②

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

A MANAGEMENT INFORMATION SYSTEM
FOR THE ANALYSIS OF THE
ARMED SERVICES APTITUDE BATTERY

by

Robert Joseph Forman

September 1983

Thesis Advisor: D.R. Dolk

Approved for public release; distribution unlimited.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. AD-A136779	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Management Information System for the Analysis of the Armed Services Aptitude Battery		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; September 1983
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert Joseph Forman		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		12. REPORT DATE September 1983
		13. NUMBER OF PAGES 141
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Armed Services Vocational Aptitude Battery (ASVAB)

Query Language Processor, Series 1100 (QLP 1100)

Data Base Management Systems

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

→ This thesis develops a management information system called ASVABMIS for use by Army agencies conducting analysis using data from the Armed Services Aptitude Battery. The thesis develops a FORTRAN program that inputs demographic data and raw scores, and computes subtest standard scores, and the Army Classification Battery Composites. The output is directed to files designed to be

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. ABSTRACT (Continued)

easily used with the Statistical Package for the Social Sciences for statistical analysis. In addition, a data base management system is integrated into the information system for data organization and management.

Approved for public release; distribution unlimited.

A Management Information System
for the Analysis of the
Armed Services Aptitude Battery

by

Robert Joseph Forman
Captain, United States Army
B.S., United States Military Academy, 1974

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MANAGEMENT

from the

NAVAL POSTGRADUATE SCHOOL

September 1983

Author: Robert J. Forman

Approved by: Daniel R. Dolk, Thesis Advisor

Stephen J. Park, Second Reader

Richard S. Elster, Chairman, Department of Administrative Science

K.T. Marshall, Dean of Information and Policy Sciences

ABSTRACT

This thesis develops a management information system called ASVABMIS for use by Army agencies conducting analysis using data from the Armed Services Aptitude Battery. The thesis develops a FORTRAN program that inputs demographic data and raw scores, and computes subtest standard scores, and the Army Classification Battery Composites. The output is directed to files designed to be easily used with the Statistical Package for The Social Sciences for statistical analysis. In addition, a data base management system is integrated into the information system for data organization and management.

<u>TABLE OF CONTENTS</u>

TABLE OF ABBREVIATIONS

ACB	Army Classification Battery
AD	Attention To Detail
ADP	Application Definition Processor
AI	Automotive Information
AR	Arithmetic Reasoning
AS	Automotive/Shop
ASVAB	Armed Services Vocational Aptitude Battery
ASVABMIS	Armed Services Vocational Aptitude Battery Management Information System
CA	Attentiveness
CC	Combat
CE	Electronics
CL	Clerical
CM	Maintenance
CO	Combat
CS	Coding Speed
DOES	Department of Evaluation and Standardization
EI	Electronics Information
EL	Electronics Repair
FA	Field Artillery
FDP	File Definition Processor
GI	General Information
GM	General Maintenance
GS	General Science

GT	General Technical
MC	Mechanical Comprehension
MK	Math Knowledge
MM	Motor Maintenance
NO	Numerical Operations
OF	Operators and Food Handlers
PC	Paragraph Comprehension
PCIOS	Processor Common Input/Output System
QLP	Query Language Processor
SC	Surveillance and Communications
SI	Shop Information
SP	Space Perception
SPSS	Statistical Package for the Social Sciences
SQT	Skill Qualification Test
SSS	Subtest Standard Score
ST	Skilled Technician
VE	Verbal
WK	Word Knowledge

I. INTRODUCTION

A. PURPOSE

The purpose of this chapter is to discuss a problem faced by the Department of Evaluation and Standardization (DOES), Fort Benning, Georgia. In addition, this chapter will determine the feasibility of providing DOES with a computer-based management information system. Finally, this chapter will conceptually develop a system design that will meet DOES' needs.

B. PROBLEM

DOES is responsible for evaluating and standardizing training programs developed by the Infantry School, Fort Benning, Georgia. DOES uses the raw scores that are a part of the Armed Services Aptitude Battery in conducting analysis of many of the training programs. Currently, DOES uses a manual system that requires a minimum of 21 entries into a conversion table, as well as the computation of 10 equations. This system becomes prohibitively time consuming with large sample sizes. For example, DOES is responsible for developing a selection criteria for the Infantry School's mechanized infantryman training program. The school trains approximately 100 soldiers during each training cycle which lasts only a few weeks. In or 'er t process the data, and to develop the selection criteria using the manual system, DOES would have to allocate much of its resources in terms of manpower and time to this project alone. Since DOES is unable to do this, the selection criteria has not been developed, and a large backlog of data exists which must be processed. Thus a speedier system is needed that can process the backlog as well as any new data using, at most, one analyst rather than the entire analytical manpower pool. ASVABMIS is the speedier system that will allow DOES to accomplish all of its requirements.

C. FEASIBILITY

The feasibility of a computer-based management information system is apparent in surveying the hardware and software available to DOES. DOES has access to a UNIVAC 1100 main-frame computer with a FORTRAN compiler. In addition, the Statistical Package for the Social Sciences is available for statistical analysis, while the SPERRY-UNIVAC QLP 1100 system is available for data management. ASVAB raw data is provided in the form of prepunched computer cards. The only component that is lacking is a software program to provide DOES with a complete system for data generation, analysis, and management.

D. SYSTEM DESIGN

In designing the software, DOES has some specific criteria that must be met. First, DOES currently has only batch processing available. Thus the program cannot be interactive. Secondly, DOES requires that the program be written in FORTRAN because of the availability of the FORTRAN compiler. Since DOES will use SPSS in batch mode, any data files generated by the program must be compatible with the SPSS format. Finally, DOES has varying needs in regard to report formats and other data management. Thus, it is difficult, if not impossible to anticipate every need in order to code every report format in the program itself. This leads to the necessity of introducing data management and organization through a data base management system. Therefore, the file definitions and query language of the QLP 1100 system must be developed in conjunction with DOES' needs.

In summary, the information system must be comprised of a FORTRAN program that inputs data, computes the necessary ASVAB scores, and generates data to be stored in a file. The files must be structured so that they provide immediate, useful information to DOES. In addition, the files must be in a form that allows integration into both SPSS for statistical analysis and QLP 1100 for data management. This design will provide DOES with both the flexibility to generate the required information in the appropriate format and provide linkages to powerful analytic packages.

Chapter I, "Introduction," has discussed one of the problems faced by DOES. In addition, it has examined the feasibility of a computer-based management information system.

Finally, the chapter has identified the user's performance specifications for the information system. Chapter II, "Systems Manual," examines the manual system in detail. It provides several examples and is designed to familiarize readers with the current system. Chapter III, "User's Manual," explains the computer program using nontechnical language. It is designed to provide the reader with a conceptual development of the computer program. Chapter IV, "The SPSS and QLP 1100 System," explains how the FORTRAN program is integrated with the SPSS program and the QLP 1100 system. Chapter V, "Conclusions," explains how the information system and its use may be expanded.

II. SYSTEMS MANUAL

A. PURPOSE

The purpose of this chapter is to provide an understanding of how the current manual system works. This is important because the procedures used in the computerized system are virtually the same.

B. INTRODUCTION

The Armed Services Vocational Aptitude Battery has six test versions that can be grouped into three different test formats. The first group is comprised of ASVAB 5. The second group is comprised of ASVAB 6 and ASVAB 7. The last group is comprised of ASVAB 8, ASVAB 9 and ASVAB 10.

The manual process used by DOES is shown in Figure 1. The input data are stored on pre-punched computer cards (see Appendix A). This data stream is decoded using the key shown in Table 1.

Once the data are decoded, they can be checked for input errors. This is important because DOES does not enter the data onto the cards, and must confirm that they are correct. Next the raw scores are converted to standard scores. This is done so that scores can be compared between test versions. Composite scores are then computed, and subsequently normalized so that they can also be compared between test

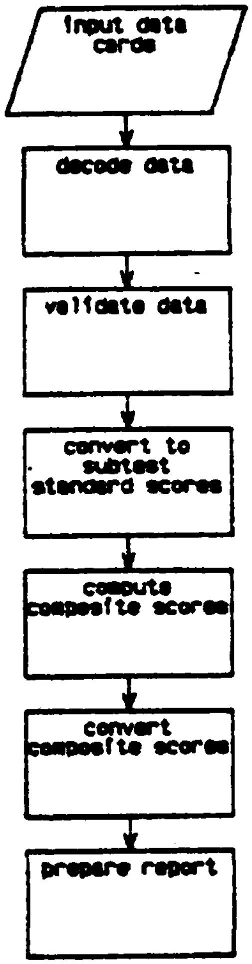

Figure 1. ASVAB Manual Process

TABLE 1

INPUT DATA KEY

COLUMNS	DESCRIPTION
1-9	Social Security Account Number
10-15	Date of Birth (YY/MM/DD)
16	Sex: 1-Male, 2-Female
17	Race: 1-White, 2-Black, 3-Other
18-23	Date of Entry (YY/MM/DD)
24-25	Highest Year of Education:
	01 1-7 Years
	02 8 Years
	03 1 Year of High School
	04 2 Years of High School
	05 3-4 Years of High School
	06 High School Diploma
	07 1 Year of College
	08 2 Years of College
	09 3-4 Years of College
	10 College Graduate
	11 Master's Degree
	12 Doctor's Degree
	13 High School GED
26-27	Entry Pay Grade:
	01 E1
	02 E2
	03 E3

TABLE 1 (CONT.)

	04	E4	
	05	E5	
	06	E6	
	07	E7	
	08	E8	
	09	E9	
28-32	Military Occupation Specialty		
33-34	Test Form:		
	35	ASVAB 5.	
	36	ASVAB 6.	
	37	ASVAB 7.	
	38	ASVAB 8.	
	39	ASVAB 9.	
	40	ASVAB 10.	
35-36	AFQT Percentile		
37-68	Subtest Raw Scores:		
	ASVAB 5	ASVAB 6/7	ASVAB 8/9/10
37-38	GI	GI	GS
39-40	NO	NO	AR
41-42	AD	AD	WK
43-44	WK	WK	PC
45-46	AR	AR	NO
47-48	SP	SP	CS
49-50	MK	MK	AS
51-52	EI	EI	MK
53-54	MC	MC	MC
55-56	GS	GS	EI

TABLE 1 (CONT.)

57-58	SI	SI	VE
59-60	AI	AI	BLANK
61-62	BLANK	MC	BLANK
63-64	BLANK	CA	BLANK
65-66	BLANK	CE	BLANK
67-68	BLANK	CC	BLANK
71-73	SEPARATION PROGRAM DESIGNATOR.		
74-79	DATE OF SEPARATION.		

versions. Finally, the data are written down in a format that is conducive to further analysis. This process is done for each data card. A typical session could involve 100 to 300 data cards depending on the size of the training group. Also the frequency of this process depends on the number of training cycles being run by the Infantry School.

C. THE ASVAB 5 PROCEDURE

In order to process an ASVAB 5 record, the data record is first decoded using Table 1. The first data record in Appendix A is provided as a typical ASVAB 5 data card. The number 35 located in columns 33 and 34 indicates that ASVAB 5 is in effect. The scores located in columns 37 through 68 can be identified and assigned to specific tests. This assignment is shown in Table 2.

Since ASVAB 5 is in effect, these raw scores must be converted into subtest standard scores. The ASVAB 5/6/7 subtest standard score conversion table is used to convert the scores (see Appendix B). The table is entered by finding the raw score in the left hand column, then moving right to find the test name and the appropriate converted score. For example, to convert GI enter the raw score column at 15, move to the right and find the column for GI and read off the standard score of 66. Proceeding in this manner, all subtest raw scores are converted as shown in Table 3.

The Army Classification Battery (ACB) uses subtest standard scores to compute the equations shown in Table 4.

TABLE 2

ASVAB 5 RAW SCORE ASSIGNMENT

SUBTEST (Variable Name)	SCORE (Data Point)
GI	15
NO	20
AD	03
WK	12
AR	06
SP	17
MK	20
EI	10
MC	09
GS	05
SI	13
AI	12

TABLE 3

SUBTEST SCORE CONVERSIONS

RAW SCORE	SUBTEST	STANDARD SCORE
15	GI	66
20	NO	39
03	AD	21
12	WK	39
06	AR	36
17	SP	59
20	MK	67
10	EI	34
09	MC	46
05	GS	36
13	SI	48
12	AI	50

TABLE 4

ASVAB 5 ACB EQUATIONS

EQUATION	SCORE
CO = AR + SI + SP + AD	164
FA = AR + GI + MK + EI	203
MM = MK + SI + EI + AI	199
GM = AR + GS + MC + AI	168
CL = AR + WK + AD	96
GT = AR + WK	75
EL = AR + EI + NC + SI	164
SC = AR + WK + MC + SP	180
ST = AR + MK + GS	139
OF = GI + AI	116

The final step is to convert the ACB scores to standardized scores using the ASVAB 5 composite conversion table (see Appendix D). The table is used exactly as the ASVAB 5 subtest standard score conversion table to give the final composite scores shown in Table 5.

TABLE 5

ASVAB 5 ACB COMPOSITE SCORES

ACB SCORE	CLASSIFICATION	COMPOSITE SCORE
164	CO	71
203	FA	102
199	MM	99
168	GM	79
96	CL	49
75	GT	71
164	EL	76
180	SC	89
139	ST	94
116	OF	116

With the calculations completed, the soldier file can be constructed using the results of the calculations and the initial data record. The soldier file for the example data record is shown below.

SSAN: 272598843

DATE OF BIRTH(YY/MM/DD): 64/4/20

SEX: MALE

RACE: BLACK

DATE OF ENTRY(YY/MM/DD): 82/2/16

HIGHEST YEAR OF EDUCATION: 1 YEAR OF HIGH SCHOOL

ENTRY PAY GRADE: E1

MOS: 11B10

TEST FORM: ASVAB 5

AFQT PERCENTILE: 35

SEPARATION PROGRAM DESIGNATOR: AKL

DATE OF SEPARATION(YY/MM/DD) 83/2/27

SUBTEST RAW SCORES

GI	NO	AD	WK	AR	SP	MK	EI	MC	GS	SI	AI
15	20	03	12	06	17	20	10	09	05	13	12

SUBTEST STANDARD SCORES

GI	NO	AD	WK	AR	SP	MK	EI	MC	GS	SI	AI
66	39	21	39	36	59	67	34	46	36	48	50

ARMY CLASSIFICATION BATTERY

CO	FA	MM	GM	CL	GT	EL	SC	ST	OF
71	102	99	79	49	71	76	89	94	116

D. THE ASVAB 6 AND ASVAB 7 PROCEDURE

The procedures for ASVAB 6 and ASVAB 7 are the same. However, these procedures differ significantly from the ASVAB

5 procedures. The first difference is that, in addition to the twelve raw scores used in ASVAB 5, four additional scores, CM, CA, CE, and CC are also used. The second difference is that the subtest raw scores are not converted to standard scores for the purpose of computing composites as they were in the ASVAB 5 procedure. They are converted for comparison with scores from other test versions. Finally, the set of equations used to compute the ACB composites are different. A typical input data record for ASVAB 6 or ASVAB 7 is provided as the second data record in Appendix A.

Since there is o need to convert the subtest raw scores, the procedure begins with the computation of the ACB composites. First, the data in columns 33 and 34 are checked to confirm that ASVAB 6 or ASVAB 7 is in effect. Using the sample input data record columns 33 and 34 have the value 36 as a data point confirming ASVAB 6. The data in columns 37 through 68 are extracted and assigned to the appropriate subtests as shown in Table 6.

Since raw scores are used, the ACB composites are immediately computed as shown in Table 7.

The final step is to convert the ACB scores to standardized scores using the ASVAB 6/7 composite conversion table (see Appendix E) as shown in Table 8.

With the conversion of the ACB scores to standard composite scores a soldier file can be constructed. However, one additional step is done. As shown above, the raw subtest

TABLE 6

ASVAB 6/7 RAW SCORE ASSIGNMENT

SUBTEST (Variable Name)	SCORE (Data Point)
GI	13
NO	40
AD	25
WK	22
AR	15
SP	16
MK	12
EI	23
MC	10
GS	09
SI	17
AI	14
CM	08
CA	12
CE	13
CC	22

TABLE 7

ASVAB 6/7 ACB EQUATIONS

EQUATION	SCORE
CO = AR + SI + SP + AD + CC	95
FA = AR + GI + MK + EI + CA	75
MM = MK + SI + EI + AI + CM	74
GM = AR + GS + MC + AI	48
CL = AR + WK + AD + CA	74
GT = AR + WK	37
EL = AR + EI + MC + ST + CE	78
SC = AR + WK + MC + SP	63
ST = AR + MK + GS	36
OF = GI + AI + CA	39

TABLE 8

ASVAB 6/7 ACB COMPOSITE SCORES

ACB SCORE	CLASSIFICATION	STANDARD SCORE
95	CO	125
75	FA	115
74	MM	106
48	GM	103
74	CL	122
37	GT	109
78	EL	114
63	SC	108
36	ST	104
39	OF	115

scores are used in computing the ACB scores. These raw scores cannot be compared to the standardized scores in other test versions. Because of this, the raw scores are converted using the ASVAB 5/6/7 standard score conversion table (see Appendix B).

The input data record and the ACB computations are used to create the soldier file.

SSAN: 016532241

DATE OF BIRTH(YY/MM/DD): 63/11/25

SEX: MALE

RACE: WHITE

DATE OF ENTRY(YY/MM/DD): 81/9/20

HIGHEST YEAR OF EDUCATION: HIGH SCHOOL DIPLOMA

ENTRY PAY GRADE: E1

MOS: 11B10

TEST FORM: ASVAB 6

AFQT PERCENTILE: 87

SEPARATION PROGRAM DESIGNATOR: AAA

DATE OF SEPARATION(YY/MM/DD): 82/11/9

SUBTEST RAW SCORES

GI	NO	AD	WK	AR	SP	MK	EI	MC	GS	SI	AI	CM	CA	CE	CC
13	40	25	22	15	16	12	23	10	09	17	14	08	12	13	22

SUBTEST STANDARD SCORES

GI	NO	AD	WK	AR	SP	MK	EI	MC	GS	SI	AI	CM	CA	CE	CC
60	59	77	53	55	57	51	56	48	45	58	55	-	-	-	-

ARMY CLASSIFICATION BATTERY

CO	FA	MM	GM	CL	GT	EL	SC	ST	OF
125	115	106	103	122	109	114	108	104	115

E. ASVAB 8, ASVAB 9, AND ASVAB 10 PROCEDURE

The procedure for ASVAB 8, ASVAB 9, and ASVAB 10 is generally the same as ASVAB 5. The subtest raw scores are converted to subtest standard scores. The converted scores are used to compute the ACB composite scores. Finally, the ACB scores are standardized. The difference between ASVAB 8, ASVAB 9, and ASVAB 10, and all other ASVAB versions is that a different set of tests are used to generate the raw scores. Because of this, different conversion tables are used, as well as a different set of equations for the computation of the ACB scores. A typical input data record for ASVAB 8, ASVAB 9, or ASVAB 10 is provided as the third data card in Appendix A.

First, columns 33 and 34 are checked to insure that either ASVAB 8, ASVAB 9, or ASVAB 10 is in effect. The data point 40 confirms that ASVAB 10 is in effect. The raw scores in columns 37 through 58 are extracted and assigned to their appropriate subtest variable. The raw scores are then standardized using the ASVAB 8/9/10 subtest standard score conversion table (see Appendix C) as shown in Table 9.

The subtest standard scores are then used to compute the ACB scores as shown in Table 10.

TABLE 9

ASVAB 8/9/10 SUBTEST SCORE CONVERSIONS

RAW SCORE (Data Point)	SUBTEST (Variable Name)	STANDARD SCORE
24	GS	65
22	AR	56
30	WK	56
14	PC	60
45	NO	59
79	CS	72
25	AS	65
20	MK	63
24	MC	65
19	EI	65
48	VE	61

TABLE 10

ASVAB 8/9/10 ACB EQUATIONS

EQUATION	SCORE
CO = AR + AS + MC + CS	258
FA = AR + MK + MC + CS	256
MM = NO + EI + MC + AS	254
GM = MK + EI + GS + AS	258
CL = NO + CS + VE	192
GT = VE + AR	117
EL = AR + EI + MK + GS	249
SC = NO + CS + VE + AS	257
ST = VE + MK + MC + GS	254
OF = NO + VE + MC + AS	250

Finally, once the ACB scores are computed, the scores are standardized using the ASVAB 8/9/10 composite conversion table (see Appendix F) as shown in Table 11.

TABLE 11

ASVAB 8/9/10 ACB COMPOSITE SCORES

ACB SCORE	CLASSIFICATION	STANDARD SCORE
258	CO	138
256	FA	128
254	MM	138
258	GM	132
192	CL	133
117	GT	117
249	EL	124
257	SC	144
254	ST	128
250	OF	135

The input data record and the ACB computations are used to generate the following soldier file.

SSAN: 213865527

DATE OF BIRTH(YY/MM/DD): 630521

SEX: FEMALE

RACE: OTHER

DATE OF ENTRY(YY/MM/DD): 821007

HIGHEST YEAR OF EDUCATION: 1 YEAR OF COLLEGE

ENTRY PAY GRADE: E2

MOS: 54E10

TEST FORM: ASVB 10

AFQT PERCENTILE: 92

SUBTEST RAW SCORES

GS	AR	WK	PC	NO	CS	AS	MK	MC	EI	VE
24	22	30	14	45	79	25	20	24	19	48

SUBTEST STANDARD SCORES

GS	AR	WK	PC	NO	CS	AS	MK	MC	EI	VE
65	56	56	60	59	72	65	63	65	65	61

ARMY CLASSIFICATION BATTERY

CO	FA	MM	GM	CL	GT	EL	SC	ST	OF
138	128	138	132	133	117	124	144	128	135

This process continues until all the input data records are processed into soldier records.

III. USER'S MANUAL

A. PURPOSE

The purpose of this chapter is to provide the user of the Armed Services Vocational Aptitude Battery Management Information System (ASVABMIS) with a manual that explains the functioning of the ASVAB program. The language is non-technical, yet specific enough to allow a thorough understanding or how ASVAB works.

B. INTRODUCTION

The relationship between ASVAB and its input file and output files is shown in Figure 2. Figure 2 shows that separate input data cards are combined into one input data file. The input data file is processed by ASVAB, and four output files called the soldier file, the raw score summary file, the standard score summary file, and the composite score summary file result.

C. THE INPUT FILES

In order to create the input data file the user must simply read into file 04 input data, the stack of input computer cards. No special preparation is necessary, and this must be done for each different set of input cards. The data cards containing the conversion tables must be read into file and permanently stored when ASVAB is installed.

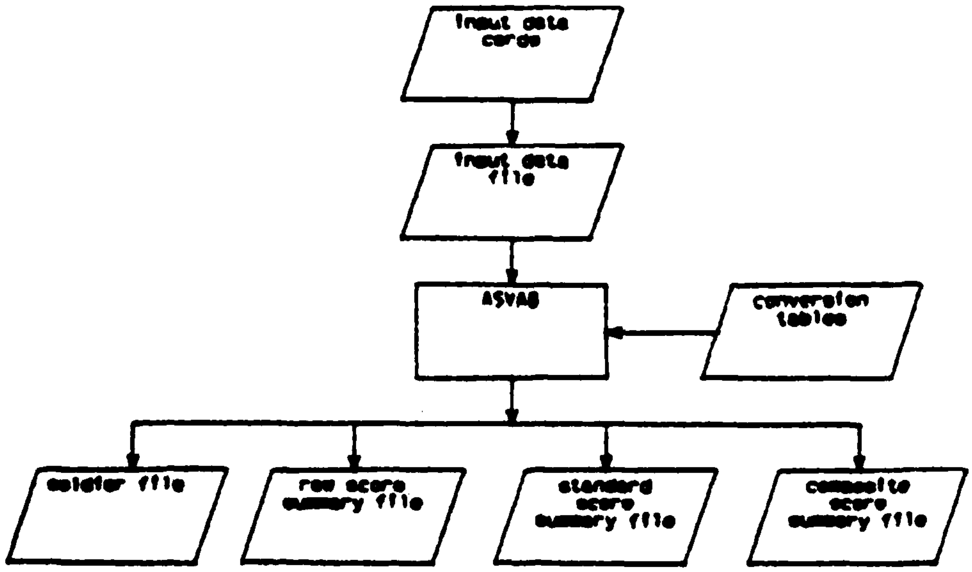

Figure 2. Relationship Between ASVAB Program and Input/Output Files

The stack of cards containing the conversion tables for ASVAB 5, ASVAB 6/7, and ASVAB 8/9/10 are read into File 01, File 02, and File 03 respectively.

