STUDENT REPORT
THE STUDENT MIX SOFTWARE SYSTEM (SMSS)
MAJOR KENNETH M. RITCHHART 86-2120
MAJOR ROBERT L. SIMMONS
"insights into tomorrow"
DISCLAIMER

The views and conclusions expressed in this document are those of the author. They are not intended and should not be thought to represent official ideas, attitudes, or policies of any agency of the United States Government. The author has not had special access to official information or ideas and has employed only open-source material available to any writer on this subject.

This document is the property of the United States Government. It is available for distribution to the general public. A loan copy of the document may be obtained from the Air University Interlibrary Loan Service (AUL/LDEX, Maxwell AFB, Alabama, 36112) or the Defense Technical Information Center. Request must include the author's name and complete title of the study.

This document may be reproduced for use in other research reports or educational pursuits contingent upon the following stipulations:

--- Reproduction rights do not extend to any copyrighted material that may be contained in the research report.

--- All reproduced copies must contain the following credit line: "Reprinted by permission of the Air Command and Staff College."

--- All reproduced copies must contain the name(s) of the report's author(s).

--- If format modification is necessary to better serve the user's needs, adjustments may be made to this report--this authorization does not extend to copyrighted information or material. The following statement must accompany the modified document: "Adapted from Air Command and Staff Research Report _(number)_ entitled _(title)_ by _(author)_ ."

--- This notice must be included with any reproduced or adapted portions of this document.
REPORT NUMBER  86-2120

TITLE   THE STUDENT MIX SOFTWARE SYSTEM (SMSS)

AUTHOR(S)  MAJOR KENNETH M. RITCHHART, USAF
            MAJOR ROBERT L. SIMMONS, USAF

FACULTY ADVISOR  MAJOR CHARLES E. WILLIAMS, ACSC/EDOWC

SPONSOR  MAJOR IRVING F. ROMER, ACSC/EDOWB

Submitted to the faculty in partial fulfillment of requirements for graduation.

AIR COMMAND AND STAFF COLLEGE
AIR UNIVERSITY
MAXWELL AFB, AL  36112
The Air Command and Staff College must assign approximately 550 students to 44 groups, called seminars, three times annually. In an effort to equally distribute key student skills throughout the seminars, the faculty spends approximately 200 manhours constructing and modifying student assignments. To reduce this manpower expenditure, a completely new automated system was developed and named the Student Mix Software System (SMSS). This report documents the creation and use of the SMSS which reduced the time required to produce the mix from over 10 days to less than 2 days.
This report documents the creation and use of the Student Mix Software System (SMSS) for the Z-120 personal computer. SMSS is used at the Air Command and Staff College (ACSC) to assign students to seminars based on user selected rules, and to prepare the required output reports. SMSS reduces the time required to produce a new mix of students from 10 days to 30 minutes, and eliminates 80% of the manual student reassignments required by the old semiautomated system. The SMSS is so successful that it is already in operation; it was used to generate the third mix student assignments for ACSC class of 1986.

The authors wish to publicly acknowledge the assistance and support of many people who patiently answered questions and explained the mix process. Special thanks and recognition is given to Major Rusty Romer, USAF, who provided the inspiration and explanations that led to the finalization of the mix rules. Additionally, we thank Major Ed Williams, USAF, who provided expert guidance, motivation, advice, feedback and served as the ACSC advisor for this project.

Anticipating the possible use by other Air University schools, the SMS was designed to be flexible and to provide the user with the ability to customize the system. Consequently, a copy of the software may be obtained by ordering the Student Mix Software System (SMSS) from Air Command and Staff College, EDD Student Mixer, Maxwell AFB, Al, 36112-5564. Requester must provide a 5 1/4", double sided, double density disk.

This research project is submitted to simplify a process that is key to the learning of all students at Air University, that of the seminar mix. We sincerely hope that future classes will benefit from the balanced seminars which the SMSS is designed to produce.
ABOUT THE AUTHOR

Major Kenneth M. Ritchhart has over twelve years of experience as an intelligence officer and automated data processing manager. He obtained his Masters Degree in Computer Science as a distinguished graduate of the University of Oklahoma in 1978, and completed over half of the requirements for a Doctorate in artificial intelligence and computer science through George Washington University. Ken is experienced in all aspects of computer science, from sophisticated research in artificial intelligence and database systems to applications software development and maintenance. He has experience in microcomputers, minicomputers, and large computer systems performing: programming, systems analysis, future planning, system acquisition, and software research. He is a 1980 graduate of the Computer Systems Staff Officer School; and in 1984 he was awarded the professional Certificate in Data Processing (CDP). His technical experience includes an extensive knowledge of varied computer hardware systems, and the ability to program in over a dozen different computer languages. Areas of special computer expertise include: artificial intelligence, computer conversion planning, data base systems, data structures, on-line interactive application systems, microcomputers, program management, and automated project management. He is a member of the Association for Computing Machinery (ACM), the Institute of Electrical and Electronic Engineers (IEEE), the Computer Society, and the American Association of Artificial Intelligence (AAAI).
Major Robert L. Simmons graduated from Kansas State University, Cum Laude, with a B.S. in Computer Science. After receiving his commission through ROTC, he entered the Air Force as a computer systems operations officer at HQ/MAC AD. While at Scott, he became the chief system analyst for the MAC computerized Aerial Port Documentation and Management System. After receiving a Masters in Business Administration from Southern Illinois University, he was stationed at Hickam AFB as the 619th MASS Data Processing Installation Chief. In 1970, Major Simmons was assigned to the Space Shuttle program at Vandenberg AFB. As the chief of computer integration he directed the installation, checkout and operation of over 100 software packages used by the Space Shuttle Launch computer. Following the Shuttle assignment he was assigned to the Office of the Secretary of the Air Force, Special Projects, as Director of Flight Software Integration. Major Simmons has completed Squadron Officers School by correspondence, Air Command and Staff College, and Air War College by seminar. His decorations include the Air Force Meritorious Service Medal with two oak leaf clusters, the Air Force Commendation Medal with one oak leaf cluster, and the Air Force Achievement Medal. He is married to the former Susan of Manhattan, Kansas and they have two children: Ryan and Sean.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Preface</th>
<th>iii</th>
</tr>
</thead>
<tbody>
<tr>
<td>About the Authors</td>
<td>iv</td>
</tr>
<tr>
<td>List of Illustrations</td>
<td>vii</td>
</tr>
</tbody>
</table>

## CHAPTER ONE—THE STUDENT MIXING PROBLEM
- Determining the SMSS Requirements 2
- Designing SMSS 3

## CHAPTER TWO—SYSTEM OVERVIEW
- Hardware 5
- Software 5
- Phase Descriptions 6
- Input Phase 7
- Preprocessing Phase 8
- Mixing Phase 10
- Post Processing Phase 12

## CHAPTER THREE—USER INSTRUCTIONS FOR PREPROCESSING PHASE 14

## CHAPTER FOUR—RUNNING THE STUDENT MIX
- Main Menu 19
- System Configuration 20
- Mix Assignment Rules 21
- Running the Mix 22
- Reviewing Mix Statistics 24
- Register Manual Changes to Seminars 27

## CHAPTER FIVE—POST PROCESSING AND REPORTS 28

## CHAPTER SIX—CONCLUSIONS 31
- SMSS vs SAM 31
- SMSS Shortcomings 32
- Suggested Improvements 32

## REFERENCES 33

## APPENDICES:
- Appendix A—Omega Data Base Values 35
- Appendix B—CONDOR Command Files 38
- Appendix C—CONDOR Data File Descriptions 51
- Appendix D—SMSS ZBASIC Source Code 58
## LIST OF ILLUSTRATIONS

### FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>System Overview</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>Input Phase</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>Preprocessing Phase</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>Mixing Phase</td>
<td>11</td>
</tr>
<tr>
<td>5</td>
<td>Post Processing and Report Generation Phase</td>
<td>12</td>
</tr>
<tr>
<td>6</td>
<td>SMSS Help Menu</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>STUMIX Help Menu</td>
<td>16</td>
</tr>
<tr>
<td>8</td>
<td>Project Help Menu</td>
<td>17</td>
</tr>
<tr>
<td>9</td>
<td>SMSS Main Menu</td>
<td>19</td>
</tr>
<tr>
<td>10</td>
<td>System Configuration Menu</td>
<td>20</td>
</tr>
<tr>
<td>11</td>
<td>Mix Assignment Rule Menu</td>
<td>21</td>
</tr>
<tr>
<td>12</td>
<td>Running the Mix Menu</td>
<td>23</td>
</tr>
<tr>
<td>13</td>
<td>Review Mix Statistics Menu</td>
<td>24</td>
</tr>
<tr>
<td>14</td>
<td>Example, Overall School Mix Statistics</td>
<td>25</td>
</tr>
<tr>
<td>15</td>
<td>Example, Wing B Mix Statistics</td>
<td>25</td>
</tr>
<tr>
<td>16</td>
<td>Example, Seminar 36 Mix Statistics</td>
<td>26</td>
</tr>
<tr>
<td>17</td>
<td>Reports Help Menu</td>
<td>29</td>
</tr>
</tbody>
</table>
Chapter One

THE STUDENT MIXING PROBLEM

Mixing students at the Air Command and Staff College (ACSC) and assigning them to wings and seminars does not seem like a terribly difficult task. After all, how difficult can it be to divide approximately 565 students into 44 seminars --you just randomly assign them, right? Unfortunately, it is not quite that simple, there are many rules that must be considered before the first student is assigned.

The basic idea behind these rules is to provide a wide range of experience for each seminar so that the students can be exposed to new and different ideas, cultures, career fields, and sister services. To do this, minority students, females, rated personnel, and students from different components need to be equally distributed throughout the student body. To insure that they have the opportunity to meet new people and make new friends and career contacts, students are not allowed to be assigned to seminars with more than one student with whom they have already served. To expand the number of faculty instructors who have the opportunity to rate each individual, the students are reassigned to different seminars each mix. Since the students present a major portion of the lessons and lead the seminars, it is also important to provide individuals to each seminar who have experience in the areas being studied such as staff communications, the Planning, Programming and Budgeting System (PPBS), acquisition and logistics, strategic operations, and tactical operations. There are also special rules that apply to international officers (IO's), and to part time students like the Squadron Officers School (SOS) staff, and Associate Research Institute (ARI) students. Obviously, a proper student mix requires the consideration of many factors.

