1983-09

A management information system for the analysis of the Armed Services Aptitude Battery

Forman, Robert Joseph

Monterey, California. Naval Postgraduate School

https://hdl.handle.net/10945/19873

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun
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.
**Title:** A Management Information System for the Analysis of the Armed Services Aptitude Battery  

**Author:** Robert Joseph Forman  

**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.
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
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.
# TABLE OF CONTENTS

## I. INTRODUCTION
- **A. PURPOSE** ................................................................. 9
- **B. PROBLEM** ............................................................... 9
- **C. FEASIBILITY** ......................................................... 10
- **D. SYSTEM DESIGN** .................................................... 10

## II. SYSTEMS MANUAL
- **A. PURPOSE** ................................................................. 13
- **B. INTRODUCTION** ............................................................. 13
- **C. THE ASVAB 5 PROCEDURE** ........................................ 18
- **D. THE ASVAB 6 AND ASVAB 7 PROCEDURE** ....................... 23
- **E. THE ASVAB 8, ASVAB 9, AND ASVAB 10 PROCEDURE** ........ 28

## III. USER'S MANUAL
- **A. PURPOSE** ................................................................. 33
- **B. INTRODUCTION** ............................................................ 33
- **C. THE INPUT FILES** ...................................................... 33
- **D. THE ALGORITHM** ........................................................ 35
- **E. THE OUTPUT FILES** ..................................................... 47

## IV. SPSS AND QLP 1100
- **A. PURPOSE** ................................................................. 49
- **B. INTRODUCTION** ............................................................ 49
- **C. SPSS** ................................................................. 50
- **D. QLP 1100** ............................................................... 54
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACB</td>
<td>Army Classification Battery</td>
</tr>
<tr>
<td>AD</td>
<td>Attention To Detail</td>
</tr>
<tr>
<td>ADP</td>
<td>Application Definition Processor</td>
</tr>
<tr>
<td>AI</td>
<td>Automotive Information</td>
</tr>
<tr>
<td>AR</td>
<td>Arithmetic Reasoning</td>
</tr>
<tr>
<td>AS</td>
<td>Automotive/Shop</td>
</tr>
<tr>
<td>ASVAB</td>
<td>Armed Services Vocational Aptitude Battery</td>
</tr>
<tr>
<td>ASVABMIS</td>
<td>Armed Services Vocational Aptitude Battery Management Information System</td>
</tr>
<tr>
<td>CA</td>
<td>Attentiveness</td>
</tr>
<tr>
<td>CC</td>
<td>Combat</td>
</tr>
<tr>
<td>CE</td>
<td>Electronics</td>
</tr>
<tr>
<td>CL</td>
<td>Clerical</td>
</tr>
<tr>
<td>CM</td>
<td>Maintenance</td>
</tr>
<tr>
<td>CO</td>
<td>Combat</td>
</tr>
<tr>
<td>CS</td>
<td>Coding Speed</td>
</tr>
<tr>
<td>DOES</td>
<td>Department of Evaluation and Standardization</td>
</tr>
<tr>
<td>EI</td>
<td>Electronics Information</td>
</tr>
<tr>
<td>EL</td>
<td>Electronics Repair</td>
</tr>
<tr>
<td>FA</td>
<td>Field Artillery</td>
</tr>
<tr>
<td>FDP</td>
<td>File Definition Processor</td>
</tr>
<tr>
<td>GI</td>
<td>General Information</td>
</tr>
<tr>
<td>GM</td>
<td>General Maintenance</td>
</tr>
<tr>
<td>GS</td>
<td>General Science</td>
</tr>
<tr>
<td>Acronym</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>GT</td>
<td>General Technical</td>
</tr>
<tr>
<td>MC</td>
<td>Mechanical Comprehension</td>
</tr>
<tr>
<td>MK</td>
<td>Math Knowledge</td>
</tr>
<tr>
<td>MM</td>
<td>Motor Maintenance</td>
</tr>
<tr>
<td>NO</td>
<td>Numerical Operations</td>
</tr>
<tr>
<td>OF</td>
<td>Operators and Food Handlers</td>
</tr>
<tr>
<td>PC</td>
<td>Paragraph Comprehension</td>
</tr>
<tr>
<td>PCIOS</td>
<td>Processor Common Input/Output System</td>
</tr>
<tr>
<td>QLP</td>
<td>Query Language Processor</td>
</tr>
<tr>
<td>SC</td>
<td>Surveillance and Communications</td>
</tr>
<tr>
<td>SI</td>
<td>Shop Information</td>
</tr>
<tr>
<td>SP</td>
<td>Space Perception</td>
</tr>
<tr>
<td>SPSS</td>
<td>Statistical Package for the Social Sciences</td>
</tr>
<tr>
<td>SQT</td>
<td>Skill Qualification Test</td>
</tr>
<tr>
<td>SSS</td>
<td>Subtest Standard Score</td>
</tr>
<tr>
<td>ST</td>
<td>Skilled Technician</td>
</tr>
<tr>
<td>VE</td>
<td>Verbal</td>
</tr>
<tr>
<td>WK</td>
<td>Word Knowledge</td>
</tr>
</tbody>
</table>
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 order to 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

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

<table>
<thead>
<tr>
<th>04</th>
<th>E4</th>
</tr>
</thead>
<tbody>
<tr>
<td>05</td>
<td>E5</td>
</tr>
<tr>
<td>06</td>
<td>E6</td>
</tr>
<tr>
<td>07</td>
<td>E7</td>
</tr>
<tr>
<td>08</td>
<td>E8</td>
</tr>
<tr>
<td>09</td>
<td>E9</td>
</tr>
</tbody>
</table>

28-32 Military Occupation Specialty

33-34 Test Form:

<table>
<thead>
<tr>
<th>35</th>
<th>ASVAB 5.</th>
</tr>
</thead>
<tbody>
<tr>
<td>36</td>
<td>ASVAB 6.</td>
</tr>
<tr>
<td>37</td>
<td>ASVAB 7.</td>
</tr>
<tr>
<td>38</td>
<td>ASVAB 8.</td>
</tr>
<tr>
<td>39</td>
<td>ASVAB 9.</td>
</tr>
<tr>
<td>40</td>
<td>ASVAB 10.</td>
</tr>
</tbody>
</table>

35-36 AFQT Percentile

37-68 Subtest Raw Scores:

<table>
<thead>
<tr>
<th>ASVAB 5</th>
<th>ASVAB 6/7</th>
<th>ASVAB 8/9/10</th>
</tr>
</thead>
<tbody>
<tr>
<td>37-38</td>
<td>GI</td>
<td>GI</td>
</tr>
<tr>
<td>39-40</td>
<td>NO</td>
<td>NO</td>
</tr>
<tr>
<td>41-42</td>
<td>AD</td>
<td>AD</td>
</tr>
<tr>
<td>43-44</td>
<td>WK</td>
<td>WK</td>
</tr>
<tr>
<td>45-46</td>
<td>AR</td>
<td>AR</td>
</tr>
<tr>
<td>47-48</td>
<td>SP</td>
<td>SP</td>
</tr>
<tr>
<td>49-50</td>
<td>MK</td>
<td>MK</td>
</tr>
<tr>
<td>51-52</td>
<td>EI</td>
<td>EI</td>
</tr>
<tr>
<td>53-54</td>
<td>MC</td>
<td>MC</td>
</tr>
<tr>
<td>55-56</td>
<td>GS</td>
<td>GS</td>
</tr>
</tbody>
</table>
TABLE 1 (CONT.)

<table>
<thead>
<tr>
<th>Year</th>
<th>Designation</th>
<th>Designation</th>
<th>Designation</th>
</tr>
</thead>
<tbody>
<tr>
<td>57-58</td>
<td>SI</td>
<td>SI</td>
<td>VE</td>
</tr>
<tr>
<td>59-60</td>
<td>AI</td>
<td>AI</td>
<td>BLANK</td>
</tr>
<tr>
<td>61-62</td>
<td>BLANK</td>
<td>MC</td>
<td>BLANK</td>
</tr>
<tr>
<td>63-64</td>
<td>BLANK</td>
<td>CA</td>
<td>BLANK</td>
</tr>
<tr>
<td>65-66</td>
<td>BLANK</td>
<td>CE</td>
<td>BLANK</td>
</tr>
<tr>
<td>67-68</td>
<td>BLANK</td>
<td>CC</td>
<td>BLANK</td>
</tr>
<tr>
<td>71-73</td>
<td>SEPARATION</td>
<td>PROGRAM</td>
<td>DESIGNATOR.</td>
</tr>
<tr>
<td>74-79</td>
<td>DATE</td>
<td>OF</td>
<td>SEPARATION.</td>
</tr>
</tbody>
</table>
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>
<thead>
<tr>
<th>SUBTEST (Variable Name)</th>
<th>SCORE (Data Point)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GI</td>
<td>15</td>
</tr>
<tr>
<td>NO</td>
<td>20</td>
</tr>
<tr>
<td>AD</td>
<td>03</td>
</tr>
<tr>
<td>WK</td>
<td>12</td>
</tr>
<tr>
<td>AR</td>
<td>06</td>
</tr>
<tr>
<td>SP</td>
<td>17</td>
</tr>
<tr>
<td>MK</td>
<td>20</td>
</tr>
<tr>
<td>EI</td>
<td>10</td>
</tr>
<tr>
<td>MC</td>
<td>09</td>
</tr>
<tr>
<td>GS</td>
<td>05</td>
</tr>
<tr>
<td>SI</td>
<td>13</td>
</tr>
<tr>
<td>AI</td>
<td>12</td>
</tr>
<tr>
<td>RAW SCORE</td>
<td>SUBTEST</td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
</tr>
<tr>
<td>15</td>
<td>GI</td>
</tr>
<tr>
<td>20</td>
<td>NO</td>
</tr>
<tr>
<td>03</td>
<td>AD</td>
</tr>
<tr>
<td>12</td>
<td>WK</td>
</tr>
<tr>
<td>06</td>
<td>AR</td>
</tr>
<tr>
<td>17</td>
<td>SP</td>
</tr>
<tr>
<td>20</td>
<td>MK</td>
</tr>
<tr>
<td>10</td>
<td>EI</td>
</tr>
<tr>
<td>09</td>
<td>MC</td>
</tr>
<tr>
<td>05</td>
<td>GS</td>
</tr>
<tr>
<td>13</td>
<td>SI</td>
</tr>
<tr>
<td>12</td>
<td>AI</td>
</tr>
</tbody>
</table>
TABLE 4