D. THE ALGORITHM

The development of the system began with the determination of what ASVAB had to accomplish. This was done by studying the manual system. The initial algorithm was developed as shown below.

```
ALGORITHM ASVAB
    READ IN CONVERSION TABLES
    ALGORITHM INPUT SOLDIER DATA
        READ IN INPUT DATA
    END ALGORITHM INPUT SOLDIER DATA

    ALGORITHM VALIDATE SOLDIER DATA
        CHECK FOR DATA INTEGRITY
    END ALGORITHM VALIDATE SOLDIER DATA

    ALGORITHM OUTPUT SOLDIER DATA
        PRINT THE INPUT DATA AS PART OF THE SOLDIER FILE
    END ALGORITHM OUTPUT SOLDIER DATA

    ALGORITHM SUBTEST STANDARD SCORES
        CONVERT RAW SCORES TO SUBTEST STANDARD SCORES
    END ALGORITHM SUBTEST STANDARD SCORES

    ALGORITHM APTITUDE AREA COMPOSITES
        COMPUTE ACB COMPOSITES
    END ALGORITHM APTITUDE AREA COMPOSITES
```

ALGORITHM ACB

CONVERT ACB COMPOSITES

END ALGORITHM ACB

ALGORITHM OUTPUT SCORES

PRINT SUBTEST STANDARD SCORES AND ACB SCORES

END ALGORITHM OUTPUT SCORES

END ALGORITHM ASVAB

The relationship between the algorithms is shown in Figure 3.

1. Algorithm ASVAB

This algorithm defines the variables that are used in ASVAB. It also reads in the conversion tables, and terminates processing when the last data card has been read.

2. Algorithm Input Soldier Data

This algorithm reads in the data from File 04 input data. The process is shown in Figure 4. Figure 3 shows that once file 04, Input Data, is read in, a decision logic is used to assign the raw scores to their proper variable names.

3. Algorithm Validate Soldier Data

This algorithm provides a rudimentary input data validation by making range checks on the data according to Table 12. If the value of a variable is outside the range shown in Table 12, an error message is printed. In general, the error statements identify the record that has

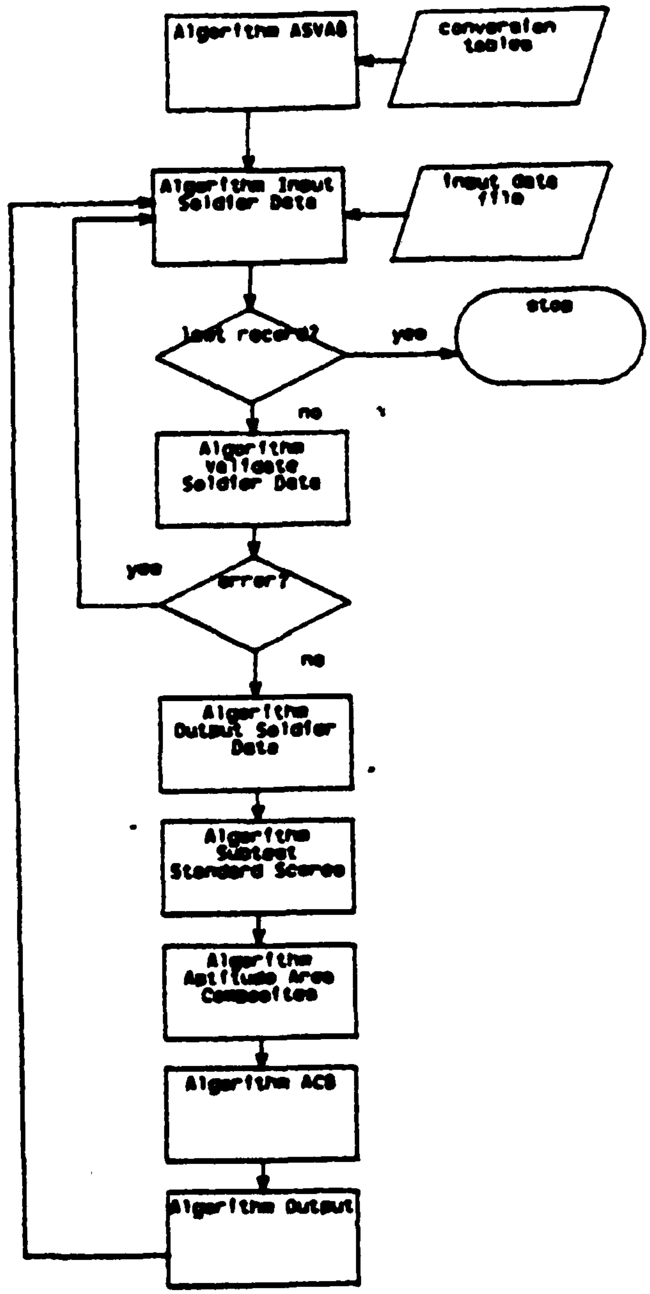

Figure 3. ASVAB Program Flow Chart

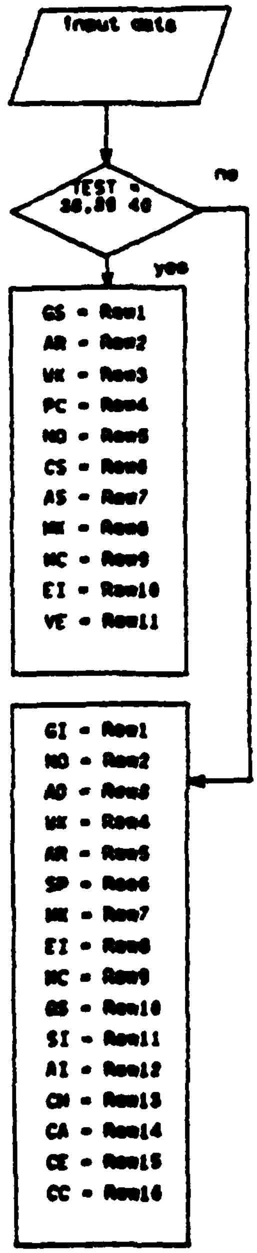

Figure 4. Algorithm Input Soldier Data Flow Chart

TABLE 12

DATA INTEGRITY TABLE

ITEM	MIN. VALUE	MAX. VALUE
MONTH (MM)	1	12
DAY (DD)	1	31
SEX	1	2
RACE	1	3
ED	1	13
TEST	35	40
ASVAB 5/6/7		
GI	0	15
NO	0	30
AD	0	30
WK	0	20
AR	0	20
SP	0	20
MK	0	20
EI	0	30
MC	0	20
GS	0	20
SI	0	20
AI	0	20
CM	0	20
CA	0	20
CE	0	20
CC	0	27

TABLE 12 (CONT.)

ITEM	MIN. VALUE	MAX. VALUE
ASVAB 8/9/10		
GS	0	25
AR	0	30
WK	0	35
PC	0	15
NO	0	50
CS	0	84
AS	0	25
MK	0	25
MC	0	25
EI	0	25
VE	0	50

the error by record number, the social security number, the incorrect variable and its column position in the record. Processing of that record is terminated and processing of a new record is begun.

4. Algorithm Output Soldier Data

This algorithm simply prints the input data. This output is provided for two reasons. First, it provides some of the information required by the user in an understandable form. Second, since the data is echoed, error messages can be confirmed. The algorithm provides the information as shown below.

SSAN	:	014560821
DATE OF BIRTH(YY/MM/DD)	:	64/4/20
SEX	:	1-MALE
RACE	:	2-BLACK
DATE OF ENTRY(YY/MM/DD)	:	82/2/16
HIGHEST YEAR OF EDUCATION	:	1 YEAR OF HIGH SCHOOL--03
ENTRY PAY GRADE: 01-E1	:	01-E1
MOS	:	11B10
TEST FORM	:	40-ASVAB 10
AFQT PERCENTILE	:	89
SEPARATION PROGRAM DESIGNATOR	:	JHK
DATE OF SEPARATION(YY/MM/DD)	:	83/5/25

SUBTEST RAW SCORES

GS	AR	WK	PC	NO	CS	AS	MK	MC	EI	VE
20	15	10	05	35	60	10	20	13	06	35

5. Algorithm Subtest Standard Scores

This algorithm converts the subtest raw scores into subtest standard scores. Instead of using the conversion tables to convert the raw scores, the standard scores are computed directly using the equation

$$SSS = [10 \times \frac{(\text{RAW SCORE} - \overline{X})}{\sigma}] + 50$$

The raw scores are read from input data. The expected value and the standard deviation for each subtest and ASVAB version have been calculated previously and are provided in the ASVAB program. Since the conversions must be done repeatedly, a function is used to calculate the conversions. This algorithm merely invokes the function for each subtest.

For example, if ASVAB 5 is in effect, to convert a raw score of 10 for GI the function is invoked by GISSS = CONVT(GI,XBAR,SIGMA). An actual computation is given below.

GISSS = CONVT(10,9.656,3.273)

$$GISSS = [10 \times \frac{(10 - 9.656)}{3.273}] + 50$$

GISSS = 51.05

The function rounds down to the nearest integer if the decimal value is less than 0.50, and rounds up if the decimal value is greater than or equal to 0.50. Since 0.05 is less than 0.50, GISSS = 51. This is the same value obtained using the conversion tables. This method is more efficient and requires less storage than the conversion table method. A decision table is used to determine which ASVAB version is in effect so that the correct raw scores, expected values, and standard deviations are used.

6. Algorithm Aptitude Area Composites

This algorithm uses the appropriately converted subtest scores from the previous algorithm to compute the Army Aptitude Composites. A decision table based on the value of TEST is used to determine which set of equations will be used. The equations for ASVAB 5, ASVAB 6/7, and ASVAB 8/9/10 are given in Tables 4, 7, and 10 respectively. This process is shown in Figure 5.

7. Algorithm ACB

This algorithm converts the scores computed in the previous algorithm into standard scores that comprise the Army Classification Battery. The process used is the table look-up process used in the manual system.

The computer stores in memory the conversion tables for ASVAB 5, ASVAB 6/7, and ASVAB 8/9/10. The three tables are stored as $m \times n$ matrices called A5, A67, and A8910, respectively. The variable m corresponds to the maximum number of rows in the matrix. Therefore, the maximum value of m must equal the maximum raw score possible. The variable n

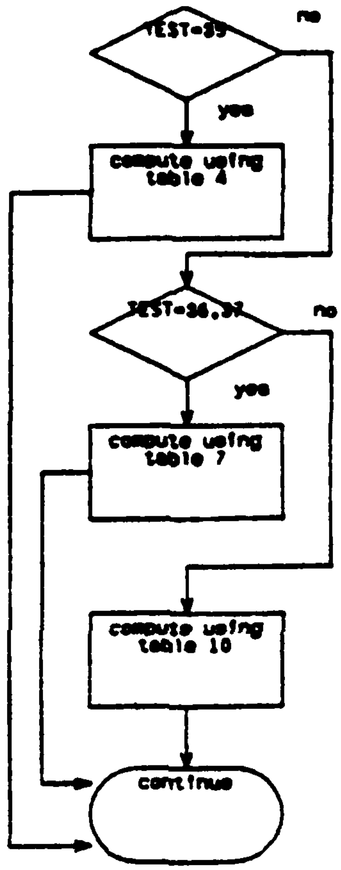

Figure 5. Algorithm Aptitude Area Composites Flow Chart

corresponds to the maximum number of columns in the matrix. Therefore, the maximum value of n must be 10 since there are 10 different scores to be converted. Table 13 shows the column assignments.

TABLE 13

MATRIX COLUMN ASSIGNMENTS

n	SCORE
1	GT
2	GM
3	EL
4	CL
5	MM
6	SC
7	CO
8	FA
9	OF
10	ST

For example, if the value of TEST is 38, 39, or 40 then ASVAB 8, 9, or 10 respectively is in effect. The algorithm makes the following assignments:

GT = A8910(GT,1)

GM = A8910(GM,2)

EL = A8910(EL,3)

CL = A8910(CL,4)

MM = A8910(MM,5)

SC = A8910(SC,6)

CO = A8910(C),7)

FA = A8910(FA,8)

OF = A8910(OF,9)

ST = A8910(ST,10)

The same procedure is used for ASVAB 5, and ASVAB 6/7, except that matrices A5 and A67 are used respectively.

8. Algorithm Output Scores

The purpose of this algorithm is to print the subtest standard scores, and the converted ACB composite scores. The algorithm output soldier data printed the following information:

SSAN	:	014560821
DATE OF BIRTH(YY/MM/DD)	:	64/4/20
SEX	:	1-MALE
RACE	:	2-BLACK
DATE OF ENTRY(YY/MM/DD)	:	82/2/16
HIGHEST YEAR OF EDUCATION	:	1 YEAR OF HIGH SCHOOL-03
ENTRY PAY GRADE	:	01-E1
MOS	:	11B10
TEST FORM - ASVAB 10	:	40-ASVAB 10
AFQT PERCENTILE	:	89
SEPARATION PROGRAM DESIGNATOR	:	JHK
DATE OF SEPARATION(YY/MM/DD)	:	83/5/25

SUBTEST RAW SCORES

GS	AR	WK	PC	NO	CS	AS	MK	MC	EI	VE
20	15	10	05	35	60	10	20	13	06	35

This algorithm adds the following lines giving the final soldier file.

SUBTEST STANDARD SCORES

GS	AR	WK	PC	NO	CS	AS	MK	MC	EI	VE
57	46	30	34	49	60	39	63	45	35	49

ARMY CLASSIFICATION BATTERY

CO	FA	MM	GM	CL	GT	EL	SC	ST	OF
93	108	78	95	102	93	101	94	107	86

In addition, the subtest raw scores, the subtest standard scores, and the ACB composite scores are each printed in a separate summary file.

E. THE OUTPUT FILES

ASVAB generates four output files. The soldier file is stored in File 9. The soldier file is the most important file and should be printed immediately by the user (see Appendix G). Invalid input data cards will be identified by error messages in the file. This allows the user to make corrections to input data.

In addition three summary files are generated (see Appendix G). A summary of the standard scores, the raw scores, and the composite scores are stored in Files 10, 11 and 12 respectively. The summary files do not contain any information that is not already contained in the soldier file. The summary files are not generated to provide information in paper copy form, but are generated in tabular form to provide input into SPSS and QLP 1100.

IV. SPSS AND QLP 1100

A. PURPOSE

The purpose of this chapter is to examine the relationship between ASVAB, the SPSS package, and the QLP 1100 system.

B. INTRODUCTION

ASVAB was designed to generate data quickly. Therefore, ASVAB alone will save the user a considerable amount of time. However, ASVAB has no analytic capability, and only limited data management capabilities. Once ASVAB is implemented, the user may find many applications which use the data in analysis and data management.

The analysis of the ASVAB data may prove to be prohibitive if a computer package is not used. For example, the computation of the mean and standard deviation of a set of composite scores is simple in concept. However in practice, given a sample size of 100 or more, the process becomes extremely time consuming if done manually. Other analysis such as analysis of variance, or regression analysis is almost impossible to do efficiently without computer assistance. Therefore, it is imperative that ASVAB output be linked to the SPSS package.

The analysis done by DOES is not strictly for internal use, but usually arises from external requests. In regard to this, the information provided in the ASVAB output files may not be sufficient or appropriate to satisfy external requirements. In addition, DOES may be required to submit various reports to the external agencies. A manual process of coding new ASVAB output formats, and typing reports is an inefficient use of time and personnel. This type of data management can be better done using the QLP 1100 system. Once again, the volume of potential, external requests demands that the QLP 1100 system be integrated with ASVAB.

C. SPSS

The user may be required to use various statistical methods on the data generated by ASVAB. SPSS is a comprehensive package that can be used to accomplish this. An SPSS program consists of specific keywords that are used to generate the analysis, as well as the data to be analyzed. The relationship between ASVAB and the SPSS program is shown in Figure 6.

One area for which the user may require SPSS is regression analysis. DOES is responsible for developing selection criteria for attendance to special infantry training. Regression analysis may be used to develop a model for selection criteria. This model will be described in order to provide an example of how ASVAB output can be used with an SPSS program.

In this example, assume a number of soldiers are selected at random to attend a special course. Upon completion of

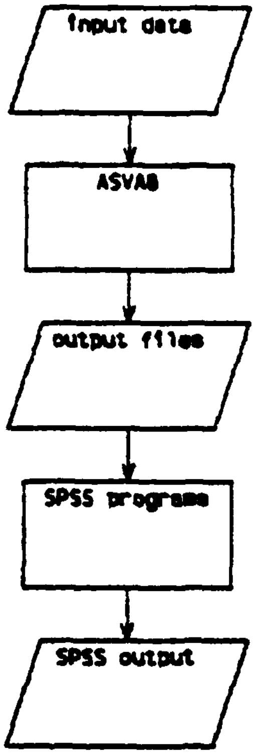

Figure 6. Relationship Between ASVAB and SPSS Programs

the course, the soldiers are administered the appropriate skill qualification test (SQT) to measure their ability to perform the tasks they were trained in. The results of the test are recorded along with the soldier's social security number. Regression analysis is used to develop a model that will predict SQT results as a function of some independent variables. In this example the 10 ACB composite scores are used to predict SQT results.

The user must process the input data cards in ASVAB. The soldier file should be printed in paper copy to check for entry errors. The SQT scores are entered in columns 11 through 15 of File 12, the composite summary. This may be done on disk from a terminal if available, or on the computer cards from the file if interactive processing is not available. The SPSS program is created on file or punched out on computer cards as appropriate. The new data are placed in the appropriate section of the file or deck (see Appendix H). The SPSS program generates all the necessary data to develop the model in final form (see Appendix J).

The R-square value is used to determine the percentage of variability in the SQT scores as explained by the variance in the ten composite scores. A high R-square value will mean that the ten composite scores are good predictors of SQT results. A low R-square value means that the ten composite scores should not be used as predictors. Further analysis such as checking for data points that are outliers, and residual analysis is done before the model developed by the SPSS package is accepted.

Once the analysis is completed and the model is validated, the SQT model becomes the basis for further selections to the training. SPSS can be used to compute the predicted SQT results for potential attendees prior to their training. In this case, the input cards are processed in ASVAB. Since the soldiers have not attended the training, no SQT results are entered in File 12. An SPSS program is created to compute the predicted SQT results from the model developed earlier (see Appendix I). The SPSS output will contain the predicted SQT results which can be used to select soldiers with the highest potential for success (see Appendix K).

In many instances the selection requirements may be more complex than just predicted SQT scores. For example, the training may involve special infantry combat training. In this case only males with infantry military occupation specialties, and high predicted SQT result can attend. The user would have to manually cross reference the list of attendees based on SQT results, with the soldier file to insure that the additional requirements are met. Alternatively, an application specific computer program can be developed to do the cross referencing. However, the development time may be greater than the time needed to manually cross reference the data. In either case time is not being used effectively.

In general, ASVABMIS does not have the flexibility to provide information on an as-needed basis. In its present form, ASVABMIS can only provide the information from its output files. A simple request for the number of black soldiers in the data pool requires additional programming, or a manual search.

In addition, ASVABMIS can not generate any reports. Once ASVAB and the SPSS package are used to generate and analyze data, any required reports must be manually created and typed.

In its present form, ASVABMIS is a rigid system unable to process requests for additional information without further programming. However, the QLP 1100 system provides a general program to handle additional requests that does not require additional programming.

D. QLP 1100

QLP 1100 provides a means to create a data base from the ASVAB output files. Since QLP 1100 is a COBOL-based system, the ASVAB output files must be converted to a COBOL format. The QLP 1100 system allows for both the conversion of FORTRAN files to COBOL files, and COBOL files to FORTRAN files.

QLP 1100 is not a true data base management system because many separate files are used, rather than a unique data base, to provide the data for manipulation. However, the end result appears the same. QLP 1100 creates a set of data that can be accessed by a query language.

In order to create the data set two steps are necessary. The first step is to define the physical and logical structure of the files that will be used. The second step is to define an application for the files by specifying which files will be included, and how the files will be used. Once these steps are completed an environment in which QLP 1100 can operate is established. Finally, the QLP 1100 query language is used to query the files as if they were a data base in order to generate reports and to manipulate data.

Two separate processors and their respective commands are used to accomplish the steps in creating the data base. The file definition processor (FDP) accomplishes the first step by defining the physical and logical attributes of the data files to be accessed, through the use of the FDP source input. More specifically, the FDP defines the file assignment characteristics, the physical file organization specifications, the name and structure specification for all records contained in the file, and the name and attribute specifications for all data items in the records. The FDP produces an internally formatted file specification which is used as input to the application definition processor (ADP).

The ADP accomplishes the second step by specifying which files, as described by the FDP, will be used in the application, as well as how the files will be used through the use of the ADP source input. The ADP produces internally formatted tables which are used by the query language processor (QLP) as input. The QLP then produces the final output as directed by the user through the use of the query language input. This process is depicted in Figure 7.

1. <u>File Definition Processor Source Input</u>

The FDP syntax contains two major subdivisions. The environment division names a file, and may specify other optional file information as deemed necessary by the user. The data division provides information about the physical structure, identification, and record name pertaining to a file. A simple FDP syntax containing the minimum required language is shown below. The line numbers are provided by the author for line identification, and are not a part of the syntax.

```
01        ENVIRONMENT DIVISION
02            SELECT file-name
03            ASSIGN TO DISK external-file-name
04        DATA DIVISION
05        FD file-name
06            LABEL RECORD IS STANDARD.
```

In this example, lines 01 and 04 specify the two subdivisions. Line 02 selects an existing file from storage by its name file-name. Line 03 allocates a storage medium which can either be disk or tape. The external file name is optional, and is the name under which the file will be stored. If the external file name is not specified, the new file will be stored using the first twelve characters of the selected file

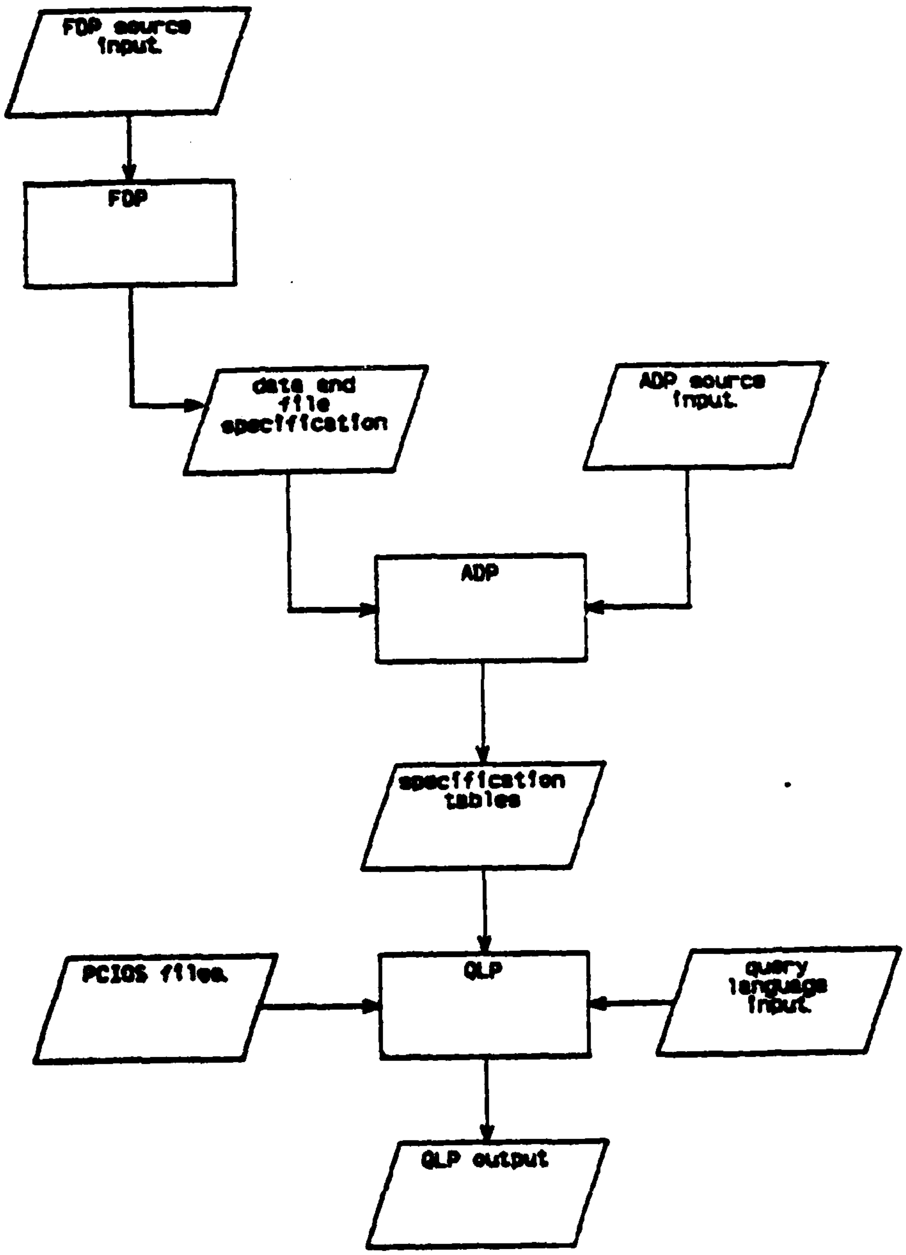

Figure 7. The QLP 1100 System

name. Line 05 is a link between the two divisions. The file name in line 05 must match the file name in line 02. Line 06 specifies that explicit labels exist for the file. The user may also use LABEL RECORD IS OMITTED which specifies that no labels exist for the file. A label is a heading, and a label record is a record that contains file headings.