All of these factors combine to make the mixing of students a difficult and time consuming task—especially by mix three when several of the rules have a chance to conflict with each other, and some of the seminars are deleted due to the departure of the IO's and the reassignment of some faculty members. Nor is the job over when the students are mixed;
reports must still be generated for the wing chiefs, and faculty instructors. Other reports, containing information on the class composition, are required by Air University.

According to the sponsor of this project, the ACSC mix master, the old Seminar Automatic Mixer (SAM) system on the Honeywell computer at Gunter AFS, was slow (seven to ten day turnaround); cumbersome (difficult to get any changes made); and unresponsive (it violated most of the rules currently required for a good mix). The output from the Honeywell normally required extensive reworking by the school mix master and by the wing mix masters involving approximately two weeks of manual work for each mix. It also took nearly two months, before the school began, for the school mix master to obtain the student data, process it by hand, manually enter it into the system, and produce the initial reports. To try and reduce the work load from this semiautomated mixing system, the mix master requested an ACSC student project to see if a software system could be created for the Z-120 personal computers at the Air Command and Staff College to efficiently mix the students and produce the required reports. The result of this student project was the Student Mix Software System (SMSS).

DETERMINING THE SMSS REQUIREMENTS

To determine the personnel factors required to properly mix the students, the ACSC mix master was repeatedly interviewed on existing AU/ACSC practices and policies. As a result of these interviews, we developed five hand written pages of rules, heuristics, and desired factors to be used in mixing the students and selecting class leaders. This information was supplemented by reviewing the related Air University and Air Command & Staff College regulations dealing with seminar organization, rank, and procedures. The resulting rules and personnel factors were then used as the basis on which SMSS was built. Before deciding to build a new software system completely from scratch, the old code from the Honeywell system was examined.

The authors reviewed the program code from the Honeywell to see if it could be used or modified to obtain the desired results. Unfortunately, this code was designed for a batch system and lacked any user interfaces or error checking procedures. The code was also extremely limited in scope and was not flexible enough to adapt to the many rules and restrictions needed for a good mix. The decision was consequently made to start from scratch and write a completely
new software system, starting with the selection of the necessary software tools.

The decision on the software support tools to be used in creating and supporting SMSS was constrained by the nature of the hardware on which SMSS was intended to operate. The ACSC mix master's Z-120 was equipped with 256KB of memory, a 10 MB hard disk, the MS-DOS operating system, and the CONDOR Relational Data Base Management System (DBMS). The Z-120 was also equipped with the MICROSOFT FORTRAN, COBOL, interpreted ZBASIC, and the compiled ZBASIC programming languages. A review of the CONDOR DBMS by the authors found that it was able to provide all of the basic functions needed by SMSS with the exception of direct calls to separate executable subroutines. Because of CONDOR's general suitability, the long lead time required to obtain a new DBMS, and the time constraints of this project, the decision was made to continue using CONDOR. A review of the programming languages available resulted in the decision to use ZBASIC. ZBASIC had several advantages over FORTRAN and COBOL. The interpreted ZBASIC simplified debugging and program development with its interactive support environment. Once the programs were developed and debugged they could be compiled to gain the advantage of faster execution and smaller memory requirements. The memory constraint was one of the major reasons for not using COBOL as the SMSS programming language. While COBOL is an excellent language for file manipulation and report generation, its large memory requirements made it unsuitable for SMSS with its extensive code requirements and with the Z-120 which had only 256KB of memory. FORTRAN was considered for the development of the mixing algorithm, but its limited character string and file manipulation facilities made it unsuitable for SMSS with its interactive user requirements. Consequently, the final support tools selected for SMSS were the MICROSOFT ZBASIC programming language and the CONDOR DBMS.

DESIGNING_SMSS

After determining the mixing constraints and selecting the support tools, an interactive user prototype of SMSS was built using ZBASIC and CONDOR. This prototype permitted the ACSC mix master to see exactly what he would be using, and permitted him to run through the menus and display screens. As a result of his interaction with the prototype, he changed several of his criteria, and asked for additional information to be displayed in SMSS. The prototype formed the basis for the final development and coding of SMSS and permitted misunderstandings to be worked out before extensive coding was
completed. It also permitted the user to refine his requirements and to contribute to the overall system design.

The SMSS hardcopy report requirements were determined by reviewing the ACSC Supplement to Air University Regulation 171-1, Student Statistical Reporting; by examining all current reports; and by interviewing the ACSC mix master about additional reports that might be needed. The report facilities of CONDOR were selected to generate these alpha and statistical reports since the user facilities in CONDOR minimized the amount of new code required.

This chapter introduced the problems which SMSS was designed to solve, and the basic methodology used in researching and designing SMSS. Chapter two provides an SMSS system overview describing the hardware and software environments and the information flow between the different data files and the various programs or CONDOR command files. Chapters three through five provide the user with guidance on how to run the preprocessing (data preparation), mixing, and post processing (report generation) programs. Finally, chapter six provides conclusions about SMSS, how well it operates in comparison to the old system, and what improvements could be made to SMSS to make it better.
Chapter Two

SYSTEM OVERVIEW

The Student Mix Software System (SMSS) gives the user the capability to automatically create a mix of students that fits a set of user definable criteria. Additionally, the user retains the option to selectively override the assignments created by SMSS. This capability is provided by a package of computer hardware, software, and procedures. This chapter will describe the hardware and software needed to execute the system and explain the information flow throughout the input, preprocessing, mixing, and post processing phases of the system. Subsequent chapters will describe specific software components and user actions. Prior to describing the four phases, a brief description of the hardware and software environment is in order.

HARDWARE

The Z-120 personal computer is in wide use throughout the Air Force, particularly Air University, and possesses significant computing capability. Consequently, the SMSS is designed to operate within this capability. The specific configuration includes a 10 megabyte hard disk, a 132 column printer, and 256KB of internal random access memory. This hardware is the minimum set needed to execute the SMSS support software packages.

SOFTWARE

The SMSS is designed to make extensive use of existing software packages. Consequently, it only executes with the support of the CONDOR III Relational Data Base Management System and the popular MS-DOS operating system. In addition to these packages, the ZBASIC programming language is used to create the mix routine. Therefore, to maintain or modify the current system, the ZBASIC compiler and related support software is needed. This brief description of the supporting software reveals that the SMSS is a combination of computer programs that have been linked together to provide the user with a flexible, convenient system.
PHASE DESCRIPTION

The linkage and relationship of the four SMSS phases (see figure 1.) can best be described from the user's viewpoint. In the input phase, the user will enter the student data into the CONDOR system and create the OMEGA and BETA data bases. After this phase is complete, the user will select, prepare, and pass the data to the mix routine. The mix phase will prompt the user to select the mix criteria and will assign the students to seminars. The final phase allows the user to view the assignment and make any desired changes. Additionally, reports will be prepared for distribution. As we discuss each phase, the purpose will be identified and a description of support software will be included.

Figure 1. System Overview
INPUT PHASE

The input phase (see figure 2) uses the general capabilities of CONDOR III. By utilizing this data base management system, a flexible input capability has been provided for the user.

Figure 2. Input Phase
Purpose

The purpose of the input phase is to insert the student data into two CONDOR data bases. The first data base is named OMEGA and is designed to contain key personnel and class assignment information on students attending Air Command and Staff College. The second data base is named BETA and contains personnel data that must be reviewed and updated to accurately reflect student skills or characteristics.

Description

The input phase, through the basic capabilities of CONDOR, places data in a data base either manually (via the ENTER command) or by reading a file (via the READ command). For additional information on these commands, see the CONDOR user manual. The information placed in the OMEGA data base is of critical importance to the entire process and must adhere to the values identified in appendix A. Upon the completion of the input phase, all students attending ACSC and their personnel data will be loaded and confirmed as correct by the user. Additionally, the specific skill data must be entered into the BETA file. When these actions are completed the input phase is complete and we are ready for the preprocessing phase.

Preprocessing Phase

The preprocessing phase (see figure 3) makes extensive use of CONDOR's advanced capabilities to minimize the user's involvement. These capabilities present the user with help screens to provide a road map through the process and rely heavily upon command files (a sequence of previously programmed commands) to generate the correct data.

Purpose

The purpose of the preprocessing phase is to create a repeatable, consistent, and simplified version of student data that can be transferred to the mix process. Additionally, it creates the interface file between CONDOR and the mix process.

Description

The overall process extracts student data from the OMEGA file, simplifies the data, and prepares the data for the mix routine. The phase begins by extracting key student data from the OMEGA file and storing the information in a temporary file called TEST1.
Since the mix process expects a simplified version of the data, a series of one character yes/no fields must be added to each record. This is accomplished by reorganizing the TEST1 data base. After adding the additional fields, the records are examined and the critical parameters are computed and stored in the newly created fields. The file is then read into a file called TEST2 and sorted by date of rank.

The next step in the preprocessing phase is to strip out redundant information. This reduction is accomplished by projecting the key mixing criteria from TEST2 into a file called CINTER. At this point, special student skill data in the BETA file is added to CINTER by CONDOR’s post process. The final step in this phase writes the interface file BASICF. This file is written in a form that can be easily read by the mixing routine.

Mixing Phase

The mixing phase (see figure 4) of the system is the heart of the process. The complexity of the routine is beyond the scope of CONDOR’s capability, so a special routine compiled in ZBASIC has been created.

Purpose

The purpose of the mix routine is to assign students to classes or seminars, based upon user selectable criteria. Through the user set criteria, various skills and backgrounds can be emphasized, and students with these skills can be evenly distributed.