ASVAB 5 ACB EQUATIONS

<table>
<thead>
<tr>
<th>EQUATION</th>
<th>SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO = AR + SI + SP + AD</td>
<td>164</td>
</tr>
<tr>
<td>FA = AR + GI + MK + EI</td>
<td>203</td>
</tr>
<tr>
<td>MM = MK + SI + EI + AI</td>
<td>199</td>
</tr>
<tr>
<td>GM = AR + GS + MC + AI</td>
<td>168</td>
</tr>
<tr>
<td>CL = AR + WK + AD</td>
<td>96</td>
</tr>
<tr>
<td>GT = AR + WK</td>
<td>75</td>
</tr>
<tr>
<td>EL = AR + EI + NC + SI</td>
<td>164</td>
</tr>
<tr>
<td>SC = AR + WK + MC + SP</td>
<td>180</td>
</tr>
<tr>
<td>ST = AR + MK + GS</td>
<td>139</td>
</tr>
<tr>
<td>OF = GI + AI</td>
<td>116</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>ACB SCORE</th>
<th>CLASSIFICATION</th>
<th>COMPOSITE SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>164</td>
<td>CO</td>
<td>71</td>
</tr>
<tr>
<td>203</td>
<td>FA</td>
<td>102</td>
</tr>
<tr>
<td>199</td>
<td>MM</td>
<td>99</td>
</tr>
<tr>
<td>168</td>
<td>GM</td>
<td>79</td>
</tr>
<tr>
<td>96</td>
<td>CL</td>
<td>49</td>
</tr>
<tr>
<td>75</td>
<td>GT</td>
<td>71</td>
</tr>
<tr>
<td>164</td>
<td>EL</td>
<td>76</td>
</tr>
<tr>
<td>180</td>
<td>SC</td>
<td>89</td>
</tr>
<tr>
<td>139</td>
<td>ST</td>
<td>94</td>
</tr>
<tr>
<td>116</td>
<td>OF</td>
<td>116</td>
</tr>
</tbody>
</table>

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 no 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>
<thead>
<tr>
<th>SUBTEST (Variable Name)</th>
<th>SCORE (Data Point)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GI</td>
<td>13</td>
</tr>
<tr>
<td>NO</td>
<td>40</td>
</tr>
<tr>
<td>AD</td>
<td>25</td>
</tr>
<tr>
<td>WK</td>
<td>22</td>
</tr>
<tr>
<td>AR</td>
<td>15</td>
</tr>
<tr>
<td>SP</td>
<td>16</td>
</tr>
<tr>
<td>MK</td>
<td>12</td>
</tr>
<tr>
<td>EI</td>
<td>23</td>
</tr>
<tr>
<td>MC</td>
<td>10</td>
</tr>
<tr>
<td>GS</td>
<td>09</td>
</tr>
<tr>
<td>SI</td>
<td>17</td>
</tr>
<tr>
<td>AI</td>
<td>14</td>
</tr>
<tr>
<td>CM</td>
<td>08</td>
</tr>
<tr>
<td>CA</td>
<td>12</td>
</tr>
<tr>
<td>CE</td>
<td>13</td>
</tr>
<tr>
<td>CC</td>
<td>22</td>
</tr>
</tbody>
</table>
### TABLE 7

ASVAB 6/7 ACB EQUATIONS

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

### TABLE 8

ASVAB 6/7 ACB COMPOSITE SCORES

<table>
<thead>
<tr>
<th>ACB SCORE</th>
<th>CLASSIFICATION</th>
<th>STANDARD SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>95</td>
<td>CO</td>
<td>125</td>
</tr>
<tr>
<td>75</td>
<td>FA</td>
<td>115</td>
</tr>
<tr>
<td>74</td>
<td>MM</td>
<td>106</td>
</tr>
<tr>
<td>48</td>
<td>GM</td>
<td>103</td>
</tr>
<tr>
<td>74</td>
<td>CL</td>
<td>122</td>
</tr>
<tr>
<td>37</td>
<td>GT</td>
<td>109</td>
</tr>
<tr>
<td>78</td>
<td>EL</td>
<td>114</td>
</tr>
<tr>
<td>63</td>
<td>SC</td>
<td>108</td>
</tr>
<tr>
<td>36</td>
<td>ST</td>
<td>104</td>
</tr>
<tr>
<td>39</td>
<td>OF</td>
<td>115</td>
</tr>
</tbody>
</table>
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

<table>
<thead>
<tr>
<th>GI</th>
<th>NO</th>
<th>AD</th>
<th>WK</th>
<th>AR</th>
<th>SP</th>
<th>MK</th>
<th>EI</th>
<th>MC</th>
<th>GS</th>
<th>SI</th>
<th>AI</th>
<th>CM</th>
<th>CA</th>
<th>CE</th>
<th>CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>40</td>
<td>25</td>
<td>22</td>
<td>15</td>
<td>16</td>
<td>12</td>
<td>23</td>
<td>10</td>
<td>09</td>
<td>17</td>
<td>14</td>
<td>08</td>
<td>12</td>
<td>13</td>
<td>22</td>
</tr>
</tbody>
</table>

SUBTEST STANDARD SCORES

<table>
<thead>
<tr>
<th>GI</th>
<th>NO</th>
<th>AD</th>
<th>WK</th>
<th>AR</th>
<th>SP</th>
<th>MK</th>
<th>EI</th>
<th>MC</th>
<th>GS</th>
<th>SI</th>
<th>AI</th>
<th>CM</th>
<th>CA</th>
<th>CE</th>
<th>CC</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>59</td>
<td>77</td>
<td>53</td>
<td>55</td>
<td>57</td>
<td>51</td>
<td>56</td>
<td>48</td>
<td>45</td>
<td>58</td>
<td>55</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
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>
<thead>
<tr>
<th>RAW SCORE (Data Point)</th>
<th>SUBTEST (Variable Name)</th>
<th>STANDARD SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>24</td>
<td>GS</td>
<td>65</td>
</tr>
<tr>
<td>22</td>
<td>AR</td>
<td>56</td>
</tr>
<tr>
<td>30</td>
<td>WK</td>
<td>56</td>
</tr>
<tr>
<td>14</td>
<td>PC</td>
<td>60</td>
</tr>
<tr>
<td>45</td>
<td>NO</td>
<td>59</td>
</tr>
<tr>
<td>79</td>
<td>CS</td>
<td>72</td>
</tr>
<tr>
<td>25</td>
<td>AS</td>
<td>65</td>
</tr>
<tr>
<td>20</td>
<td>MK</td>
<td>63</td>
</tr>
<tr>
<td>24</td>
<td>MC</td>
<td>65</td>
</tr>
<tr>
<td>19</td>
<td>EI</td>
<td>65</td>
</tr>
<tr>
<td>48</td>
<td>VE</td>
<td>61</td>
</tr>
</tbody>
</table>
TABLE 10

ASVAB 8/9/10 ACB EQUATIONS

<table>
<thead>
<tr>
<th>EQUATION</th>
<th>SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO = AR + AS + MC + CS</td>
<td>258</td>
</tr>
<tr>
<td>FA = AR + MK + MC + CS</td>
<td>256</td>
</tr>
<tr>
<td>MM = NO + EI + MC + AS</td>
<td>254</td>
</tr>
<tr>
<td>GM = MK + EI + GS + AS</td>
<td>258</td>
</tr>
<tr>
<td>CL = NO + CS + VE</td>
<td>192</td>
</tr>
<tr>
<td>GT = VE + AR</td>
<td>117</td>
</tr>
<tr>
<td>EL = AR + EI + MK + GS</td>
<td>249</td>
</tr>
<tr>
<td>SC = NO + CS + VE + AS</td>
<td>257</td>
</tr>
<tr>
<td>ST = VE + MK + MC + GS</td>
<td>254</td>
</tr>
<tr>
<td>OF = NO + VE + MC + AS</td>
<td>250</td>
</tr>
</tbody>
</table>
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**

<table>
<thead>
<tr>
<th>ACB SCORE</th>
<th>CLASSIFICATION</th>
<th>STANDARD SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>258</td>
<td>CO</td>
<td>138</td>
</tr>
<tr>
<td>256</td>
<td>FA</td>
<td>128</td>
</tr>
<tr>
<td>254</td>
<td>MM</td>
<td>138</td>
</tr>
<tr>
<td>258</td>
<td>GM</td>
<td>132</td>
</tr>
<tr>
<td>192</td>
<td>CL</td>
<td>133</td>
</tr>
<tr>
<td>117</td>
<td>GT</td>
<td>117</td>
</tr>
<tr>
<td>249</td>
<td>EL</td>
<td>124</td>
</tr>
<tr>
<td>257</td>
<td>SC</td>
<td>144</td>
</tr>
<tr>
<td>254</td>
<td>ST</td>
<td>128</td>
</tr>
<tr>
<td>250</td>
<td>OF</td>
<td>135</td>
</tr>
</tbody>
</table>

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
```
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>
<thead>
<tr>
<th>ITEM</th>
<th>MIN. VALUE</th>
<th>MAX. VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>MONTH (MM)</td>
<td>1</td>
<td>12</td>
</tr>
<tr>
<td>DAY (DD)</td>
<td>1</td>
<td>31</td>
</tr>
<tr>
<td>SEX</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>RACE</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>ED</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>TEST</td>
<td>35</td>
<td>40</td>
</tr>
</tbody>
</table>

**ASVAB 5/6/7**

<table>
<thead>
<tr>
<th>ITEM</th>
<th>MIN. VALUE</th>
<th>MAX. VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>GI</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>NO</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>AD</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>WK</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>AR</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>SP</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>MK</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>EI</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>MC</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>GS</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>SI</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>AI</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>CM</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>CA</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>CE</td>
<td>0</td>
<td>20</td>
</tr>
<tr>
<td>CC</td>
<td>0</td>
<td>27</td>
</tr>
<tr>
<td>ITEM</td>
<td>MIN. VALUE</td>
<td>MAX. VALUE</td>
</tr>
<tr>
<td>----------</td>
<td>------------</td>
<td>------------</td>
</tr>
<tr>
<td>ASVAB 8/9/10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>GS</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>AR</td>
<td>0</td>
<td>30</td>
</tr>
<tr>
<td>WK</td>
<td>0</td>
<td>35</td>
</tr>
<tr>
<td>PC</td>
<td>0</td>
<td>15</td>
</tr>
<tr>
<td>NO</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>CS</td>
<td>0</td>
<td>84</td>
</tr>
<tr>
<td>AS</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>MK</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>MC</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>EI</td>
<td>0</td>
<td>25</td>
</tr>
<tr>
<td>VE</td>
<td>0</td>
<td>50</td>
</tr>
</tbody>
</table>
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
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

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

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

\[
GISSS = \text{CONVT}(GI, \text{XBAR}, \text{SIGMA})
\]

An actual computation is given below.

\[
GISSS = \text{CONVT}(10, 9.656, 3.273)
\]

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

\[
GISSS = 51.05
\]

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 sub-test 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 x 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**

<table>
<thead>
<tr>
<th>n</th>
<th>SCORE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GT</td>
</tr>
<tr>
<td>2</td>
<td>GM</td>
</tr>
<tr>
<td>3</td>
<td>EL</td>
</tr>
<tr>
<td>4</td>
<td>CL</td>
</tr>
<tr>
<td>5</td>
<td>MM</td>
</tr>
<tr>
<td>6</td>
<td>SC</td>
</tr>
<tr>
<td>7</td>
<td>CO</td>
</tr>
<tr>
<td>8</td>
<td>FA</td>
</tr>
<tr>
<td>9</td>
<td>OF</td>
</tr>
<tr>
<td>10</td>
<td>ST</td>
</tr>
</tbody>
</table>

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:

\[
\begin{align*}
GT &= A8910(GT,1) \\
GM &= A8910(GM,2) \\
EL &= A8910(EL,3) \\
CL &= A8910(CL,4)
\end{align*}
\]
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
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
```