The FDP syntax may be expanded based on the type of file to be created. If the file is to be a sequential file the syntax may be expanded as shown below.

```
01        ENVIRONMENT DIVISION
02        INPUT-OUTPUT SECTION
03        FILE CONTROL
04            SELECT file-name
05            ASSIGN TO DISK external-file-name
06            FILE QUALIFIER IS qualifier-name
07            ORGANIZATION IS SEQUENTIAL
08            SORT KEY IS ASCENDING data-name
09        DATA DIVISION
11        FD file-name
10            LABEL RECORD IS STANDARD
```

In this example, lines 01, 04, 05, 09, 10 and 11 are unchanged from the previous basic syntax. Lines 02 and 03 are descriptive and add further clarity. Line 06 specifies a file qualifier that is used in the calling sequence when the FDP is invoked. Line 07 specifies that the file is to be sequentially ordered. Line 08 specifies that the file will be sorted in an ascending order according to the values assigned to the key which is specified by the data name. For example, lines 07 and 08 could be used to sort a file in ascending order according to the social security account number by specifying:

```
ORGANIZATION IS SEQUENTIAL
SORT KEY IS ASCENDING SSAN
```

The user may also specify that the file be sorted in descending order by substituting DESCENDING for ASCENDING in line 08.

Similarly, the basic syntax may be expanded if the file is to be an indexed file rather than a sequential file. If the file is an indexed file the syntax may be expanded as shown below. In either case the user must decide which type of file is the most appropriate.

```
01      ENVIRONMENT DIVISION
02      INPUT-OUTPUT SECTION
03      FILE CONTROL
04          SELECT file-name
05          ASSIGN TO DISK external-file-name
06          FILE QUALIFIER IS qualifier-name
07          ORGANIZATION IS INDEXED
08          RECORD KEY IS data-name
09      DATA DIVISION
10      FD file-name
11          LABEL RECORD IS STANDARD
```

In this example, lines 01, 04, 05, 09, 10 and 11 are unchanged from the basic syntax. Lines 02, 03, and 06 have the same meaning as in the previous sequential example. Line 07 specifies that the file is random access rather than sequential. Line 08 specifies that a record can be identified by its key which has unique values, and is specified by the data name provided by the user. For example, since social security account numbers are unique they can be used to identify any record in a file.

Since the data division specifies the physical structure, identification, and record name of a file, the data division may be further specified with optional syntax if required. However, this syntax is normally used when default specifications are not appropriate. Some of the available optional syntax language is described below.

BLOCK CONTAINS integer-1 TO integer-2 RECORDS

This syntax specifies the block size in terms of the range of record numbers in the block.

DATA RECORD IS record-name-1

DATA RECORDS ARE record-name-1, record-name-2

This syntax is used when there is more than one record type in the file, and specifies the record selection by its name. More than one record type may be selected in multiple type files by listing the appropriate record names.

```
level number record-name
    level number data-name PICTURE IS character string
```

This syntax further describes the selected data record. The level number specifies the hierarchy of the data. For example, a record has a higher hierarchy than a data element, and would therefore have a level number of 01, while all the data elements would have level numbers of 02.

The PICTURE syntax describes the general characteristics of the data element as defined by the character string. It is in effect a format statement. The character string is a combination of symbols comprised of A, S, V, X and 9. These symbols are shown in Table 14.

TABLE 14

PICTURE CHARACTERS

CHARACTER	DESCRIPTION
A	A character position which can only contain a letter of the alphabet or a space.
S	A character position used to denote the presence of an operational sign, i.e., plus or minus.
V	A character position used to denote the location of a decimal point.
X	A character position which contains any allowable character from the computer's character set, i.e., $, /.
9	A character representing a numeral.

An example of the picture clause is shown below.

```
01  RA
    02  RA1  PICTURE IS  9999
    02  RA2  PICTURE IS  S9V999
    02  RA3  PICTURE IS  AAAA
    02  RA4  PICTURE IS  A999
```

This example specifies that the record RA has the highest hierarchy. The record has four data elements specified as RA1, RA2, RA3 and RA4. Furthermore, the syntax specifies that RA1 is a numeric character made up of four numerals. This specification may also be shown in a shorter notation by using a 9(4) character string. RA2 is a signed numeric character having a decimal point following the first numeral, and three numeral decimal values. RA3 is an alphabetic character with four positions. It may optionally be specified by an A(4) character string. Finally, RA4 is an alphanumeric character specifying one alphabetical character followed by three numerals. Its optional specification could have been shown as A9(3).

Combining the previous examples, a complete FDP source input is shown below.

```
ENVIRONMENT DIVISON
INPUT-OUTPUT SECTION
FILE CONTROL
    SELECT file-name
    ASSIGN TO DISK external-file-name
```

```
    FILE QUALIFIER IS qualifier-name
    ORGANIZATION IS INDEXED
    RECORD KEY IS data-name-2
DATA DIVISION
FD file-name
    LABEL RECORD IS OMITTED
    BLOCK CONTAINS integer-1 to integer-2 RECORDS
    DATA RECORD IS record-name
01 record-name
    02 data-name-1 PICTURE IS AAA
    02 data-name-2 PICTURE IS 999
```

In this example, the syntax specifies that a file called file-name will be taken from storage and assigned to disk storage under the name external-file-name. The file is assigned a qualifier name which is used in the FDP calling sequence when invoking the FDP. The file is an indexed file, and each record in the file can be identified by the key named date-name-2.

The data division further specifies that the file has no explicit labels. In addition, the range of record numbers in a block is specified. Also, the file contains multiple record types, but only one record type having the name record name will be used. Finally, the record has two data elements, the first of which is a three position alphabetic character. The last data element is specified to be a three numeral numeric character.

When invoked, the FDP will accept the FDP source input and initiate a completely independent run. That is, different constructs of FDP source input will produce different FDP run results. The output from a FDP run consists of an internally formatted description of the file specification defined by the FDP source input. This output, along with the ADP source input, is used as input into the ADP.

2. Application Definition Processor Source Input

The ADP links the FDP output and the separate files to produce a processor common input/output system (PCIOS) file. The PCIOC file becomes the data base to be accessed by the QLP. This is accomplished through the ADP source input.

The ADP syntax has two divisions. The first division is the identification division which simply contains the name of the application. The second division is the data division which contains the names of the files to be used in the application, as well as specifying the location of the FDP file description, and the usage restriction on the file. A complete ADP source input syntax is shown below.

```
01      IDENTIFICATION DIVISION
02          APPLICATION NAME IS application-name
03      DATA DIVISION
04          FILE SECTION
05              FILE NAME IS file-name-1
06                  DEFINED IN FILE file-name-2
```

```
07                          ELEMENT element-name-1
08                          QUALIFIER qualifier-name-1
09                      USE IS RETRIEVAL
```

In this example, lines 01, 03 and 04 are descriptive and provide clarity. Line 02 assigns a name to this specific application. Different applications can be applied to the same FDP run. Therefore, the application name is used to identify specific runs. The application name is also used to define where the application run will reside. This is important when linking to the QLP. Line 05 specifies which files will be used in this application and is the same file specified in the SELECT clause, and the FD clause in the FDP source input. Lines 06, 07, and 08 specify precisely where the FDP run output is located. This information is included in the FDP calling sequence. Finally, line 09 defines how the application will be used. The word RETRIEVAL means that the file can only be read. The words UPDATE or OUTPUT may be substituted for RETRIEVAL. The word OUTPUT specifies that the file is intended for initial creation. Finally, the word UPDATE specifies that the file may be both read and written.

The ADP source input along with the FDP output produces an independent ADP run. The output from the ADP run produces internally formatted tables that are used as input into the QLP.

3. FDP and ADP Call Commands

Before the FDP or ADP source input can be used to produce processor runs, each processor must be invoked using call commands. The basic syntax for the call commands is shown below and are described in Table 15.

@ file-name. FDP, options, SI, RO, SO

@ file-name. ADP, options, SI, RO, SO

TABLE 15

CALL SYNTAX

ELEMENT	DESCRIPTION
file-name	Is the name of the file containing the absolute FDP or ADP.
options	Are specified through the choice of one or more option letters.
D	Produce an allocation summary for each record in the file. This may be used only on the ADP.
I	Insert a new symbolic element in the program file.
P	Card image input, if any, is in fieldata. The output symbolic element is in fieldata.
Q	Output symbolic element is in ACIII. Card image input, if any, is in ASCII.
S	The source input will be echoed to the user.
U	Update and produce a new cycle of the symbolic element.
W	List correction lines.

The characters SI, RO, and SO are parameters that stand for source input, relocatable output, and source output, respectively. The meaning of the parameters are shown in Table 16.

TABLE 16

PARAMETER DESCRIPTIONS

PARAMETER	DESCRIPTION
SI	If a new element is being introduced from the run, this parameter specifies the file into which the new element is placed and the name it is given.
RO	This parameter specifies the name and program file into which the element produced by the processor is placed.
SO	This parameter specifies the name and the file for the updated symbolic element produced.

An example of a processor call is shown below.

```
@ FDP.FDP,S FDP*FDPEXAMPLE.SOURCE,FDP*FDPEXAMPLE.RELOCATABLE
```

In this example, the file name is FDP, and the option is S. The source input contains the actual source description for this file. More specifically the qualifier is FDP, the file name is FDPEXAMPLE, and the element is RELOCATABLE. The relocatable output is specified by the statement FDP*FDPEXAMPLE.RELOCATABLE and is the location where the FDP output will reside.

The processor calls are combined with the FDP and ADP source input to completely define the file and application definitions. This combination creates a PCIOS file which can be accessed by the QLP.

The procedure for creating the PCIOS file from the 3 ASVAB summary files raw, standard, and composite is explained below.

STEP 1: Produce an FDP description of file raw.

```
@FDP.FDP,S FDP*SCORES.SOURCE,FDP*SCORES.RELOC-RAW
  ENVIRONMENT DIVISION
  INPUT-OUTPUT SECTION
  FILE-CONTROL
      SELECT RAW
      ASSIGN TO DISK
      FILE QUALIFIER IS FDP
      ORGANIZATION IS INDEXED
      RECORD KEY IS SSAN
  DATA DIVISION
  FILE SECTION
  FD RAW
      LABEL RECORDS ARE STANDARD
```

STEP 2: Produce an FDP description of file standard.

```
@FDP.FDP,S FDP*SCORES.SOURCE,FDP*SCORES.RELOC-STAND
  ENVIRONMENT DIVISION
  INPUT-OUTPUT SECTION
  FILE-CONTROL
```

```
    SELECT STANDARD
    ASSIGN TO DISK
    FILE QUALIFIER IS FDP
    ORGANIZATION IS INDEXED
    RECORD KEY IS SSAN
DATA DIVISION
FILE SECTION
FD STANDARD
    LABEL RECORDS ARE STANDARD
```

STEP 3: Produce an FDP description of the file composite.

```
@FDP.FDP,S FDP*SCORES.SOURCE,FDP*SCORES.RELOC-COMP
  ENVIRONMENT DIVISION
  INPUT-OUTPUT SECTION
  FILE CONTROL
      SELECT COMPOSITE
      ASSIGN TO DISK
      ORGANIZATION IS INDEXED
      RECORD KEY IS SSAN
  DATA DIVISION
  FILE SECTION
  FD COMPOSITE
      LABEL RECORDS ARE STANDARD
```

STEP 4: The ADP is used to identify which files are to be used in this application.

```
@ADP.ADP,ADP*SCORES.SOURCE-BATTERY,ADP*SCORES,ACB-BATT
  IDENTIFICATION DIVISION
      APPLICATION NAME IS ACB-BATT
  DATA DIVISION
      FILE SECTION
          FILE NAME IS RAW
              DEFINED IN FILE SCORES
                  ELEMENT RELOC-RAW
                      QUALIFIER IS FDP
              USE IS RETRIEVAL
              FILE NAME IS STANDARD
                   DEFINED IN FILE SCORES
                        ELEMENT RELOC-STAND
                        QUALIFIER IS FDP
              USE IS RETRIEVAL
              FILE NAME IS COMPOSITE
                  DEFINED IN FILE SCORES
                       ELEMENT RELOC-COMP
                       QUALIFIER IS FDP
              USE IS RETRIEVAL
```

STEP 5: The QLP is invoked

```
  INVOKE ACB-BATT OF ADP*SCORES
```

With this accomplished all files and applications are defined. The application tables produced by the ADP are saved in ADP*SCORES.ACB-BATT, and the QLP has been invoked and is prepared to accept QLP 1100 query commands.

4. <u>Query Language Processor, 1100 Series</u>

The QLP accepts as input the ADP output and its specific QLP 1100 language. The QLP output consists of reports and data files that can be stored or printed. The QLP 1100 language is subdivided into eight functional facilities. Each facility has several commands associated with it. The facilities are:

Data system interface facilities
Conversational facilities
Data selection facilities
Operational facilities
Definitional facilities
Logic control and manipulative facilities
Data access and data base network facilities
Savefile facilities

The last four facilities are complex and are intended for use by experienced programmers. They will be given only a cursory explanation. The first four facilities will be explained in more detail.

a. Data System Interface Facilities

(1) INVOKE. INVOKE establishes a link between the user and the data files to be accessed. The syntax is:

INVOKE application-name OF qualifier-name*file-name

An example of the syntax is:

INVOKE ACB-BATT OF ADP*SCORES

This command links the user to application ACB-BATT and the stored output file ADP*SCORES.

(2) EXIT. EXIT releases the link between the user and the file, and terminates QLP processing.

b. Conversational Facilities

(1) LIST. LIST provides the user with a list of data values as specified by the WHERE clause. The syntax is:

LIST data-identifier WHERE expression

An example of the syntax is:

LIST SSAN WHERE AFQT > 90

This command will provide the response:

SSAN = 242968813

SSAN = 016558905

The LIST command may be expanded according to the user's needs. An example of an expanded syntax is:

LIST SSAN, MOS WHERE AFQT > 90

This command will provide the response:

```
SSAN = 242968813

MOS = 11B10

SSAN = 016558905

MOS = 11C30
```

The SORT command may be used in conjunction with the LIST command in the following manner:

```
LIST SSAN,MOS,AFQT SORTED ON ASCENDING AFQT
WHERE AFQT > 90
```

This command will provide the response:

```
SSAN = 016558905

MOS = 11C30

AFQT = 96

SSAN = 242968813

MOS = 11B10

AFQT = 92
```

(2) COUNT. COUNT tallies the number of occurrences specified by the COUNT clause, and qualified by the WHERE clause. The syntax is:

```
COUNT data-identifier WHERE expression
```

An example of the syntax is:

```
COUNT SSAN WHERE AFQT < 89
```

This command will provide the response:

312 SSAN RECORDS SELECTED

(3) CHANGE. CHANGE modifies, as specified, the data elements qualified by the WHERE clause. The syntax is:

CHANGE data-identifier=expression WHERE expression

An example of the syntax is:

CHANGE DES='AAA' WHERE SSAN = 041358991

This command changes the DES value to AAA for the record identified by the social security number 041358991.

(4) REPEAT. REPEAT executes the previous command with a new WHERE clause. The syntax is:

REPEAT WHERE expression

The REPEAT command may be used when a previous command did not provide valid results. For example, if the syntax LIST SSAN WHERE AFQT > 101 did not provide valie results because there is no such percentile, the list command may be repeated by commanding:

REPEAT WHERE AFQT > 75

c. Data Selection Facilities

WHERE specifies the criteria for data selection. The syntax is:

```
WHERE expression
```

d. Operational Facilities

OUTPUT defines the destination of subsequent query output to a terminal, printer, or file. The syntax is:

```
OUTPUT TO device
```

For example, in order to send a list command to the printer and then return all output to the originating device the following commands are issued:

```
OUTPUT TO PRINTER
LIST SSAN WHERE AFQT > 70
OUTPUT TO ORIGINATOR
```

e. Definitional Facilities

FORMAT allows the user to specify a simple tabular output form with column headings. The syntax is:

```
FORMAT format-name
    format-clause
END FORMAT
```

The FORMAT syntax is used with other syntax to generate the tables. For example, in order to generate a table with headings, the LIST command is used as shown below.

```
FORMAT FORM
    'SSAN' SSAN 1, 'PERCENTILE' AFQT 20
END FORMAT
LIST USING FORMAT FORM WHERE AFQT > 90
```

The FORMAT syntax specifies that there are two columns to be printed. The first column occupies print positions 1 through 19, while the second column starts in column 20. The table will appear as shown below.

SSAN	PERCENTILE
242968813	92
016558905	96

f. Logic Control and Manipulative Facilities

These facilities allow the user to create an algorithm of commands, known as a macro, that can accomplish an application that can not be accomplished with one command. The commands that accomplish this are DO, IF, LEAVE, and RETURN.

g. Data Access and Data Base Network Facilities

These facilities allow the user to process records one at a time, by specifying pointers to specific records.

h. SAVEFILE Facilities

SAVE allows the user to save files. The syntax is:

```
SAVE savefile-name INTO PERMANENT SAVEFILE.
```

In order to save a file named FILE1 the following command is issued:

```
SAVE FILE1 INTO PERMANENT SAVEFILE
```

5. <u>Summary Example</u>

In this example the user has a requirement to provide a selection listing of potential attendees for a special school. The selection listing must provide the social security numbers of the attendees selected, along with a justification for the selection. In addition the number of potential attendees who were selected must be provided.

The user determines that the regression model developed from previous classes will be used to predict SQT scores. The selection criteria are determined from predicted SQT scores first, and AFQT percentile second. The user also determines that only potential attendees with a predicted SQT score of 50 or better are qualified to attend.

The user processes the input data cards in ASVAB, and uses the SPSS program to compute the predicted SQT result. The data in the composite score summary file, and the predicted SQT results are combined into one file using the QLP 1100 command;

```
BUILD COBOL FILE REGDATA ON DISK FROM SSAN, SQT, AFQT,
CO, FA, MM, GM, CL_ GT, EL, SC, ST, OF
```

The user must select the file REGDATA in the FDP and ADP source input. The QLP is then invoked. In order to satisfy the requirement the user would issue the QLP 1100 commands shown below.

```
FORMAT REQUIRE5
    'SSAN'SSAN1, 'SQT'/'RANK' SQT12, 'AFQT'/'RANK' AFQT16
END FORMAT
OUTPUT TO PRINTER
LIST USING FORMAT REQUIRE5 SORTED ON DESCENDING SQT
SORTED ON DESCENDING AFQT WHERE SQT > 50
COUNT SSAN WHERE SQT > 50
OUTPUT TO ORIGINATOR
EXIT
```

This QLP 1100 query session will produce the output shown below.

```
SSAN        SQT       AFQT
            RANK      RANK
232489961   85        60
016558843   75        55
114863105   75        50
   .
   .
   .
404358221   51        45
97 SSAN RECORDS SELECTED
```

The entire ASVABMIS system as used in the above example is shown in Figure 8. This figure shows that ASVAB can be integrated with SPSS and QLP 1100 into one information system.

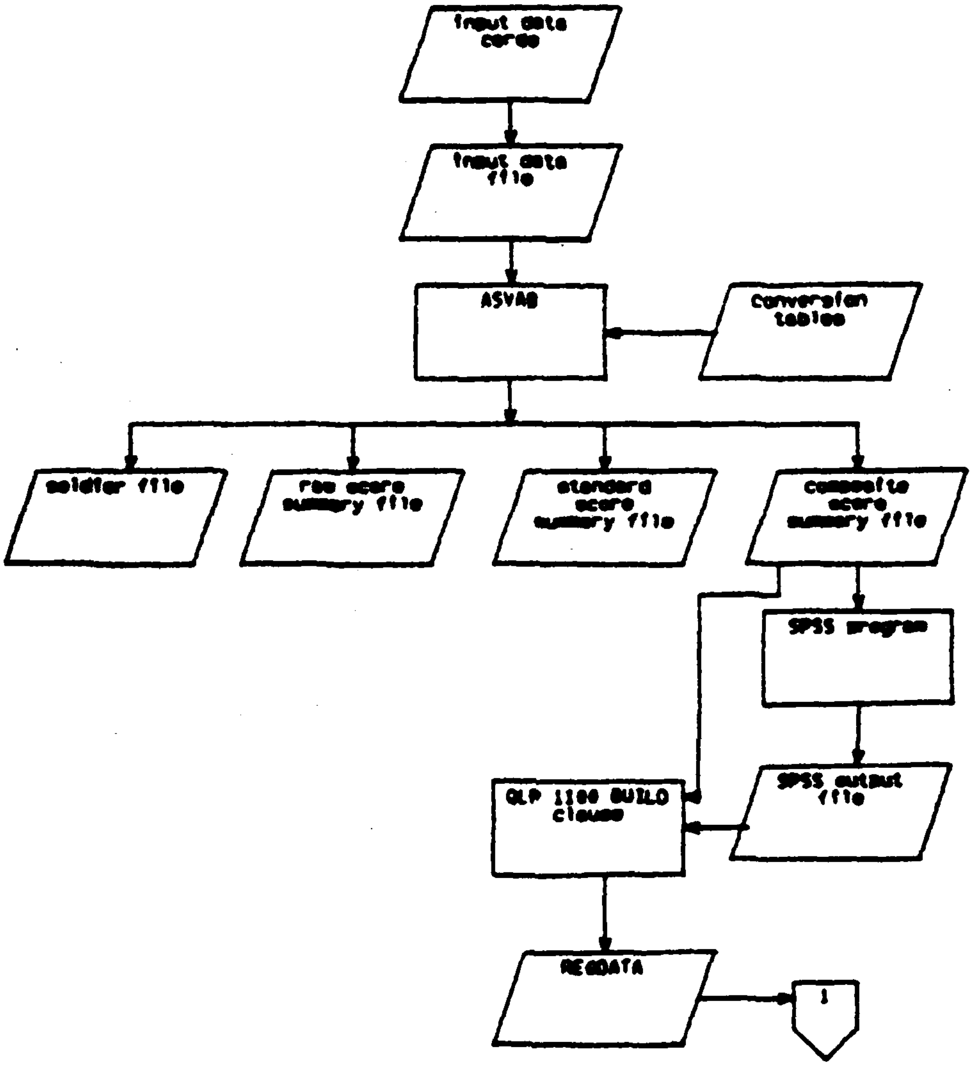

Figure 8. The ASVABMIS System

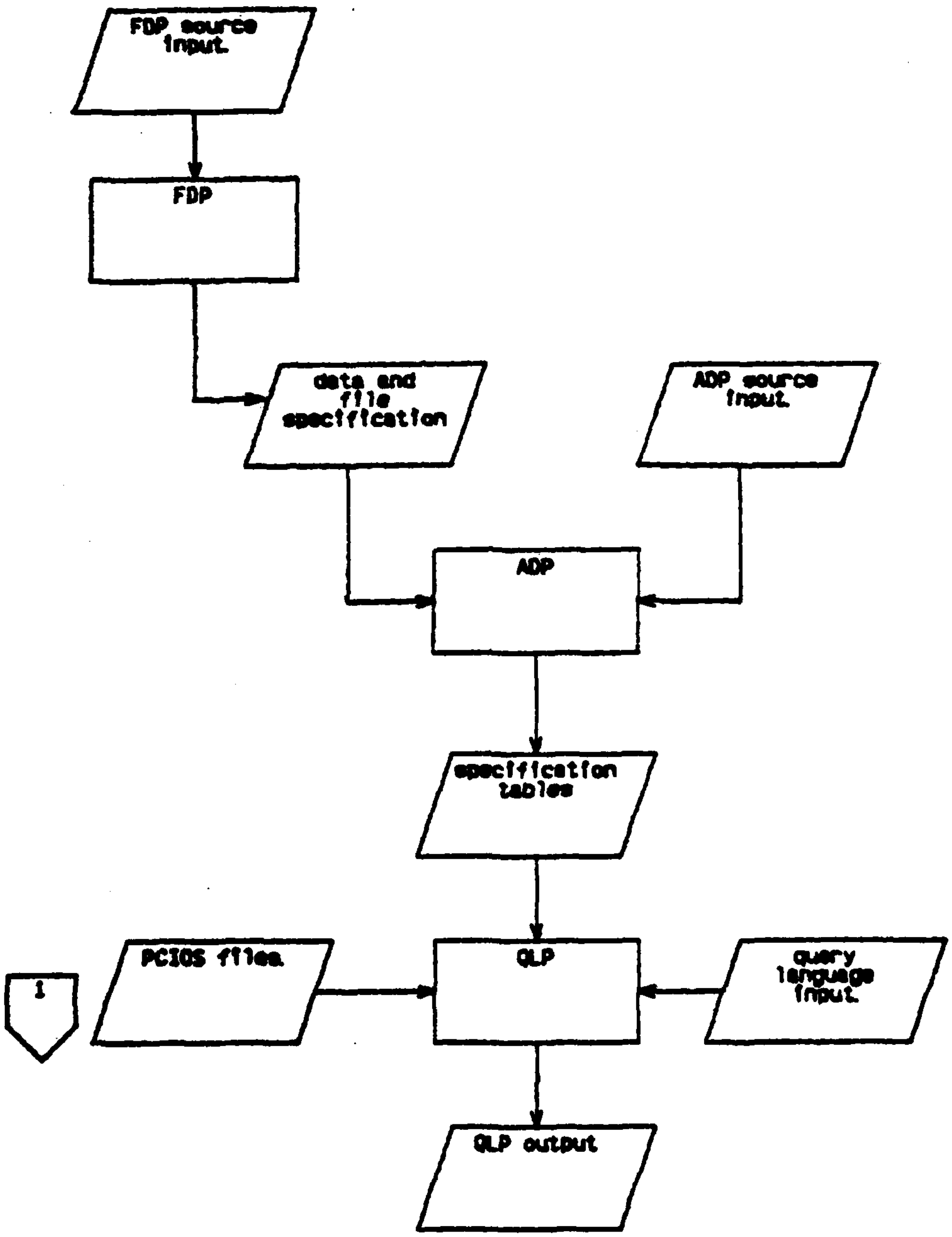

Figure 8. (CONT.)

