A/C & SQ.
1830 IF A0="SODIR" THEN V2=V1 ELSE V2=V0 INAME OF A/C
1840 IF A0="SODIR" THEN V3=V0 ELSE V3=V1 INAME OF SQ.
1850 IF D(2)="1" THEN 1900
1860 FOR M1=1 TO N2
1870 IF V2=B(M1) THEN 1900
1880 NEXT M1
1890 RETURN
1900 IF D(3)="1" THEN 1950
1910 FOR M1=1 TO N3
1920 IF V3=C(M1) THEN 1950
1930 NEXT M1
1940 RETURN
1950 T2=1
1960 RETURN

2000 ICHECKS WUC AGAINST M(N4)
2010 T1=1
2030 W=SUBSTR(H(12),25,5)
2040 FOR M2=1 TO N4
2050 IF W=M(M2) THEN T1=0
2060 NEXT M2
2070 IF D(7)="3" THEN RETURN
2080 T1=1-T1
2090 RETURN

5000 PRINT
5010 PRINT A1:" NUMBER "V0:"
5020 PRINT
5030 IF D(16)="2" THEN PRINT IN IMAGE B$A2+" NUMBER "+V1+"
5040 PRINT
5050 PRINT"BEG. BEG. END END"
5060 PRINT"DATE TIME DATE TIME"
5070 PRINT
5080 GO TO 1240
5100 PRINT
5110 PRINT IN IMAGE B$A2+" NUMBER "+V1+"
5120 PRINT
5130 GO TO 1240

6000 PRINT IN IMAGE"XXX XXXX XXXX XXXX":D1,E1,F1,G1
6010 GO TO 1470

```

7. /RESTRICT/

```

12 T3=0,X=0 IT3=RESTRICTION FLAG,X RESTRICTION VARIABLE
80 GOSUB 9000
1300 X=0,T3=0 !INITIALIZE RESTRICTION
1500 ON T3 GO TO 10000,10010,10020,10030,10040,10050
1530 S3=0,X=0
1610 IF D(9)="2" THEN 8200 ELSE 8000
8000 IF D(10)="1" THEN 8060
8010 W=SUBSTR(H(I2),21,3)
8020 FOR J=1 TO N5
8030 IF W=F(J) THEN 8060
8040 NEXT J
8050 GO TO 1620

8060 IF D(9)="3" THEN 8100 ELSE X=X+1
8070 GO TO 1620

8100 X=X+VAL(SUBSTR(H(I2),17,2))+VAL(SUBSTR(H(I2),19,2))/60
8110 GO TO 1620

8200 IF D(10)="1" THEN 8300 ELSE IF D(10)="3" THEN 8400
8210 FOR J=0 TO 4
8220 FOR I4=1 TO N5
8230 IF SUBSTR(H(I2),24+3*J,1)=F(I4) THEN
      X=X+VAL(SUBSTR(H(I2),25+3*J,2))
8240 NEXT I4
8250 NEXT J
8260 GO TO 1620

8300 X=X+VAL(SUBSTR(H(I2),25+3*J,2)) FOR J=0 TO 4
8310 GO TO 1620

8400 FOR J=0 TO 4
8410 FOR I4=1 TO N5
8420 IF SUBSTR(H(I2),24+3*J,1)=F(I4) THEN 8450
8430 NEXT I4
8440 X=X+VAL(SUBSTR(H(I2),25+3*J,2))
8450 NEXT J
8460 GO TO 1620

9000 I9=INDEX(D(11),"<")
9010 IF I9>0 THEN 9100
9020 I9=INDEX(D(11),">")
9030 IF I9>0 THEN 9200
9040 I9=INDEX(D(11),"=")
9050 X0=VAL(SUBSTR(D(11),I9+1))
9060 T3=1 !X=X0
9070 RETURN

9100 I8=INDEX(D(11),">")
9110 IF I8>0 THEN 9300
9120 I8=INDEX(D(11),"=")
9130 IF I8>0 THEN 9400
9140 X0=VAL(SUBSTR(D(11),I9+1))
9150 T3=2 !X=X0
9160 RETURN

```