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 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. File Definition Processor Source Input

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**

<table>
<thead>
<tr>
<th>CHARACTER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A character position which can only contain a letter of the alphabet or a space.</td>
</tr>
<tr>
<td>S</td>
<td>A character position used to denote the presence of an operational sign, i.e., plus or minus.</td>
</tr>
<tr>
<td>V</td>
<td>A character position used to denote the location of a decimal point.</td>
</tr>
<tr>
<td>X</td>
<td>A character position which contains any allowable character from the computer's character set, i.e., $, /.</td>
</tr>
<tr>
<td>9</td>
<td>A character representing a numeral.</td>
</tr>
</tbody>
</table>
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 DIVISION
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
```
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>
<thead>
<tr>
<th>ELEMENT</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>file-name</td>
<td>Is the name of the file containing the absolute FDP or ADP.</td>
</tr>
<tr>
<td>options</td>
<td>Are specified through the choice of one or more option letters.</td>
</tr>
<tr>
<td>D</td>
<td>Produce an allocation summary for each record in the file. This may be used only on the ADP.</td>
</tr>
<tr>
<td>I</td>
<td>Insert a new symbolic element in the program file.</td>
</tr>
<tr>
<td>P</td>
<td>Card image input, if any, is in fielddata. The output symbolic element is in fielddata.</td>
</tr>
<tr>
<td>Q</td>
<td>Output symbolic element is in ACIII. Card image input, if any, is in ASCII.</td>
</tr>
<tr>
<td>S</td>
<td>The source input will be echoed to the user.</td>
</tr>
<tr>
<td>U</td>
<td>Update and produce a new cycle of the symbolic element.</td>
</tr>
<tr>
<td>W</td>
<td>List correction lines.</td>
</tr>
</tbody>
</table>
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>
<thead>
<tr>
<th>PARAMETER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>SI</td>
<td>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.</td>
</tr>
<tr>
<td>RO</td>
<td>This parameter specifies the name and program file into which the element produced by the processor is placed.</td>
</tr>
<tr>
<td>SO</td>
<td>This parameter specifies the name and the file for the updated symbolic element produced.</td>
</tr>
</tbody>
</table>

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*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. **Query Language Processor, 1100 Series**

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 value 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.
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.

<table>
<thead>
<tr>
<th>SSAN</th>
<th>PERCENTILE</th>
</tr>
</thead>
<tbody>
<tr>
<td>242968813</td>
<td>92</td>
</tr>
<tr>
<td>016558905</td>
<td>96</td>
</tr>
</tbody>
</table>

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. Summary Example

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.

```plaintext
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.

<table>
<thead>
<tr>
<th>SSAN</th>
<th>SQT</th>
<th>AFQT</th>
</tr>
</thead>
<tbody>
<tr>
<td>232489961</td>
<td>85</td>
<td>60</td>
</tr>
<tr>
<td>016558843</td>
<td>75</td>
<td>55</td>
</tr>
<tr>
<td>114863105</td>
<td>75</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>404358221</td>
<td>51</td>
<td>45</td>
</tr>
</tbody>
</table>

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
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 B

### ASVAB 5/6/7 SSS CONVERSION TABLE

<table>
<thead>
<tr>
<th>GI</th>
<th>WK</th>
<th>AR</th>
<th>SP</th>
<th>AD</th>
<th>MK</th>
<th>EI</th>
<th>MC</th>
<th>GS</th>
<th>AI</th>
<th>SI</th>
<th>VJ</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>20</td>
<td>23</td>
<td>23</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
<td>20</td>
</tr>
<tr>
<td>1</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>2</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
<td>27</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>4</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
<td>33</td>
</tr>
<tr>
<td>5</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
<td>36</td>
</tr>
<tr>
<td>7</td>
<td>42</td>
<td>42</td>
<td>42</td>
<td>42</td>
<td>42</td>
<td>42</td>
<td>42</td>
<td>42</td>
<td>42</td>
<td>42</td>
<td>42</td>
</tr>
<tr>
<td>8</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>9</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
<td>48</td>
</tr>
<tr>
<td>10</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
<td>51</td>
</tr>
<tr>
<td>11</td>
<td>54</td>
<td>54</td>
<td>54</td>
<td>54</td>
<td>54</td>
<td>54</td>
<td>54</td>
<td>54</td>
<td>54</td>
<td>54</td>
<td>54</td>
</tr>
<tr>
<td>12</td>
<td>57</td>
<td>57</td>
<td>57</td>
<td>57</td>
<td>57</td>
<td>57</td>
<td>57</td>
<td>57</td>
<td>57</td>
<td>57</td>
<td>57</td>
</tr>
<tr>
<td>13</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
<td>60</td>
</tr>
<tr>
<td>14</td>
<td>63</td>
<td>63</td>
<td>63</td>
<td>63</td>
<td>63</td>
<td>63</td>
<td>63</td>
<td>63</td>
<td>63</td>
<td>63</td>
<td>63</td>
</tr>
<tr>
<td>15</td>
<td>66</td>
<td>66</td>
<td>66</td>
<td>66</td>
<td>66</td>
<td>66</td>
<td>66</td>
<td>66</td>
<td>66</td>
<td>66</td>
<td>66</td>
</tr>
<tr>
<td>16</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>69</td>
<td>69</td>
</tr>
<tr>
<td>17</td>
<td>72</td>
<td>72</td>
<td>72</td>
<td>72</td>
<td>72</td>
<td>72</td>
<td>72</td>
<td>72</td>
<td>72</td>
<td>72</td>
<td>72</td>
</tr>
<tr>
<td>18</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>75</td>
</tr>
<tr>
<td>19</td>
<td>78</td>
<td>78</td>
<td>78</td>
<td>78</td>
<td>78</td>
<td>78</td>
<td>78</td>
<td>78</td>
<td>78</td>
<td>78</td>
<td>78</td>
</tr>
<tr>
<td>20</td>
<td>81</td>
<td>81</td>
<td>81</td>
<td>81</td>
<td>81</td>
<td>81</td>
<td>81</td>
<td>81</td>
<td>81</td>
<td>81</td>
<td>81</td>
</tr>
<tr>
<td>21</td>
<td>84</td>
<td>84</td>
<td>84</td>
<td>84</td>
<td>84</td>
<td>84</td>
<td>84</td>
<td>84</td>
<td>84</td>
<td>84</td>
<td>84</td>
</tr>
<tr>
<td>22</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>87</td>
<td>87</td>
</tr>
<tr>
<td>23</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
<td>90</td>
</tr>
<tr>
<td>24</td>
<td>93</td>
<td>93</td>
<td>93</td>
<td>93</td>
<td>93</td>
<td>93</td>
<td>93</td>
<td>93</td>
<td>93</td>
<td>93</td>
<td>93</td>
</tr>
<tr>
<td>25</td>
<td>96</td>
<td>96</td>
<td>96</td>
<td>96</td>
<td>96</td>
<td>96</td>
<td>96</td>
<td>96</td>
<td>96</td>
<td>96</td>
<td>96</td>
</tr>
<tr>
<td>27</td>
<td>102</td>
<td>102</td>
<td>102</td>
<td>102</td>
<td>102</td>
<td>102</td>
<td>102</td>
<td>102</td>
<td>102</td>
<td>102</td>
<td>102</td>
</tr>
<tr>
<td>28</td>
<td>105</td>
<td>105</td>
<td>105</td>
<td>105</td>
<td>105</td>
<td>105</td>
<td>105</td>
<td>105</td>
<td>105</td>
<td>105</td>
<td>105</td>
</tr>
<tr>
<td>30</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
<td>111</td>
</tr>
<tr>
<td>31</td>
<td>114</td>
<td>114</td>
<td>114</td>
<td>114</td>
<td>114</td>
<td>114</td>
<td>114</td>
<td>114</td>
<td>114</td>
<td>114</td>
<td>114</td>
</tr>
<tr>
<td>32</td>
<td>117</td>
<td>117</td>
<td>117</td>
<td>117</td>
<td>117</td>
<td>117</td>
<td>117</td>
<td>117</td>
<td>117</td>
<td>117</td>
<td>117</td>
</tr>
<tr>
<td>33</td>
<td>120</td>
<td>120</td>
<td>120</td>
<td>120</td>
<td>120</td>
<td>120</td>
<td>120</td>
<td>120</td>
<td>120</td>
<td>120</td>
<td>120</td>
</tr>
<tr>
<td>34</td>
<td>123</td>
<td>123</td>
<td>123</td>
<td>123</td>
<td>123</td>
<td>123</td>
<td>123</td>
<td>123</td>
<td>123</td>
<td>123</td>
<td>123</td>
</tr>
<tr>
<td>35</td>
<td>126</td>
<td>126</td>
<td>126</td>
<td>126</td>
<td>126</td>
<td>126</td>
<td>126</td>
<td>126</td>
<td>126</td>
<td>126</td>
<td>126</td>
</tr>
<tr>
<td>36</td>
<td>129</td>
<td>129</td>
<td>129</td>
<td>129</td>
<td>129</td>
<td>129</td>
<td>129</td>
<td>129</td>
<td>129</td>
<td>129</td>
<td>129</td>
</tr>
<tr>
<td>37</td>
<td>132</td>
<td>132</td>
<td>132</td>
<td>132</td>
<td>132</td>
<td>132</td>
<td>132</td>
<td>132</td>
<td>132</td>
<td>132</td>
<td>132</td>
</tr>
<tr>
<td>38</td>
<td>135</td>
<td>135</td>
<td>135</td>
<td>135</td>
<td>135</td>
<td>135</td>
<td>135</td>
<td>135</td>
<td>135</td>
<td>135</td>
<td>135</td>
</tr>
<tr>
<td>40</td>
<td>141</td>
<td>141</td>
<td>141</td>
<td>141</td>
<td>141</td>
<td>141</td>
<td>141</td>
<td>141</td>
<td>141</td>
<td>141</td>
<td>141</td>
</tr>
<tr>
<td>41</td>
<td>144</td>
<td>144</td>
<td>144</td>
<td>144</td>
<td>144</td>
<td>144</td>
<td>144</td>
<td>144</td>
<td>144</td>
<td>144</td>
<td>144</td>
</tr>
<tr>
<td>42</td>
<td>147</td>
<td>147</td>
<td>147</td>
<td>147</td>
<td>147</td>
<td>147</td>
<td>147</td>
<td>147</td>
<td>147</td>
<td>147</td>
<td>147</td>
</tr>
<tr>
<td>43</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td>150</td>
<td>150</td>
</tr>
<tr>
<td>45</td>
<td>156</td>
<td>156</td>
<td>156</td>
<td>156</td>
<td>156</td>
<td>156</td>
<td>156</td>
<td>156</td>
<td>156</td>
<td>156</td>
<td>156</td>
</tr>
<tr>
<td>46</td>
<td>159</td>
<td>159</td>
<td>159</td>
<td>159</td>
<td>159</td>
<td>159</td>
<td>159</td>
<td>159</td>
<td>159</td>
<td>159</td>
<td>159</td>
</tr>
<tr>
<td>47</td>
<td>162</td>
<td>162</td>
<td>162</td>
<td>162</td>
<td>162</td>
<td>162</td>
<td>162</td>
<td>162</td>
<td>162</td>
<td>162</td>
<td>162</td>
</tr>
<tr>
<td>48</td>
<td>165</td>
<td>165</td>
<td>165</td>
<td>165</td>
<td>165</td>
<td>165</td>
<td>165</td>
<td>165</td>
<td>165</td>
<td>165</td>
<td>165</td>
</tr>
<tr>
<td>49</td>
<td>168</td>
<td>168</td>
<td>168</td>
<td>168</td>
<td>168</td>
<td>168</td>
<td>168</td>
<td>168</td>
<td>168</td>
<td>168</td>
<td>168</td>
</tr>
<tr>
<td>50</td>
<td>171</td>
<td>171</td>
<td>171</td>
<td>171</td>
<td>171</td>
<td>171</td>
<td>171</td>
<td>171</td>
<td>171</td>
<td>171</td>
<td>171</td>
</tr>
</tbody>
</table>