V. CONCLUSIONS

The management information system has been developed in order to provide the user with a means to generate, analyze and manage data concerning the Armed Services Aptitude Battery. The ASVAB program is complete and can be immediately installed by the user. This thesis provides the necessary program documentation, and the user need not acquire additional documents. The FORTRAN output files can be used immediately with SPSS programs. However, since only a few sample SPSS programs are provided, the user may find it necessary to obtain the SPSS manuals listed in the bibliography in order to develop other programs. The sample SPSS programs were processed on a computer and are valid. The user can install them as they are written.

The treatment of the QLP 1100 system was introductory only. The user will find it necessary to obtain QLP 1100 manuals in order to further develop this part of the system. The chapter covering the QLP 1100 system could not provide all the necessary information for a complete understanding of QLP 1100. However, the coverage of QLP 1100 is specific enough to demonstrate the ease of operation, and efficiency of the system. This in itself should prompt further development of the QLP 1100 system.

The ASVABMIS system has been designed for use by U.S. Army agencies. The ASVAB program can be expanded for use by the Air Force and Marine Corps. The same raw scores are used by all 3 services. The major difference is the type of composites computed.

The ASVABMIS system was developed specifically for the Infantry Center's Department of Evaluation and Standardization. The applications discussed in this thesis are relevant to similar departments of the Armor, Field Artillery, and other centers.

A future expansion of the system worthy of note is the installation of ASVABMIS on a mini or micro computer. The ASVAB program can be written in BASIC. In addition, vendor-developed statistical packages and data base systems can be used on small computers in a manner similar to SPSS and QLP 1100.

In conclusion, the ASVABMIS system developed in this thesis will provide the user with the ability to generate and analyze ASVAB data. A limited ability to manage data is also provided. However, extensive data management will require further development by the user.

APPENDIX A

INPUT DATA FILE

```
27255884364042012820216030111B103547152003120617201009051312   AKL830227
016532241631125111810920060111B10368713402522151612231009171408121322   AAA821109
21386552763052123821007070254E1040922422301445792520241948   AAA831015
23544232163052311821113060111B1041852321291344722020211823   AAA831110
015342237640809118109250601 11B10377515392521151612231112171310121332   AKL840915
```

APPENDIX B

ASVAB 5/6/7 SSS CONVERSION TABLE

	GI	WK	AR	SP	AD	MK	EI	MC	GS	AI	SI	VE
0	20	23	23	20	20	26	20	25	24	26	20	20
1	24	24	25	21	20	28	20	27	26	28	21	20
2	27	26	27	24	20	30	21	30	29	30	23	21
3	30	27	29	26	21	32	22	32	31	32	25	22
4	33	28	32	28	24	35	24	34	33	34	28	23
5	36	30	34	31	26	37	26	37	36	36	30	24
6	39	31	36	33	29	39	27	39	38	38	32	25
7	42	33	38	35	31	41	29	41	40	40	35	26
8	45	34	40	38	34	43	31	43	42	42	37	27
9	48	35	42	40	36	45	32	46	45	44	39	28
10	51	37	44	42	39	47	34	48	47	46	42	29
11	54	38	46	45	41	49	36	50	49	48	44	30
12	57	39	48	47	44	51	37	53	52	50	46	31
13	60	41	51	50	46	53	39	55	54	52	48	32
14	63	42	53	52	49	55	41	57	56	55	51	33
15	66	44	55	54	51	57	42	60	58	57	53	34
16		45	57	57	54	59	44	62	61	59	55	35
17		46	59	59	57	61	46	64	63	61	58	36
18		48	61	61	59	63	48	66	65	63	60	37
19		49	63	64	62	65	49	69	68	65	62	38
20		50	65	66	64	67	51	71	70	67	65	39
21		52			67		53					40
22		53			69		54					41
23		55			72		56					42
24		56			74		58					43
25		57			77		59					44
26		59			79		61					45
27		60			80		63					46
28		62			80		64					47
29		63			80		66					48
30		64			80		68					49
31												50
32												51
33												52
34												53
35												54
36												55
37												56
38												57
39												58
40												59
41												60
42												61
43												62
44												63
45												64
46												65
47												66
48												67
49												68
50												69

APPENDIX C

ASVAB 8/9/10 SSS CONVERSION TABLE

	GS	AR	WK	PC	NO	CS	AS	MK	MC	EI	VE
0	20	25	20	20	20	23	21	29	22	21	20
1	20	27	20	22	20	24	22	31	24	23	20
2	22	28	20	25	20	24	24	32	26	26	20
3	24	29	20	28	20	25	26	34	28	28	20
4	26	31	22	31	20	26	28	36	29	30	20
5	28	32	23	34	20	26	30	37	31	33	21
6	30	34	24	37	21	27	31	39	33	35	22
7	32	35	26	40	22	28	33	41	35	37	23
8	34	36	27	43	23	28	35	42	37	40	23
9	36	38	28	46	24	29	37	44	38	42	24
10	38	39	30	49	25	29	39	46	40	44	25
11	40	41	31	52	26	30	40	48	42	47	26
12	42	42	32	54	27	31	42	49	44	49	27
13	44	43	33	57	28	31	44	51	45	51	28
14	46	45	35	60	29	32	46	53	47	54	29
15	48	46	36	63	30	33	47	54	49	56	30
16	50	48	37		31	33	49	56	51	58	31
17	52	49	39		32	34	51	58	53	61	32
18	54	50	40		33	34	53	59	54	63	33
19	56	52	41		34	35	55	61	56	65	34
20	57	53	43		35	36	56	63	58	67	35
21	59	54	44		36	36	58	64	60		36
22	61	56	45		36	37	60	66	62		37
23	63	57	47		37	38	62	68	63		38
24	65	59	48		38	38	64	69	65		39
25	67	60	49		39	39	65	71	67		39
26		61	50		40	39					40
27		63	52		41	40					41
28		64	53		42	41					42
29		66	54		43	41					43
30		67	56		44	42					44
31			57		45	42					45
32			58		46	43					46
33			60		47	44					47
34			61		48	44					48
35			62		49	45					49
36					50	46					50
37					51	46					51
38					52	47					52
39					53	47					53
40					54	48					54
41					55	49					55
42					56	49					56
43					57	50					56
44					58	51					57
45					59	51					58
46					60	52					59
47					61	52					60
48					62	53					61
49					62	54					62
50					63	54					63
51						55					
52						55					
53						56					
54						57					
55						57					
56						58					
57						59					

	GS	AR	WK	PC	NO	CS	AS	MK	MC	EI	VE
58						59					
59						60					
60						60					
61						61					
62						62					
63						62					
64						63					
65						64					
66						64					
67						65					
68						65					
69						66					
70						67					
71						67					
72						68					
73						69					
74						69					
75						70					
76						70					
77						71					
78						72					
79						72					
80						73					
81						73					
82						74					
83						75					
84						75					

APPENDIX D

ASVAB 5 COMPOSITE CONVERSION TABLE

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
1	41	41	41	41	41	41	41	41	41	41
2	41	41	41	41	41	41	41	41	41	41
3	41	41	41	41	41	41	41	41	41	41
4	41	41	41	41	41	41	41	41	41	41
5	41	41	41	41	41	41	41	41	41	41
6	41	41	41	41	41	41	41	41	41	41
7	41	41	41	41	41	41	41	41	41	41
8	41	41	41	41	41	41	41	41	41	41
9	41	41	41	41	41	41	41	41	41	41
10	41	41	41	41	41	41	41	41	41	41
11	41	41	41	41	41	41	41	41	41	41
12	41	41	41	41	41	41	41	41	41	41
13	41	41	41	41	41	41	41	41	41	41
14	41	41	41	41	41	41	41	41	41	41
15	41	41	41	41	41	41	41	41	41	41
16	41	41	41	41	41	41	41	41	41	41
17	41	41	41	41	41	41	41	41	41	41
18	41	41	41	41	41	41	41	41	41	41
19	41	41	41	41	41	41	41	41	41	41
20	41	41	41	41	41	41	41	41	41	41
21	41	41	41	41	41	41	41	41	41	41
22	41	41	41	41	41	41	41	41	41	41
23	41	41	41	41	41	41	41	41	41	41
24	41	41	41	41	41	41	41	41	41	41
25	41	41	41	41	41	41	41	41	41	41
26	41	41	41	41	41	41	41	41	41	41
27	41	41	41	41	41	41	41	41	41	41
28	41	41	41	41	41	41	41	41	41	41
29	41	41	41	41	41	41	41	41	41	41
30	41	41	41	41	41	41	41	41	41	41
31	41	41	41	41	41	41	41	41	41	41
32	41	41	41	41	41	41	41	41	41	41
33	41	41	41	41	41	41	41	41	41	41
34	41	41	41	41	41	41	41	41	41	41
35	41	41	41	41	41	41	41	41	41	41
36	41	41	41	41	41	41	41	41	41	41
37	41	41	41	41	41	41	41	41	41	41
38	41	41	41	41	41	41	41	41	41	41
39	41	41	41	41	41	41	41	41	41	41
40	41	41	41	41	41	41	41	41	41	41
41	41	41	41	41	41	41	41	41	41	41
42	41	41	41	41	41	41	41	41	41	41
43	41	41	41	41	41	41	41	41	41	41
44	41	41	41	41	41	41	41	41	41	41
45	41	41	41	41	41	41	41	41	41	41
46	41	41	41	41	41	41	41	41	41	41
47	41	41	41	41	41	41	41	41	41	41
48	41	41	41	41	41	41	41	41	41	41
49	41	41	41	41	41	41	41	41	41	41
50	41	41	41	41	41	41	41	41	41	41
51	41	41	41	41	41	41	41	41	41	41
52	41	41	41	41	41	41	41	41	41	41
53	41	41	41	41	41	41	41	41	41	41
54	41	41	41	41	41	41	41	41	41	41
55	41	41	41	41	41	41	41	41	41	41
56	41	41	41	41	41	41	41	41	41	41
57	41	41	41	41	41	41	41	41	41	41
58	41	41	41	41	41	41	41	41	42	41

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
59	43	41	41	41	41	41	41	41	45	41
60	45	41	41	41	41	41	41	41	47	41
61	49	41	41	41	41	41	41	41	49	41
62	53	41	41	41	41	41	41	41	52	41
63	55	41	41	41	41	41	41	41	55	41
64	57	41	41	41	41	41	41	41	57	41
65	59	41	41	41	41	41	41	41	59	41
66	61	41	41	41	41	41	41	41	60	41
67	62	41	41	41	41	41	41	41	61	41
68	63	41	41	41	41	41	41	41	62	41
69	65	41	41	41	41	41	41	41	63	41
70	66	41	41	41	41	41	41	41	64	41
71	67	41	41	41	41	41	41	41	65	41
72	68	41	41	41	41	41	41	41	66	41
73	69	41	41	41	41	41	41	41	67	41
74	70	41	41	41	41	41	41	41	68	41
75	71	41	41	41	41	41	41	41	69	41
76	72	41	41	41	41	41	41	41	70	41
77	73	41	41	41	41	41	41	41	71	41
78	74	41	41	41	41	41	41	41	72	41
79	75	41	41	41	41	41	41	41	73	41
80	76	41	41	41	41	41	41	41	74	41
81	77	41	41	41	41	41	41	41	75	41
82	78	41	41	41	41	41	41	41	77	41
83	79	41	41	41	41	41	41	41	78	41
84	81	41	41	41	41	41	41	41	79	41
85	82	41	41	41	41	41	41	41	80	41
86	83	41	41	41	41	41	41	41	81	41
87	84	41	41	41	41	41	41	41	82	41
88	85	41	41	41	41	41	41	41	84	41
89	87	41	41	42	41	41	41	41	85	41
90	88	41	41	43	41	41	41	41	86	41
91	89	41	41	43	41	41	41	41	88	41
92	91	41	41	44	41	41	41	41	89	41
93	92	41	41	45	41	41	41	41	90	41
94	93	41	41	46	41	41	41	41	92	41
95	95	41	41	47	41	41	41	41	93	41
96	96	41	41	49	41	41	41	41	94	41
97	97	41	41	51	41	41	41	41	95	43
98	98	41	41	53	41	41	41	41	97	45
99	99	41	41	54	41	41	41	41	98	48
100	100	41	41	55	41	41	41	41	99	51
101	100	41	41	56	41	41	41	41	100	54
102	101	41	41	56	41	41	41	41	101	56
103	102	41	41	57	41	41	41	41	102	57
104	103	41	41	58	41	41	41	41	103	59
105	104	41	41	59	41	41	41	41	104	60
106	105	41	41	60	41	41	41	41	105	61
107	106	41	41	61	41	41	41	41	106	63
108	107	41	41	62	41	41	41	41	107	64
109	108	41	41	63	41	41	41	41	108	65
110	109	41	41	63	41	41	41	41	109	66
111	110	41	41	64	41	41	41	41	110	67
112	111	41	41	65	41	41	41	41	111	68
113	112	41	41	65	41	41	41	41	112	69
114	113	41	41	66	41	41	41	41	113	69
115	114	41	41	67	41	41	41	41	115	70
116	115	41	41	68	41	41	41	41	116	71
117	116	41	41	68	41	41	41	41	118	72
118	117	41	42	69	41	41	41	41	119	73
119	118	41	43	70	41	41	41	41	120	74
120	119	41	44	70	41	41	42	42	121	75
121	120	41	45	71	42	41	42	43	122	76
122	121	41	45	72	42	41	43	44	123	77
123	122	41	46	73	43	42	43	45	124	78
124	124	41	46	73	44	43	44	46	125	79

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
125	125	41	47	74	45	43	44	47	126	80
126	127	41	48	75	46	44	44	48	127	81
127	129	41	49	76	47	44	45	49	128	82
128	133	42	51	77	49	45	45	51	129	83
129	138	44	53	78	51	46	46	52	131	84
130	138	45	55	79	53	47	47	54	133	85
131	138	46	56	80	54	48	48	55	135	86
132	138	47	57	80	55	50	49	56	137	87
133	138	49	58	81	56	51	50	57	138	88
134	138	51	59	82	57	53	51	58	138	89
135	138	53	59	83	58	54	53	59	138	90
136	138	55	60	84	59	55	55	60	138	91
137	138	56	60	85	60	56	56	60	138	92
138	138	57	61	86	60	58	56	61	138	93
139	138	58	62	88	61	59	57	61	138	94
140	138	59	62	89	62	60	57	62	138	95
141	138	60	63	90	62	61	58	63	138	95
142	138	61	63	91	63	62	58	63	138	96
143	138	62	64	93	63	62	59	64	138	97
144	138	62	65	94	64	63	60	65	138	97
145	138	63	65	95	64	64	60	65	138	98
146	138	64	66	96	65	64	61	65	138	99
147	138	64	66	97	65	65	62	66	138	99
148	138	65	67	98	66	66	62	66	138	100
149	138	66	67	99	67	66	63	67	138	101
150	138	66	68	100	67	67	63	67	138	101
151	138	67	68	100	68	68	64	68	138	102
152	138	68	69	101	69	68	65	69	138	102
153	138	69	70	102	69	69	65	69	138	103
154	138	70	70	102	70	70	66	70	138	104
155	138	70	71	103	70	70	66	71	138	104
156	138	71	71	104	71	71	67	71	138	105
157	138	72	72	105	71	72	67	72	138	105
158	138	73	73	106	72	72	68	72	138	106
159	138	73	73	106	73	73	68	73	138	107
160	138	74	74	107	73	74	69	74	138	107
161	138	75	74	108	74	74	70	74	138	108
162	138	75	75	109	75	75	70	75	138	109
163	138	76	76	110	76	76	71	75	138	109
164	138	76	76	111	76	76	71	76	138	110
165	138	77	77	112	77	77	72	77	138	111
166	138	78	77	113	77	78	72	77	138	111
167	138	79	78	114	78	78	73	78	138	112
168	138	79	79	115	78	79	74	79	138	113
169	138	80	79	116	79	80	75	79	138	113
170	138	81	80	117	80	80	75	80	138	114
171	138	82	81	118	80	81	76	81	138	115
172	138	83	82	118	81	82	77	82	138	115
173	138	83	83	119	81	83	77	82	138	116
174	138	84	83	120	82	83	78	83	138	116
175	138	85	84	120	83	84	79	84	138	117
176	138	86	84	121	84	85	80	84	138	117
177	138	87	85	122	84	86	80	85	138	118
178	138	88	85	123	85	87	81	85	138	119
179	138	89	86	123	86	88	82	86	138	119
180	138	89	87	124	87	89	83	87	138	120
181	138	90	88	124	87	90	84	88	138	120
182	138	91	89	125	88	90	85	89	138	121
183	138	91	90	125	89	91	85	90	138	122
184	138	92	90	126	89	92	86	90	138	122
185	138	93	91	127	90	93	87	91	138	123
186	138	93	92	128	91	93	88	92	138	123
187	138	94	92	129	92	94	89	93	138	124
188	138	95	93	130	92	95	90	94	138	125
189	138	95	94	130	93	95	91	94	138	126
190	138	96	94	131	94	96	92	95	138	126

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
191	138	97	95	132	95	97	93	95	138	127
192	138	97	96	133	95	97	94	96	138	128
193	138	98	96	133	96	98	95	97	138	129
194	138	98	97	134	96	98	96	97	138	130
195	138	99	97	134	97	99	97	98	138	131
196	138	100	98	135	98	99	97	98	138	132
197	138	100	98	136	98	100	98	99	138	133
198	138	100	99	137	99	100	99	99	138	135
199	138	101	100	137	99	101	100	100	138	137
200	138	101	100	138	100	101	100	100	138	138
201	138	102	100	138	100	102	101	101	138	138
202	138	102	101	138	101	102	102	101	138	138
203	138	103	101	138	101	103	102	102	138	138
204	138	103	102	138	102	103	103	102	138	138
205	138	104	102	138	103	104	104	103	138	138
206	138	104	103	138	104	104	104	103	138	138
207	138	105	103	138	104	105	105	103	138	138
208	138	105	104	138	105	105	105	104	138	138
209	138	106	105	138	105	106	106	105	138	138
210	138	106	105	138	106	106	107	106	138	138
211	138	107	106	138	106	107	108	106	138	138
212	138	107	107	138	107	108	108	107	138	138
213	138	108	107	138	108	108	109	108	138	138
214	138	109	108	138	108	109	110	108	138	138
215	138	109	109	138	109	110	110	109	138	138
216	138	110	110	138	110	110	111	110	138	138
217	138	110	110	138	110	111	112	110	138	138
218	138	111	111	138	111	111	112	111	138	138
219	138	112	111	138	111	112	113	111	138	138
220	138	112	111	138	112	112	114	112	138	138
221	138	113	112	138	112	113	114	112	138	138
222	138	113	112	138	113	113	115	113	138	138
223	138	114	113	138	113	114	116	113	138	138
224	138	114	114	138	114	115	117	114	138	138
225	138	115	115	138	115	115	117	114	138	138
226	138	115	115	138	115	116	118	115	138	138
227	138	116	116	138	116	116	119	115	138	138
228	138	116	116	138	117	117	119	116	138	138
229	138	117	117	138	117	117	120	116	138	138
230	138	117	117	138	118	118	120	117	138	138
231	138	118	118	138	118	118	121	117	138	138
232	138	118	118	138	119	119	122	118	138	138
233	138	119	119	138	120	119	122	119	138	138
234	138	119	119	138	120	120	123	120	138	138
235	138	119	120	138	121	120	123	120	138	138
236	138	120	120	138	121	120	124	120	138	138
237	138	120	121	138	122	121	124	121	138	138
238	138	120	121	138	122	121	125	121	138	138
239	138	121	122	138	123	122	125	122	138	138
240	138	121	122	138	123	122	126	122	138	138
241	138	121	123	138	123	123	126	123	138	138
242	138	122	123	138	124	123	127	123	138	138
243	138	122	123	138	124	124	128	123	138	138
244	138	123	124	138	125	124	129	124	138	138
245	138	123	124	138	125	125	129	124	138	138
246	138	124	125	138	126	126	130	125	138	138
247	138	124	125	138	126	127	130	125	138	138
248	138	125	126	138	127	127	131	126	138	138
249	138	125	126	138	128	128	132	126	138	138
250	138	126	127	138	129	129	133	127	138	138
251	138	127	128	138	130	130	134	127	138	138
252	138	128	128	138	130	132	135	128	138	138
253	138	129	129	138	131	133	136	128	138	138
254	138	129	130	138	133	134	137	129	138	138
255	138	130	131	138	135	135	137	130	138	138
256	138	130	132	138	137	136	138	131	138	138