Description

The mix phase first prompts the operator to select the criteria and the mix to be assigned. Based upon this information, the student file is read and the total number of students possessing the selected criteria tabulated. Next, the seminar leaders and assistant seminar leaders are assigned. If it is the first mix, the class commander and senior ranking officer for each wing are selected. The next step is to assign students to the various classes or seminars, based upon the criteria. The goal of the assignment algorithm is to equally distribute the critical skills within limitations of specific rules. After completing the assignment, a summary report is created which displays the number of students possessing the selected characteristics by seminar. Finally, the results are written to BASICOT. At this point, the post processing and report generation phase begins.
FIGURE 4.  MIXING PHASE

11
Post Processing and Report Generation Phase

This phase (see figure 5) is the last phase and uses the capability of CONDOR to produce the report documents.

Figure 5. Post Processing Phase
Purpose

The purpose of the last phase is to review and manually modify any assignments, finalize the assignments, and to generate reports for distribution.

Description

The first task is to read student records, including their seminar assignment data, from the BASICOT file back to CINTER. This action brings the mix assignment data back into CONDOR for subsequent processing. The next step updates OMEGA with the tentative assignment information. The user has the option to select seven different report formats. After a review of the data, the user can change the OMEGA data base by utilizing CONDOR's UPDATE command. At this point, the responsibility to maintain skill balance is completely controlled by the user. Once the user is satisfied with the assignments, a final set of reports can be run and the final assignments transferred to the CINTER data base. This last step will enable mix two and subsequent mix processing to skip the preprocessing phase.

The system overview was designed to give a user a brief description of the Student Mix Software System. The description followed the input, process, and output steps that are inherit in all automated systems. As the system was described, the uses of the data base management system, CONDOR III, and the computer language, ZBASIC, were identified. Additionally, the purpose of each phase was explained to aid the user in understanding the system. The next chapters will cover the user's actions necessary to complete a mix process.
Chapter Three

USER INSTRUCTIONS FOR PREPROCESSING PHASE

These instructions are written assuming that the user or operator is familiar with the Z-120 computer. He/She should be able to operate the equipment and possess a working knowledge of the MS-DOS operating system. Additionally, the operator should be familiar with the CONDOR III Data Base Management System.

The SMSS system was developed to minimize user actions. However, if problems develop or minor modifications are needed, a knowledgeable user is essential. To assist the knowledgeable user, the CONDOR command files and the basic source code in appendix B and D contain comments to explain the program logic. With this caveat about appropriate user knowledge, the following instructions should be sufficient to successfully execute the preprocessing phase of the SMSS.

The preprocessing phase should not be initiated until the OMEGA and BETA data bases have been loaded and verified. The preprocessing phase must be completed once per student body or class. Normally, the process will be executed just before the first mix. After completing the entire mix process, all critical information will be posted to OMEGA and BETA. Consequently, the preprocessing routines need not be executed again, unless some information within OMEGA or BETA is changed.

The procedure to execute the preprocessing phase is straightforward. As mentioned before, help screens have been developed to aid the user. The following steps are described to demonstrate the actions the user must take.
Step 1. After entering CONDOR, the main help menu should be called.

Enter: HELP SMSS <CR>

The following menu is displayed:

```
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. UPDATE ACSC STUDENT DATA</td>
<td>2. PREPARE DATA FOR MIX MASTER</td>
</tr>
<tr>
<td>3. UPDATE ACSC STUDENT FILE WITH MIX MASTER RESULTS</td>
<td>4. CREATE REPORTS</td>
</tr>
<tr>
<td>5. POST FINAL MIX ASSIGNMENTS TO ALL FILES</td>
<td>6. EXIT</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------</td>
</tr>
</tbody>
</table>
```

Enter number:

Figure 6. SMSS Help Menu
Step 2. To initiate preprocessing phase, select item two.

Enter:  2 <CR>

The following menu is displayed:

- TO RUN THE STUDENT MIX ROUTINE IT IS FIRST NECESSARY TO CREATE AN INTERFACE FILE WHICH CAN BE READ BY THE MIXMASTER. THIS IS DONE BY FOLLOWING THE HELP SCREEN COMMANDS. THE PROCESS HAS BEEN BROKEN INTO STEPS TO PROVIDE YOU WITH THE OPPORTUNITY TO CONFIRM THAT EVERYTHING IS PROCEEDING AS EXPECTED. I DO NOT WANT TO DESTROY ANY OF THE DATA BASE THAT YOU WORKED SO HARD TO CREATE.
- LET'S GO . . .

1. PROJECT SELECTED OMEGA DATA TO A WORK FILE SO THAT MIXING DATA CAN BE DERIVED
2. COMPUTE KEY MIX DATA
3. SORT DATA BY DATE OF RANK FOR MIX MASTER
4. POST STUDENT SKILLS TO INTERFACE FILE
5. CREATE AN INTERFACE FILE FOR THE MIX MASTER
6. EXIT

Enter number:

Figure 7. STUMIX Help Menu
Step 3. The process outlined in the help menu must be completed in sequence in order to produce an interface file for the mix routine.

Enter: 1 <CR>

The following menu is displayed:

```
1. LET'S GO
2. RETURN TO PREVIOUS MENU
'. EXIT

Enter number:
```

Figure 8. PROJECT Help Menu

Step 4. As indicated by the menu, some help from the operator is required. CONDOR's REORG command requires the operator to define the data type. The SMSS will initiate the process and do most of the work; however, the operator must provide the requested information.

Enter: 1 <CR>

The process will take approximately 15 minutes.

Step 5. After the process is complete, the menu will reappear. To continue with the preprocessing phase, return to the previous menu.

Enter: 2 <CR>

The main preprocessing menu (see step 3) will be displayed.

Step 6. Continue the process by initiating the commands in sequence.

Enter: 2 <CR>

This phase of the process will take about 15 minutes. Then the menu will reappear.
Step 7. Continue the process.
   Enter: 3<CR>
This step of the process will take about 5 minutes and will conclude by displaying the now familiar menu.

Step 8. Continue the process.
   Enter: 4<CR>
This step will take about 10 minutes and will conclude by displaying the menu.

Step 9. Continue the process.
   Enter: 5<CR>
This step will take about 20 minutes and will conclude by displaying the menu.

Step 10. At this point the BASICIF file has been created and the next phase of the system can be initiated. Consequently, we should return to the system level and execute the SMSS basic program.
   Enter: 6<CR>
The CONDOR prompt should appear.

Step 11. To return to the MS-DOS operating system:
   Enter: SYSTEM <CR>
The system prompt will appear.
Chapter Four

RUNNING THE STUDENT MIX

MAIN MENU

The Student Mix Software System is started by entering E:SMSS in response to the system prompt. This will bring up the main system menu allowing you to configure the system, review or change the mix rules, run the mix subroutine, review the mix distribution statistics, register manual changes, or return to the system level. An example of the main menu is shown below. You should make your choice by entering 1 thru 5 and then pushing the "RETURN" key.

Figure 9.  SMSS Main Menu
SYSTEM CONFIGURATION

This menu allows you to configure the system by specifying the name of the school, entering the name of the major units (i.e. WINGS, FLIGHTS, ETC), the number of major units (up to 5), the name of subunits (i.e. SEMINARS, SECTIONS, ETC), and the total number of subunits (up to 60). If you do not wish to change the default values simply hit "RETURN" and go on to the next selection field. You will then have the opportunity to enter the individual wing names, the number of subunits (seminars) in each wing, the seniority assigned to each wing, the first seminar in the wing, the last seminar in each wing, and any missing seminars which have been deleted for this mix.

Figure 10. System Configuration Menu
MIX_ASSIGNMENT_RULES

This menu is used to select which rules are to be applied during the actual student mixing process. Rules may be specified as: A--Always apply, P--Preferred, or D--Don’t care. You can simply review these rules and then return to the previous menu by entering ‘X’ at the choice prompt; or you can enter ‘C’ and elect to change the existing defaults. If you elect to change the defaults, you will be led through the rules one at a time and may change the rule by entering a new value or you may leave it the same by simply entering ‘RETURN’. Please note that you can not use the full screen editor and go directly to the rule you want to change. You must progress thru the rules one at a time.

---

1) PRIORITY GIVEN TO COMMUNICATION SKILLS: D
2) PRIORITY GIVEN TO PPBS SKILLS: D
3) PRIORITY GIVEN TO TACTICAL OPERATION SKILLS: A
4) PRIORITY GIVEN TO STRATEGIC OPERATION SKILLS: A
5) SOS STUDENTS DO NOT CHANGE SEMINARS: D
6) IO’S DON’T CHANGE SEMINARS (OR ‘X’ TO DELETE): M
7) ARI STUDENTS DO NOT CHANGE SEMINARS: A
8) CAN NOT BE ASSIGNED TO THE SAME SEMINAR: A
9) MAX # OF STUDENTS PREVIOUSLY ASSIGNED WITH: I
10) PRIORITY GIVEN TO ACQUISITION/LOG SKILLS: D
11) EVENLY DISTRIBUTE THE FOLLOWING STUDENTS:
    ARMY A NAVY A RATED A
    RES/NG/MC A BY SEX A BY RACE M
    NON LINE A BY RANK D USAFA D
    ED.LEVEL D SR ORG EXP D SINGLE A

ENTER C) TO CHANGE THE DEFAULTS
X) TO EXIT BACK TO THE PREVIOUS MENU

ENTER YOUR CHOICE: C

Figure 11. Mix Assignment Rules Menu
RUNNING THE MIX MENU

This menu provides you with the ability to run the mix programs or return to the main menu. You may select mix 1, 2, 3, or Special. The special option should be used if you are running a new mix like mix 4, if you are running a partial mix like remixing wing B only, or if you simply want the rules left as they were—since the other options reset the rules to the standard defaults normally desired for that mix. BEFORE YOU RUN THE MIX PROGRAM YOU MUST HAVE EXTRACTED THE DATA FROM THE CONDOR DBMS AND PASSED IT TO SMSS USING THE PREPROCESSOR. This preprocessing only needs to be done once, but it must be done before running this program; if you have not done this do not proceed, instead return to the main menu, exit SMSS, go to the DBMS HELP menus and run the preprocessor.