84
### APPENDIX C

#### ASVAB 8/9/10 SSS CONVERSION TABLE

<table>
<thead>
<tr>
<th>GS</th>
<th>AR</th>
<th>WK</th>
<th>PC</th>
<th>NO</th>
<th>CS</th>
<th>AS</th>
<th>MK</th>
<th>MC</th>
<th>EI</th>
<th>VE</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>20</td>
<td>21</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
<td>29</td>
</tr>
<tr>
<td>1</td>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
</tr>
<tr>
<td>2</td>
<td>40</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
</tr>
<tr>
<td>3</td>
<td>50</td>
<td>51</td>
<td>52</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td>59</td>
</tr>
<tr>
<td>4</td>
<td>60</td>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
<td>66</td>
<td>67</td>
<td>68</td>
<td>69</td>
</tr>
<tr>
<td>5</td>
<td>70</td>
<td>71</td>
<td>72</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td>76</td>
<td>77</td>
<td>78</td>
<td>79</td>
</tr>
<tr>
<td>6</td>
<td>80</td>
<td>81</td>
<td>82</td>
<td>83</td>
<td>84</td>
<td>85</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>20</th>
<th>21</th>
<th>22</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>26</th>
<th>27</th>
<th>28</th>
<th>29</th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td>31</td>
<td>32</td>
<td>33</td>
<td>34</td>
<td>35</td>
<td>36</td>
<td>37</td>
<td>38</td>
<td>39</td>
</tr>
<tr>
<td>40</td>
<td>41</td>
<td>42</td>
<td>43</td>
<td>44</td>
<td>45</td>
<td>46</td>
<td>47</td>
<td>48</td>
<td>49</td>
</tr>
<tr>
<td>50</td>
<td>51</td>
<td>52</td>
<td>53</td>
<td>54</td>
<td>55</td>
<td>56</td>
<td>57</td>
<td>58</td>
<td>59</td>
</tr>
<tr>
<td>60</td>
<td>61</td>
<td>62</td>
<td>63</td>
<td>64</td>
<td>65</td>
<td>66</td>
<td>67</td>
<td>68</td>
<td>69</td>
</tr>
<tr>
<td>70</td>
<td>71</td>
<td>72</td>
<td>73</td>
<td>74</td>
<td>75</td>
<td>76</td>
<td>77</td>
<td>78</td>
<td>79</td>
</tr>
<tr>
<td>80</td>
<td>81</td>
<td>82</td>
<td>83</td>
<td>84</td>
<td>85</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

85
## APPENDIX D

### ASVAB 5 COMPOSITE CONVERSION TABLE

<table>
<thead>
<tr>
<th>SS</th>
<th>GT</th>
<th>GM</th>
<th>EL</th>
<th>CL</th>
<th>MM</th>
<th>SC</th>
<th>CO</th>
<th>FA</th>
<th>OF</th>
<th>ST</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>2</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>3</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>4</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>5</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>6</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>7</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>8</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>9</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>10</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>11</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>12</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>13</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>14</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>15</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>16</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>17</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>18</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>19</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>20</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>21</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>22</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>23</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>24</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>25</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>26</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>27</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>28</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>29</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>30</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>31</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>32</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>33</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>34</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>35</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>36</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>37</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>38</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>39</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>40</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>42</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>43</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>44</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>45</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>46</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>47</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>48</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>49</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>50</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>51</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>52</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>53</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>54</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>55</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>56</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>57</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>58</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
<td>41</td>
</tr>
<tr>
<td>RAW</td>
<td>GT</td>
<td>GM</td>
<td>EL</td>
<td>CL</td>
<td>MM</td>
<td>SC</td>
<td>CO</td>
<td>FA</td>
<td>OF</td>
<td>ST</td>
</tr>
<tr>
<td>-----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
<td>----</td>
</tr>
<tr>
<td>0</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>1</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>2</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>3</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>4</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>5</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>6</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>7</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>8</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>9</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>10</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>11</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>12</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>13</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>14</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>15</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>16</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>17</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>18</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>19</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>20</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>21</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>22</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>23</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>24</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>26</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>27</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>28</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>29</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>30</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>31</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>32</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>33</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>34</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>35</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>36</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>37</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>38</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>40</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>41</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>42</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>43</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>44</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>45</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>46</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>47</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>48</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>49</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>50</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>51</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>52</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>54</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>55</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>56</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
<tr>
<td>57</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
<td>53</td>
</tr>
</tbody>
</table>

**APPENDIX E**

**ASVAB 6/7 COMPOSITE CONVERSION TABLE**
APPENDIX F

ASVAB 8/9/10 COMPOSITE CONVERSION TABLE

<table>
<thead>
<tr>
<th>SS</th>
<th>GT</th>
<th>GM</th>
<th>EL</th>
<th>CL</th>
<th>MM</th>
<th>SC</th>
<th>CO</th>
<th>FA</th>
<th>OF</th>
<th>ST</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
<td>40</td>
</tr>
</tbody>
</table>
APPENDIX G

ASVAB PROGRAM OUTPUT FILES

SSAN: 272598843
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 - 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: 11810
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 SEPARATION(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 CLASSIFICATION BATTERY COMPOSITES
CO FA MM GM CL GT EL SC ST OF
138 128 138 132 133 117 124 144 148 135
ERROR IN RECORD NUMBER: 4 WITH SSAN: 235442321
ERROR: TEST FORM INCORRECT, COLUMNS 33-34

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

SUBTEST RAW SCORES

<table>
<thead>
<tr>
<th>GS</th>
<th>AR</th>
<th>WK</th>
<th>PC</th>
<th>NO</th>
<th>CS</th>
<th>AS</th>
<th>MK</th>
<th>MC</th>
<th>EI</th>
<th>VE</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>21</td>
<td>29</td>
<td>13</td>
<td>44</td>
<td>72</td>
<td>20</td>
<td>20</td>
<td>21</td>
<td>18</td>
<td>23</td>
</tr>
</tbody>
</table>
**ERROR IN RECORD NUMBER:**  5  **WITH SSAN:**  015342237  
**ERROR:**  CC INCORRECT, COLUMNS 67-68

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSAN</td>
<td>015342237</td>
</tr>
<tr>
<td>DATE OF BIRTH(YY/MM/DD)</td>
<td>64/8/9</td>
</tr>
<tr>
<td>SEX</td>
<td>1 - MALE</td>
</tr>
<tr>
<td>RACE</td>
<td>1 - WHITE</td>
</tr>
<tr>
<td>DATE OF ENTRY(YY/MM/DD)</td>
<td>81/9/25</td>
</tr>
<tr>
<td>HIGHEST YEAR OF EDUCATION</td>
<td>HIGH SCHOOL DIPLOMA - 6</td>
</tr>
<tr>
<td>ENTRY PAY GRADE</td>
<td>E1 1</td>
</tr>
<tr>
<td>MOS</td>
<td>I1B10</td>
</tr>
<tr>
<td>37 TEST FORM</td>
<td>ASVAB 7</td>
</tr>
<tr>
<td>AFQT PERCENTILE</td>
<td>75</td>
</tr>
<tr>
<td>SEPARATION PROGRAM DESIGNATOR</td>
<td>AKL</td>
</tr>
<tr>
<td>DATE OF SEPARATION(YY/MM/DD)</td>
<td>84/9/15</td>
</tr>
</tbody>
</table>