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
257	138	131	134	138	138	136	138	133	138	138
258	138	131	135	138	139	137	139	135	138	138
259	138	132	136	138	139	137	139	137	138	138
260	138	133	137	138	139	138	139	139	138	138
261	138	134	138	138	139	138	139	139	138	138
262	138	135	139	138	139	139	139	139	138	138
263	138	137	139	138	139	139	139	139	138	138
264	138	139	139	138	139	139	139	139	138	138
265	138	139	139	138	139	139	139	139	138	138
266	138	139	139	138	139	139	139	139	138	138
267	138	139	139	138	139	139	139	139	138	138
268	138	139	139	138	139	139	139	139	138	138
269	138	139	139	138	139	139	139	139	138	138
270	138	139	139	138	139	139	139	139	138	138
271	138	139	139	138	139	139	139	139	138	138
272	138	139	139	138	139	139	139	139	138	138
273	138	139	139	138	139	139	139	139	138	138
274	138	139	139	138	139	139	139	139	138	138
275	138	139	139	138	139	139	139	139	138	138
276	138	139	139	138	139	139	139	139	138	138
277	138	139	139	138	139	139	139	139	138	138
278	138	139	139	138	139	139	139	139	138	138
279	138	139	139	138	139	139	139	139	138	138
280	138	139	139	138	139	139	139	139	138	138
281	138	139	139	138	139	139	139	139	138	138
282	138	139	139	138	139	139	139	139	138	138
283	138	139	139	138	139	139	139	139	138	138
284	138	139	139	138	139	139	139	139	138	138
285	138	139	139	138	139	139	139	139	138	138
286	138	139	139	138	139	139	139	139	138	138
287	138	139	139	138	139	139	139	139	138	138
288	138	139	139	138	139	139	139	139	138	138
289	138	139	139	138	139	139	139	139	138	138
290	138	139	139	138	139	139	139	139	138	138
291	138	139	139	138	139	139	139	139	138	138
292	138	139	139	138	139	139	139	139	138	138
293	138	139	139	138	139	139	139	139	138	138
294	138	139	139	138	139	139	139	139	138	138
295	138	139	139	138	139	139	139	139	138	138
296	138	139	139	138	139	139	139	139	138	138
297	138	139	139	138	139	139	139	139	138	138
298	138	139	139	138	139	139	139	139	138	138
299	138	139	139	138	139	139	139	139	138	138
300	138	139	139	138	139	139	139	139	138	138
301	138	139	139	138	139	139	139	139	138	138
302	138	139	139	138	139	139	139	139	138	138
303	138	139	139	138	139	139	139	139	138	138
304	138	139	139	138	139	139	139	139	138	138
305	138	139	139	138	139	139	139	139	138	138
306	138	139	139	138	139	139	139	139	138	138
307	138	139	139	138	139	139	139	139	138	138
308	138	139	139	138	139	139	139	139	138	138
309	138	139	139	138	139	139	139	139	138	138
310	138	139	139	138	139	139	139	139	138	138
311	138	139	139	138	139	139	139	139	138	138
312	138	139	139	138	139	139	139	139	138	138
313	138	139	139	138	139	139	139	139	138	138
314	138	139	139	138	139	139	139	139	138	138
315	138	139	139	138	139	139	139	139	138	138
316	138	139	139	138	139	139	139	139	138	138
317	138	139	139	138	139	139	139	139	138	138
318	138	139	139	138	139	139	139	139	138	138
319	138	139	139	138	139	139	139	139	138	138
320	138	139	139	138	139	139	139	139	138	138

APPENDIX E

ASVAB 6/7 COMPOSITE CONVERSION TABLE

RAW	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
0	53	53	53	53	53	53	53	53	53	53
1	53	53	53	53	53	53	53	53	53	53
2	53	53	53	53	53	53	53	53	53	53
3	53	53	53	53	53	53	53	53	53	53
4	53	53	53	53	53	53	53	53	53	53
5	53	53	53	53	53	53	53	53	53	53
6	53	53	53	53	53	53	53	53	53	53
7	53	53	53	53	53	53	53	53	53	53
8	53	53	53	53	53	53	53	53	53	53
9	53	53	53	53	53	53	53	53	53	53
10	53	53	53	53	53	53	53	53	53	53
11	55	53	53	53	53	53	53	53	53	53
12	60	53	53	53	53	53	53	53	53	54
13	61	53	53	53	53	53	53	53	55	58
14	64	53	53	53	53	53	53	53	58	60
15	65	53	53	53	53	53	53	53	60	62
16	67	53	53	53	53	53	53	53	62	64
17	69	53	53	53	53	53	53	53	64	66
18	71	58	53	53	53	53	53	53	66	69
19	73	61	53	53	53	53	53	53	68	72
20	75	62	53	53	53	53	53	53	71	75
21	77	64	53	53	53	53	53	53	73	76
22	78	65	53	53	53	53	53	53	76	78
23	80	66	53	53	53	53	53	53	78	80
24	82	68	53	53	53	54	53	53	80	82
25	85	70	53	53	53	55	53	53	82	84
26	87	72	54	53	53	57	53	53	84	87
27	89	73	55	53	53	59	53	53	88	89
28	92	75	56	54	55	61	53	54	90	91
29	94	76	56	55	57	63	53	57	93	93
30	96	78	57	57	59	64	53	58	95	95
31	98	79	60	59	60	65	53	60	98	96
32	100	81	61	61	61	66	53	61	100	98
33	102	82	62	61	62	67	53	62	102	100
34	104	83	64	62	63	69	53	64	105	101
35	106	85	65	63	64	70	53	65	108	102
36	108	87	66	64	65	72	53	66	110	104
37	109	89	68	65	66	73	53	67	112	105
38	111	90	69	67	67	75	53	69	114	107
39	112	92	70	68	68	76	55	71	115	108
40	113	93	72	70	70	77	55	72	118	109
41	115	94	73	71	71	78	57	73	121	111
42	116	96	74	73	72	80	57	74	123	112
43	117	97	75	75	73	81	59	76	125	113
44	119	98	76	77	75	82	61	77	126	114
45	121	99	77	78	76	84	62	78	127	115
46	123	101	78	79	77	85	63	79	130	116
47	124	102	80	81	78	87	64	81	132	118
48	126	103	81	82	79	89	65	82	132	119
49	131	104	82	84	80	91	66	83	133	120
50	135	106	83	86	81	92	66	84	133	121
51	135	107	85	88	81	94	68	86	134	122
52	135	108	86	90	82	95	70	88	135	123
53	135	109	87	92	84	96	70	89	135	125
54	135	110	89	94	85	98	72	91	135	126
55	135	111	90	96	86	99	74	92	135	128
56	135	112	91	98	88	100	75	92	135	130
57	135	113	92	99	89	101	76	95	135	131

RAW	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
58	135	114	93	101	90	102	77	96	135	135
59	135	115	95	102	91	103	78	98	135	135
60	135	116	96	103	92	105	79	99	135	135
61	135	117	97	105	93	106	82	100	135	135
62	135	118	98	107	94	107	82	101	135	135
63	135	119	99	108	95	108	83	102	135	135
64	135	120	100	109	96	109	85	103	135	135
65	135	122	101	11C	97	110	86	105	135	135
66	135	123	102	111	98	111	88	106	135	135
67	135	124	103	113	99	112	90	107	135	135
68	135	125	104	114	100	112	91	109	135	135
69	135	126	105	115	101	113	93	110	135	135
70	135	126	106	117	102	115	94	111	135	135
71	135	127	108	117	103	116	95	111	135	135
72	135	128	109	119	104	117	96	112	135	135
73	135	128	110	120	105	118	98	113	135	135
74	135	130	111	122	106	119	99	114	135	135
75	135	131	112	123	107	120	100	115	135	135
76	135	131	112	124	108	121	102	116	135	135
77	135	132	113	125	109	122	104	117	135	135
78	135	135	114	125	110	123	105	118	135	135
79	135	135	115	126	111	124	106	119	135	135
80	135	135	116	127	112	125	108	120	135	135
81	135	135	116	128	113	126	109	121	135	135
82	135	135	117	129	113	127	110	122	135	135
83	135	135	118	130	114	128	111	122	135	135
84	135	135	119	130	115	130	112	123	135	135
85	135	135	120	131	116	132	113	124	135	135
86	135	135	121	132	117	132	114	125	135	135
87	135	135	122	132	118	135	115	125	135	135
88	135	135	123	132	119	135	117	126	135	135
89	135	135	123	133	120	135	118	127	135	135
90	135	135	124	135	121	135	120	127	135	135
91	135	135	125	135	121	135	121	128	135	135
92	135	135	126	135	122	135	122	129	135	135
93	135	135	126	135	123	135	123	130	135	135
94	135	135	127	135	124	135	124	130	135	135
95	135	135	127	135	125	135	125	131	135	135
96	135	135	128	135	126	135	126	132	135	135
97	135	135	129	135	126	135	127	135	135	135
98	135	135	129	135	127	135	127	135	135	135
99	135	135	132	135	128	135	128	135	135	135
100	135	135	132	135	129	135	128	135	135	135
101	135	135	133	135	130	135	128	135	135	135
102	135	135	134	135	131	135	129	135	135	135
103	135	135	135	135	132	135	129	135	135	135
104	135	135	135	135	132	135	129	135	135	135
105	135	135	135	135	135	135	130	135	135	135
106	135	135	135	135	135	135	130	135	135	135
107	135	135	135	135	135	135	130	135	135	135
108	135	135	135	135	135	135	132	135	135	135
109	135	135	135	135	135	135	135	135	135	135
110	135	135	135	135	135	135	135	135	135	135

APPENDIX F

ASVAB 8/9/10 COMPOSITE CONVERSION TABLE

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
1	40	40	40	40	40	40	40	40	40	40
2	40	40	40	40	40	40	40	40	40	40
3	40	40	40	40	40	40	40	40	40	40
4	40	40	40	40	40	40	40	40	40	40
5	40	40	40	40	40	40	40	40	40	40
6	40	40	40	40	40	40	40	40	40	40
7	40	40	40	40	40	40	40	40	40	40
8	40	40	40	40	40	40	40	40	40	40
9	40	40	40	40	40	40	40	40	40	40
10	40	40	40	40	40	40	40	40	40	40
11	40	40	40	40	40	40	40	40	40	40
12	40	40	40	40	40	40	40	40	40	40
13	40	40	40	40	40	40	40	40	40	40
14	40	40	40	40	40	40	40	40	40	40
15	40	40	40	40	40	40	40	40	40	40
16	40	40	40	40	40	40	40	40	40	40
17	40	40	40	40	40	40	40	40	40	40
18	40	40	40	40	40	40	40	40	40	40
19	40	40	40	40	40	40	40	40	40	40
20	40	40	40	40	40	40	40	40	40	40
21	40	40	40	40	40	40	40	40	40	40
22	40	40	40	40	40	40	40	40	40	40
23	40	40	40	40	40	40	40	40	40	40
24	40	40	40	40	40	40	40	40	40	40
25	40	40	40	40	40	40	40	40	40	40
26	40	40	40	40	40	40	40	40	40	40
27	40	40	40	40	40	40	40	40	40	40
28	40	40	40	40	40	40	40	40	40	40
29	40	40	40	40	40	40	40	40	40	40
30	40	40	40	40	40	40	40	40	40	40
31	40	40	40	40	40	40	40	40	40	40
32	40	40	40	40	40	40	40	40	40	40
33	40	40	40	40	40	40	40	40	40	40
34	40	40	40	40	40	40	40	40	40	40
35	40	40	40	40	40	40	40	40	40	40
36	40	40	40	40	40	40	40	40	40	40
37	40	40	40	40	40	40	40	40	40	40
38	40	40	40	40	40	40	40	40	40	40
39	40	40	40	40	40	40	40	40	40	40
40	40	40	40	40	40	40	40	40	40	40
41	40	40	40	40	40	40	40	40	40	40
42	40	40	40	40	40	40	40	40	40	40
43	40	40	40	40	40	40	40	40	40	40
44	40	40	40	40	40	40	40	40	40	40
45	40	40	40	40	40	40	40	40	40	40
46	40	40	40	40	40	40	40	40	40	40
47	40	40	40	40	40	40	40	40	40	40
48	40	40	40	40	40	40	40	40	40	40
49	40	40	40	40	40	40	40	40	40	40
50	40	40	40	40	40	40	40	40	40	40
51	40	40	40	40	40	40	40	40	40	40
52	41	40	40	40	40	40	40	40	40	40
53	41	40	40	40	40	40	40	40	40	40
54	41	40	40	40	40	40	40	40	40	40
55	41	40	40	40	40	40	40	40	40	40
56	41	40	40	40	40	40	40	40	40	40
57	43	40	40	40	40	40	40	40	40	40
58	46	40	40	40	40	40	40	40	40	40

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
59	50	40	40	40	40	40	40	40	40	40
60	52	40	40	40	40	40	40	40	40	40
61	54	40	40	40	40	40	40	40	40	40
62	56	40	40	40	40	40	40	40	40	40
63	57	40	40	40	40	40	40	40	40	40
64	59	40	40	40	40	40	40	40	40	40
65	61	40	40	40	40	40	40	40	40	40
66	62	40	40	40	40	40	40	40	40	40
67	63	40	40	40	40	40	40	40	40	40
68	64	40	40	40	40	40	40	40	40	40
69	65	40	40	40	40	40	40	40	40	40
70	66	40	40	40	40	40	40	40	40	40
71	67	40	40	40	40	40	40	40	40	40
72	68	40	40	41	40	40	40	40	40	40
73	70	40	40	41	40	40	40	40	40	40
74	71	40	40	41	40	40	40	40	40	40
75	72	40	40	41	40	40	40	40	40	40
76	73	40	40	41	40	40	40	40	40	40
77	74	40	40	41	40	40	40	40	40	40
78	75	40	40	41	40	40	40	40	40	40
79	76	40	40	43	40	40	40	40	40	40
80	77	40	40	45	40	40	40	40	40	40
81	78	40	40	47	40	40	40	40	40	40
82	79	40	40	48	40	40	40	40	40	40
83	80	40	40	50	40	40	40	40	40	40
84	81	40	40	51	40	40	40	40	40	40
85	82	40	40	52	40	40	40	40	40	40
86	83	40	40	53	40	40	40	40	40	40
87	84	40	40	54	40	40	40	40	40	40
88	85	40	40	55	40	40	40	40	40	40
89	86	40	40	56	40	40	40	40	40	40
90	87	40	40	56	40	40	40	40	40	40
91	89	40	40	57	40	40	40	40	40	40
92	90	40	40	57	40	40	40	40	40	40
93	91	40	40	58	40	40	40	40	40	40
94	92	40	40	58	40	40	40	40	40	40
95	93	40	40	59	40	40	40	40	40	40
96	94	40	40	59	40	40	40	40	40	40
97	95	40	40	60	40	40	40	40	40	40
98	96	40	40	61	40	41	40	40	40	40
99	97	40	40	61	40	41	40	40	41	40
100	98	40	40	62	40	41	40	40	41	40
101	99	40	40	62	41	41	40	40	41	40
102	100	40	40	63	41	41	40	40	41	40
103	102	40	40	63	41	41	40	40	41	40
104	103	40	40	64	41	41	40	40	41	40
105	104	40	40	64	41	41	40	40	41	40
106	105	40	40	65	41	41	40	40	41	41
107	105	40	40	65	41	41	41	40	41	41
108	106	41	40	66	41	41	41	40	41	41
109	107	41	40	66	41	41	41	40	41	41
110	109	41	41	67	41	43	41	40	41	41
111	110	41	41	67	41	45	41	40	41	41
112	111	41	41	68	41	47	41	40	41	41
113	112	41	41	69	41	48	41	40	41	41
114	113	41	41	70	41	49	41	41	42	41
115	114	41	41	70	41	50	41	41	45	41
116	115	41	41	71	41	51	41	41	47	41
117	117	41	41	71	43	52	41	41	49	41
118	118	41	41	72	45	53	41	41	50	41
119	119	41	41	73	48	54	41	41	52	41
120	120	41	41	73	50	54	41	41	53	43
121	121	41	41	74	51	55	41	41	54	45
122	122	41	41	74	52	55	41	41	54	47
123	123	44	41	75	53	56	43	41	55	49
124	125	47	41	75	54	56	45	41	56	50

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
125	126	50	43	76	54	57	48	41	56	51
126	128	51	45	76	55	57	50	41	57	52
127	130	53	48	77	56	58	52	41	57	53
128	135	54	50	78	56	58	53	44	58	55
129	143	54	51	78	57	59	54	47	58	56
130	155	55	52	79	57	59	55	50	59	56
131	155	56	53	80	58	59	56	51	59	57
132	155	57	54	80	58	60	57	53	60	57
133	155	57	55	81	59	60	57	54	60	58
134	155	58	56	82	59	61	58	55	61	59
135	155	59	57	83	60	61	59	56	61	60
136	155	59	57	84	61	62	59	56	62	61
137	155	60	58	85	61	62	60	57	62	61
138	155	61	59	85	62	63	61	57	63	62
139	155	61	59	86	62	63	61	58	63	62
140	155	62	60	87	63	64	62	59	64	63
141	155	62	61	88	63	64	62	59	64	64
142	155	63	62	89	64	65	62	60	65	64
143	155	64	62	90	64	65	63	61	65	65
144	155	65	63	91	65	65	64	62	66	65
145	155	65	64	92	65	66	65	62	66	66
146	155	66	65	92	66	66	65	63	67	66
147	155	67	65	93	66	66	66	64	67	67
148	155	68	66	94	67	67	66	65	68	68
149	155	68	67	95	68	67	67	65	68	69
150	155	69	68	95	68	68	68	66	69	69
151	155	70	69	96	69	68	68	66	69	70
152	155	70	70	97	69	69	69	67	70	70
153	155	71	70	98	70	69	69	68	70	71
154	155	71	71	99	70	70	70	69	71	71
155	155	72	72	100	71	70	70	70	71	72
156	155	73	73	100	71	71	71	70	72	73
157	155	74	73	101	72	71	71	71	72	73
158	155	74	74	102	73	72	72	72	73	74
159	155	75	74	103	73	72	73	73	73	74
160	155	75	75	104	74	73	74	74	74	75
161	155	76	75	105	74	73	74	74	74	75
162	155	76	76	106	75	74	75	75	75	76
163	155	78	76	107	75	74	75	75	75	76
164	155	78	77	108	76	75	76	76	76	77
165	155	79	78	109	76	75	76	76	76	77
166	155	79	78	110	77	75	77	77	76	78
167	155	80	79	111	77	76	77	78	77	79
168	155	80	80	112	78	76	78	78	77	79
169	155	81	80	113	79	76	79	79	78	80
170	155	81	81	113	79	77	79	79	78	80
171	155	82	81	114	80	77	80	80	79	81
172	155	82	82	115	80	78	80	81	79	81
173	155	82	83	116	81	78	81	82	80	82
174	155	83	84	117	81	79	82	83	81	83
175	155	84	85	118	82	79	82	84	81	84
176	155	84	85	119	83	80	83	84	82	84
177	155	85	86	120	84	80	84	85	82	85
178	155	86	87	121	84	81	85	86	83	85
179	155	86	88	121	85	82	85	87	84	86
180	155	87	88	122	85	82	86	87	85	87
181	155	88	89	123	86	83	87	88	85	88
182	155	88	90	123	87	84	87	89	86	88
183	155	89	91	124	87	84	88	90	86	89
184	155	90	91	125	88	85	89	90	87	90
185	155	90	92	126	89	85	89	91	88	91
186	155	91	92	127	90	86	90	92	89	91
187	155	91	93	127	90	87	91	92	89	92
188	155	92	93	128	91	88	92	93	90	92
189	155	92	94	129	91	88	92	94	90	93
190	155	93	94	130	92	89	93	94	91	93

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
191	155	93	95	131	92	90	93	95	91	94
192	155	94	95	133	93	91	94	95	92	94
193	155	94	96	135	93	92	94	96	92	95
194	155	95	97	137	94	92	95	96	93	95
195	155	96	98	138	94	93	95	97	93	96
196	155	96	98	141	95	93	96	98	94	96
197	155	97	99	144	95	94	96	98	95	97
198	155	98	99	147	96	94	97	99	95	98
199	155	98	100	149	96	95	98	99	96	98
200	155	99	100	152	97	96	98	100	96	99
201	155	99	101	155	98	96	99	100	97	99
202	155	100	101	155	98	97	100	101	98	100
203	155	100	102	155	99	98	100	102	98	100
204	155	101	102	155	99	99	101	102	99	101
205	155	101	103	155	100	99	101	103	99	101
206	155	102	103	155	100	100	102	103	100	102
207	155	102	104	155	101	100	102	104	100	103
208	155	103	104	155	102	101	103	104	101	103
209	155	103	105	155	103	102	103	105	102	104
210	155	104	105	155	103	103	104	105	103	104
211	155	104	106	155	104	103	104	106	103	105
212	155	105	106	155	104	104	105	107	104	105
213	155	105	107	155	105	104	105	107	104	106
214	155	106	107	155	105	105	106	108	105	107
215	155	107	108	155	106	106	107	108	106	107
216	155	107	109	155	107	107	108	109	107	108
217	155	108	109	155	107	107	109	109	107	109
218	155	109	110	155	108	108	109	110	108	109
219	155	109	110	155	109	109	110	111	109	110
220	155	110	111	155	110	110	111	111	110	111
221	155	111	111	155	111	111	111	112	111	112
222	155	112	112	155	112	112	112	112	112	112
223	155	113	112	155	112	113	113	113	113	113
224	155	113	113	155	113	113	113	113	113	113
225	155	114	113	155	114	114	114	114	114	114
226	155	114	114	155	114	115	114	114	115	114
227	155	115	114	155	115	116	115	115	116	115
228	155	116	115	155	116	117	116	115	116	115
229	155	116	115	155	116	118	116	116	117	116
230	155	117	116	155	117	118	117	116	118	116
231	155	117	116	155	118	119	117	117	119	117
232	155	118	117	155	118	120	118	117	120	117
233	155	118	117	155	119	121	119	118	121	118
234	155	119	118	155	120	121	120	118	121	118
235	155	119	118	155	121	122	121	119	122	119
236	155	120	118	155	121	122	121	119	123	119
237	155	120	119	155	122	123	122	120	123	120
238	155	121	119	155	123	123	122	120	124	120
239	155	121	120	155	123	124	123	121	124	121
240	155	122	120	155	124	124	123	121	125	121
241	155	122	121	155	124	125	124	121	125	122
242	155	122	121	155	125	126	124	122	126	122
243	155	123	121	155	126	126	125	122	127	123
244	155	123	122	155	126	127	126	123	127	123
245	155	123	122	155	127	127	126	123	128	123
246	155	124	123	155	127	128	127	123	129	124
247	155	124	123	155	128	129	127	124	130	124
248	155	125	123	155	129	130	128	124	131	125
249	155	125	124	155	130	131	128	125	133	125
250	155	126	124	155	131	133	129	125	135	126
251	155	126	124	155	133	134	130	126	137	126
252	155	127	125	155	135	135	130	126	141	127
253	155	127	125	155	137	136	131	127	144	127
254	155	128	126	155	138	137	133	127	147	128
255	155	129	126	155	141	138	135	128	150	129
256	155	130	127	155	143	141	136	128	152	130

SSS	GT	GM	EL	CL	MM	SC	CO	FA	OF	ST
257	155	131	127	155	145	144	137	129	153	131
258	155	132	128	155	148	147	138	130	155	132
259	155	133	129	155	150	149	139	130	155	133
260	155	135	13C	155	151	150	141	131	155	135
261	155	137	131	155	153	151	143	133	155	136
262	155	138	132	155	155	152	144	134	155	138
263	155	141	133	155	155	153	145	135	155	141
264	155	143	134	155	155	154	147	136	155	145
265	155	145	135	155	155	155	149	137	155	147
266	155	147	137	155	155	155	150	138	155	150
267	155	149	141	155	155	155	151	139	155	153
268	155	151	145	155	155	155	152	141	155	155
269	155	153	147	155	155	155	152	142	155	155
270	155	155	15C	155	155	155	153	143	155	155
271	155	155	153	155	155	155	154	145	155	155
272	155	155	155	155	155	155	154	147	155	155
273	155	155	155	155	155	155	155	149	155	155
274	155	155	155	155	155	155	155	150	155	155
275	155	155	155	155	155	155	155	151	155	155
276	155	155	155	155	155	155	155	152	155	155
277	155	155	155	155	155	155	155	153	155	155
278	155	155	155	155	155	155	155	153	155	155
279	155	155	155	155	155	155	155	154	155	155
280	155	155	155	155	155	155	155	155	155	155
281	155	155	155	155	155	155	155	155	155	155