If you elect to run the mix you will then be given a chance to change the rules selection criteria. You may elect to change any or all of the rules to be applied during the mix process. When you enter 'X' to return from the Rules Menu, you will immediately start on the mixing process. The first program to be called will be SMSSMIXI which will read the data from the file provided by the CONDOR Preprocessor and write the student data out in a form usable by the SMSS mixing program. It will also initialize all the variables and arrays to be used during the mix process, and read in the rules. This process takes approximately four minutes. During the reading process a '.' will be printed on the screen after every ten records to let you know what the program is doing.

The next program to be executed is SMSSMIX which performs the actual mixing. It applies the rules you selected, one at a time. It takes the information supplied by SMSSMIXI and allocates each seminar a fair share of each student category (i.e. pilots, minorities, PPBS skills, etc.). It then randomly assigns the most senior student not having already been a Seminar Leader (SL), as the SL for that seminar, and assigns the next most senior individual as the Alternate Seminar Leader (ASL). After the SL/ASL process, the international officers (IO’s) are posted (they do not normally change seminars). Next, the ARI and SOS students are posted or assigned depending on the rules. The system performs assignments in the following order:

Assign Seminar Leaders & Alternate Seminar Leaders
Post International Officers
Post or assign ARI & SOS
Assign Army
Assign Communication Skills
Assign PPBS Skills
Assign Tactical Operation Skills
Assign Strategic Operation Skills
Assign Acquisition/Logistic Skills
Assign Navy, Reserve, National Guard, USMC
Assign Females
Assign Singles
Assign Minorities
Assign Pilots
Assign Navigators
Assign No Master's Education
Assign USAFA Graduates
Assign Senior Organizational Experience
Assign All Others

The mixing program takes approximately 30 minutes to run. During this process it prints out the category and the individual being processed. When the mixing is completed, the system will call the SMSSMIXD program to take the results and write them out in a form that is readable by the CONDOR DBMS; this takes three minutes. The system then returns to the SMSS program and allows the user to review the results through the statistical menu described in figure 13.

![RUNNING THE MIX MENU - SMSS]

This menu provides you with the capability to run the mix programs and produce a student distribution based on the rules that you select. The rules will be reset to the standard defaults for the mix you select. You can then customize or change the rules as you see fit. You should use option 'D) SPECIAL' - if you want to do something like adding a mix 4 or using the existing rules without resetting. Please note that you must run the CONDOR DBMS preprocessor to extract the right data before executing this program.

Options:
A) MIX 1
B) MIX 2
C) MIX 3
D) SPECIAL
X) RETURN TO MAIN MENU

Enter your choice: C

Figure 12. Running the Mix Menu

23
REVIEW_MIX_STATISTICS_MENU

This menu provides you with the ability to review the results of the mix process at the overall school level, at the wing level, or by reviewing individual seminars. Examples of statistical reviews for these three levels are provided below. You may request a paper print out of these results by answering "Y" to the question at the bottom of the screen. If you are not happy with the results of the mix you can rerun it by returning to the 'RUNNING THE MIX MENU'. The results will not be posted back to CONDOR until you run the CONDOR Processor which is the topic of chapter five.

STATISTICAL OPTIONS:
A) REVIEW AIR COMMAND & STAFF COLLEGE
B) REVIEW WINGS BY SEMINAR
X) RETURN TO PREVIOUS MENU

ENTER YOUR CHOICE: ___________

Figure 13. Review Mix Statistics Menu
### Figure 14. Example, Overall School Mix Statistics

<table>
<thead>
<tr>
<th>ATTRIBUTE</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMM SKILLS</td>
<td>12</td>
<td>10</td>
<td>11</td>
<td>10</td>
</tr>
<tr>
<td>PPBS SKILLS</td>
<td>14</td>
<td>11</td>
<td>15</td>
<td>13</td>
</tr>
<tr>
<td>TAC OPS SKILLS</td>
<td>22</td>
<td>19</td>
<td>20</td>
<td>21</td>
</tr>
<tr>
<td>STRAT OPS SKILL</td>
<td>25</td>
<td>22</td>
<td>17</td>
<td>22</td>
</tr>
<tr>
<td>ACO/LOG SKILLS</td>
<td>10</td>
<td>8</td>
<td>11</td>
<td>7</td>
</tr>
<tr>
<td>PILOT</td>
<td>36</td>
<td>34</td>
<td>38</td>
<td>37</td>
</tr>
<tr>
<td>NAVIGATOR</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>SINGLE/UNAC</td>
<td>30</td>
<td>32</td>
<td>28</td>
<td>26</td>
</tr>
<tr>
<td>USAFA GRADS</td>
<td>14</td>
<td>20</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>ARMY</td>
<td>11</td>
<td>11</td>
<td>11</td>
<td>11</td>
</tr>
<tr>
<td>RES/NG/USN/USMC</td>
<td>9</td>
<td>9</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>MINORITIES</td>
<td>7</td>
<td>7</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>FEMALES</td>
<td>5</td>
<td>7</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>RANK - CAPT</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>88xx/89xx/9xxx</td>
<td>3</td>
<td>6</td>
<td>6</td>
<td>5</td>
</tr>
<tr>
<td>SR ORG EXP</td>
<td>77</td>
<td>73</td>
<td>78</td>
<td>77</td>
</tr>
<tr>
<td>ARI/SOS</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>NO MASTERS ED</td>
<td>18</td>
<td>22</td>
<td>25</td>
<td>16</td>
</tr>
</tbody>
</table>

*Would you like a hardcopy print of this? (Y/N) Y*

### Figure 15. Example, Wing B Mix Statistics

<table>
<thead>
<tr>
<th>ATTRIBUTE</th>
<th>23</th>
<th>24</th>
<th>25</th>
<th>27</th>
<th>28</th>
<th>29</th>
<th>30</th>
<th>31</th>
<th>32</th>
<th>33</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMM SKILLS</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>PPBS SKILLS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TAC OPS SKILLS</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>STRAT OPS SKILL</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>ACO/LOG SKILLS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>PILOT</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>NAVIGATOR</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>SINGLE/UNAC</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>USAFA GRADS</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>ARMY</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>RES/NG/USN/USMC</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>MINORITIES</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>FEMALES</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>RANK - CAPT</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>88xx/89xx/9xxx</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>4</td>
<td>8</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>SR ORG EXP</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>8</td>
<td>10</td>
<td>8</td>
<td>4</td>
<td>8</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>ARI/SOS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>NO MASTERS ED</td>
<td>3</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>

*Would you like a hardcopy print of this? (Y/N) Y*
The mixing program does not make a perfect mix, and manual changes are usually required. An easy way of checking out the ramifications of those changes is to produce the wing statistical summary in SMSS. But, before a correct statistical summary can be produced, the changes need to be posted. Selecting this option from the main menu will result in running the SMSSUPDT program which will read in the BASICIF input file from CONDOR and post the manual changes so that the statistics review menu can be used with the latest manual updates. Before selecting this option you should have completed your manual changes to the OMEGA and CINTER files using the CONDOR DBMS. These changes should then be written out to the BASICIF file using the appropriate help menu just as if you were going to run the mix over again. Instead of running the mix over, this option posts back the manual changes you have made so that you can determine their effect on the overall wing statistical distribution of key characteristics.

The next chapter will describe the CONDOR Post Processor which posts the results of the mix programs back to the CONDOR DBMS by reading the BASICOT ASCII file into CINTER. The Post Processor also contains a help menu for producing the required output reports.
POST PROCESSING AND REPORTS

As described in chapter two, this phase of the system posts the mix assignment back to OMEGA and prepares a number of user selectable reports. Based upon these reports, the user can make any changes he/she feels are appropriate. After completing all changes, the assignments are posted to CINTER for next mix processing and final reports are generated.

As was the case for the preprocessing phase, the user should possess the skills described in chapter three.

The post processing phase should not be initiated until the mix phase has concluded with the creation of the file BASICOT.

Step 1. To enter the post process phase, the user must be within CONDOR. The post processing phase is initiated by calling up the main SMSS menu.

Enter: HELP SMSS <CR>
This action will present the SMSS screen (see figure 6.)

Step 2. The first step is to retrieve the newly created assignment information and post it to the OMEGA file. This is accomplished by selecting option 3.

Enter: 3<CR>
The posting process will take approximately 10 minutes. When it is finished the main SMSS menu will reappear.
Step 3. At this point, the information can be reviewed in different formats. To obtain a list of the various formats, the reports menu must be called.

Enter: 4<CR>

The following display will appear:

```
REPORTS FORMAT

1. PRINT ALPHA REPORT FOR ENTIRE CLASS
2. PRINT ALPHA REPORT BY WING
3. PRINT MIX 1 ALPHA REPORT BY MIX 1 SEMINARS
4. PRINT MIX 2 ALPHA REPORT BY MIX 2 SEMINARS
5. PRINT MIX 2 ALPHA REPORT BY MIX 1 SEMINARS
6. PRINT MIX 2 ALPHA REPORT BY MIX 2 SEMINARS
7. PRINT MIX 3 ALPHA REPORT BY MIX 2 SEMINARS
8. PRINT STATISTICAL ABSTRACT
9. EXIT
```

Enter number:

Figure 17. REPORTS Help Menu

Step 4. Make a selection of the appropriate report.

Enter: X<CR> (where X = the report number)

After sorting in the appropriate order, the report will begin printing. When the report is finished the report menu will reappear.

Step 5. After reviewing the reports, changes to the mix assignment can be made by updating the OMEGA file. To begin this process exit the reports phase and return to the SMSS main menu.

Enter: 9<CR>

The CONDOR Prompt should appear.