**SUBTEST RAW SCORES**

<table>
<thead>
<tr>
<th>Subtest</th>
<th>GI</th>
<th>NO</th>
<th>AD</th>
<th>WK</th>
<th>AR</th>
<th>SP</th>
<th>MK</th>
<th>EI</th>
<th>MC</th>
<th>GS</th>
<th>SI</th>
<th>AI</th>
<th>CM</th>
<th>CA</th>
<th>CE</th>
<th>CC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>15</td>
<td>39</td>
<td>25</td>
<td>21</td>
<td>15</td>
<td>16</td>
<td>12</td>
<td>23</td>
<td>11</td>
<td>12</td>
<td>17</td>
<td>13</td>
<td>10</td>
<td>12</td>
<td>13</td>
<td>32</td>
</tr>
<tr>
<td>SSAN</td>
<td>SUBTEST RAW SCORES</td>
<td>GI</td>
<td>NC</td>
<td>AD</td>
<td>WK</td>
<td>AR</td>
<td>SP</td>
<td>MK</td>
<td>EI</td>
<td>MC</td>
<td>GS</td>
<td>SI</td>
<td>AI</td>
<td>CM</td>
<td>CA</td>
<td>GE</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>272598843</td>
<td>15</td>
<td>20</td>
<td>3</td>
<td>12</td>
<td>6</td>
<td>17</td>
<td>20</td>
<td>10</td>
<td>9</td>
<td>5</td>
<td>13</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>016532241</td>
<td>13</td>
<td>40</td>
<td>25</td>
<td>22</td>
<td>15</td>
<td>16</td>
<td>12</td>
<td>23</td>
<td>10</td>
<td>9</td>
<td>17</td>
<td>14</td>
<td>8</td>
<td>12</td>
<td>13</td>
<td>22</td>
</tr>
<tr>
<td>213865527</td>
<td>24</td>
<td>22</td>
<td>30</td>
<td></td>
<td>14</td>
<td>45</td>
<td>79</td>
<td>25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SSAN</td>
<td>SUBTEST STANDARD SCORES</td>
<td>GI</td>
<td>NC</td>
<td>AD</td>
<td>WK</td>
<td>AR</td>
<td>SP</td>
<td>MK</td>
<td>EI</td>
<td>MC</td>
<td>GS</td>
<td>SI</td>
<td>AI</td>
<td>CM</td>
<td>CA</td>
<td>GE</td>
</tr>
<tr>
<td>272598843</td>
<td>66</td>
<td>39</td>
<td>21</td>
<td>39</td>
<td>36</td>
<td>59</td>
<td>67</td>
<td>34</td>
<td>46</td>
<td>36</td>
<td>48</td>
<td>50</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>016532241</td>
<td>60</td>
<td>59</td>
<td>77</td>
<td>53</td>
<td>55</td>
<td>57</td>
<td>51</td>
<td>56</td>
<td>48</td>
<td>45</td>
<td>58</td>
<td>55</td>
<td>44</td>
<td>58</td>
<td>60</td>
<td>63</td>
</tr>
<tr>
<td>213865527</td>
<td>59</td>
<td>59</td>
<td>56</td>
<td>56</td>
<td>63</td>
<td>65</td>
<td>65</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SSAN</th>
<th>ARMY CLASSIFICATION BATTERY COMPOSITES</th>
<th>AFQT</th>
<th>CO</th>
<th>FA</th>
<th>MM</th>
<th>GM</th>
<th>CL</th>
<th>GT</th>
<th>EL</th>
<th>SC</th>
<th>ST</th>
<th>OF</th>
</tr>
</thead>
<tbody>
<tr>
<td>272598843</td>
<td></td>
<td>47</td>
<td>71</td>
<td>102</td>
<td>99</td>
<td>79</td>
<td>49</td>
<td>71</td>
<td>76</td>
<td>89</td>
<td>94</td>
<td>116</td>
</tr>
<tr>
<td>016532241</td>
<td></td>
<td>87</td>
<td>125</td>
<td>115</td>
<td>106</td>
<td>103</td>
<td>122</td>
<td>109</td>
<td>114</td>
<td>108</td>
<td>104</td>
<td>115</td>
</tr>
<tr>
<td>213865527</td>
<td></td>
<td>92</td>
<td>138</td>
<td>128</td>
<td>138</td>
<td>132</td>
<td>133</td>
<td>117</td>
<td>124</td>
<td>144</td>
<td>128</td>
<td>135</td>
</tr>
</tbody>
</table>
APPENDIX H

SPSS REGRESSION PROGRAM LISTING

DATA LIST
FIXEC/ 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
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/
- ST SURVEILLANCE, COMMUNICATIONS/
- OP OPERATOR AND FOOD HANDLER

#### COMPUTE
- SQT=3+(0.085*GT)

#### WRITE CASES
- \((F9.0, 3X, F5.0)\)
- SSAN, SQT

#### READ INPUT DATA
- 427310253
- 520683135
- 527724668
- 535883688
- 553177646
- 560138625
- 573555686
- 555393568
- 039360566
- 025292862
- 056444837
- 121545078
- 255043799
- 426150571
- 441707290
- 257044264
- 260252797
- 265533874
- 455548554
- 508888285
- 050852622
- 420829366
- 435106687
- 423547244
- 565374369
- 235665486
- 562254590

#### END INPUT DATA

#### FINISH
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

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>B</th>
<th>SE B</th>
<th>BETA</th>
<th>T</th>
<th>SIG T</th>
</tr>
</thead>
<tbody>
<tr>
<td>CL</td>
<td>0.16172</td>
<td>0.05820</td>
<td>0.47852</td>
<td>2.779</td>
<td>0.0100</td>
</tr>
<tr>
<td>(CONSTANT)</td>
<td>-7.73595</td>
<td>5.61678</td>
<td>0.47852</td>
<td>-1.377</td>
<td>0.1802</td>
</tr>
</tbody>
</table>

1SPSS BATCH SYSTEM

DEPENDENT VARIABLE: SQT SKILL QUALIFICATION TEST

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

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>B</th>
<th>SE B</th>
<th>BETA</th>
<th>T</th>
<th>SIG T</th>
</tr>
</thead>
<tbody>
<tr>
<td>MULTIPLE R</td>
<td>0.59073</td>
<td>0.34897</td>
<td>0.29688</td>
<td>2.018</td>
<td>0.0223</td>
</tr>
<tr>
<td>R SQUARE</td>
<td>0.34897</td>
<td>0.29688</td>
<td>2.018</td>
<td>0.0223</td>
<td></td>
</tr>
<tr>
<td>ADJUSTED R SQUARE</td>
<td>0.29688</td>
<td>2.018</td>
<td>0.0223</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STANDARD ERROR</td>
<td>3.94529</td>
<td>3.94529</td>
<td>3.94529</td>
<td>3.94529</td>
<td></td>
</tr>
</tbody>
</table>

ANALYSIS OF VARIANCE

DF | SUM OF SQUARES
---|-------------------
1  | 136.86306
26 | 460.85122

F = 7.72145
SIGNIF F = 0.0100

-- VARIABLES IN THE EQUATION --

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>B</th>
<th>SE B</th>
<th>BETA</th>
<th>T</th>
<th>SIG T</th>
</tr>
</thead>
<tbody>
<tr>
<td>CL</td>
<td>0.16172</td>
<td>0.05820</td>
<td>0.47852</td>
<td>2.779</td>
<td>0.0100</td>
</tr>
<tr>
<td>(CONSTANT)</td>
<td>-7.73595</td>
<td>5.61678</td>
<td>0.47852</td>
<td>-1.377</td>
<td>0.1802</td>
</tr>
</tbody>
</table>

1SPSS REGRESSION PROGRAM OUTPUT
\[
F = 6.70020 \quad \text{SIGNIF } F = 0.0047
\]

<table>
<thead>
<tr>
<th>VARIABLE</th>
<th>B</th>
<th>SE B</th>
<th>BETA</th>
<th>T</th>
<th>SIG T</th>
</tr>
</thead>
<tbody>
<tr>
<td>CL</td>
<td>0.29638</td>
<td>0.08313</td>
<td>0.87697</td>
<td>3.565</td>
<td>0.0015</td>
</tr>
<tr>
<td>GT</td>
<td>-0.15531</td>
<td>0.07235</td>
<td>-0.52797</td>
<td>-2.147</td>
<td>0.0417</td>
</tr>
<tr>
<td>(CONSTANT)</td>
<td>-5.34173</td>
<td>5.38036</td>
<td>-0.52797</td>
<td>-0.993</td>
<td>0.3303</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VARIAB</th>
<th>CO</th>
<th>FA</th>
<th>MM</th>
<th>GM</th>
<th>EL</th>
<th>SC</th>
<th>ST</th>
<th>OF</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>SSAN</th>
<th>SQT</th>
</tr>
</thead>
<tbody>
<tr>
<td>42731000</td>
<td>12</td>
</tr>
<tr>
<td>52068308</td>
<td>12</td>
</tr>
<tr>
<td>52771456</td>
<td>13</td>
</tr>
<tr>
<td>53980339</td>
<td>12</td>
</tr>
<tr>
<td>55317760</td>
<td>11</td>
</tr>
<tr>
<td>56013845</td>
<td>11</td>
</tr>
<tr>
<td>56617795</td>
<td>11</td>
</tr>
<tr>
<td>57355680</td>
<td>12</td>
</tr>
<tr>
<td>45553936</td>
<td>11</td>
</tr>
<tr>
<td>03936835</td>
<td>14</td>
</tr>
<tr>
<td>09252016</td>
<td>10</td>
</tr>
<tr>
<td>09644552</td>
<td>13</td>
</tr>
<tr>
<td>12154507</td>
<td>9</td>
</tr>
<tr>
<td>25504379</td>
<td>12</td>
</tr>
<tr>
<td>42615040</td>
<td>8</td>
</tr>
<tr>
<td>44170724</td>
<td>10</td>
</tr>
<tr>
<td>25704425</td>
<td>10</td>
</tr>
<tr>
<td>26025278</td>
<td>13</td>
</tr>
<tr>
<td>26553864</td>
<td>10</td>
</tr>
<tr>
<td>45564854</td>
<td>10</td>
</tr>
<tr>
<td>50888268</td>
<td>14</td>
</tr>
<tr>
<td>05052622</td>
<td>10</td>
</tr>
<tr>
<td>22082936</td>
<td>11</td>
</tr>
<tr>
<td>25108686</td>
<td>13</td>
</tr>
<tr>
<td>42394700</td>
<td>11</td>
</tr>
<tr>
<td>56537420</td>
<td>11</td>
</tr>
<tr>
<td>23556547</td>
<td>12</td>
</tr>
<tr>
<td>56225433</td>
<td>11</td>
</tr>
</tbody>
</table>

---

VARIABLES AND CONSTANTS

SSAN SOCIAL SECURITY ACCOUNT NUMBER
DOBYY YEAR OF BIRTH
DOBMM MONTH OF BIRTH
DOBDY DAY OF BIRTH
SEX SEX
RACE RACE
DEEY YEAR OF ENTRY
DEEMM MONTH OF ENTRY
DEDD DAY OF ENTRY
ED HIGHEST YEAR OF EDUCATION
GRADE ENTRY PAY GRADE
MOS MILITARY OCCUPATION SPECIALITY
TEST TEST FORM
AFQT ARMED FORCES QUALIFICATION PERCENTILE
GI GENERAL INFORMATION
NG NUMERICAL OPERATIONS
AD ATTENTION TO DETAIL
WK WORD KNOWLEDGE
AR ARITHMETIC REASONING
SP SPACE PERCEPTION
MK MATH KNOWLEDGE
EI ELECTRONIC INFORMATION
MC MECHANICAL COMPREHENSION
GS GENERAL SCIENCE
SH SHOP INFORMATION
AI AUTOMOTIVE INFORMATION
CM
MAINTENANCE.
CA
ATTENTIVENESS.
CE
ELECTRONICS.
CC
COMBAT.
PC
PARAGRAPH COMPREHENSION.
CS
CODING SPEED.
AS
AUTOMOTIVE/SHOP.
VE
VERBAL.
CO
COMBAT.
FA
FIELD ARTILLERY.
EL
ELECTRONICS.
OF
OPERATORS AND FOOD HANDLERS.
GM
GENERAL MAINTENANCE.
MM
MOTOR MAINTENANCE.
CL
CLERICAL.
ST
SKILLED TECHNICAL.
GT
GENERAL TECHNICAL.
SC
SURVEILLANCE/COMMUNICATION.
A5
ASVAB 5 CONVERSION TABLE.
A67
ASVAB 6/7 CONVERSION TABLE.
A8910
ASVAB 8/9/10 CONVERSION TABLE.
SSS
VARIABLE SUFFIX: SUBTEST STANDARD SCORE.
R
VARIABLE SUFFIX: VARIABLE AS A REAL NUMBER.
END
TRAILER VARIABLE

---------------FILE DEFINITIONS---------------
FILE 01 INPUT, ASVAB 5 COMPOSITE CONVERSION TABLE.
FILE 02 INPUT, ASVAB 6/7 COMPOSITE CONVERSION TABLE.
FILE 03 INPUT, ASVAB 8/9/10 COMPOSITE CONVERSION TABLE.
FILE 04 INPUT, THE DEMOGRAPHIC DATA AND THE RAW SCORES.
FILE 09 OUTPUT, THE SOLDIER FILE.
FILE 10 OUTPUT, THE ASVAB STANDARD SCORES.
FILE 11 OUTPUT, THE ASVAB RAW SCORES.
FILE 12 OUTPUT, THE ABC COMPOSITE SCORES.