```

9200 I8=INDEX(D(11),"")
9210 IF I8>0 THEN 9500
9220 X0=VAL(SUBSTR(D(11),I9+1))
9230 T3=3 !X>X0
9240 RETURN

9300 IF I8>I9 THEN I9=I8
9310 X0=VAL(SUBSTR(D(11),I9+1))
9320 T3=4 !X<>X0
9330 RETURN

9400 IF I8>I9 THEN I9=I8
9410 X0=VAL(SUBSTR(D(11),I9+1))
9420 T3=5 !X<=X0
9430 RETURN

9500 IF I8>I9 THEN I9=I8
9510 X0=VAL(SUBSTR(D(11),I9+1))
9520 T3=6 !X>=X0
9530 RETURN

10000 IF X=X0 THEN 1510 ELSE 1530
10010 IF X<X0 THEN 1510 ELSE 1530
10020 IF X>X0 THEN 1510 ELSE 1530
10030 IF X<>X0 THEN 1510 ELSE 1530
10040 IF X<=X0 THEN 1510 ELSE 1530
10050 IF X>=X0 THEN 1510 ELSE 1530

```

8. /COUNT/

```

15 H1=0,T0=5,H2=0
20 IF D(12)="4" THEN LOAD /AV-CT/
21 IF D(13)="1" THEN A4="EVENTS" ELSE IF D(13)="2" THEN A4="FLIGHTS"
   ELSE A4="LANDINGS"
22 IF D(16)="1" THEN B4=" EACH EVENT." ELSE IF D(16)="2" THEN B4
   =" EACH A/C." ELSE IF D(16)="3" THEN B4=" EACH SQUADRON." ELSE
   B4=" ALL OCCURENCES."
30 IF D(16)="3" THEN A1="SQUADRON" ELSE A1="A/C"
35 IF D(16)="3" THEN A2="A/C" ELSE A2="SQUADRON"
40 IF D(16)="3" THEN A0="SQDIR" ELSE A0="ACDIR"
45 IF D(16)="3" THEN S0=0 ELSE S0=1
60 PRINT"THE COUNTS OF THE QUALIFYING "A4" FOR" B4
75 IF D(16)="1" THEN GOSUB 5000 ELSE IF D(16)="2" THEN GOSUB 5100
   ELSE IF D(16)="3" THEN GOSUB 5200 1HEADING

300 IF D(16)="2" THEN 6100 ELSE IF D(16)="3" THEN 6200
610 IF D(16)="4" THEN 6300 ELSE PRINT"EXECUTION COMPLETE."
1110 !
1510 IF D(13)="1" THEN H1=H1+1
1520 IF D(16)="1" THEN 6000 ELSE H1=H1+H2
1530 S3=0,H2=0
1620 IF D(13)="2" THEN 4000 ELSE IF D(13)="3" THEN 4100
4000 IF D(14)="1" THEN 4070
4010 U=SUBSTR(H(12),21,3)
4020 FOR I4=1 TO N6
4030 IF U=G(I4) THEN IF D(14)="2" THEN 4070
4040 NEXT I4
4050 IF D(14)="3" THEN H2=H2+1
4060 G0 TO 1630
4070 H2=H2+1
4080 G0 TO 1630
4100 IF D(14)="2" THEN 4200 ELSE IF D(14)="3" THEN 4300
4110 H2=H2+VAL(SUBSTR(H(12),25+3+J,2)) FOR J=0 TO 4
4120 G0 TO 1630
4200 FOR J=0 TO 4
4210 FOR I4=1 TO N6
4220 IF SUBSTR(H(12),24+3+J,1)=G(I4) THEN H2=H2+VAL(SUBSTR(H(12),25+3+J,2))
4230 NEXT I4
4240 NEXT J
4250 G0 TO 1630
4300 FOR J=0 TO 4
4310 FOR I4=1 TO N6
4320 IF SUBSTR(H(12),24+3+J,1)=G(I4) THEN 4350
4330 NEXT I4
4340 H2=H2+VAL(SUBSTR(H(12),25+3+J,2))
4350 NEXT J
4360 G0 TO 1630
5000 PRINT"      A/C      SQUAD.      EVENT      COUNT"
5010 PRINT"      NUMBER      NUMBER      BEG. DATE      OF"
5020 PRINT"      (ENDING)      "A4
5030 RETURN

```

5100 PRINT"  
5110 PRINT"  
5120 PRINT"  
5130 PRINT  
5140 RETURN

A/C  
NUMBER

NUMBER OF"  
QUALIFYING"  
"1A41".

5200 PRINT" SQUADRON  
5210 PRINT" NUMBER  
5220 PRINT"  
5230 PRINT  
5240 RETURN

NUMBER OF"  
QUALIFYING"  
"1A41".

6000 PRINT IN IMAGE"  
V2,V3,D1,H2

XXXXXX

XXX

XXXX

XXXX"

6010 H2=0

6020 GO TO 1530

6100 PRINT IN IMAGE"

XXXXXX

XXXX"V2,H1

6110 H1=0

6120 GO TO 310

6200 PRINT IN IMAGE"

XXX

XXXX"V3,H1

6210 H1=0

6220 GO TO 310

6300 PRINT"THE TOTAL NUMBER OF OCCURENCES IN QUALIFYING EVENTS IS":H1

6310 GO TO 620

9. /DURATION/