Enter: HELP SMSS<CR>

The SMSS menu will appear (see figure 6).
Step 6. To initiate the update process:
   Enter: 1<CR>
   The OMEGA format will appear and the update process can begin. For additional information on this subject, see the CONDOR III user manual.

Step 7. After the update process has been completed, the final results should be posted to CINTER. This step will eliminate the long process of creating the CINTER file from scratch and allow future mi> assignment to be initiated at the mi> phase. To post the results:
   Enter: 5<CR>
   The post process should take about 10 minutes. When the process is completed the SMSS main menu will reappear.

Step 8. At this point the final reports can be run in the same manner as described in step 4. Once the reports are completed, the Student Mi> Software System has completed its task.
Chapter Six

CONCLUSIONS

The success or failure of SMSS may be judged against two criteria. First, how much better is SMSS than the old Seminar Assignment Mix (SAM) system on the Honeywell. Second, how completely does SMSS meet the mixing objectives of the school. Any areas where SMSS falls short of the school objectives can be identified as areas of potential improvement.

SMSS_VS_SAM

According to the ACSC Mix Master, the SAM system was basically unresponsive to the current needs of the school. It required 7 to 10 days to get results back, it was basically unresponsive to required changes, and the product required extensive rework (nearly two weeks work involving from 25-40% of the students being reassigned) by the school and wing mix masters to produce a useable mix. The total manpower requirements of the SAM system were approximately 200 manhours to produce each final mix. This manpower intensive system, combined with the long lead time required to obtain the initial SAM product, meant that the system was not responsive to new requirements or new student data.

SMSS allows the mix master to turn the rules on or off as desired. It takes only 30 minutes to run the mix and obtain the initial results. If the SMSS results are undesirable, or if new data is available, the mix can be re-run immediately. The SMSS output is not quite perfect, it still requires some manual rework to produce a final mix. According to the ACSC Mix Master, the SMSS output will require several hours of manual rework by each of the wing mix masters as opposed to several weeks work on SAM. The SMSS output will require 5-10 moves per wing, which will involve reassigning approximately 5% of the students by hand. The CONDOR SMSS Statistical Output reports also permit the mix master to produce the ACSC Student Statistics Report in less than an hour as opposed to several weeks when done by hand. By any measure of comparison, SMSS is a great success when compared to the old.
semi-automated system. SMSS provide the mix master with more flexibility, greater user control, much faster results, and a better product.

**SMSS-SHORTCOMINGS**

If SMSS is compared to the school mix objectives, it does have a few shortcomings. It does not produce a perfectly even mix. There are also minor perturbations in the output with some seminars getting more of one student characteristic than another by a factor of more than one. There are also a few students who are assigned to seminars with two or more previous classmates. The CONDOR SMSS output report procedure produces all of the information required for the Student Statistical Report, but it does not automatically produce the final report in the correct format. Overall, SMSS does meet the school objectives of an improved product with much greater speed (about 400 times faster), and greater flexibility. It also permits accurate automatic statistical summaries for school reporting.

**SUGGESTED IMPROVEMENTS**

Two programs could be written in ZBASIC to improve the performance of the SMSS. The first of these would be an adjustment program which would walk through the seminars and compare the student characteristics within a wing. Whenever it discovered two seminars with an imbalance of personnel characteristics it would try to swap students until it balanced the allocations. This program should also check and reassign anyone who was allocated to a seminar with more than one previous classmate.

The other program should be able to take the information provided by the CONDOR statistical summary queries and produce the final ACSC Student Statistical Report in the form required by the Air University (AU). This would further eliminate the work load on the ACSC Mix Master and reduce errors by eliminating manual processing and typing.

SMSS has done a lot to reduce the workload on the school and wing mix masters. It provides a much better product, with great flexibility, in much less time than previously required. With the addition of the improvements suggested above, SMSS will provide ACSC and the AU with a fast and efficient system to meet their student mixing and reporting requirements.
REFERENCES

Air Command and Staff College. *Automatic Data Processing Systems and Procedures*; ACSC Regulation 171-1; Maxwell AFB, AL: 26 May 1981

Air Command and Staff College. *Seminar Organization & Responsibilities*; ACSC Regulation 35-2; Maxwell AFB, AL: 1 Aug 1985

Air Command and Staff College. *Student Statistical Reporting*; ACSC Supplement to Air University Regulation 178-1; Maxwell AFB, AL: 1 Aug 1984

Air University. *Academic Rank*; AU Regulation 30-5; Maxwell AFB, AL: 30 Apr 1984

Romer, Irving F., Major, USAF. ACSC Mix Master, Maxwell AFB, AL. *Interviews, Seminar Mix Selection Criteria*; 4-20 Nov 1985


Young and Gilliland, Majors, USAF. *Background Paper on the Seminar Assignment Program*; Unpublished Paper, Maxwell AFB, AL: 2 Dec 1977


APPENDICES

Appendix A—Omega Data Base Values ......................... 35
Appendix B—CONDOR Command Files ......................... 38
Appendix C—CONDOR Data File Descriptions ............... 51
Appendix D—SMSS ZBASIC Source Code ....................... 58
APPENDIX

A

Omega Data Base Values
OMEGA DATA BASE VALUES

1. NAME: LAST, FIRST, MI, ETC
2. SSAN: STANDARD 9 DIGIT, NO HYPHENS
3. RANK: 03, 04, ETC
4. DOR: 000000 (840101)
5. COMP: USAF, USA, USN, USMC, ANG, AFRES, CIV, USCG
6. AERO: PLT, NAV, SR.PLT, SR.NAV, CMD.PLT, MS.NAV, NO RATING
7. PAFSC: (AN) K1065C
8. PAS: SAC, MAC, TAC, ETC., REFER TO HANDOUT IN MIXER FOLDER FOR ACTUAL COMMAND CODES.
9. SEX: M, F
10. RACE: 3 DIGIT, CAU, BLK, OTH
11. MAR. ST: S, M, D (WE WILL HAVE TO UPDATE FOR UNACCOMPANIED)
12. DOB: 000000 (500101)
13. COMM: OTS, ROTC, USAFA, USMA, USNA, DIR
15. PLSD: 000000 (700101)
16. DAFSC: (AN) K1065C
17. 2.AFSC: """
18. 3.AFSC: """
19. H.ORG: DOD, HAF, SOA, MAJCOM, NAF, ADV, GRP, WNG, SON, DET
20. PME.1: SOS C/R (INSERT IF SOS BY C OR R COMPLETED)
21. PME.2: ACSC S/C (INSERT IF ACSC BY S OR C COMPLETED)
22. PME.3: OTHER (AWC, ICAF, NDU, AFSC, ACGS, NWCS, NWC)
23. RTFD: 0000 (8705)

24. MOF: 000 (UP TO A 3 DIGIT NUMBER)

25. SEI: ALL GROUPS OF 3 NUMERICS, NO SPACES BETWEEN GROUPS

26. 1AC.HRS.DATE: MAKE SURE NUMERIC CODE TRANSLATED

27. 2AC.HRS.DATE: " " " " 

28. 3AC.HRS.DATE: " " " " 

29. 4AC.HRS.DATE: " " " " 

30. 5AC.HRS.DATE: " " " " 

31. WING: A, B, C, D

32. MIX.1: NUMERIC FROM 01-44

33. MIX.2: " " " 

34. MIX.3: " " " 

35. ST.NO.: (STUDENT NUMBER, FILLED IN AFTER MIXING)

36. AY: CLASS YEAR
APPENDIX B

CONDOR Command Files
ABSTRACT1.CMD FILE

:THIS COMMAND FILE CREATES SECTIONS I AND SECTIONS II OF THE CLASS STATISTICS REPORT.

:IDENTIFY IO'S

COPY ABSTRACT = OMEGA OK

IDENTIFY IO'S

SELECT ABSTRACT WHERE DOR = 

CHANGE RESULT ST COMP #10

POST ABSTRACT RESULT BY SSAN REP COMP

SORT ABSTRACT BY COMP

PRINTS SECTION I

TABULATE ABSTRACT BY COMP [P]

PRINTS SECTION II

SORT ABSTRACT BY RANK, COMP

TABULATE ABSTRACT BY RANK, COMP [P]

RUN ABSTRACT1

*END

ABSTRACT3.CMD FILE

:THIS COMMAND FILE CREATES SECTION III AND V OF THE STUDENT STATISTICS REPORT.

:IO'S ARE NOT INCLUDED IN SECTION III-XIV

DELETE ABSTRACT WHERE COMP = 10

SELECT ABSTRACT WHERE AERO = *PIL*

CHANGE RESULT ST AERO = PILOT

SAVE AERODB

SELECT ABSTRACT WHERE AERO = *NAV*

CHANGE RESULT ST AERO = NAVIGATOR

APPEND AERODB RESULT

SELECT ABSTRACT WHERE AERO = *NO*

CHANGE RESULT ST AERO = "NON RATED"

APPEND AERODB RESULT

SORT AERODB BY AERO, COMP

TABULATE AERODB BY AERO, COMP [S]

SAVE RPT1 OK

CLEAN UP FILES

DESTROY AERODB OK
RUN ABSTRACT6
*END

ABSTRACT6.CMD FILE

;THIS COMMAND FILE CREATES SECTION VI OF THE STUDENT STATISTICS REPORT
;
;COMBINES SINGLE AND DEVORICED INTO SINGLE
;
SELECT ABSTRACT WHERE MAR.ST = D,S
CHANGE RESULT ST MAR.ST = S
SAVE MARSTDB OK
;
;COMBINES MARRIED AND UNACCOMPANIED INTO MARRIED
;
SELECT ABSTRACT WHERE MAR.ST = M,U
CHANGE RESULT ST MAR.ST = M
APPEND MARSTDB RESULT
;
;IDENTIFIES THE NUMBER OF MARRIED OFFICERS THAT ARE UNACCOMPANIED
;
SELECT ABSTRACT WHERE MAR.ST = U
APPEND MARSTDB RESULT
;
;PRINT RESULTS
SORT MARSTDB BY MAR.ST
TABULATE MARSTDB BY MAR.ST [P]
;
;CLEAN UP FILES
DESTROY MARSTDB OK
RUN ABSTRACT7
*END