---------------DECLARE VARIABLES---------------
INTEGER DOBYY, DOBMM, DOBDD, SEX, RACE, DOETY, DOEMM, DOEDD, ED,
*GRADE, TEST, GS, AR, WK, PC, NO, CS, AS, MK, MC, EI, VE, DE, ETSYY, EISMH,
*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, GS55,
*ARSSS, MKSSS, PCSSS, NGSSS, CSSSS, ASSSS, MKSSS, MCSSS, EISSS, VESSS, GISSS,
*ADSSS, SPSSS, SISSS, AISSS, CMSSS, CASSS, CESSS, CCSSS, CO, FA, MM, GM, CL, GT
INTEGER EL, SC, ST, OF, AFQT, MOS1, MOS2, J, K, A5(320, 10), A67(111, 10),
* A8910(281, 10), SSAN1, SSAN2, SSAN3, END, ERROR

REAL CONVT, GIR, ADR, SPR, SIR, AIR, CMR, CAR, CCR, GSR, ARR, WKR, PCR,
* NOR, CSR, ASR, MKR, MCR, EIR, VER

ALGORITHM ASVAB IS THE MASTER ALGORITHM. IT CONTAINS 7 SUB
ALGORITHMS. ALGORITHM ASVAB INITIALIZES A COUNTER (I), INPUTS
THE CONVERSION TABLES FOR ACB COMPOSITE SCORES, AND WRITES THE
HEADINGS FOR ALL FILES.

ALGORITHM ASVAB

CALL TABLE(A5, A67, A8910, I, J, K)

ALGORITHM INPUT SOLDIER DATA READS THE SOLDIER DEMOGRAPHIC DATA
AND THE ASVAB RAW SCORES.

ALGORITHM INPUT SOLDIER DATA

END = 0
CONTINUE

CALL IN(SSAN1, SSAN2, SSAN3, DOBYY, DOBMM, DOBDD, SEX, RACE, DOEYY,
*DQEMM, DOBDD, 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------------------ TERMINATE PROGRAM AFTER LAST CARD
C

IF(END .EQ. 1) GO TO 280
I=I+1
END ALGORITHM INPUT SOLDIER DATA

ALGORITHM VALIDATE SOLDIER DATA VALIDATES THE INPUT DATA BY
MAKING RANGE CHECKS ON THE FOLLOWING VARIABLES: DOB, SEX, RACE,
CE, ED, ETS.

ALGORITHM VALIDATE SOLDIER DATA
ERROR=0
CALL VAL(SSAN1, SSAN2, SSAN3, DOBYY, DOBMM, DOBDD, SEX, RACE, DCEYY,
*DOBMM, DOBDD, ED, ETS, GS, AR, MK, MC, EI, VE, GI, AD, SP, SI, AI, CM, CA,
*CE, WK, PC, NC, CS, AS, ETSY, ETSMM, ETSDD, I, ERROR)
END ALGORITHM VALIDATE SOLDIER DATA

ALGORITHM OUTPUT SOLDIER DATA PRINTS, OR 'ECHOS', THE INPUT
DATA FOR VALIDATION PURPOSES.

ALGORITHM OUTPUT SOLDIER DATA
CALL ECHO(SSAN1, SSAN2, SSAN3, DOBYY, DOBMM, DOBDD, SEX, RACE, DCEYY,
*DOBMM, DOBDD, ED, GRADE, MOS1, MOS2, ETS, AFQT, DES, ETSY, ETSMM, ETSDD,
*GS, AR, MK, MC, EI, VE, GI, AD, SP, SI, AI, CM, CA, CE, CC, WK, PC, NO, CS, AS, MK, MC,*

IF (ERROR .EQ. 1) GO TO 5

END ALGORITHM OUTPUT SOLDIER DATA

ALGORITHM SUBTEST STANDARD SCORES CONVERTS THE SUBTEST RAW
SCORES TO SUBTEST STANDARD SCORES.

ALGORITHM SUBTEST STANDARD SCORES

CALL SSS( GI, AD, SP, SI, AI, CM, CA, CE, CC, GS, AR, WK, PC, NO, CS, AS, MK, MC, *
*EI, VE, TEST, GISS, ADSSS, SPSS, SISS, AISSS, CMSSS, CASSS, CEASSS, *
*CASSS, GSSSS, ARSSS, WKS, PCSSS, NOSSS, CSSSS, Asss, MKSSS, MCSSS, *
*EISSS, VEISSS)

END ALGORITHM SUBTEST STANDARD SCORES

ALGORITHM APTITUDE AREA COMPOSITES COMPUTES THE ACB COMPOSITE
SCORES.

ALGORITHM APTITUDE AREA COMPOSITES

CALL COMP(GSSS, ARSSS, WKS, PCSSS, NOSSS, CSSSS, Asss, MKSSS, *
*MCSSS, EIASSS, VEISSS, GISS, ADSSS, SPSS, SISS, AISSS, CMSSS, *
*CASSS, CEASSS, CASSS, GI, AR, WK, PC, NO, CS, AS, MK, MC, EI, VE, GI, AD, SP, *
*SI, AI, CM, CE, CA, CC, CO, FA, EL, OF, CM, MM, CL, GT, ST, SC, TEST)
END ALGORITHM APTITUDE AREA COMPOSITES

ALGORITHM ACB CONVERTS THE ACB SCORES USING THE CONVERSION TABLES A5, A67, A8910, INTO STANDARD SCORES.

ALGORITHM ACB
CALL ACB(A5,A67,A8910,TEST,GT,GM,EL,CL,MM,SC,CO,FA,GF,ST)
END ALGORITHM ACB

ALGORITHM OUTPUT WRITES THE SUBTEST RAW SCORES, THE SUBTEST STANDARD SCORES, AND THE ACB COMPOSITE SCORES TO SUMMARY FILES.

ALGORITHM OUTPUT
END ALGORITHM OUTPUT

GO TO 5
CONTINUE
C ENC ALGORITHM ASVAB

STOP
END

SUBPROGRAMS

SUBROUTINE TABLE INITIALIZES A COUNTER, READS IN THE CONVERSION TABLES FROM FILES 1, 2, AND 3, AND PRINTS FILE HEADINGS FOR FILES 10, 11, AND 12.

SUBROUTINE TABLE (A5, A67, A8910, I, J, K)

INTEGER A5(320, 10), A67(111, 10), A8910(281, 10), I, J, K

INITIALIZE COUNTER

I = 0

INPUT CONVERSION TABLES

ASVAB 5
DO 1 J=1,320
READ(1,499) (A5(J,K),K=1,10)
CONTINUE

C-----------------------
C ASVAB 6/7
C-----------------------
DO 2 J=1,111
READ(2,499) (A67(J,K),K=1,10)
CONTINUE

C-----------------------
C ASVAB 8/9/10
C-----------------------
DO 3 J=1,281
READ(3,459) (A8910(J,K),K=1,10)
CONTINUE

C-----------------------
C PRINT HEADINGS
C-----------------------
WRITE(10,850)
WRITE(10,917)
WRITE(11,871)
WRITE(11,917)
WRITE(12,910)
WRITE(12,916)
RETURN

499 FORMAT(1015)
871 FORMAT('/16X,'SUBTEST RAW SCORES')
890 FORMAT('/12X,'SUBTEST STANDARD SCORES')
910 FORMAT('/6X,'ARMY CLASSIFICATION BATTERY COMPOSITES')
916 FORMAT(11X,'AFQT CO FA MM GM CL GT EL SC ST OF')
917 FORMAT(11X,'SSAN GI NO AD WK AR SP MK EI MC GS SI AI CM CA CE C
*CC PC CS AS VE')
END

C
C

SUBROUTINE IN INPUTS THE SOLDIER DEMOGRAPHIC DATA AND THE RAW C
SCORES FROM FILE 4.
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)

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, NC, CS, AS

--- INPUT DEMOGRAPHIC DATA AND RAW SCORES ---

READ(4,500,END=100) 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, CES, ETSYY, ETSMM, ETSDD