```

15 H1=0,T0=6
20 IF D(12)="5" THEN LOAD /AV-DUR/
21 IF D(13)="1" THEN A4="DURATION OF EVENTS" ELSE A4="FLIGHT TIME"
22 IF D(16)="1" THEN B4="EACH EVENT." ELSE IF D(16)="2" THEN
    B4="EACH A/C." ELSE IF D(16)="3" THEN B4="EACH SQUADRON." ELSE
    B4="ALL EVENTS."
23 IF D(13)="1" THEN A3="EVENTS." ELSE A3="FLIGHTS."
30 IF D(16)="3" THEN A1="SQUADRON" ELSE A1="A/C"
35 IF D(16)="3" THEN A2="A/C" ELSE A2="SQUADRON"
40 IF D(16)="3" THEN A0="SQDIR" ELSE A0="ACDIR"
45 IF D(16)="3" THEN S0=0 ELSE IF D(16)="1" THEN S0=0
60 PRINT "THE ACCUMULATED "+A4+" FOR "+B4
71 PRINT
75 IF D(16)="1" THEN 5000
76 PRINT

300 IF D(16)="2" THEN 6100 ELSE IF D(16)="3" THEN 6200
610 IF D(16)="4" THEN 6300 ELSE PRINT "EXECUTION COMPLETE."
1110 IF D(16)="2" THEN 5100 ELSE IF D(16)="3" THEN 5200
1510 IF D(13)="1" THEN 4000 ELSE H1=H1+H2
1520 IF D(16)="1" THEN 6000
1530 H2=0
1530 H2=0
1620 IF D(13)="2" THEN 4200
4000 H2=VAL(F1)-VAL(D1)
4010 IF H2>=635 THEN H2=H2-634
4020 Y9=VAL(RIGHT(E1,2)),Z9=VAL(LEFT(E1,2))
4030 Y8=VAL(RIGHT(G1,2)),Z8=VAL(LEFT(G1,2))+24*H2
4040 IF Y8>=Y9 THEN 4060
4050 Y8=Y8+60,Z8=Z8-1
4060 H2=Z8-Z9+(Y8-Y9)/60
4070 H1=H1+H2
4080 GO TO 1520

4200 IF D(14)="1" THEN 4350
4210 W=SUBSTR(H(I2),21,3)
4220 FOR J=1 TO N6
4230 IF W=G(J) THEN 4300
4240 NEXT J
4250 IF D(14)="3" THEN 4350 ELSE 1630

4300 IF D(14)="2" THEN 4350 ELSE 1630

4350 H2=H2+VAL(SUBSTR(H(I2),17,2))+VAL(SUBSTR(H(I2),19,2))/60
4360 GO TO 1630

5000 PRINT "    A/C          SQUAD.  BEG. DATE    ACCUM. DURATION"
5005 PRINT "          (ENDING)        DAYS      HOURS"
5010 PRINT
5020 GO TO 76

```

5100 PRINT" A/C ACCUM. DURATION"  
 5110 PRINT" NUMBER OF QUALIFYING ":A3  
 5120 PRINT" DAYS HOURS"  
 5130 PRINT  
 5140 GO TO 1120

5200 PRINT" SQUADRON ACCUM. DURATION"  
 5210 PRINT" NUMBER OF QUALIFYING ":A3  
 5220 PRINT" DAYS HOURS"  
 5230 PRINT  
 5240 GO TO 1120

6000 H4=H2 MOD 24,H3=FIX(H2/24)  
 6010 PRINT IN IMAGE" XXXXXX XXX XXXX XX.X":  
 V2,V3,D1,H3,H4  
 6020 H2=0  
 6030 GO TO 1520

6100 H4=H1 MOD 24,H3=FIX(H1/24)  
 6110 PRINT IN IMAGE" XXXXXX XXXX XX.X":V2,H3,H4  
 6120 H1=0  
 6130 GO TO 310

6200 H4=H1 MOD 24,H3=FIX(H1/24)  
 6210 PRINT IN IMAGE" XXX XXXX XX.X":V3,H3,H4  
 6220 H1=0  
 6230 GO TO 310

6300 PRINT"THE ACCUMULATED DURATION OF ALL QUALIFYING ":A3:" IS":H1  
 6310 PRINT  
 6320 GO TO 620

10. /AV-CT/