ABSTRACT7.CMD FILE

;THIS COMMAND FILE CREATES SECTION VII OF THE STUDENT STATISTICS REPORT
;
SORT ABSTRACT BY SEX
TABULATE ABSTRACT BY SEX [P]
RUN ABSTRACT8
*END
ABSTRACT8.CMD FILE

;THIS COMMAND FILE CREATES SECTION VIII OF THE STUDENT STATISTICS REPORT
;
SORT ABSTRACT BY RACE
TABULATE ABSTRACT BY RACE [P]
RUN ABSTRACT9
*END

ABSTRACT9.CMD FILE

;THIS COMMAND FILE CREATES SECTION IX OF THE STUDENT STATISTICS REPORT
;
;ELIMINATE CIV FROM THIS SECTION
SELECT ABSTRACT WHERE COMP NE CIV
SORT RESULT BY COMM
TABULATE RESULT BY COMM [P]
RUN ABSTRACT11
*END

ABSTRACT10.CMD FILE

;THIS COMMAND FILE CREATES SECTION X OF THE STUDENT STATISTICS REPORT
;
SORT ABSTRACT BY H.ORG
TABULATE ABSTRACT BY H.ORG [P]
*END

ABSTRACT11.CMD FILE

;THIS COMMAND FILE CREATES SECTION XI OF THE STUDENT STATISTICS REPORT
;
;COMPUTE OFFICERS WHO COMPLETED SOS AND EQUIVALENT SCHOOLS IN RESIDENCE
;
SELECT ABSTRACT WHERE PME.1 = *R
SORT RESULT BY PME.1
TABULATE RESULT BY PME.1 [P]
;
;COMPUTE OFFICERS WHO COMPLETED PME BY CORRESPONDANCE
;
SELECT ABSTRACT WHERE PME.1 = *C
SORT RESULT BY PME.1
TABULATE RESULT BY PME.1 [P]
;
SELECT ABSTRACT WHERE PME.2 = "????C"
SORT RESULT BY PME.2
TABULATE RESULT BY PME.2 [P]
;
SELECT ABSTRACT WHERE PME.3 = "????C"
SORT RESULT BY PME.3
TABULATE RESULT BY PME.3 [P]
RUN ABSTRA12
*END

ABSTRA12.CMD FILE

;THIS COMMAND FILE CREATES SECTION XII OF THE STUDENT STATISTICS REPORT
;
COMPUTE ABSTRACT WHERE ED.LEVEL = "PDG ST ED.LEVEL = MAS+
SORT ABSTRACT BY ED.LEVEL, COMP
TABULATE ABSTRACT BY ED.LEVEL, COMP [P]
RUN ABSTRA13
*END

ABSTRA13.CMD FILE

;THIS COMMAND FILE CREATES SECTION XIII OF THE STUDENTS STATISTICS REPORT
;
;DELETE NON AIR FORCE PERSONNEL FROM STATISTICS
;
SELECT ABSTRACT WHERE COMP = "USAF,ANG,AFRES"
SAVE OCCTEMP OK
;
SELECT OCCTEMP WHERE PAFSC = "00"???
CHANGE RESULT ST PAFSC = 00
SAVE RPTDB OK
;
SELECT OCCTEMP WHERE PAFSC = "02"???
CHANGE RESULT ST PAFSC = 02
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = "05"???
CHANGE RESULT ST PAFSC = 05
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = "09"???
CHANGE RESULT ST PAFSC = 09
APPEND RPTDB RESULT
;

42
SELECT OCCTEMP WHERE PAFSC = ?1????
CHANGE RESULT ST PAFSC = 10-20
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?20????
CHANGE RESULT ST PAFSC = 10-20
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?21????
CHANGE RESULT ST PAFSC = 10-20
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?22????
CHANGE RESULT ST PAFSC = 10-20
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?25????
CHANGE RESULT ST PAFSC = 25
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?26????
CHANGE RESULT ST PAFSC = 26
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?27????
CHANGE RESULT ST PAFSC = 27
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?28????
CHANGE RESULT ST PAFSC = 28
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?31????
CHANGE RESULT ST PAFSC = 31
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?40????
CHANGE RESULT ST PAFSC = 40
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?49????
CHANGE RESULT ST PAFSC = 49
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?55????
CHANGE RESULT ST PAFSC = 55
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = ?60????
CHANGE RESULT ST PAFSC = 60
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 62
CHANGE RESULT ST PAFSC = 62
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 64
CHANGE RESULT ST PAFSC = 64
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 65
CHANGE RESULT ST PAFSC = 65
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 66
CHANGE RESULT ST PAFSC = 66
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 67
CHANGE RESULT ST PAFSC = 67
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 70
CHANGE RESULT ST PAFSC = 70
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 73
CHANGE RESULT ST PAFSC = 73
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 74
CHANGE RESULT ST PAFSC = 74
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 75
CHANGE RESULT ST PAFSC = 75
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 79
CHANGE RESULT ST PAFSC = 79
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 80
CHANGE RESULT ST PAFSC = 80
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 81
CHANGE RESULT ST PAFSC = 81
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 82
CHANGE RESULT ST PAFSC = 82
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 87
CHANGE RESULT ST PAFSC = 87
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 88
CHANGE RESULT ST PAFSC = 88
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 89
CHANGE RESULT ST PAFSC = 89
APPEND RPTDB RESULT
;
SELECT OCCTEMP WHERE PAFSC = 90-99
CHANGE RESULT ST PAFSC = 90-99
APPEND RPTDB RESULT
;
PRINT RESULTS
TABULATE RPTDB BY PAFSC [P]
;
CLEAN UP FILES
DESTROY RPTDB OK
DESTROY OCCTEMP OK
RUN ABSTRA14
*END

ABSTRA14.CMD FILE

; THIS COMMAND FILE CREATS SECTION XIV OF THE STUDENT STATISTICAL REPORT
;
;
SELECT ABSTRACT WHERE COMP = ANG, AFRES, USAF
SAVE RPTDB OK
SORT RPTDB BY PAS
TABULATE RPTDB BY PAS [P]
DESTROY RPTDB OK
RUN ABSTRA15
*END

ABSTRA15.CMD FILE

; THIS FILE CREATES SECTION XV OF THE STUDENT STATISTICS REPORT
;
; SELECT ONLY INTERNATIONAL OFFICER
;
SELECT OMEGA WHERE DOR = " "
SORT RESULT BY COMP
TABULATE RESULT BY COMP [P]
*END
ALPHA.CMD FILE

SORT OMEGA BY NAME
REPORT ALPHA
*END

CPTMIXPR.CMD FILE

COMPUTE TEST1 WHERE DOR = "", O ST IO = Y
SELECT TEST1 WHERE IO = Y
SAVE IOFILE
DELETE TEST1 WHERE IO = Y
COMPUTE TEST1 WHERE ED.LEVEL = BAC, BAC+, " " ST NOMAST = Y
COMPUTE TEST1 WHERE H.ORG = MAJCOM,50A,HAF,DOD ST SRORG = Y
COMPUTE TEST1 WHERE AERO = *PIL* ST PILOT = Y
COMPUTE TEST1 WHERE AERO = *NAV* ST NAVIGATOR = Y
COMPUTE TEST1 WHERE MAR.ST = U,D,S ST SINGLE = Y
COMPUTE TEST1 WHERE COMM = "AF ACAD" ST USAFA = Y
COMPUTE TEST1 WHERE COMP = USN ST NAVY = Y
COMPUTE TEST1 WHERE COMP = USA ST ARMY = Y
COMPUTE TEST1 WHERE COMP = AFRES, ANG, USMC, USG ST RESNGUSMC = Y
COMPUTE TEST1 WHERE SEX = F ST FEMALE = Y
COMPUTE TEST1 WHERE RANK = 03 ST RANKC = Y
COMPUTE TEST1 WHERE PAFSC = ?B*,?B9*,?9* OR COMP = CIV ST NONLINE = Y
COMPUTE TEST1 WHERE PME.3 NE " " ST SRPME = Y
COMPUTE TEST1 WHERE COMP = USAF ST USAF = Y
COMPUTE TEST1 WHERE RACE NE CAU ST MINORITY = Y
COMPUTE TEST1 WHERE PAFSC = ?11*,?1515?,?1555? ST TACOPSK = Y
COMPUTE TEST1 WHERE 2.AFSC = ?11*,?1515?,?1555? ST TACOPSK = Y
APPEND TEST1 IOFILE
COMPUTE TEST1 ST NRANK = @RANK
COMPUTE TEST1 ST NDOR = @DOR
DESTROY IOFILE
*END

PSTFINAL.CMD FILE

; THIS COMMAND FILE WILL POST THE USER MODIFIED RESULTS OF THE
; MIXING PROCESS BACK TO CINTER
; THIS ACTION ELEMINATES THE NECESSITY TO PREPARE THE DATA
; FOR THE
; MIX ROUTINE IN FUTURE MIXES
;
COPY CINTERBU = CINTER OK
POST CINTER OMEGA BY SSAN REP MIX.1, MIX.2, MIX.3, ASL1, ASL2, ASL3, SL1, SL2, SL3, CC, SRO
*END

PSTOMEGA.CMD FILE

COPY OMEGABU = OMEGA OK
COPY CINTERBU = CINTER OK
EMPTY CINTER OK
READ CINTER BASICOT [E] POST OMEGA BINTER BY NAME, SSAN REP MIX.1, MIX.2, MIX.3, ACL1, ASL2, ASL3, SL1, SL2, SL3, SRO, CC
*END

PSTSKILL.CMD FILE

;This command file post the student skills contained
; in the BETA file to the CINTER file
;
COPY CINTERBU = CINTER OK
POST CINTER BETA BY SSAN REP COMMSK, STRATOPS, TACKOPS, PPBS, ACQLOG
*END