APPENDIX G

ASVAB PROGRAM OUTPUT FILES

```
SSAN                            : 272598843
CATE OF BIRTH(YY/MM/DD)         : 64/ 4/20
SEX                             : 1 - MALE
RACE                            : 2 - BLACK
CATE OF ENTRY(YY/MM/DD)         : 82/ 2/16
HIGHEST YEAR OF EDUCATION       : 1 YEAR OF HIGH SCHOOL -   3
ENTRY PAY GRADE                 : E1  1
MOS                             : 11B10
35 TEST FORM                    : ASVAB 5
AFQT PERCENTILE                 : 47
SEPARATION PROGRAM DESIGNATOR: AKL
DATE OF SEPARATION(YY/MM/DD)  : 83/ 2/27

                 SUBTEST RAW SCORES
 GI   NO   AD   WK   AR   SP   MK   EI   MC   GS   SI   AI
 15   20    3   12    6   17   20   10    9    5   13   12

              SUBTEST STANDARD SCORES
 GI   NO   AD   WK   AR   SP   MK   EI   MC   GS   SI   AI
 66   39   21   39   36   59   67   34   46   36   48   50

       ARMY  CLASSIFICATION BATTERY COMPOSITES
CO    FA    MM    GM    CL    GT    EL    SC    ST    OF
71   102    99    79    49    71    76    89    94   116
```

```
SSAN                           : 016532241
DATE OF BIRTH(YY/MM/DD)        : 63/11/25
SEX                            : 1 - MALE
RACE                           : 1 - WHITE
DATE OF ENTRY(YY/MM/DD)        : 81/ 9/20
HIGHEST YEAR OF EDUCATION      : HIGH SCHOOL DIPLOMA -  6
ENTRY PAY GRADE                : E1  1
MOS                            : 11B10
36 TEST FORM                   : ASVAB 6
AFQT PERCENTILE                : 87
SEPARATION PROGRAM DESIGNATOR: AAA
DATE OF SEPARATION(YY/MM/DD) : 82/11/ 9
```

SUBTEST RAW SCORES

GI	NO	AD	WK	AR	SP	MK	EI	MC	GS	SI	AI	CM	CA	CE	CC
13	40	25	22	15	16	12	23	10	9	17	14	8	12	13	22

SUBTEST STANDARD SCORES

GI	NO	AC	WK	AR	SP	MK	EI	MC	GS	SI	AI	CM	CA	CE	CC
60	59	77	53	55	57	51	56	48	45	58	55	44	58	60	63

ARMY CLASSIFICATION BATTERY COMPOSITES

CO	FA	MM	GM	CL	GT	EL	SC	ST	OF
125	115	106	103	122	109	114	108	104	115

```
SSAN                            : 213865527
DATE OF BIRTH(YY/MM/DD)         : 63/ 5/21
SEX                             : 2 - FEMALE
RACE                            : 3 - OTHER
DATE OF ENTRY(YY/MM/DD)         : 82/10/ 7
HIGHEST YEAR OF EDUCATION       : 1 YEAR OF COLLEGE -  7
ENTRY PAY GRADE                 : E2  2
MOS                             : 54E10
40 TEST FORM                    : ASVAB 10
AFQT PERCENTILE                 : 92
SEPARATION PROGRAM DESIGNATOR: AAA
DATE OF SEPARATICN(YY/MM/DD) : 83/10/15
```

SUBTEST RAW SCORES

GS	AR	WK	PC	NO	CS	AS	MK	MC	EI	VE
24	22	30	14	45	79	25	20	24	19	48

SUBTEST STANDARD SCORES

GS	AR	WK	PC	NO	CS	AS	MK	MC	EI	VE
65	56	56	60	59	72	65	63	65	65	61

ARMY CLASSIFICATICN BATTERY COMPOSITES

CO	FA	MM	GM	CL	GT	EL	SC	ST	OF
138	128	138	132	133	117	124	144	128	135

```
ERRCR IN RECORD NUMBER:    4 WITH SSAN: 235442321
ERROR: TEST FORM INCORRECT, COLUMNS 33-34


SSAN                          : 235442321
DATE OF BIRTH(YY/MM/DD)       : 63/ 5/23
SEX                           : 1 - MALE
RACE                          : 1 - WHITE
DATE OF ENTRY(YY/MM/DD)       : 82/11/13
HIGhEST YEAR OF EDUCATION     : HIGH SCHOOL DIPLOMA -  6
ENTRY PAY GRADE               : E1  1
MOS                           : 11B10
41 TEST FORM                  : ASVAB 10
AFCT PERCENTILE               : 85
SEPARATION PROGRAM DESIGNATOR: AAA
DATE OF SEPARATION(YY/MM/DD)  : 83/11/10

             SLBTEST RAW SCORES
   GS   AR   WK   PC   NO   CS   AS   MK   MC   EI   VE
   23   21   29   13   44   72   20   20   21   18   23
```

```
ERROR IN RECORD NUMBER:    5 WITH SSAN: 015342237
ERROR: CC INCORRECT, COLUMNS 67-68


SSAN                            : 015342237
DATE OF BIRTH(YY/MM/DD)         : 64/ 8/ 9
SEX                             : 1 - MALE
RACE                            : 1 - WHITE
CATE OF ENTRY(YY/MM/DD)         : 81/ 9/25
HIGHEST YEAR OF EDUCATION       : HIGH SCHOOL DIPLOMA -  6
ENTRY PAY GRADE                 : E1  1
MOS                             : 11B10
37 TEST FORM                    : ASVAB 7
AFQT PERCENTILE                 : 75
SEPARATION PROGRAM DESIGNATOR: AKL
DATE OF SEPARATICN(YY/MM/DD)  : 84/ 9/15

               SUBTEST RAW SCORES
  GI  NO  AD  WK  AR  SP  MK  EI  MC  GS  SI  AI  CM  CA  CE  CC
  15  39  25  21  15  16  12  23  11  12  17  13  10  12  13  32
```

SUBTEST RAW SCORES

SSAN	GI	NO	AD	WK	AR	SP	MK	EI	MC	GS	SI	AI	CM	CA	CE	CC	PC	CS	AS	VE
272598843	15	20	3	12	6	17	20	10	9	5	13	12								
016532241	13	40	25	22	15	16	12	23	10	9	17	14	8	12	13	22				
213865527		24		22	30		14	45	79	25							20	24	19	48

SUBTEST STANDARD SCORES

SSAN	GI	NO	AD	WK	AR	SP	MK	EI	MC	GS	SI	AI	CM	CA	CE	CC	PC	CS	AS	VE
272598843	66	39	21	39	36	59	67	34	46	36	48	50								
016532241	60	59	77	53	55	57	51	56	48	45	58	55	44	58	60	63				
213865527		59		56	56		63	65	65	65							60	72	65	61

ARMY CLASSIFICATION BATTERY COMPOSITES

SSAN	AFQT	CO	FA	MM	GM	CL	GT	EL	SC	ST	OF
272598843	47	71	102	99	79	49	71	76	89	94	116
016532241	87	125	115	106	103	122	109	114	108	104	115
213865527	92	138	128	138	132	133	117	124	144	128	135

APPENDIX H

SPSS REGRESSION PROGRAM LISTING

```
DATA LIST        FIXED/ 1 SQT 11-15 AFQT 17-18
                 CO 20-22 FA 24-26 MM 28-30 GM 32-34
                 CL 36-38 GT  40-42 EL 44-46 SC 48-50
                 ST 52-54 OF 56-58
VAR LABELS       SQT    SKILL QUALIFICATION TEST/
                 AFQT   AFQT PERCENTILE/
                 CO     COMBAT/
                 FA     FIELD ARTILLERY/
                 MM     MOTOR MAINTENANCE/
                 GM     GENERAL MAINTENANCE/
                 CL     CLERICAL/
                 GT     GENERAL TECHNICAL/
                 EL     ELECTRONIC/
                 SC     SURVEILLANCE, COMMUNICATIONS/
                 ST     SKILLED TECHNICAL/
                 OF     OPERATOR AND FOOD HANDLER
NEW REGRESSION   VARIABLES = SQT,CO TO OF/
                 DEP = SQT/STEPWISE/
                 RESIDUALS/CASEWISE = ALL/
                 SCATTER = (*RESID,*PRED)/SAVE = RESID PRED/
READ INPUT DATA
427310253       19 58 121 113 123 120 116 104 121 111 109 120
520683135        5 48 108  97 114 114  89 104 107  99 109 110
527714668        8 70 105 103 112 113  95 117 113 100 107 108
539803606        3 66 106 102 109 107 109 109 105 111 112 114
553177646        6 26  98  91  91  94  89  89  87  94  94  93
560138625       17 44 114 112 107  99 106  94 101 108  92 101
560177925        5 28 101  98 102 105  78  95 103  84  99  93
573555886       11 70 114 109 112 109  98 117 110 104 113 116
455393568        5 25  96  84  88  82  81  89  93  81  96  71
039368355       14 95 118 125 115 118 122 124 125 128 126 105
092520162        6 21  90  65  91  89  84  87  85  81  64  82
096449837        3 60 106 111 107  99  99 112 112 101 111  88
121545078        3 21  91  98  82  83  88  75  96  81  91  80
255043799       15 35  82 110  80  83 114 111  92  87 102 108
426150571        9 12  93  66  80  75  70  60  85  70  69  66
441707290       14 23  85 109 106 101  96  82 100  87 105 114
257044264        3 21  77  78  67  64  84  87  78  75  76  66
260252757       14 75 109 105  98 108 113 113 101 111 113 100
265538874        6 15  82  71  90  87  82  78  89  73  78  71
455948594        7 18  82  91 105  99  92  85  99  77  82 112
508882657        5 77  95 125  97 113 123 126 116 116 128 100
050526224        8 25  85  92  85  87  84  80  90  84  93  88
220829366        2 41  83  79  73  78  82  98  81  91  87  71
251086870        6 53  82 106 109 109 101 115 106 100 105 110
423947244        7 44  91  84  81  65  96  96  81  91  75  80
565374369        7 38  82  95 107 103  81  98  99  92 100 108
235965486        2 31  90 111 105 104 101 108 104  87 107 102
562254590        6 38 106  92 102 110 102  98 105  98  98 102
END INPUT DATA
FINISH
```

APPENDIX I

SPSS REGRESSION MODEL PROGRAM LISTING

```
DATA LIST          FIXED/ 1 SSAN 1-9 SQT 11-15 AFQT 17-18
                   CO 20-22 FA 24-26 MM 28-30 GM 32-34
                   CL 36-38 GT 40-42 EL 44-46 SC 48-50
                   ST 52-54 OF 56-58
VAR LABELS         SSAN   SOCIAL SECURITY NUMBER/
                   SQT    SKILL QUALIFICATION TEST/
                   AFQT   AFQT PERCENTILE/
                   CO     COMBAT/
                   FA     FIELD ARTILLERY/
                   MM     MOTOR MAINTENANCE/
                   GM     GENERAL MAINTENANCE/
                   CL     CLERICAL/
                   GT     GENERAL TECHNICAL/
                   EL     ELECTRONIC/
                   SC     SURVEILLANCE, COMMUNICATIONS/
                   ST     SKILLED TECHNICAL/
                   OF     OPERATOR AND FOOD HANDLER
COMPUTE            SQT=3+(0.085*GTT)
WRITE CASES        (F9.0,3X,F5.0) SSAN,SQT
READ INPUT DATA
427310253          15 58 121 113 123 120 116 104 121 111 109 120
520683135          13 48 108  97 114 114  89 104 107  99 109 110
527714668          21 70 105 103 112 113  95 117 113 100 107 108
539803606          20 66 106 102 109 107 109 109 105 111 112 114
553177646           6 26  98  91  91  94  89  89  87  94  94  93
560138625          12 44 114 112 107  99 106  94 101 108  92 101
560177925           5 28 101  98 102 105.  78  95 103  84  99  93
573559886          23 70 114 109 112 109  98 117 110 104 113 116
455393568           5 25  96  84  88  82  81  89  93  31  96  71
039368355          28 95 118 125 115 118 122 124 125 128 126 105
092520162           6 21  90  65  91  89  84  87  85  81  64  82
.96449837          19 60 106 111 107  99  99 112 112 101 111  88
121545078           3 21  91  98  82  83  38  75  96  81  91  80
255043799           5 35  82 110  80  83 114 111  92  87 102 108
426150571           1 12  93  66  80  75  70  60  85  70  69  66
441707290           7 23  85 109 106 101  96  82 100  87 105 114
257044264           7 21  77  78  67  64  84  87  78  75  76  66
260252797          18 75 109 105  98 108 113 113 101 111 113 100
265538874           6 15  82  71  90  87  82  78  89  73  78  71
455948594           7 18  82  91 105  99  92  85  99  77  82 112
508882857          19 77  95 125  97 113 123 126 116 116 128 100
050526224          10 25  85  92  85  87  84  80  90  84  93  88
220829366          12 41  83  79  73  78  82  98  81  91  87  71
251086870          16 53  82 106 109 109 101 115 106 100 105 110
423947244          13 44  91  84  81  65  96  96  81  91  75  80
565374369          10 38  82  95 107 103  81  98  99  92 100 108
235965486           9 31  90 111 105 104 101 108 104  87 107 102
562254590          11 38 106  92 102 110 102  98 105  98  98 102
END INPUT DATA
FINISH
```

APPENDIX J

SPSS REGRESSION PROGRAM OUTPUT

```
VARIABLE LIST NUMBER  1.  LISTWISE DELETION OF MISSING DATA.
EQUATION NUMBER  1.
DEPENDENT VARIABLE..  SQT         SKILL QUALIFICATION TEST

BEGINNING BLOCK NUMBER  1.  METHOD:  STEPWISE

VARIABLE(S) ENTERED ON STEP NUMBER  1..   CL          CLERICAL

MULTIPLE R           0.47852        ANALYSIS OF VARIANCE
R SQUARE             0.22898                            DF      SUM OF SQUARES
ADJUSTED R SQUARE    0.19932        REGRESSION           1           136.86306
STANDARD ERROR       4.21011        RESIDUAL            26           460.85122

                                    F =       7.72145      SIGNIF F = 0.0100

---------------- VARIABLES IN THE EQUATION -----------------               ------
VARIABLE                 B        SE B        BETA           T  SIG T      VARIAB

CL                 0.16172     0.05820     0.47852       2.779 0.0100      CO
(CONSTANT)        -7.73595     5.61678                  -1.377 0.1802      FA
                                                                           MM
                                                                           GM
                                                                           GTT
                                                                           EL
                                                                           SC
                                                                           ST
                                                                           OF
1SPSS BATCH SYSTEM

DEPENDENT VARIABLE..  SQT         SKILL QUALIFICATION TEST

VARIABLE(S) ENTERED ON STEP NUMBER  2..   GT          GENERAL TECHNICAL

MULTIPLE R           0.59073        ANALYSIS OF VARIANCE
R SQUARE             0.34897                            DF      SUM OF SQUARES
ADJUSTED R SQUARE    0.29688        REGRESSION           2           208.58148
STANDARD ERROR       3.94529        RESIDUAL            25           389.13281
```

F = 6.70020 SIGNIF F = 0.0047

------------------ VARIABLES IN THE EQUATION ------------------

VARIABLE	B	SE B	BETA	T	SIG T
CL	0.29638	0.08313	0.87697	3.565	0.0015
GT	-0.15531	0.07235	-0.52797	-2.147	0.0417
(CONSTANT)	-5.34173	5.38036		-0.993	0.3303

VARIAB

CO
FA
MM
GM
EL
SC
ST
OF

APPENDIX K

SPSS REGRESSION MODEL PROGRAM OUTPUT

SSAN	SQT
427310080	12
520683008	12
527714560	13
539803392	12
553177600	11
560138496	11
560177920	11
573559808	13
455393536	11
039368352	14
092520160	10
096449824	13
121545072	9
255043792	12
426150400	8
441707264	10
257044256	10
260252784	13
265538864	10
455948544	10
508882688	14
050526224	10
220829360	11
251086864	13
423947008	11
565374208	11
235965472	12
562254336	11