```
20 TO=7,N9=0
21 IF D(13)="1" THEN A4="FLIGHTS" ELSE IF D(13)="2" THEN
A4="LANDINGS" ELSE A4="EVENTS"
22 IF D(16)="1" THEN B4=" PER EVENT," ELSE IF D(16)="2" THEN B4=
" PER A/C," ELSE IF D(16)="3" THEN B4=" PER SQUADRON," ELSE
B4=" IN ALL OCCURENCES,"
23 IF D(15)="5" THEN B4="A/C" ELSE B4="EVENT"
24 IF D(15)="1" THEN C4="." ELSE IF D(15)="2" THEN C4="A/C."
ELSE IF D(15)="3" THEN C4="SQUADRON." ELSE IF D(15)="4"
THEN C4="A/C IN SQUADRON." ELSE IF D(15)="5" THEN
C4="SQUADRON."
26 IF D(15)="1" THEN E4="" ELSE E4=" FOR EACH "
30 !
35 !
40 IF D(15)="1" THEN A0="ACDIR" ELSE IF D(15)="2" THEN A0="ACDIR"
ELSE A0="SODIR"
45 IF D(15)="1" THEN S0=1 ELSE IF D(15)="2" THEN S0=1 ELSE S0=0
60 PRINT"THE AVERAGE NUMBER OF "A4;" PER "B4;E4;C4
75 !
300 IF D(15)="2" THEN 6100 ELSE IF D(15)="3" THEN 6200 ELSE IF
D(15)="5" THEN 6200
610 IF D(15)="1" THEN 6300 ELSE PRINT "EXECUTION COMPLETE."
910 IF D(15)="4" THEN 6100 ELSE IF D(15)="5" THEN N9=N9+1
1110 IF S4=0 THEN IF D(15)="2" THEN 5110 ELSE IF D(15)="3"
THEN 5200 ELSE IF D(15)="4" THEN 5100 ELSE IF D(15)="5"
THEN 5300
1510 IF D(13)="3" THEN H1=H1+1 ELSE H1=H1+H2
1520 IF VAL(D(15))<5 THEN N9=N9+1
1620 IF D(13)="1" THEN 4000 ELSE IF D(13)="2" THEN 4100
5100 PRINT" SQUADRON NO.:"V3
5110 PRINT" A/C AVERAGE NO. OF"
5120 PRINT" NUMBER "A4;" PER EVENT"
5130 GO TO 1120

5200 PRINT" SQUADRON AVERAGE NO. OF"
5210 PRINT" NUMBER "A4;" PER EVENT"
5220 GO TO 1120

6100 IF N9=0 THEN 6150
6110 PRINT IN IMAGE" XXXXXX XXX.XX":V2,H1/N9
6120 H1=0,N9=0
6130 GO TO 910

6150 PRINT IN IMAGE" XXXXXX NONE":V2
6160 H1=0,N9=0
6170 GO TO 910

6200 IF N9=0 THEN 6250
6210 PRINT IN IMAGE" XXX XXX.XX":V3,H1/N9
6220 H1=0,N9=0
6230 GO TO 910

6250 PRINT IN IMAGE" XXX NONE":V3
6260 H1=0,N9=0
6270 GO TO 910
```



6300 IF N9=0 THEN PRINT "NO QUALIFYING EVENTS. EXECUTION COMPLETE."  
ELSE 6350  
6310 GO TO 620

6350 PRINT "THE AVERAGE NUMBER OF "A4" PER EVENT= "H1/N9  
6360 GO TO 620

11. /AV-DUR/