REDEFINE.CMD FILE

*MESSAGE ENTER THE NAME OF THE DATASET YOU WANT TO REDEFINE
*GET $1
*MESSAGE ENTER THE NAME FOR THE BACKUP COPY OF $1
*GET $2
COPY $2 = $1
WRITE $1 [B]
DEFINE $1
READ $1
*END

JEM1SEM1.CMD FILE

COPY RPTFILE = OMEGA OK
*MESSAGE YOU MAY SELECT ALL WINGS OR INDIVIDUAL WINGS
*MESSAGE TO SELECT ALL WINGS USE "X" OTHERWISE USE WING LETTER
*MESSAGE ENTER WING LETTER A,B,C,D OR X
*GET $1
*IF $1 = A,B,C,D

47
SELECT RPTFILE WHERE WING = $1
SAVE RPTFILE OK
*ENDIF
SORT RPTFILE BY MIX.1
REPORT SEM1ALPH ALL [P]
*END

SEM2SEM1.CMD FILE
COPY RPTFILE = OMEGA OK
*MESSAGE YOU MAY SELECT ALL WINGS OR INDIVIDUAL WINGS
*MESSAGE TO SELECT ALL WINGS USE "X" OTHERWISE USE WING LETTER
*MESSAGE ENTER WING LETTER A,B,C,D OR X
*GET $1
*IF $1 = A,B,C,D
   SELECT RPTFILE WHERE WING = $1
   SAVE RPTFILE OK
*ENDIF
SORT RPTFILE BY MIX.1
REPORT SEM2ALPH ALL [P]
*END

SEM2SEM2.CMD FILE
COPY RPTFILE = OMEGA OK
*MESSAGE YOU MAY SELECT ALL WINGS OR INDIVIDUAL WINGS
*MESSAGE TO SELECT ALL WINGS USE "X" OTHERWISE USE WING LETTER
*MESSAGE ENTER WING LETTER A,B,C,D OR X
*GET $1
*IF $1 = A,B,C,D
   SELECT RPTFILE WHERE WING = $1
   SAVE RPTFILE OK
*ENDIF
SORT RPTFILE BY MIX.2
REPORT SEM2ALPH ALL [P]
*END

SEM3SEM2.CMD FILE
COPY RPTFILE = OMEGA OK
*MESSAGE YOU MAY SELECT ALL WINGS OR INDIVIDUAL WINGS
*MESSAGE TO SELECT ALL WINGS USE "X" OTHERWISE USE WING LETTER
*MESSAGE ENTER WING LETTER A,B,C,D OR X
*GET $1
*IF $1 = A,B,C,D
   SELECT RPTFILE WHERE WING = $1
   SAVE RPTFILE OK
*ENDIF
SORT RPTFILE BY MIX.2
REPORT SEM2ALPH ALL [P]
SME3SEM3.CMD FILE

; SINCE IO'S GRADUATE BEFORE THE THIRD MIX, THEY ARE ELEDINATED
; SELECT OMEGA WHERE DOR NE " "
; SAVE RPTFILE OK
*MESSAGE YOU MAY SELECT ALL WINGS OR INDIVIDUAL WINGS
*MESSAGE TO SELECT ALL WINGS USE "X" OTHERWISE USE WING LETTER
*MESSAGE ENTER WING LETTER A,B,C,D OR X
*GET $1
*IF $1 = A,B,C,D
   SELECT RPTFILE WHERE WING = $1
   SAVE RPTFILE OK
*ENDIF
SORT RPTFILE BY MIX.3
REPORT SEM3ALPH ALL [P]
*END

SLTMIXDA.CMD FILE

COPY TEST1BU = TEST1 OK
DESTROY TEST1 OK
PROJECT OMEGA BY NAME, SSAN, RANK, ST.NO, COMP, AERO, PAFSC,
2.AFSC, SEX, RACE, MAR.ST, COMM, ED.LEVEL, H.ORG, PME.3,
SAVE 1HALF OK
PROJECT OMEGA BY SSAN, WING, MIX.1, MIX.2, MIX.3, PAS, DOR,
SAVE 2HALF OK
JOIN 1HALF 2HALF MATCHING SSAN
SAVE TEST1 OK
DESTROY 1HALF OK
DESTROY 2HALF OK
REORG TEST1 TEST2.FRM
*END

SORT.CMD FILE

WRITE TEST1 TEMP
EMPTY TEST2 OK
READ TEST2 TEMP
COMPUTE TEST2 ST SORTKEY = (12-NRANK)*1000000 + NDOR
SORT TEST2 SORTKEY
*END
APPENDIX

CONDOR Data File Descriptions
Attribute summary of dataset OMEGA

1. NAME: AN,27,0,27,"
2. SSAN: AN,9,0,9,"
3. RANK: AN,2,0,2,"
4. DOR: AN,6,0,6,"
5. COMP: AN,5,0,5,"
6. AERO: AN,9,0,9,"
7. AFSC: AN,6,0,6,"
8. PAS: AN,3,0,3,"
9. SEX: AN,1,0,1,"
10. RACE: AN,3,0,3,"
11. MAR.ST: AN,1,0,1,"
12. DOB: AN,6,0,6,"
13. COMM: AN,7,0,7,"
14. ED.LEVEL: AN,4,0,4,"
15. PLSD: AN,6,0,6,"
16. DAFSC: AN,6,0,6,"
17. 2.AFSC: AN,6,0,6,"
18. 3.AFSC: AN,6,0,6,"
19. H.ORG: AN,3,0,3,"
20. PME.1: AN,5,0,5,"
21. PME.2: AN,5,0,5,"
22. PME.3: AN,5,0,5,"
23. RTFD: AN,4,0,4,"
24. MOF: AN,3,0,3,"
25. SEI: AN,24,0,24,"
26. AC.HRS.DATE: AN,15,0,15,"
27. AC.HRS.DATE: AN,15,0,15,"
28. AC.HRS.DATE: AN,15,0,15,"
29. AC.HRS.DATE: AN,15,0,15,"
30. AC.HRS.DATE: AN,15,0,15,"
31. WING: AN,1,0,1,"
32. MIX.1: AN,2,0,2,"
33. MIX.2: AN,2,0,2,"
34. MIX.3: AN,2,0,2,"
35. ST.NO: AN,4,0,4,"
36. AY: AN,2,0,2,

Record size (bytes) = 251
Total records = 565

Attribute summary of dataset BETA

1. NAME: A,27,0,27,"
2. SSAN: N,9,0,9,"
3. 1: AN,6,0,6,"
4. 11: AN,31,0,31,"
5. 21: AN,4,0,4,"
Record size (bytes) = 510

### Attribute summary of dataset TEST1

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. NAME</td>
<td>AN</td>
<td>27,0,27,</td>
</tr>
<tr>
<td>2. SSAN</td>
<td>AN</td>
<td>9,0,9,</td>
</tr>
<tr>
<td>3. RANK</td>
<td>AN</td>
<td>2,0,2,</td>
</tr>
<tr>
<td>4. ST.NO</td>
<td>AN</td>
<td>4,0,4,</td>
</tr>
<tr>
<td>5. COMP</td>
<td>AN</td>
<td>5,0,5,</td>
</tr>
</tbody>
</table>
6. AERO: AN, 9, 0, 9,
7. PAFSC: AN, 6, 0, 6,
8. PAS: AN, 3, 0, 3,
9. SEX: AN, 1, 0, 1,
10. RACE: AN, 3, 0, 3,
11. MAR. ST: AN, 1, 0, 1,
12. COMM: AN, 7, 0, 7,
13. ED. LEVEL: AN, 4, 0, 4,
14. H. ORG: AN, 3, 0, 3,
15. PME. 3: AN, 5, 0, 5,
16. DOH: AN, 6, 0, 6,
17. MIX. 1: AN, 2, 0, 2,
18. MIX. 2: AN, 2, 0, 2,
19. MIX. 3: AN, 2, 0, 2,
20. MIX. X: AN, 2, 0, 2,
21. PAS: AN, 3, 0, 3,
22. WING: AN, 1, 0, 1,
23. USAF: AN, 1, 0, 1,
24. NOMAST: AN, 1, 0, 1,
25. SRORG: AN, 1, 0, 1,
26. PILOT: AN, 1, 0, 1,
27. NAVIGATOR: AN, 1, 0, 1,
28. SINGLE: AN, 1, 0, 1,
29. USAFA: AN, 1, 0, 1,
30. NAVY: AN, 1, 0, 1,
31. ARMY: AN, 1, 0, 1,
32. RESNGUSMC: AN, 1, 0, 1,
33. MINORITY: AN, 1, 0, 1,
34. FEMALE: AN, 1, 0, 1,
35. RANKC: AN, 1, 0, 1,
36. NONLINE: AN, 1, 0, 1,
37. SRFME: AN, 1, 0, 1,
38. TOPPER: AN, 1, 0, 1,
39. COMMSK: AN, 1, 0, 1,
40. TACOPSX: AN, 1, 0, 1,
41. STRATOPSX: AN, 1, 0, 1,
42. PPSSK: AN, 1, 0, 1,
43. ACOLOG: AN, 1, 0, 1,
44. SL1: AN, 1, 0, 1,
45. SL2: AN, 1, 0, 1,
46. SL3: AN, 1, 0, 1,
47. SLX: AN, 1, 0, 1,
48. ASL1: AN, 1, 0, 1,
49. ASL2: AN, 1, 0, 1,
50. ASL3: AN, 1, 0, 1,
51. ASLX: AN, 1, 0, 1,
52. SOS: AN, 1, 0, 1,
53. AR1: AN, 1, 0, 1,
54. SRO: AN, 1, 0, 1,
55. CC: AN, 1, 0, 1,
56. IO: AN, 1, 0, 1,
57. NRANK: AN, 2, 0, 2,
Record size (bytes) = 158
Total records = 158