APPENDIX L

ASVAB PROGRAM LISTING

```
C
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                      C
C     THIS PROGRAM READS IN THE SOLDIER DEMOGRAPHIC DATA, AND THE      C
C   ARMED SERVICES VOCATIONAL APPTITUDE BATTERY (ASVAB) SUBTEST        C
C   RAW SCORES.  THE RAW SCORES ARE CONVERTED INTO SUBTEST STANDARD    C
C   SCORES, AND THE ARMY CLASSIFICATION BATTERY (ACB) IS COMPUTED,     C
C   AND CONVERTED TO STANDARD SCORES.  THE OUTPUT CONSISTS OF THE      C
C   SOLDIER DEMOGRAPHIC DATA, THE ASVAB RAW SCORES, THE ASVAB SUB-     C
C   TEST STANDARD SCORES, AND THE ACB COMPOSITE SCORES.                C
C                                                                      C
C                                                                      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C------------------------------
C     VARIABLES AND CONSTANTS
C------------------------------
C
C     SSAN          SOCIAL SECURITY ACCOUNT NUMBER
C     DOBYY         YEAR OF BIRTH
C     DOBMM         MONTH OF BIRTH
C     DOBDD         DAY OF BIRTH
C     SEX           SEX
C     RACE          RACE
C     DCEYY         YEAR OF ENTRY
C     DOEMM         MONTH OF ENTRY
C     DOEDD         DAY OF ENTRY.
C     ED            HIGHEST YEAR OF EDUCATION.
C     GRADE         ENTRY PAY GRADE.
C     MOS           MILITARY OCCUPATION SPECIALITY.
C     TEST          TEST FORM.
C     AFQT          ARMED FORCES QUALIFICATION PERCENTILE.
C     GI            GENERAL INFORMATION.
C     NO            NUMERICAL OPERATIONS.
C     AD            ATTENTION TO DETAIL.
C     WK            WORD KNOWLEDGE.
C     AR            ARITHMETIC REASONING.
C     SP            SPACE PERCEPTION.
C     MK            MATH KNOWLEDGE.
C     EI            ELECTRONIC INFORMATION.
C     MC            MECHANICAL COMPREHENSION.
C     GS            GENERAL SCIENCE.
C     SI            SHOP INFORMATION.
C     AI            AUTOMOTIVE INFORMATION.
```

```
C       CM              MAINTENANCE.
C       CA              ATTENTIVENESS.
C       CE              ELECTRONICS.
C       CC              COMBAT.
C       PC              PARAGRAPH COMPREHENSION.
C       CS              CODING SPEED.
C       AS              AUTOMOTIVE/SHOP.
C       VE              VERBAL.
C       CO              COMBAT.
C       FA              FIELD ARTILLERY.
C       EL              ELECTRCNICS.
C       OF              OPERATORS AND FOOD HANDLERS.
C       GM              GENERAL MAINTENANCE.
C       MM              MOTOR MAINTENANCE.
C       CL              CLERICAL.
C       ST              SKILLED TECHNICAL.
C       GT              GENERAL TECHNICAL.
C       SC              SURVEILLANCE/COMMUNICATION.
C       A5              ASVAB 5 CONVERSION TABLE.
C       A67             ASVAB 6/7 CONVERSION TABLE.
C       A8910           ASVAB 8/9/10 CONVERSION TABLE.
C       SSS             VARIABLE SUFFIX: SUBTEST STANDARC SCORE.
C       R               VARIABLE SUFFIX: VARIABLE AS A REAL NUMBER.
C       END             TRAILER VARIABLE
C
C---------------------
C     FILE DEFINITIONS
C---------------------
C
C       FILE 01         INPUT, ASVAB 5 COMPOSITE CONVERSION TABLE.
C       FILE 02         INPUT, ASVAB 6/7 COMPOSITE CONVERSION TABLE.
C       FILE 03         INPUT, ASVAB 8/9/10 COMPOSITE CONVERSION TABLE.
C       FILE 04         INPUT, THE DEMOGRAPHIC DATA AND THE RAW SCORES.
C       FILE 09         OUTPUT, THE SOLDIER FILE.
C       FILE 10         OUTPUT, THE ASVAB STANDARD SCORES.
C       FILE 11         OUTPUT, THE ASVAB RAW SCORES.
C       FILE 12         OUTPUT, THE ACB COMPOSITE SCORES.
C
C----------------------
C       DECLARE VARIABLES
C----------------------
C
        INTEGER DOBYY,DOBMM,DOBDD,SEX,RACE,DOEYY,DOEMM,DOEDD,ED,
      *GRADE,TEST,GS,AR,WK,PC,NO,CS,AS,MK,MC,EI,VE,DES,ETSYY,ETSMM,
      *ETSDD,GI,AD,SP,SI,AI,CM,CA,CE,CC,RAW1,RAW2,RAW3,RAW4,RAW5,RAW6,
      *RAW7,RAW8,RAW9,RAW10,RAW11,RAW12,RAW13,RAW14,RAW15,RAW16,I,GSSSS,
      *ARSSS,WKSSS,PCSSS,NCSSS,CSSSS,ASSSS,MKSSS,MCSSS,EISSS,VESSS,GISSS,
      *ADSSS,SPSSS,SISSS,AISSS,CMSSS,CASSS,CESSS,CCSSS,CO,FA,MM,GM,CL,GT
```

```
C
      INTEGER EL,SC,ST,OF,AFQT,MOS1,MOS2,J,K,A5(320,10),A67(111,10),
     *A8910(281,10),SSAN1,SSAN2,SSAN3,END,ERROR
C
      REAL CONVT,GIR,ADR,SPR,SIR,AIR,CMR,CAR,CER,CCR,GSR,ARR,WKR,PCR,
     *NOR,CSR,ASR,MKR,MCR,EIR,VER
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C                                                                     C
C     ALGORITHM ASVAB IS THE MASTER ALGORITHM.  IT CONTAINS 7 SUB     C
C    ALGORITHMS.  ALGORITHM ASVAB INITIALIZES A COUNTER (I), INPUTS   C
C    THE CONVERSICN TABLES FOR ACB COMPOSITE SCORES, AND WRITES THE   C
C    HEADINGS FOR ALL FILES.                                          C
C                                                                     C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C ALGCRITHM ASVAB
C
C
      CALL TABLE(A5,A67,A8910,I,J,K)
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C                                                                     C
C     ALGORITHM INPUT SOLDIER DATA READS THE SOLDIER DEMOGRAPHIC DATA C
C    AND THE ASVAB RAW SCORES.                                        C
C                                                                     C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C    ALGORITHM INPUT SOLDIER DATA
C
      END = 0
C
5     CCNTINUE
C
      CALL IN(SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,RACE,DOEYY,
     *DOEMM,DOEDD,ED,GRADE,MOS1,MOS2,TEST,AFQT,RAW1,RAW2,RAW3,RAW4,RAW5,
     *RAW6,RAW7,RAW8,RAW9,RAW10,RAW11,RAW12,RAW13,RAW14,RAW15,RAW16,END,
     *DES,ETSYY,ETSMM,ETSDD,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,
     *CC,WK,PC,NG,CS,AS)
C
```

```
C-----------------------------------
C     TERMINATE PROGRAM AFTER LAST CARD
C-----------------------------------
C
      IF(END .EQ. 1) GO TO 280
C
      I=I+1
C
C    END ALGORITHM INPUT SOLDIER DATA
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C                                                                     C
C     ALGORITHM VALIDATE SOLDIER DATA VALIDATES THE INPUT DATA BY     C
C    MAKING RANGE CHECKS ON THE FOLLOWING VARIABLES: DOB, SEX,RACE,   C
C    DOE,ED,TEST,GS, AR, MK, MC, EI, VE, GI, AD, SP, SI, AI,CM, CA,   C
C    CE, CC ETS.                                                      C
C                                                                     C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C    ALGORITHM VALIDATE SOLDIER DATA
C
      ERROR=0
C
      CALL VAL(SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,RACE,DOEYY,
     *DOEMM,DOEDD,ED,TEST,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,
     *CC,WK,PC,NO,CS,AS,ETSYY,ETSMM,ETSDD,I,ERROR)
C
C  END ALGORITHM VALIDATE SOLDIER DATA
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C                                                                     C
C     ALGORITHM OUTPUT SOLDIER DATA PRINTS, OR 'ECHOS', THE INPUT     C
C    DATA FOR VALICATION PURPOSES.                                    C
C                                                                     C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C    ALGCRITHM OUTPUT SOLDIER DATA
C
      CALL ECHO(SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,RACE,DOEYY,
     *DOEMM,DOEDD,ED,GRADE,MOS1,MOS2,TEST,AFQT,DES,ETSYY,ETSMM,ETSDD,
```

```
     *GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,CC,WK,PC,NO,CS,AS)
C
      IF(ERROR .EQ. 1) GO TO 5
C
C    END ALGORITHM OUTPUT SOLDIER DATA
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                      C
C                                                                      C
C      ALGORITHM SUBTEST STANDARD SCORES CONVERTS THE SUBTEST RAW      C
C    SCORES TO SUBTEST STANDARD SCORES.                                C
C                                                                      C
C                                                                      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C    ALGORITHM SUBTEST STANDARD SCORES
C
C
      CALL SSS(GI,AD,SP,SI,AI,CM,CA,CE,CC,GS,AR,WK,PC,NO,CS,AS,MK,MC,
     *EI,VE,TEST,GISSS,ADSSS,SPSSS,SISSS,AISSS,CMSSS,CASSS,CESSS,
     *CCSSS,GSSSS,ARSSS,WKSSS,PCSSS,NOSSS,CSSSS,ASSSS,MKSSS,MCSSS,
     *EISSS,VESSS)
C
C
C    END ALGORITHM SUBTEST STANDARD SCORES
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                      C
C                                                                      C
C      ALGORITHM APTITUDE AREA COMPOSITES COMPUTES THE ACB COMPOSITE   C
C    SCORES.                                                           C
C                                                                      C
C                                                                      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C
C
C    ALGORITHM APTITUDE AREA COMPOSITES
C
      CALL COMP(GSSSS,ARSSS,WKSSS,PCSSS,NOSSS,CSSSS,ASSSS,MKSSS,
     *MCSSS,EISSS,VESSS,GISSS,ADSSS,SPSSS,SISSS,AISSS,CMSSS,
     *CASSS,CESSS,CCSSS,GS,AR,WK,PC,NO,CS,AS,MK,MC,EI,VE,GI,AD,SP,
     *SI,AI,CM,CE,CA,CC,CO,FA,EL,OF,GM,MM,CL,GT,ST,SC,TEST)
C
C
```

```
C
C     END ALGORITHM APTITUDE AREA COMPOSITES
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                      C
C                                                                      C
C       ALGORITHM ACB CONVERTS THE ACB SCORES USING THE CONVERSION     C
C     TABLES  A5, A67, A8910, INTO STANDARD SCORES.                    C
C                                                                      C
C                                                                      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C     ALGORITHM ACB
C
        CALL ACB(A5,A67,A8910,TEST,GT,GM,EL,CL,MM,SC,CO,FA,OF,ST)
C
C     END ALGORITHM ACB
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                      C
C                                                                      C
C       ALGORITHM OUTPUT WRITES THE SUBTEST RAW SCORES, THE SUBTEST    C
C     STANDARD SCORES, AND THE ACB COMPOSITE SCORES TO SUMMARY FILES.  C
C                                                                      C
C                                                                      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C     ALGCRITHM OUTPUT
C
C
C
        CALL OUT(SSAN1,SSAN2,SSAN3,AFQT,GSSSS,ARSSS,MKSSS,MCSSS,
     *EISSS,VESSS,GISSS,ADSSS,SPSSS,SISSS,AISSS,CMSSS,CASSS,CESSS,
     *CCSSS,NOSSS,ASSSS,WKSSS,CSSSS,PCSSS,TEST,CO,FA,MM,GM,CL,GT,
     *EL,SC,ST,OF,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,CC,WK,PC,
     *NO,CS,AS)
C
C     END ALGORITHM OUTPUT
C
C
        GO TO 5
280     CCNTINUE
C
C
```

```
C ENC ALGORITHM ASVAB
C
C
      STOP
      END
C
C
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                              SUBPROGRAMS                             C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                      C
C                                                                      C
C     SUBROUTINE TABLE INITIALIZES A COUNTER, READS IN THE CONVERSION  C
C    TABLES FROM FILES 1, 2, AND 3, AND PRINTS FILE HEADINGS FOR FILES C
C    10, 11, AND 12.                                                   C
C                                                                      C
C                                                                      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      SUBROUTINE TABLE(A5,A67,A8910,I,J,K)
C
      INTEGER A5(320,10), A67(111,10), A8910(281,10),I,J,K
C
C-----------------------
C     INITIALIZE COUNTER
C-----------------------
C
      I=0
C
C----------------------------
C     INPUT CONVERSION TABLES
C----------------------------
C
C
C
C------------
C     ASVAB 5
C------------
C
```

```
      DO 1 J=1,320
         READ(1,499) (A5(J,K),K=1,10)
1     CONTINUE
C
C----------------
C     ASVAB 6/7
C----------------
C
      DO 2 J=1,111
         READ(2,499) (A67(J,K),K=1,10)
2     CONTINUE
C
C----------------
C     ASVAB 8/9/10
C----------------
C
      DO 3 J=1,281
         READ(3,499) (A8910(J,K),K=1,10)
3     CONTINUE
C
C------------------
C     PRINT HEADINGS
C------------------
C
      WRITE(10,890)
      WRITE(10,917)
      WRITE(11,871)
      WRITE(11,917)
      WRITE(12,910)
      WRITE(12,916)
      RETURN
C
499   FORMAT(10I5)
871   FORMAT(/16X,'SUBTEST RAW SCORES')
890   FORMAT(/12X,'SUBTEST STANDARD SCORES')
910   FORMAT(/6X, 'ARMY CLASSIFICATION BATTERY COMPOSITES')
916   FORMAT(' SSAN',11X,'AFQT CO   FA   MM   GM   CL   GT   EL   SC   ST   OF')
917   FORMAT('  SSAN         GI NO AD WK AR SP MK EI MC GS SI AI CM CA CE
     *CC PC CS AS VE')
      END
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                    C
C                                                                    C
C     SUBROUTINE IN INPUTS THE SOLDIER DEMOGRAPHIC DATA AND THE RAW  C
C   SCORES FROM FILE 4.                                              C
C                                                                    C
```

```
C                                                                      C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      SUBROUTINE IN(SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,RACE,DOEYY,
     *DOEMM,DOEDD,ED,GRADE,MOS1,MOS2,TEST,AFQT,RAW1,RAW2,RAW3,RAW4,RAW5,
     *RAW6,RAW7,RAW8,RAW9,RAW10,RAW11,RAW12,RAW13,RAW14,RAW15,RAW16,END,
     *DES,ETSYY,ETSMM,ETSDD,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,
     *CC,WK,PC,NC,CS,AS)
C
      INTEGER SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,RACE,DOEYY,
     *DOEMM,DOEDD,ED,GRADE,MOS1,MOS2,TEST,AFQT,RAW1,RAW2,RAW3,RAW4,RAW5,
     *RAW6,RAW7,RAW8,RAW9,RAW10,RAW11,RAW12,RAW13,RAW14,RAW15,RAW16,END,
     *DES,ETSYY,ETSMM,ETSDD,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,
     *CC,WK,PC,NO,CS,AS
C
C-----------------------------------------
C     INPUT DEMOGRAPHIC DATA AND RAW SCORES
C-----------------------------------------
C
      READ(4,500,END=100) SSAN1,SSAN2,SSAN3,DOBYY,DCBMM,DOBDD,SEX,RACE,
     *DOEYY,DOEMM,DOEDD,ED,GRADE,MOS1,MOS2,TEST,AFQT,RAW1,RAW2,RAW3,
     *RAW4,RAW5,RAW6,RAW7,RAW8,RAW9,RAW10,RAW11,RAW12,RAW13,RAW14,RAW15,
     *RAW16,CES,ETSYY,ETSMM,ETSDD
C
C
C-----------------------------------
C     ASSIGN RAW SCORES TO VARIABLES
C-----------------------------------
C
C
C
C----------------
C     ASVAB 8/9/10
C----------------
C
      IF(TEST .LE. 37) GO TO 6
         GS=RAW1
         AR=RAW2
         WK=RAW3
         PC=RAW4
         NO=RAW5
         CS=RAW6
         AS=RAW7
         MK=RAW8
         MC=RAW9
         EI=RAW10
         VE=RAW11
      GC TO 7
```

```
C
C-----------------
C       ASVAB 5/6/7
C-----------------
C
6       CCNTINUE
           GI=RAW1
           NO=RAW2
           AD=RAW3
           WK=RAW4
           AR=RAW5
           SP=RAW6
           MK=RAW7
           EI=RAW8
           MC=RAW9
           GS=RAW10
           SI=RAW11
           AI=RAW12
           CM=RAW13
           CA=RAW14
           CE=RAW15
           CC=RAW16
7       CCNTINUE
C
        RETURN
100     END = 1
        RETURN
C
500     FORMAT(3A3,3I2,2I1,5I2,A4,A1,18I2,2X,A3,3I2)
        END
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C                                                                     C
C       SUBROUTINE VAL VALIDATES THE DEMOGRAPHIC DATA AND THE RAW     C
C     SCORES. IF AN ERROR IS FOUND IT TERMINATES PROCESSING OF THE    C
C     RECORD.                                                         C
C                                                                     C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
        SUBROUTINE VAL(SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,RACE,
       *DOEYY,DOEMM,DOEDD,ED,TEST,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,
       *CM,CA,CE,CC,WK,PC,NO,CS,AS,ETSYY,ETSMM,ETSDD,I,ERROR)
C
        INTEGER SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,RACE,DOEYY,
```

```
     *DOBMM,DOEDD,ED,TEST,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,
     *CC,WK,PC,NO,CS,AS,ETSYY,ETSMM,ETSDD,I,ERROR
C
C---------------------------------
C     VALIDATE DATE OF BIRTH
C---------------------------------
C
      IF(DOBYY .GT. 31) GO TO 8
         WRITE(9,510) I,SSAN1,SSAN2,SSAN3
         WRITE(9,515)
         ERROR = 1
8     CONTINUE
C
C-------------------------
C     VALIDATE SEX
C-------------------------
C
      IF((SEX .EQ. 1) .OR.(SEX .EQ. 2)) GO TO 9
         WRITE(9,510) I,SSAN1,SSAN2,SSAN3
         WRITE(9,520)
         ERROR = 1
9     CCNTINUE
C
C-------------------------
C     VALIDATE RACE
C-------------------------
C
      IF((RACE .EQ. 1).CR.(RACE .EQ.2).OR.(RACE .EQ. 3)) GO TO 10
         WRITE(9,510) I,SSAN1,SSAN2,SSAN3
         WRITE(9,525)
         ERROR = 1
10    CCNTINUE
C
C---------------------------------
C     VALIDATE DATE OF ENTRY
C---------------------------------
C
      IF(DOEYY .GT. 31) GO TO 11
         WRITE(9,510) I,SSAN1,SSAN2,SSAN3
         WRITE(9,530)
         ERROR = 1
11    CONTINUE
C
C----------------------------------------------
C     VALIDATE HIGHEST YEAR OF EDUCATION
C----------------------------------------------
C
      IF((ED .GE. 1) .OR. (ED .LE. 13)) GO TO 12
```

```
         WRITE(9,510) I,SSAN1,SSAN2,SSAN3
         WRITE(9,535)
         ERROR = 1
12    CONTINUE
C
C-------------------------
C     VALIDATE TEST VERSION
C-------------------------
C
      IF((TEST .GE. 35) .AND.(TEST .LE. 40)) GO TO 13
         WRITE(9,510) I, SSAN1,SSAN2,SSAN3
         WRITE(9,540)
         ERROR = 1
13    CONTINUE
C
C-------------------------------
C     VALIDATE SUBTEST RAW SCORES
C-------------------------------
C
C
C------------------
C        ASVAB 5/6/7
C------------------
      IF(TEST .GT. 37) GO TO 30
         IF(GI .LE. 15) GO TO 14
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,545)
            ERROR = 1
14       CONTINUE
         IF(NO .LE. 50) GO TO 15
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,550)
            ERROR = 1
15       CONTINUE
         IF(AD .LE. 30) GO TO 16
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,555)
            ERROR = 1
16       CONTINUE
         IF(WK .LE. 30) GO TO 17
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,560)
            ERROR = 1
17       CONTINUE
         IF(AR .LE. 20) GO TO 18
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,565)
            ERROR = 1
```

```
18      CONTINUE
        IF(SP  .LE. 20) GO TO 19
           WRITE(9,510) I, SSAN1,SSAN2,SSAN3
           WRITE(9,570)
           ERROR = 1
19      CONTINUE
        IF(MK  .LE. 20) GO TO 20
           WRITE(9,510) I,SSAN1,SSAN2,SSAN3
           WRITE(9,575)
           ERROR = 1
20      CONTINUE
        IF(EI  .LE. 30) GO TO 21
           WRITE(9,510) I,SSAN1,SSAN2,SSAN3
           WRITE(9,580)
           ERROR = 1
21      CONTINUE
        IF(MC  .LE. 20) GO TO 22
           WRITE(9,510) I,SSAN1,SSAN2,SSAN3
           WRITE(9,585)
           ERROR = 1
22      CONTINUE
        IF(GS  .LE. 20) GO TO 23
           WRITE(9,510) I,SSAN1,SSAN2,SSAN3
           WRITE(9,590)
           ERROR = 1
23      CONTINUE
        IF(SI  .LE. 20) GO TO 24
           WRITE(9,510) I,SSAN1,SSAN2,SSAN3
           WRITE(9,595)
           ERROR = 1
24      CONTINUE
        IF(AI  .LE. 20) GO TO 25
           WRITE(9,510) I, SSAN1,SSAN2,SSAN3
           WRITE(9,600)
           ERROR = 1
25      CONTINUE
        IF(CM  .LE. 20) GO TO 26
           WRITE(9,510) I,SSAN1,SSAN2,SSAN3
           WRITE(9,605)
           ERROR = 1
26      CONTINUE
        IF(CA  .LE. 20) GO TO 27
           WRITE(9,510) I,SSAN1,SSAN2,SSAN3
           WRITE(9,610)
           ERROR = 1
27      CONTINUE
        IF(CE  .LE. 20) GO TO 28
           WRITE(9,510) I,SSAN1,SSAN2,SSAN3
```

```
            WRITE(9,615)
            ERROR = 1
28        CONTINUE
          IF(CC  .LE. 27) GO TO 29
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,620)
            ERROR = 1
29        CONTINUE
      GO TO 42
C
C---------------------
C         ASVAB 8/9/10
C---------------------
30    CCNTINUE
          IF(GS  .LE. 25) GO TO 31
            WRITE(9,510) I, SSAN1,SSAN2,SSAN3
            WRITE(9,625)
            ERROR = 1
31        CONTINUE
          IF(AR  .LE. 30) GO TO 32
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,630)
            ERROR = 1
32        CONTINUE
          IF(WK  .LE. 35) GO TO 33
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,635)
            ERROR = 1
33        CONTINUE
          IF(PC  .LE. 15) GO TO 34
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,640)
            ERROR = 1
34        CONTINUE
          IF(NO  .LE. 50) GO TO 35
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,645)
            ERROR = 1
35        CONTINUE
          IF(CS  .LE. 84) GO TO 36
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,650)
            ERROR = 1
36        CONTINUE
          IF(AS  .LE. 25) GO TO 37
            WRITE(9,510) I,SSAN1,SSAN2,SSAN3
            WRITE(9,655)
            ERROR = 1
```

```
37        CONTINUE
          IF(MK  .LE. 25) GO TO 38
             WRITE(9,510) I,SSAN1,SSAN2,SSAN3
             WRITE(9,660)
             ERROR = 1
38        CONTINUE
          IF(MC  .LE. 25) GO TO 39
             WRITE(9,510) I,SSAN1,SSAN2,SSAN3
             WRITE(9,665)
             ERROR = 1
39        CONTINUE
          IF(EI  .LE. 20) GO TO 40
             WRITE(9,510) I,SSAN1,SSAN2,SSAN3
             WRITE(9,670)
             ERROR = 1
40        CONTINUE
          IF(VE  .LE. 50) GO TO 41
             WRITE(9,510) I,SSAN1,SSAN2,SSAN3
             WRITE(9,675)
             ERROR = 1
41        CONTINUE
42     CONTINUE
C
C--------------------------------
C      VALIDATE DATE OF SEPARATION
C--------------------------------
C
       IF(ETSYY .EQ. 00) GO TO 43
       IF(ETSYY .GT. 31) GO TO 43
          WRITE(9,510) I,SSAN1,SSAN2,SSAN3
          WRITE(9,680)
          ERROR = 1
43     CONTINUE
       RETURN
C
510    FORMAT(///' ERROR IN RECORD NUMBER: ',I3,' WITH SSAN: ',3A3)
515    FORMAT(' ERROR: DATE OF BIRTH INCORRECT, COLUMNS 10-15')
520    FORMAT(' ERROR: SEX INCORRECT, COLUMN 16')
525    FORMAT(' ERROR: RACE INCORRECT, COLUMN 17')
530    FORMAT(' ERROR: DATE OF ENTRY INCORRECT, COLUMNS 18-23')
535    FORMAT(' ERROR: HIGHEST YEAR OF EDUCATION INCORRECT, COLUMNS 24-
      *25')
540    FORMAT(' ERROR: TEST FORM INCORRECT, COLUMNS 33-34')
545    FORMAT(' ERROR: GI INCORRECT, COLUMNS 37-38')
550    FORMAT(' ERROR: NO INCORRECT, COLUMNS 39-40')
555    FORMAT(' ERROR: AD INCORRECT, COLUMNS 41-42')