```
20 T0=N,N9=0
21 IF VAL(D(15))<5 THEN A4="AVERAGE FLIGHT DURATION PER EVENT "
   ELSE A4="AVERAGE DURATION OF EVENTS "
22 IF D(15)="1" THEN B4="." ELSE IF D(15)="2" THEN B4="FOR EACH A/C."
   ELSE IF D(15)="3" THEN B4="FOR EACH SQUADRON." ELSE IF D(15)="4"
   THEN B4="FOR EACH A/C IN SQUADRON."
23 IF D(15)="5" THEN B4="FOR EACH A/C." ELSE IF D(15)="6" THEN
   B4="FOR EACH SQUADRON." ELSE IF D(15)="7" THEN B4=
   "FOR EACH A/C IN SQUADRON." ELSE B4="."
40 IF D(15)="3" THEN A0="SQDIR" ELSE IF D(15)="4" THEN A0="SQDIR"
   ELSE IF D(15)="6" THEN A0="SQDIR" ELSE IF D(15)="7" THEN A0="SQDIR"
   ELSE A0="ACDIR"
60 PRINT A4:B4
70 I
300 IF D(15)="2" THEN 6000 ELSE IF D(15)="3" THEN 6100 ELSE
   IF D(15)="5" THEN 6000 ELSE IF D(15)="6" THEN 6100
610 IF D(15)="1" THEN 6300 ELSE IF D(15)="8" THEN 6300 ELSE
   PRINT "EXECUTION COMPLETE."
910 IF D(15)="4" THEN 6000 ELSE IF D(15)="7" THEN 6000
1110 IF S4=0 THEN IF D(15)="2" THEN 5010 ELSE IF D(15)="3" THEN
   5100 ELSE IF D(15)="4" THEN 5000 ELSE IF D(15)="5" THEN 5210
   ELSE IF D(15)="6" THEN 5300 ELSE IF D(15)="7" THEN 5200
1510 IF VAL(D(15))>4 THEN 4000 ELSE H1=H1+H2
1520 N9=N9+1
1620 IF VAL(D(15))<5 THEN 4200
5000 PRINT" SQUADRON NO.: "I:V3
5010 PRINT"      A/C                AVERAGE"
5020 PRINT"      NUMBER            FLIGHT DURATION"
5030 PRINT"                                PER EVENT"
5040 PRINT"                                DAYS HOURS"
5050 G0 T0 1240

5100 PRINT" SQUADRON                AVERAGE"
5110 PRINT"      NUMBER            FLIGHT DURATION"
5120 PRINT"                                PER EVENT"
5130 PRINT"                                DAYS HOURS"
5140 G0 T0 1240

5200 PRINT" SQUADRON NO.: "I:V3
5210 PRINT"      A/C                AVERAGE"
5220 PRINT"      NUMBER            DURATION OF EVENTS"
5230 PRINT"                                DAYS HOURS"
5240 G0 T0 1240

5300 PRINT" SQUADRON                AVERAGE"
5310 PRINT"      NUMBER            DURATION OF EVENTS"
5320 PRINT"                                DAYS HOURS"
5330 G0 T0 1240

6000 IF N9=0 THEN 6050 ELSE H1=H1/N9
6010 H4=H1 MOD 24,H3=FIX(H1/24)
6020 PRINT IN IMAGE"      XXXXX                XX      XX.XX":V2,H3,H4
6030 H1=0,N9=0
6040 G0 T0 310
```

```

6050 PRINT IN IMAGE"  XXXXXX      NONE":V2
6060 H1=0,N9=0
6070 GO TO 310

6100 IF N9=0 THEN 6150 ELSE H1=H1/N9
6110 H4=H1 MOD 24,H3=FIX(H1/24)
6120 PRINT IN IMAGE"  XXX      XXX  XX.XX":V3,H3,H4
6130 H1=0,N9=0
6140 GO TO 310

6150 PRINT IN IMAGE"  XXX      NONE":V3
6160 H1=0,N9=0
6170 GO TO 310

6300 IF N9=0 THEN 6350, ELSE H1=H1/N9
6310 H3=FIX(H1/24),H4=H1 MOD 24
6320 PRINT"THE ":A4:"= ":H3:" DAYS,"H4:" HOURS."
6330 GO TO 620

6350 PRINT"NO APPLICABLE DATA."
6360 GO TO 620

```