--- ASSIGN RAW SCORES TO VARIABLES ---

--- ASVAB 8/9/10 ---

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
**ASVAB 5/6/7**

```c
CC
6  CONTINUE
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  CONTINUE
RETURN
100 END = 1
RETURN
C 500 FORMAT(3A3,3I2,2I1,5I2,A4,A1,18I2,2X,A3,3I2)
END

SUBROUTINE VAL VALIDATES THE DEMOGRAPHIC DATA AND THE RAW SCORES. IF AN ERROR IS FOUND IT TERMINATES PROCESSING OF THE RECORD.

SUBROUTINE VAL(SSAN1,SSAN2,SSAN3,DOBYY,DOBMM,DOBDD,SEX,RACE,
*DOEYY,DOEMM,DOEDD,ED,TEST,GS,AR,MK,MC,FI,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,
*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, ETSSDI, I, ERROR

VALIDATE DATE OF BIRTH

IF(DOBYY .GT. 31) GO TO 8
WRITE(9,510) I, SSAN1, SSAN2, SSAN3
WRITE(9,515)
ERROR = 1
CONTINUE

VALIDATE SEX

IF((SEX .EQ. 1) .OR. (SEX .EQ. 2)) GO TO 9
WRITE(9,510) I, SSAN1, SSAN2, SSAN3
WRITE(9,520)
ERROR = 1
CONTINUE

VALIDATE RACE

IF((RACE .EQ. 1) .OR. (RACE .EQ. 2) .OR. (RACE .EQ. 3)) GO TO 10
WRITE(9,510) I, SSAN1, SSAN2, SSAN3
WRITE(9,525)
ERROR = 1
CONTINUE

VALIDATE DATE OF ENTRY

IF(DOEEY .GT. 31) GO TO 11
WRITE(9,510) I, SSAN1, SSAN2, SSAN3
WRITE(9,530)
ERROR = 1
CONTINUE

VALIDATE HIGHEST YEAR OF EDUCATION

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

VALIDATE TEST VERSION

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

VALIDATE SUBTEST RAW SCORES

ASVAB 5/6/7

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
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 8/5/10
C
C-------------------------------
30 CONTINUE
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
CONTINUE
IF(MK .LE. 25) GO TO 38
WRITE(9,510) I,SSAN1,SSAN2,SSAN3
WRITE(9,660)
ERROR = 1
CONTINUE
IF(MC .LE. 25) GO TO 39
WRITE(9,510) I,SSAN1,SSAN2,SSAN3
WRITE(9,665)
ERROR = 1
CONTINUE
IF(EI .LE. 20) GO TO 40
WRITE(9,510) I,SSAN1,SSAN2,SSAN3
WRITE(9,670)
ERROR = 1
CONTINUE
IF(VE .LE. 50) GO TO 41
WRITE(9,510) I,SSAN1,SSAN2,SSAN3
WRITE(9,675)
ERROR = 1
CONTINUE

VALIDATE DATE OF SEPARATION

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
CONTINUE
RETURN

FORMAT('/// ERROR IN RECORD NUMBER: ',I3,' WITH SSAN: ',A3)
FORMAT(' ERROR: DATE OF BIRTH INCOMPLETE, COLUMNS 10-15')
FORMAT(' ERROR: SEX INCOMPLETE, COLUMN 16')
FORMAT(' ERROR: RACE INCOMPLETE, COLUMN 17')
FORMAT(' ERROR: DATE OF ENTRY INCOMPLETE, COLUMNS 18-23')
FORMAT(' ERROR: HIGHEST YEAR OF EDUCATION INCOMPLETE, COLUMNS 24-25')
FORMAT(' ERROR: TEST FORM INCOMPLETE, COLUMNS 33-34')
FORMAT(' ERROR: GI INCOMPLETE, COLUMNS 37-38')
FORMAT(' ERROR: NO INCOMPLETE, COLUMNS 39-40')
FORMAT(' ERROR: AD INCOMPLETE, COLUMNS 41-42')
FORMAT(' ERROR: WK INCOMPLETE, COLUMNS 43-44')
FORMAT(' ERROR: AR INCOMPLETE, COLUMNS 45-46')
SUBROUTINE ECHO WRITES THE DEMOGRAPHIC DATA AND RAW SCORES TO FILE 9.

SUBROUTINE ECHO( SSAN1, SSAN2, SSAN3, DOBYY, DOBMM, DOBD, SEX,
  * RACE, DOEYY, DOEMM, DOED, ED, GRADE, MOS1, MOS2, TEST, AFQ1, DES, ETSYY,
  * ETSMM, ETSDD, GS, AR, MK, MC, EI, VE, GI, AD, SP, SI, AI, CM, CA, CE, CC, WK, PC,
  * NC, CS, AS)

INTEGER SSAN1, SSAN2, SSAN3, DOBYY, DOBMM, DOBD, SEX, RACE, DOEYY,
  * DOEMM, DOED, ED, GRADE, MOS1, MOS2, TEST, AFQ1, DES, ETSYY, ETSMM, ETSDD,
  * GS, AR, MK, MC, EI, VE, GI, AD, SP, SI, AI, CM, CA, CE, CC, WK, PC, NO, CS, AS

PRINT SSAN, AND DATE OF BIRTH
C WRITE(9,681) SSAN1, SSAN2, SSAN3, DOBYY, DOBMM, DOBDD
C -----------------------
C PRINT SEX
C -----------------------
C IF(SEX .NE. 1) GO TO 45
  WRITE(9,685)
  GO TO 46
45  CONTINUE
  WRITE(9,690)
46  CONTINUE
C -----------------------
C PRINT RACE
C -----------------------
C IF(RACE .NE. 1) GO TO 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 PRINT DATE OF ENTRY
C -----------------------
C WRITE(9,710) DOBYY, DOBMM, DOBDD
C -----------------------
C PRINT HIGHEST YEAR OF EDUCATION
C -----------------------
C IF(ED .LT. 1) GO TO 120
  IF(ED .GT. 13) GO TO 120
  GO 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  CONTINUE
WRITE(9,720) ED
GO TO 120
CONTINUE
WRITE(9,725) ED
GO TO 120
CONTINUE
WRITE(9,730) ED
GO TO 120
CONTINUE
WRITE(9,735) ED
GO TO 120
CONTINUE
WRITE(9,740) ED
GO TO 120
CONTINUE
WRITE(9,745) ED
GO TO 120
CONTINUE
WRITE(9,750) ED
GO TO 120
CONTINUE
WRITE(9,755) ED
GO TO 120
CONTINUE
WRITE(9,760) ED
GO TO 120
CONTINUE
WRITE(9,765) ED
GO TO 120
CONTINUE
WRITE(9,770) ED
GO TO 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
CONTINUE
WRITE(9,775) GRADE
GO TO 170
CONTINUE
WRITE(9,780) GRADE
GO TO 170
CONTINUE
WRITE(9,785) GRADE
GO TO 170
CONTINUE
WRITE(9,790) GRADE
GO TO 170
CONTINUE
WRITE(9,795) GRADE
GO TO 170
CONTINUE
WRITE(9,800) GRADE
GO TO 170
CONTINUE
WRITE(9,810) GRADE
GO TO 170
CONTINUE
WRITE(9,815) GRADE
GO TO 170
CONTINUE
WRITE(9,820) GRADE
CONTINUE

-----------------
PRINT MOS
-----------------
WRITE(9,825) MOS1,MOS2

-----------------
PRINT TEST VERSION
-----------------
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 CONTINUE
C-------- PRINT AFQT PERCENTILE
C--------
C        WRITE(9,860) AFQT
C-------- PRINT SEPARATION PROGRAM DESIGNATOR
C--------
C        WRITE(9,865) DES
C-------- PRINT DATE OF SEPARATION
C--------
C        WRITE(9,870) ETSYY,ETSMM,ETSDD
C        WRITE(9,871)
C-------- PRINT ASVAB 5 RAW SCORES
C--------
C        IF(TEST .NE. 35) GO TO 205
C          WRITE(9,874)
C          WRITE(9,875) GI,NO,AD,WK,AR,SP,HK,EI,MC,GS,S1,Al
C          GO TO 215
C 205      CONTINUE
C-------- PRINT ASVAB 6/7 RAW SCORES
C--------
C        IF(TEST .GT. 37) GO TO 210
C          WRITE(9,879)
C          WRITE(9,880) GI,NO,AD,WK,AR,SP,HK,EI,MC,GS,S1,Al,CM,CA,CE,CC
C          GO TO 215
C 210      CONTINUE
C-------- PRINT ASVAB 8/9/10 RAW SCORES
C--------
C        WRITE(9,884)
C        WRITE(9,885) GS,AR,WK,PC,NO,CS,AS,MK,MC,EI,VE
C 215      CONTINUE
C        RETURN
C
681 FORMAT('///: SSN',25X::',3A3:' DATE OF BIRTH(YY/MM/DD)',6X,
*:12,'/','12,'/','12)
685 FORMAT('SEX',12)' MALE')
690 FORMAT('SEX',12)' FEMALE')
695 FORMAT('RACE',12)' WHITE')
700 FORMAT('RACE',12)' BLACK')
705 FORMAT('RACE',12)' OTHER')
710 FORMAT('DATE OF ENTRY(YY/MM/DD)',12,'/','12,'/','12)
715 FORMAT('HIGHEST YEAR OF EDUCATION',8)' 1-7 YEARS - '12)
720 FORMAT('HIGHEST YEAR OF EDUCATION',8)' 8 YEARS - '12)
725 FORMAT('HIGHEST YEAR OF EDUCATION',8)' 1 YEAR OF HIGH SCHOOL - '12)
730 FORMAT('HIGHEST YEAR OF EDUCATION',8)' 2 YEARS OF HIGH SCHOOL - '12)
735 FORMAT('HIGHEST YEAR OF EDUCATION',8)' HIGH SCHOOL DIPLOMA - '12)
740 FORMAT('HIGHEST YEAR OF EDUCATION',8)' 1 YEAR OF COLLEGE - '12)
745 FORMAT('HIGHEST YEAR OF EDUCATION',8)' 2 YEARS OF COLLEGE - '12)
750 FORMAT('HIGHEST YEAR OF EDUCATION',8)' 3-4 YEARS OF COLLEGE - '12)
755 FORMAT('HIGHEST YEAR OF EDUCATION',8)' COLLEGE GRADUATE - '12)
760 FORMAT('HIGHEST YEAR OF EDUCATION',8)' MASTERS DEGREE - '12)
765 FORMAT('HIGHEST YEAR OF EDUCATION',8)' DOCTOR'S DEGREE - '12)
770 FORMAT('HIGHEST YEAR OF EDUCATION',8)' HIGH SCHOOL GED - '12)
775 FORMAT('ENTRY PAY GRADE',12)' E1 - '12)
780 FORMAT('ENTRY PAY GRADE',12)' E2 - '12)
785 FORMAT('ENTRY PAY GRADE',12)' E3 - '12)
790 FORMAT('ENTRY PAY GRADE',12)' E4 - '12)
795 FORMAT('ENTRY PAY GRADE',12)' E5 - '12)
800 FORMAT('ENTRY PAY GRADE',12)' E6 - '12)
805 FORMAT('ENTRY PAY GRADE',12)' E7 - '12)
810 FORMAT('ENTRY PAY GRADE',12)' E8 - '12)
815 FORMAT('ENTRY PAY GRADE',12)' E9 - '12)
820 FORMAT('ENTRY PAY GRADE',12)' 'A1 - '12)
825 FORMAT('MOS',12)' ASVAB 5')
830 FORMAT('MOS',12)' ASVAB 6')
835 FORMAT('MOS',12)' ASVAB 7')
840 FORMAT('MOS',12)' ASVAB 8')