560    FORMAT(' ERROR: WK INCORRECT, COLUMNS 43-44')
565    FORMAT(' ERROR: AR INCORRECT, COLUMNS 45-46')
```

```
570   FORMAT(' ERROR: SP INCORRECT, COLUMNS 47-48')
575   FORMAT(' ERROR: MK INCORRECT, COLUMNS 49-50')
580   FORMAT(' ERROR: EI INCORRECT, COLUMNS 51-52')
585   FORMAT(' ERROR: MC INCORRECT, COLUMNS 53-54')
590   FORMAT(' ERROR: GS INCORRECT, COLUMNS 55-56')
595   FORMAT(' ERROR: SI INCORRECT, COLUMNS 57-58')
600   FORMAT(' ERROR: AI INCORRECT, COLUMNS 59-60')
605   FORMAT(' ERROR: CM INCORRECT, COLUMNS 61-62')
610   FORMAT(' ERROR: CA INCORRECT, COLUMNS 63-64')
615   FORMAT(' ERROR: CE INCORRECT, COLUMNS 65-66')
620   FORMAT(' ERROR: CC INCORRECT, COLUMNS 67-68')
625   FORMAT(' ERROR: GS INCORRECT, COLUMNS 37-38')
630   FORMAT(' ERROR: AR INCORRECT, COLUMNS 39 40')
635   FORMAT(' ERROR: WK INCORRECT, COLUMNS 41-42')
640   FORMAT(' ERROR: PC INCORRECT, COLUMNS 43-44')
645   FORMAT(' ERROR: NO INCORRECT, COLUMNS 45-46')
650   FORMAT(' ERROR: CS INCORRECT, COLUMNS 47-48')
655   FORMAT(' ERROR: AS INCORRECT, COLUMNS 49-50')
660   FORMAT(' ERROR: MK INCORRECT, COLUMNS 51-52')
665   FORMAT(' ERROR: MC INCORRECT, COLUMNS 53-54')
670   FORMAT(' ERROR: EI INCORRECT, COLUMNS 55-56')
675   FORMAT(' ERROR: VE INCORRECT, COLUMNS 57-58')
680   FORMAT(' ERROR: DATE OF SEPARATION INCORRECT, COLUMNS 74-79')
      END
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C                                                                     C
C     SUBROUTINE ECHO WRITES THE DEMOGRAPHIC DATA AND RAW SCORES TO   C
C   FILE 9.                                                           C
C                                                                     C
C                                                                     C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      SUBROUTINE ECHO(SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,
     *RACE,DOEYY,DOEMM,DOEDD,ED,GRADE,MOS1,MOS2,TEST,AFQT,DES,ETSYY,
     *ETSMM,ETSDD,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,CC,WK,PC,
     *NC,CS,AS)
C
      INTEGER SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,RACE,DOEYY,
     *DOEMM,DOEDD,ED,GRADE,MOS1,MOS2,TEST,AFQT,DES,ETSYY,ETSMM,ETSDD,
     *GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,CC,WK,PC,NO,CS,AS
C
C-----------------------------------
C     PRINT SSAN, AND DATE OF BIRTH
C-----------------------------------
```

```
C
      WRITE(9,681) SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD
C
C----------------
C     PRINT SEX
C----------------
C
      IF(SEX .NE. 1) GO TO 45
         WRITE(9,685)
      GO TO 46
45    CCNTINUE
         WRITE(9,690)
46    CONTINUE
C
C-----------------
C     PRINT RACE
C-----------------
C
      IF(RACE .NE. 1) GO TC 47
         WRITE(9,695)
      GO TO 50
47    IF(RACE .NE. 2) GO TO 48
         WRITE(9,700)
      GO TO 50
48    CONTINUE
         WRITE(9,705)
50    CONTINUE
C
C--------------------------
C     PRINT DATE CF ENTRY
C--------------------------
C
      WRITE(9,71C) DOEYY,DCEMM,DOEDD
C
C-------------------------------------
C     PRINT HIGHEST YEAR OF EDUCATION
C-------------------------------------
C
      IF(ED .LT. 1) GO TO 120
      IF(ED .GT. 13) GO TO 120
      GC TO (55,60,65,70,75,80,85,90,95,100,105,110,115),ED
55    CONTINUE
         WRITE(9,711) ED
      GO TO 120
60    CONTINUE
         WRITE(9,715) ED
      GO TO 120
65    CCNTINUE
```

```
        WRITE(9,720) ED
      GO TO 120
70    CCNTINUE
        WRITE(9,725) ED
      GO TO 120
75    CCNTINUE
        WRITE(9,730) ED
      GO TO 120
80    CCNTINUE
        WRITE(9,735) ED
      GO TO 120
85    CONTINUE
        WRITE(9,740) ED
      GC TO 120
90    CONTINUE
        WRITE(9,745) ED
      GO TO 120
95    CONTINUE
        WRITE(9,750) ED
      GO TO 120
100   CONTINUE
        WRITE(9,755) ED
      GO TO 120
105   CONTINUE
        WRITE(9,760) ED
      GC TO 120
110   CCNTINUE
        WRITE(9,765) ED
      GO TO 120
115   CONTINUE
        WRITE(9,770) ED
120   CONTINUE
C
C-----------------
C     PRINT GRADE
C-----------------
C
      IF(GRADE .LT. 1) GO TO 170
      IF(GRADE .GT. 9) GO TO 170
      GO TO (125,130,135,140,145,150,155,160,165),GRADE
125   CONTINUE
        WRITE(9,775) GRADE
      GO TO 170
130   CCNTINUE
        WRITE(9,780) GRADE
      GC TO 170
135   CONTINUE
        WRITE(9,785) GRADE
```

```
      GO TO 170
140   CCNTINUE
         WRITE(9,790) GRADE
      GO TO 170
145   CONTINUE
         WRITE(9,795) GRADE
      GO TO 170
150   CCNTINUE
         WRITE(9,800) GRADE
      GO TO 170
155   CCNTINUE
         WRITE(9,810) GRADE
      GO TO 170
160   CCNTINUE
         WRITE(9,815) GRADE
      GO TO 170
165   CCNTINUE
         WRITE(9,820) GRADE
170   CCNTINUE
C
C---------------
C     PRINT MOS
C---------------
C
      WRITE(9,825) MOS1,MOS2
C
C-----------------------
C     PRINT TEST VERSION
C-----------------------
C
      IF(TEST .NE. 35) GO TO 175
         WRITE(9,830) TEST
      GO TO 200
175   IF(TEST .NE. 36) GO TO 180
         WRITE(9,835) TEST
      GO TO 200
180   IF(TEST .NE. 37) GO TO 185
         WRITE(9,840) TEST
      GO TO 200
185   IF(TEST .NE. 38) GO TO 190
         WRITE(9,845) TEST
      GO TO 200
190   IF(TEST .NE. 39) GO TO 195
         WRITE(9,850) TEST
      GO TO 200
195   CONTINUE
         WRITE(9,855) TEST
200   CCNTINUE
```

```
C
C-----------------------------
C     PRINT AFQT PERCENTILE
C-----------------------------
C
      WRITE(9,860) AFQT
C
C----------------------------------------
C     PRINT SEPARATION PROGRAM DESIGNATOR
C----------------------------------------
C
      WRITE(9,865) DES
C
C--------------------------------
C     PRINT DATE OF SEPARATION
C--------------------------------
C
      WRITE(9,870) ETSYY,ETSMM,ETSDD
      WRITE(9,871)
C
C------------------------------
C     PRINT ASVAB 5 RAW SCCRES
C------------------------------
C
      IF(TEST .NE. 35) GO TO 205
         WRITE(9,874)
         WRITE(9,875) GI,NO,AD,WK,AR,SP,MK,EI,MC,GS,SI,AI
      GO TO 215
205   CONTINUE
C
C--------------------------------
C     PRINT ASVAB 6/7 RAW SCORES
C--------------------------------
C
      IF(TEST .GT. 37) GO TO 210
         WRITE(9,879)
         WRITE(9,880) GI,NO,AD,WK,AR,SP,MK,EI,MC,GS,SI,AI,CM,CA,CE,CC
      GO TO 215
210   CONTINUE
C
C-----------------------------------
C     PRINT ASVAB 8/9/10 RAW SCORES
C-----------------------------------
C
         WRITE(9,884)
         WRITE(9,885) GS,AR,WK,PC,NO,CS,AS,MK,MC,EI,VE
215   CONTINUE
      RETURN
```

```
C
681   FORMAT(///,' SSAN',25X,': ',3A3,/' DATE OF BIRTH(YY/MM/DD)',6X,
     *': ',I2,'/',I2,'/',I2)
685   FORMAT(' SEX                           : 1 - MALE')
690   FORMAT(' SEX                           : 2 - FEMALE')
695   FORMAT(' RACE                          : 1 - WHITE')
700   FORMAT(' RACE                          : 2 - BLACK')
705   FORMAT(' RACE                          : 3 - OTHER')
710   FORMAT(' DATE OF ENTRY(YY/MM/DD)       : ',I2,'/',I2,'/',I2)
711   FORMAT(' HIGHEST YEAR OF EDUCATION     : 1-7 YEARS - ',I2)
715   FORMAT(' HIGHEST YEAR OF EDUCATION     : 8 YEARS - ',I2)
720   FORMAT(' HIGHEST YEAR OF EDUCATION     : 1 YEAR OF HIGH SCHOOL - '
     *,I2)
730   FORMAT(' HIGHEST YEAR OF EDUCATION     : 3-4 YEARS OF HIGH',
     *' SCHOOL - ',I2)
725   FORMAT(' HIGHEST YEAR OF EDUCATION     : 2 YEARS OF HIGH SCHOOL - '
     *,I2)
735   FORMAT(' HIGHEST YEAR OF EDUCATION     : HIGH SCHOOL DIPLOMA - '
     *,I2)
740   FORMAT(' HIGHEST YEAR OF EDUCATION     : 1 YEAR CF COLLEGE - '
     *,I2)
745   FORMAT(' HIGHEST YEAR OF EDUCATION     : 2 YEARS OF COLLEGE - '
     *,I2)
750   FORMAT(' HIGHEST YEAR OF EDUCATION     : 3-4 YEARS OF COLLEGE - '
     *,I2)
755   FORMAT(' HIGHEST YEAR OF EDUCATION     : COLLEGE GRADUATE - ',I2)
760   FORMAT(' HIGHEST YEAR OF EDUCATION     : MASTERS DEGREE - ', I2)
765   FORMAT(' HIGHEST YEAR OF EDUCATION     : DOCTORS DEGREE - ',I2)
770   FORMAT(' HIGHEST YEAR OF EDUCATION     : HIGH SCHOOL GED - ',I2)
775   FORMAT(' ENTRY PAY GRADE               : E1 ',I2)
780   FORMAT(' ENTRY PAY GRADE               : E2 ',I2)
785   FORMAT(' ENTRY PAY GRADE               : E3 ',I2)
790   FORMAT(' ENTRY PAY GRADE               : E4 ',I2)
795   FORMAT(' ENTRY PAY GRADE               : E5 ',I2)
800   FORMAT(' ENTRY PAY GRADE               : E6 ',I2)
810   FORMAT(' ENTRY PAY GRADE               : E7 ',I2)
815   FORMAT(' ENTRY PAY GRADE               : E8 ',I2)
820   FORMAT(' ENTRY PAY GRADE               : E9 ',I2)
825   FORMAT(' MOS                           : ',A4,A1)
830   FORMAT(1X,I2,' TEST FORM                  : ASVAB 5')
835   FORMAT(1X,I2,' TEST FORM                  : ASVAB 6')
840   FORMAT(1X,I2,' TEST FORM                  : ASVAB 7')
845   FORMAT(1X,I2,' TEST FORM                  : ASVAB 8')
850   FORMAT(1X,I2,' TEST FORM                  : ASVAB 9')
855   FORMAT(1X,I2,' TEST FORM                  : ASVAB 10')
860   FORMAT(' AFQT PERCENTILE               : ',I2)
865   FORMAT(' SEPARATION PROGRAM DESIGNATOR: ', A3)
870   FORMAT(' DATE OF SEPARATION(YY/MM/DD) : ',I2,'/',I2,'/',I2)
```

```
871   FORMAT(/16X,'SUBTEST RAW SCORES')
874   FORMAT('    GI   NO   AD   WK   AR   SP   MK   EI   MC   GS   SI   AI')
875   FORMAT(1X,12(2X,I2))
879   FORMAT('    GI   NO   AD   WK   AR   SP   MK   EI   MC   GS   SI   AI   CM   CA
     * CE  CC')
880   FORMAT(1X,16(2X,I2))
884   FORMAT('    GS   AR   WK   PC   NO   CS   AS   MK   MC   EI   VE')
885   FORMAT(1X,11(2X,I2))
      END
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                 C
C                                                                 C
C     SUBROUTINE SSS CONVERTS THE SUBTEST RAW SCCRES INTO SUBTEST  C
C   STANDARD SCORES.                                              C
C                                                                 C
C                                                                 C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      SUBROUTINE SSS(GI,AD,SP,SI,AI,CM,CA,CE,CC,GS,AR,WK,PC,NC,CS,AS,
     *MK,MC,EI,VE,TEST,GISSS,ADSSS,SPSSS,SISSS,AISSS,CMSSS,CASSS,CESSS,
     *CCSSS,GSSSS,ARSSS,WKSSS,PCSSS,NOSSS,CSSSS,ASSSS,MKSSS,MCSSS,
     *EISSS,VESSS)
C
      INTEGER  GI,AD,SP,SI,AI,CM,CA,CE,CC,GS,AR,WK,PC,NO,CS,AS,MK,MC,
     *EI,VE,TEST,GISSS,ADSSS,SPSSS,SISSS,AISSS,CMSSS,CASSS,CESSS,
     *CCSSS,GSSSS,ARSSS,WKSSS,PCSSS,NOSSS,CSSSS,ASSSS,MKSSS,MCSSS,
     *EISSS,VESSS
C
      REAL CONVT,GIR,ADR,SPR,SIR,AIR,CMR,CAR,CER,CCR,GSR,ARR,WKR,PCR,
     *NOR,CSR,ASR,MKR,MCR,EIR,VER
C
C
C-----------------------
C     CCNVERT ASVAB 8/9/10
C-----------------------
C
      IF((TEST .NE. 38) .AND.(TEST .NE. 39) .AND.(TEST .NE. 40))
     *GO TO 220
         GSR=FLOAT(GS)
         GSSSS=CCNVT(GSR,16.199,5.087)
         ARR=FLOAT(AR)
         ARSSS=CCNVT(ARR,17.766,7.2)
         WKR=FLOAT(WK)
         WKSSS=CCNVT(WKR,25.670,7.658)
         PCR=FLOAT(PC)
```

```
          PCSSS=CCNVT(PCR,10.483,3.436)
          NOR=FLOAT(NO)
          NOSSS=CCNVT(NOR,36.027,10.387)
          CSR=FLOAT(CS)
          CSSSS=CONVT(CSR,43.149,16.121)
          ASR=FLOAT(AS)
          ASSSS=CCNVT(ASR,16.413,5.597)
          MKR=FLOAT(MK)
          MKSSS=CCNVT(MKR,12.468,5.950)
          MCR=FLOAT(MC)
          MCSSS=CCNVT(MCR,15.519,5.570)
          EIR=FLOAT(EI)
          EISSS=CCNVT(EIR,12.451,4.316)
          VER=FLOAT(VE)
          VESSS=CCNVT(VER,36.153,10.613)
      GO TO 225
C
C-----------------------
C     CCNVERT ASVAB 5/6/7
C-----------------------
C
220   CCNTINUE
          GIR=FLOAT(GI)
          GISSS=CCNVT(GIR,9.656,3.273)
          NOR=FLOAT(NO)
          NOSSS=CCNVT(NOR,30.826,10.049)
          ADR=FLOAT(AD)
          ADSSS=CCNVT(ADR,14.410,3.957)
          WKR=FLOAT(WK)
          WKSSS=CCNVT(WKR,19.638,7.251)
          ARR=FLOAT(AR)
          ARSSS=CONVT(ARR,12.739,4.729)
          SPR=FLOAT(SP)
          SPSSS=CCNVT(SPR,13.197,4.257)
          MKR=FLOAT(MK)
          MKSSS=CCNVT(MKR,11.636,4.932)
          EIR=FLOAT(EI)
          EISSS=CCNVT(EIR,19.474,5.955)
          MCR=FLOAT(MC)
          MCSSS=CCNVT(MCR,10.842,4.361)
          GSR=FLOAT(GS)
          GSSSS=CCNVT(GSR,11.317,4.362)
          SIR=FLOAT(SI)
          SISSS=CCNVT(SIR,13.648,4.302)
          AIR=FLOAT(AI)
          AISSS=CCNVT(AIR,11.773,4.946)
225   CCNTINUE
C
```

```
C------------------------
C     CCNVERT ASVAB 6/7
C------------------------
C
      IF((TEST .NE. 36) .AND. (TEST .NE. 37)) GO TO 230
         CMR=FLOAT(CM)
         CMSSS=CCNVT(CMR,10.97,4.69)
         CAR=FLOAT(CA)
         CASSS=CCNVT(CAR,9.68,2.91)
         CER=FLOAT(CE)
         CESSS=CCNVT(CER,8.92,4.21)
         CCR=FLOAT(CC)
         CCSSS=CCNVT(CCR,15.51,4.89)
230   CONTINUE
      RETURN
      END
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                    C
C                                                                    C
C     FUNCTION CCNVT CONVERTS THE SUBTEST RAW SCORES TO SUBTEST      C
C   STANDARD SCORES.                                                 C
C                                                                    C
C                                                                    C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      REAL FUNCTICN CONVT(RAW,XBAR,SIGMA)
C
C-----------------------------
C     VARIABLES AND CONSTANTS
C-----------------------------
C
C     RAW           SUBTEST RAW SCORE.
C     XBAR          SUBTEST POPULATION MEAN.
C     SIGMA         SUBTEST POPULATION STANDARD DEVIATION.
C     T             TEMPORARY VARIABLE.
C     TT            TEMPORARY VARIABLE.
C     TTT           TEMPCRARY VARIABLE.
C
C------------------------
C     DECLARE VARIABLES
C------------------------
C
      INTEGER TT
      REAL T,DIFF,TTT,XBAR,SIGMA,RAW
C
```

```
      T=(10.0*(RAW-XBAR)/SIGMA)+50.0
      TT=IFIX(T)
      TTT=FLOAT(TT)
      DIFF=T-TTT
      IF(DIFF .GE. 0.5) GO TO 10
         CONVT=TT
      GO TO 20
10    CONTINUE
         CONVT=TT+1
20    CONTINUE
      IF(CONVT .LT. 20) CONVT=20
      IF(CONVT .GT. 80) CONVT=80
      RETURN
      END
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C                                                                     C
C     SUBROUTINE COMP COMPUTES THE ACB COMPOSITE SCORES.              C
C                                                                     C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      SUBROUTINE COMP(GSSSS,ARSSS,WKSSS,PCSSS,NOSSS,CSSSS,ASSSS,MKSSS,
     *MCSSS,EISSS,VESSS,GISSS,ADSSS,SPSSS,SISSS,AISSS,CMSSS,
     *CASSS,CESSS,CCSSS,GS,AR,WK,PC,NO,CS,AS,MK,MC,EI,VE,GI,AD,SP,
     *SI,AI,CM,CE,CA,CC,CO,FA,EL,OF,GM,MM,CL,GT,ST,SC,TEST)
C
      INTEGER GSSSS,ARSSS,WKSSS,PCSSS,NOSSS,CSSSS,ASSSS,MKSSS,
     *MCSSS,EISSS,VESSS,GISSS,ADSSS,SPSSS,SISSS,AISSS,CMSSS,TEST,
     *CASSS,CESSS,CCSSS,GS,AR,WK,PC,NO,CS,AS,MK,MC,EI,VE,GI,AD,SP,
     *SI,AI,CM,CA,CE,CC,CO,FA,EL,OF,GM,MM,CL,GT,ST,SC
C
C-------------------
C     COMPUTE ASVAB 5
C-------------------
C
      IF(TEST .NE. 35) GO TO 235
         CO=ARSSS+SISSS+SPSSS+ADSSS
         FA=ARSSS+GISSS+MKSSS+EISSS
         MM=MKSSS+SISSS+EISSS+AISSS
         GM=ARSSS+GSSSS+MCSSS+AISSS
         CL=ARSSS+WKSSS+ADSSS
         GT=ARSSS+WKSSS
         EL=ARSSS+EISSS+MCSSS+SISSS
         SC=ARSSS+WKSSS+MCSSS+SPSSS
```

```
      ST=ARSSS+MKSSS+GSSSS
      OF=GISSS+AISSS
      GO TO 245
C
C------------------------
C     COMPUTE ASVAB 6/7
C------------------------
C
235   IF((TEST .NE. 36) .AND. (TEST .NE. 37)) GO TO 240
         CO=AR+SI+SP+AD+CC
         FA=AR+GI+MK+EI+CA
         MM=MK+SI+EI+AI+CM
         GM=AR+GS+MC+AI
         CL=AR+WK+AD+CA
         GT=AR+WK
         EL=AR+EI+MC+SI+CE
         SC=AR+WK+MC+SP
         ST=AR+MK+GS
         OF=GI+AI+CA
      GO TO 245
C
C----------------------------
C     COMPUTE ASVAB 8/9/10
C----------------------------
C
240   CONTINUE
         CO=ARSSS+ASSSS+MCSSS+CSSSS
         FA=ARSSS+MKSSS+MCSSS+CSSSS
         EL=ARSSS+EISSS+MKSSS+GSSSS
         OF=NOSSS+VESSS+MCSSS+ASSSS
         SC=NOSSS+CSSSS+VESSS+ASSSS
         MM=NOSSS+EISSS+MCSSS+ASSSS
         GM=MKSSS+EISSS+GSSSS+ASSSS
         CL=NOSSS+CSSSS+VESSS
         ST=VESSS+MKSSS+MCSSS+GSSSS
         GT=VESSS+ARSSS
245   CONTINUE
      RETURN
      END
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C                                                                     C
C     SUBROUTINE ACB CONVERTS THE ACB SCORES INTO STANDARD SCORES.    C
C                                                                     C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
C
C
      SUBROUTINE ACB(A5 ,A67,A8910,TEST,GT,GM,EL,CL,MM,SC,CO,FA,OF,ST)
C
      INTEGER A5(320,10 ),A67(111,10),A8910(281,10),GT,GM,EL,CL,MM,SC,
     *CO,FA,OF,ST,TEST
C
C
C-----------------------
C     CCNVERT ASVAB 5
C-----------------------
C
      IF(TEST .NE. 35) GO TO 250
         GT=A5(GT,1)
         GM=A5(GM,2)
         EL=A5(EL,3)
         CL=A5(CL,4)
         MM=A5(MM,5)
         SC=A5(SC,6)
         CO=A5(CC,7)
         FA=A5(FA,8)
         OF=A5(OF,9)
         ST=A5(ST,10)
      GC TO 260
C
C-------------------------
C     CCNVERT ASVAB 6/7
C-------------------------
C
250   IF((TEST .NE. 36).AND.(TEST .NE. 37)) GO TO 255
         GT=A67(GT+1,1)
         GM=A67(GM+1,2)
         EL=A67(EL+1,3)
         CL=A67(CL+1,4)
         MM=A67(MM+1,5)
         SC=A67(SC+1,6)
         CO=A67(CC+1,7)
         FA=A67(FA+1,8)
         OF=A67(CF+1,9)
         ST=A67(ST+1,10)
      GO TO 260
C
C----------------------------
C     CCNVERT ASVAB 8/9/10
C----------------------------
C
255   CCNTINUE
         GT=A8910(GT,1)
```

```
      GM=A8910(GM,2)
      EL=A8910(EL,3)
      CL=A8910(CL,4)
      MM=A8910(MM,5)
      SC=A8910(SC,6)
      CO=A8910(CO,7)
      FA=A8910(FA,8)
      OF=A8910(OF,9)
      ST=A8910(ST,10)
260   CCNTINUE
C
      RETURN
      END
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C                                                                     C
C                                                                     C
C     SUBROUTINE OUT WRITES THE SUBTEST RAW SCORES TO FILE 11, THE    C
C    SUBTEST STANDARD SCORES TO FILES 9 AND 10, AND THE ACB COMPOSITE C
C    SCORES TO FILES 9 AND 12.                                        C
C                                                                     C
C                                                                     C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
      SUBROUTINE OUT(SSAN1,SSAN2,SSAN3,AFQT,GSSSS,ARSSS,MKSSS,MCSSS,
     *EISSS,VESSS,GISSS,ADSSS,SPSSS,SISSS,AISSS,CMSSS,CASSS,CESSS,
     *CCSSS,NOSSS,ASSSS,WKSSS,CSSSS,PCSSS,TEST,CO,FA,MM,GM,CL,GT,
     *EL,SC,ST,OF,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,CC,WK,
     *PC,NO,CS,AS)
C
      INTEGER SSAN1,SSAN2,SSAN3,AFQT,GSSSS,ARSSS,MKSSS,MCSSS,CSSSS,
     *EISSS,VESSS,GISSS,ADSSS,SPSSS,SISSS,AISSS,CMSSS,CASSS,CESSS,
     *CCSSS,NOSSS,ASSSS,WKSSS,PCSSS,CO,FA,MM,GM,CL,GT,EL,SC,ST,OF,
     *TEST,GS,AR,MK,MC,EI,VE,GI,AD,SP,SI,AI,CM,CA,CE,CC,WK,PC,NO,CS,AS
C
C------------------
C     PRINT HEADINGS
C------------------
C
      WRITE(9,890)
C
C---------------------------------------
C     PRINT SUBTEST RAW AND STANDARD SCORES
C---------------------------------------
C
C
```

```
C-------------------
C     PRINT ASVAB 5
C-------------------
C
      IF(TEST .NE. 35) GO TO 265
         WRITE(9,874)
         WRITE(9,875) GISSS,NOSSS,ADSSS,WKSSS,ARSSS,SPSSS,MKSSS,EISSS,
     *   MCSSS,GSSSS,SISSS,AISSS
         WRITE(1C,876) SSAN1,SSAN2,SSAM3,GISSS,NOSSS,ADSSS,WKSSS,ARSSS,
     *   SPSSS,MKSSS,EISSS,MCSSS,GSSSS,SISSS,AISSS
         WRITE(11,876) SSAN1,SSAN2,SSAN3,GI,NO,AD,WK,AR,SP,MK,EI,MC,
     *   GS,SI,AI
      GO TO 275
C
C---------------------
C     PRINT ASVAB 6/7
C---------------------
C
265   IF(TEST .GT. 37) GO TO 270
         WRITE(9,879)
         WRITE(9,880) GISSS,NOSSS,ADSSS,WKSSS,ARSSS,SPSSS,MKSSS,EISSS,
     *   MCSSS,GSSSS,SISSS,AISSS,CMSSS,CASSS,CESSS,CCSSS
         WRITE(10,881) SSAN1,SSAN2,SSAN3,GISSS,NOSSS,ADSSS,WKSSS,ARSSS,
     *   SPSSS,MKSSS,EISSS,MCSSS,GSSSS,SISSS,AISSS,CMSSS,CASSS,CESSS,
     *   CCSSS
         WRITE(11,881) SSAN1,SSAN2,SSAN3,GI,NO,AD,WK,AR,SP,MK,EI,MC,
     *   GS,SI,AI,CM,CA,CE,CC
      GO TO 275
C
C------------------------
C     PRINT ASVAB 8/9/10
C------------------------
C
270   CONTINUE
         WRITE(9,884)
         WRITE(9,885) GSSSS,ARSSS,WKSSS,PCSSS,NOSSS,CSSSS,ASSSS,MKSSS,
     *   MCSSS,EISSS,VESSS
         WRITE(10,886) SSAN1,SSAN2,SSAN3,NOSSS,WKSSS,ARSSS,MKSSS,EISSS,
     *   MCSSS,GSSSS,PCSSS,CSSSS,ASSSS,VESSS
         WRITE(11,886) SSAN1,SSAN2,SSAN3,GS,AR,WK,PC,NO,CS,AS,MK,
     *   MC,EI,VE
275   CONTINUE
C
C-------------------------------
C     PRINT ACB CCMPOSITE SCORES
C-------------------------------
C
      WRITE(9,910)
```

```
      WRITE(9,915)
      WRITE(9,920) CO,FA,MM,GM,CL,GT,EL,SC,ST,OF
      WRITE(12,921) SSAN1,SSAN2,SSAN3,AFQT,CO,FA,MM,GM,CL,GT,EL,SC,
     *ST,OF
      RETURN
C
874   FORMAT('    GI   NO   AD   WK   AR   SP   MK   EI   MC   GS   SI   AI')
875   FORMAT(1X,12(2X,I2))
876   FORMAT(1X,3A3,3X,12(I2,1X))
879   FORMAT('    GI   NO   AD   WK   AR   SP   MK   EI   MC   GS   SI   AI   CM   CA
     * CE   CC')
880   FORMAT(1X,16(2X,I2))
881   FORMAT(1X,3A3,3X,16(I2,1X))
884   FCRMAT('    GS   AR   WK   PC   NO   CS   AS   MK   MC   EI   VE')
885   FORMAT(1X,11(2X,I2))
886   FORMAT(1X,3A3,6X,I2,4X,2(I2,1X),3X,4(I2,1X),18X,4(I2,1X))
890   FORMAT(/12X,'SUBTEST STANDARD SCORES')
910   FORMAT(/6X,'ARMY CLASSIFICATION BATTERY COMPOSITES')
915   FORMAT('   CO    FA    MM    GM    CL    GT    EL    SC    ST    OF')
920   FORMAT(1X,10(I3,2X))
921   FORMAT(1X,3A3,6X,11(I3,1X))
      END
```

BIBLIOGRAPHY

Graham, N., Introduction to Computer Science A Structured Approach, West Publishing Company, 1979.

Hussain, D. and Hussain, K.M., Information Processing Systems for Management, Richard D. Irwin, Incorporated, 1981.

Lipshutz, S. and Poe, A., Theory and Problems of Programming with FORTRAN, McGraw-Hill, 1978.

Nie, N., and others, Statistical Package for the Social Sciences, 2d Ed., McGraw-Hill, 1975.

Sperry Univac, Query Language Processor (QLP 1100) Level 4R1 Application Development Facilities, 1980.

Sperry Univac, Query Language Processor (QLP 1100) Level 4R1 Conversational Commands, 1980.

Sperry Univac, Query Language Processor (QLP 1100) Level 4R1 File and Application Definition Processors, 1980.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
3.	Commandant U.S. Army Infantry School Attn: ATSH-ES Fort Benning, Georgia 31905	1
4.	LTC. Stephen Paek, Code 55Ph Naval Postgraduate School Monterey, California 93943	1
5.	Captain Robert Forman 6550 Big Creek Parkway Parma Heights, Ohio 44130	1