845 FORMAT('MOS',12)' ASVAB 9')
850 FORMAT('MOS',12)' ASVAB 10')
855 FORMAT('MOS',12)' 'A1 - '12)
860 FORMAT('AFQT PERCENTILE',12)' 'A3)
865 FORMAT('SEPARATION PROGRAM DESIGNATOR',12)' 'A3)
870 FORMAT('DATE OF SEPARATION(YY/MM/DD)',12,'/','12,'/','12)
FORMAT(/16,'SUBTEST RAW SCORES')
FORMAT('GI NO AD WK AR SP MK EI MC GS SI AI')
FORMAT(1X,12(2X,12))
FORMAT('GI NO AD WK AR SP MK EI MC GS SI AI CM CA')
FORMAT(1X,16(2X,12))
FORMAT('GS AR WK PC NO CS AS MK MC EI VE')
FORMAT(1X,11(2X,12))
END

SUBROUTINE SSS CONVERTS THE SUBTEST RAW SCORES INTO SUBTEST
STANDARD SCORES.

SUBROUTINE SSS(GI,AD,SP,SI,AM,CM,CA,CE,CC,GS,AR,WK,PC,MC,AS,
*MK,MC,EL,VE,TEST,GISS,ADSS,SPSS,SISS,AISS,CMSS,CASS,CESS,
*CSSS,GSSS,ASSS,WKSS,PCSS,NOSSCSSS,ASSS,ASSS,ASSS,MCSSSS,
*EISSS,VESSS)

INTEGER GI,AD,SP,SI,AM,CM,CA,CE,CC,GS,AR,WK,PC,MC,AS,
*MK,MC,EL,VE,TEST,GISS,ADSS,SPSS,SISS,AISS,CMSS,CASS,CESS,
*CSSS,GSSS,ASSS,WKSS,PCSS,NOSSCSSS,ASSS,ASSS,ASSS,MCSSSS,
*EISSS,VESSS

REAL CONVT,GIR,ADR,SPR,SIR,AIR,CMR,CAR,CCR,GSR,ARR,WKR,PCR,
*NOR,CSR,ASR,MRK,MC,EL,VE'

IF((TEST .NE. 38) .AND.(TEST .NE. 39) .AND.(TEST .NE. 40))
*GO TO 220
GSR=FLOAT(GS)
GSSS=CCNV(GSR,16.199,5.087)
ARR=FLOAT(AR)
ARSSS=CCNV(ARR,17.766,7.2)
WKR=FLOAT(WK)
WKSSS=CCNV(WKR,25.670,7.658)
PCR=FLOAT(PC)
PCSSS = CCNV T(PGR, 10.483, 3.436)
NOR = FLOAT(NO)
NOSSS = CCNV T(NOR, 36.027, 10.387)
CSR = FLOAT(CS)
C SSSS = CCNV T(CSR, 43.149, 16.121)
AS = FLOAT(A S)
ASSS = CCNV T(AS, 16.413, 5.597)
MKR = FLOAT(MK)
MKSSS = CCNV T(MKR, 12.468, 5.950)
MCR = FLOAT(MC)
MCSSS = CCNV T(MCR, 15.519, 5.570)
EIR = FLOAT(EI)
EISSS = CCNV T(EIR, 12.451, 4.316)
VER = FLOAT(VE)
VESSS = CCNV T(VER, 36.153, 10.613)
GO TO 225

C
---------------
C
CCNV T ASVAB 5/6/7
C
---------------
C
220 CCNTINUE
GIR = FLOAT(GI)
GISSS = CCNV T(GIR, 9.656, 3.273)
NOR = FLOAT(NO)
NOSSS = CCNV T(NOR, 30.826, 10.049)
ADR = FLOAT(AD)
ADSSS = CCNV T(ADR, 14.410, 3.957)
WKR = FLOAT(WK)
WKSSS = CCNV T(WKR, 19.638, 7.251)
ARR = FLOAT(AR)
ARSSS = CCNV T(ARR, 12.739, 4.729)
SPR = FLOAT(SP)
SPSSS = CCNV T(SPR, 13.197, 4.257)
MKR = FLOAT(MK)
MKSSS = CCNV T(MKR, 11.636, 4.932)
EIR = FLOAT(EI)
EISSS = CCNV T(EIR, 19.474, 5.955)
MCR = FLOAT(MC)
MCSSS = CCNV T(MCR, 10.842, 4.361)
GSR = FLOAT(GS)
GSSS = CCNV T(GSR, 11.317, 4.362)
STR = FLOAT(SI)
SISSS = CCNV T(SIR, 13.648, 4.302)
AIR = FLOAT(AI)
AISSS = CCNV T(AIR, 11.773, 4.946)

225 CCNTINUE
CC CONVERT ASVAB 6/7

IF((TEST .NE. 36) .AND. (TEST .NE. 37)) GO TO 230

CMR=FLOAT(CM)
CMSS=CCNV(T,CMR,10.97,4.69)
CAR=FLOAT(CA)
CAS=CCNV(T,CAR,9.68,2.91)
CER=FLOAT(CE)
CESS=CCNV(T,CER,8.92,4.21)
CCR=FLOAT(CC)
CSS=CCNV(T,CCR,15.51,4.89)

230 CONTINUE
RETURN
END

FUNCTION CCNV(T, RAW, XBAR, SIGMA)

REAL FUNCTION CCNV(T, RAW, XBAR, SIGMA)

VARIABLES AND CONSTANTS

RAW SUBTEST RAW SCORE.
XBAR SUBTEST POPULATION MEAN.
SIGMA SUBTEST POPULATION STANDARD DEVIATION.
T TEMPORARY VARIABLE.
TT TEMPORARY VARIABLE.
TTT TEMPORARY VARIABLE.

DECLARE VARIABLES

INTEGER TT
REAL T,DIFF,TTT,XBAR,SIGMA,RAW
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) CCNVT=80
RETURN
END

SUBROUTINE COMP COMPUTES THE ACB COMPOSITE SCORES.

SUBROUTINE COMP(GS5S, 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, EL, VE, GI, AD, SP,
* SI, AI, CM, CE, CA, CC, CO, FA, EL, OF, GM, MM, CL, GT, ST, SC, TEST)
INTEGER GS5S, 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, EL, VE, GI, AD, SP,
* SI, AI, CM, CA, CE, CC, CO, FA, EL, OF, GM, MM, CL, GT, ST, SC

--------------------------
CCOMPUTE ASVAB 5
--------------------------

IF( TEST .NE. 35) GO TO 235
CO=ARSSS+SISSS+SPSSS+ADSSS
FA=ARSSS+GISSS+MKSSS+EISSS
MM=MKSSS+SISSS+EISSS+AISSS
GM=ARSSS+GISSS+MCSSS+AISSS
CL=ARSSS+WKSSS+ADSSS
GT=ARSSS+WKSSS
EL=ARSSS+EISSS+MCSSS+SISSS
SC=ARSSS+WKSSS+MCSSS+SPSSS
SUBROUTINE ACB CONVERTS THE ACB SCORES INTO STANDARD SCORES.
SUBROUTINE ACB(A5, A67, A8910, TEST, GT, GM, EL, CL, MM, SC, CO, FA, OF, ST)

INTEGER A5(320, 10), A67(111, 10), A8910(281, 10), GT, GM, EL, CL, MM, SC,
* CO, FA, OF, ST, TEST

-------------------------------------

CCNVERT ASVAB 5

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(CO, 7)
  FA = A5(FA, 8)
  OF = A5(OF, 9)
  ST = A5(ST, 10)
GO TO 260

-------------------------------------

CCNVERT ASVAB 6/7

-------------------------------------

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(CO+1, 7)
  FA = A67(FA+1, 8)
  OF = A67(OF+1, 9)
  ST = A67(ST+1, 10)
GO TO 260

-------------------------------------

CCNVERT ASVAB 8/9/10

-------------------------------------

255 CONTINUE
  ST = 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)

CONTINUE
C
RETURN
END

SUBROUTINE OUT WRITES THE SUBTEST RAW SCORES TO FILE 11, THE
SUBTEST STANDARD SCORES TO FILES 9 AND 10, AND THE ACB COMPOSITE
SCORES TO FILES 9 AND 12.

SUBROUTINE OUT(SSID1, SSID2, SSID3, AFQT, GSSSS, ARSSS, MKSSS, MCSSS,
* EISSS, VESSS, GISSS, ADSSS, SPSSS, SISSS, AISSS, CMSSS, CASSS, CSSS,
* CSSSS, NOSSS, AASSS, WSSSS, 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)

INTEGER SSID1, SSID2, SSID3, AFQT, GSSSS, ARSSS, MKSSS, MCSSS, CSSSS,
* EISSS, VESSS, GISSS, ADSSS, SPSSS, SISSS, AISSS, CMSSS, CASSS, CSSS,
* CSSSS, NOSSS, AASSS, WSSSS, CSSSS, 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

PRINT HEADINGS

WRITE (9, 890)

PRINT SUBTEST RAW AND STANDARD SCORES
--- PRINT ASVAB 5 ---

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(10, 876) SSAN1, SSAN2, SSAN3, 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

--- PRINT ASVAB 6/7 ---

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, CESSS
WRITE(10, 881) SSAN1, SSAN2, SSAN3, GISSS, NOSSS, ADSSS, WKSSS, ARSSS,
* SPSSS, MKSSS, EISSS, MCSSS, GSSSS, SISSS, AISSS, CMSSS, CASSS, CESSS,
* CESSS
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

--- PRINT ASVAB 8/9/10 ---

270 CONTINUE
WRITE(9, 884)
WRITE(9, 885) GISSS, ARSSS, WKSSS, PCSSS, NOSSS, CSSSS, ASSSS, MKSSS,
* MCSSS, EISSS, VESSS
WRITE(10, 886) SSAN1, SSAN2, SSAN3, NOSSS, WKSSS, ARSSS, MKSSS, EISSS,
* MCSSS, GISSS, PCSSS, CSSSS, ASSSS, VESSS
WRITE(11, 886) SSAN1, SSAN2, SSAN3, GS, AR, WK, PC, NO, CS, AS, MK,
* MC, EI, VE
275 CONTINUE

--- PRINT ACB COMPOSITE SCORES ---

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(I2,12))
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(I2,12))
881 FORMAT(1X,3A3,3X,16(I2,1X))
884 FCMAT(' GS AR WK PC NO CS AS MK MC EI VE')
885 FORMAT(1X,11(I2,12))
886 FORMAT(1X,3A3,6X,12,4X,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


<table>
<thead>
<tr>
<th>Initial Distribution List</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>No. Copies</strong></td>
</tr>
<tr>
<td>1. Defense Technical Information Center 2</td>
</tr>
<tr>
<td>Cameron Station 2</td>
</tr>
<tr>
<td>Alexandria, Virginia 22314</td>
</tr>
<tr>
<td>2. Library, Code 0142 2</td>
</tr>
<tr>
<td>Naval Postgraduate School 2</td>
</tr>
<tr>
<td>Monterey, California 93943</td>
</tr>
<tr>
<td>3. Commandant 1</td>
</tr>
<tr>
<td>U.S. Army Infantry School 1</td>
</tr>
<tr>
<td>Attn: ATSH-ES 1</td>
</tr>
<tr>
<td>Fort Benning, Georgia 31905</td>
</tr>
<tr>
<td>4. LTC. Stephen Paek, Code 55Ph 1</td>
</tr>
<tr>
<td>Naval Postgraduate School 1</td>
</tr>
<tr>
<td>Monterey, California 93943</td>
</tr>
<tr>
<td>5. Captain Robert Forman 1</td>
</tr>
<tr>
<td>6550 Big Creek Parkway 1</td>
</tr>
<tr>
<td>Parma Heights, Ohio 44130</td>
</tr>
</tbody>
</table>
A management information system for the analysis of the Armed Services Aptitude Battery.