Attribute summary of dataset TEST2

1. NAME: AN,27,0,27,
2. SSAN: AN,9,0,9,
3. RANK: AN,2,0,2,
4. ST.NO: AN,4,0,4,
5. COMP: AN,5,0,5,
6. AERO: AN,9,0,9,
7. PAFSC: AN,6,0,6,
8. PAS: AN,3,0,3,
9. SEX: AN,1,0,1,
10. RACE: AN,3,0,3,
11. MAR.ST: AN,1,0,1,
12. COMM: AN,7,0,7,
13. ED.LEVEL: AN,4,0,4,
14. H.ORG: AN,3,0,3,
15. PME.3: AN,5,0,5,
16. DOR: AN,6,0,6,
17. MIX.1: AN,2,0,2,
18. MIX.2: AN,2,0,2,
19. MIX.3: AN,2,0,2,
20. MIX.X: AN,2,0,2,
21. PAS: AN,3,0,3,
22. WING: AN,1,0,1,
23. USAF: AN,1,0,1,
24. NOMAST: AN,1,0,1,
25. SRORG: AN,1,0,1,
26. PILOT: AN,1,0,1,
27. NAVIGATOR: AN,1,0,1,
28. SINGLE: AN,1,0,1,
29. USAFA: AN,1,0,1,
30. NAVY: AN,1,0,1,
31. ARMY: AN,1,0,1,
32. RESNGUSMC: AN,1,0,1,
33. MINORITY: AN,1,0,1,
34. FEMALE: AN,1,0,1,
35. RANKC: AN,1,0,1,
36. NONLINE: AN,1,0,1,
37. SRPME: AN,1,0,1,
38. TOPPER: AN,1,0,1,
39. COMMSK: AN,1,0,1,
40. TACOPSK: AN,1,0,1,
41. STRATOPS: AN,1,0,1,
42. PPBSK: AN,1,0,1,
43. ACQLOG: AN,1,0,1,
<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th>Length</th>
<th>Minimum</th>
<th>Maximum</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAME</td>
<td>AN</td>
<td>27</td>
<td>0</td>
<td>27</td>
<td></td>
</tr>
<tr>
<td>SSAN</td>
<td>AN</td>
<td>9</td>
<td>0</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>ST.NO</td>
<td>AN</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>DOR</td>
<td>AN</td>
<td>6</td>
<td>0</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>MIX.1</td>
<td>AN</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>MIX.2</td>
<td>AN</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>MIX.3</td>
<td>AN</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>MIX.X</td>
<td>AN</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>WING</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>USAF</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>NOMAST</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>SORG</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PILOT</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>NAVIGATOR</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>SINGLE</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>USAFA</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>NAVY</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>ARMY</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>RESNGUSMC</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>MINORITY</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>FEMALE</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>RANK.C</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>NONLINE</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>SRPME</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>TOPPER</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>COMMSK</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>STRATOPSK</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>TACKOPSK</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>PPBSSK</td>
<td>AN</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

Record size (bytes) = 158
Total records = 141

Attribute summary of dataset CINTER
<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>30. ACQLOG:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31. SL1:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32. SL2:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33. SL3:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34. SLX:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35. ASL1:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36. ASL2:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37. ASL3:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38. ASLX:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39. SOS:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40. ARI:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41. SRO:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42. CC:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43. IO:</td>
<td>AN, 1, 0, 1, &quot;</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Record size (bytes) = 90
Total records = 141
APPENDIX

SMSS ZBASIC Source Code
REM ************************************************************************
REM ** PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS) **
REM ** FILE NAME: SMSS.BAS DATE: 18 FEB 1986 **
REM ** FUNCTION: MAIN SYSTEM MENU **
REM ** COMPUTER: ZENITH 120 LANGUAGE: BASIC **
REM ** AUTHOR: KEN RITCHHART **
REM ************************************************************************
DIM WING$(5, 6)
DIM STAT(18,5,12)
DIM STATS(22)
DIM RULES(22)
DIM SEMI(5,12,15,3)
COMMON MIX, SFLG, WING$(,), STAT(), SEMI(), RULE$(,), PLE$, TRONS, SU%, MU%
GOSUB 945
GDAT = 0: TRONS = 1
BLK$ = "" "
EMESS1$="ERROR - THE FIRST & LAST "+SUNIT$+" NUMBERS DON'T AGREE WITH TOT.
EMESS2$="YOU ARE MISSING SOME "+SUNIT$+" FROM THIS "+MUNIT$+"" ENTER THE MISSING UNIT"
GOSUB 2075
IF SFLG = 1 THEN GDAT = 1: GOSUB 2075
CLS
GOSUB 3125
SCREEN 0,1: LOCATE 1,1
PRINT "SMSS MAIN MENU ";
PRINT ": SCREEN ,0
LOCATE 3,31: PRINT "WELCOME TO THE"
LOCATE 4,25: PRINT SCHOOL$
LOCATE 6,23: SCREEN ,1
PRINT "STUDENT MIX SOFTWARE SYSTEM (SMSS)"
SCREEN 0: LOCATE 8,25: PRINT "CREATED BY: MAJOR KEN RITCHHART"
LOCATE 9,37: PRINT "MAJOR BOB SIMMONS"
LOCATE 11,10: PRINT "ENTER THE NUMBER FOR THE DESIRED OPTION - 
LOCATE 13,20: PRINT "1) SYSTEM CONFIGURATION"
LOCATE 14,20: PRINT "2) MIX ASSIGNMENT RULES"
LOCATE 15,20: PRINT "3) RUN THE MIX PROGRAMS"
LOCATE 16,20: PRINT "4) REVIEW MIX DISTRIBUTION STATISTICS"
LOCATE 18,20: PRINT "6) EXIT TO SYSTEM LEVEL"
LOCATE 17,20: PRINT "5) REGISTER MANUAL CHANGES TO SEMINARS"
LOCATE 20,10: PRINT "ENTER YOUR CHOICE: ";
INPUT C
LOCATE 14,2": PRINT "
IF C = 1 THEN GOSUB 370
IF C = 2 THEN GOSUB 980
IF C = 3 THEN GOSUB 1585
IF C = 4 THEN GOSUB 2075
350 IF C = 5 THEN CLOSE: CHAIN "SMSSUPDT"
355 IF C = 6 THEN CLOSE: CLS: SYSTEM
360 GOTO 220
365 END
370 REM
375 REM "******************************************************************************
380 REM **
385 REM ** PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS) **
390 REM ** FILE NAME: SMSS1.BAS DATE: 27 NOV 1985 **
395 REM ** FUNCTION: SYSTEM CONFIGURATION MENU **
400 REM ** COMPUTER: ZENITH 120 LANGUAGE: BASIC **
405 REM ** AUTHOR: KEN RITCHHART **
410 REM **
415 REM "******************************************************************************
420 CLS
425 GOSUB 3125
430 SCREEN ,1: LOCATE 1,1
435 PRINT ": SYSTEM CONFIGURATION MENU - SMSS1"
440 PRINT ": SCREEN ,0
445 LOCATE 3,10: PRINT "THE DEFAULTS FOR THIS SYSTEM SET UP FOR ACSC. YOU CAN"
450 LOCATE 4,10: PRINT "CHANGE THE NAMES OF THE ORGANIZATIONAL UNITS AND THE"
455 LOCATE 5,10: PRINT "NUMBER OF UNITS BELOW - OR HIT RETURN TO LEAVE AS IS."
460 SCREEN ,1
465 LOCATE 7,10: PRINT "SCHOOL NAME:"
470 LOCATE 8,10: PRINT "MAJOR UNITS:"
475 LOCATE 8,50: PRINT "NUMBER OF UNITS:"
480 LOCATE 9,10: PRINT "NEXT SUBUNITS:"
485 LOCATE 9,50: PRINT "TOTAL # OF SUBUNITS:"
490 SCREEN ,0
495 LOCATE 7,25: PRINT "SCHOOL":
500 LOCATE 8,25: PRINT "MUNIT":
505 LOCATE 8,70: PRINT "MU"
510 LOCATE 9,25: PRINT "UNITS:"
515 LOCATE 9,73: PRINT "SU": SCREEN ,1
520 LOCATE 12,8: PRINT "MUNIT:"
525 LOCATE 11,18: PRINT "NUMBER OF"
530 LOCATE 12,18: PRINT "SUNITS:""S"
535 LOCATE 11,30: PRINT "MUNIT:
540 LOCATE 12,7: PRINT "TENORITY"
545 LOCATE 11,43: PRINT "FIRST"
550 LOCATE 12,43: PRINT "SUNITS"
555 LOCATE 11,55: PRINT "LAST"
560 LOCATE 12,55: PRINT "SUNITS"
565 LOCATE 11,66: PRINT "MISSING"
570 LOCATE 12,66: PRINT "SUNITS:""S": SCREEN ,0
575 FOR I% = 1 TO MU%
580 LOCATE 13+I%,10: PRINT WING$(I%,1)
585 LOCATE 13+I%,22: PRINT WING$(I%,2)
590 LOCATE 13+I%,34: PRINT WING$(I%,3)
595 LOCATE 13+I%,45: PRINT WING$(I%,4)
LOCATE 13+I%, 58: PRINT WINGS(I%, 5);
605 LOCATE 13+I%, 67: PRINT WINGS(I%, 6);
610 NEXT I%
615 LOCATE 20, 10: PRINT "ENTER C) TO CHANGE THE DEFAULTS";
620 LOCATE 21, 16: PRINT "X) TO EXIT BACK TO THE PREVIOUS MENU.");
625 LOCATE 23, 10: PRINT "ENTER YOUR CHOICE: ";
630 IF FLAG = 1 THEN FLAG = 0: RETURN
635 INPUT CH$
640 LOCATE 22, 16: PRINT "
645 IF CH$ = "C" THEN GOTO 675
650 IF CH$ = "c" THEN GOTO 675
655 IF CH$ = "X" THEN RETURN
660 IF CH$ = "x" THEN RETURN
665 LOCATE 22, 16: SCREEN , 1: PRINT "INCORRECT CHOICE - TRY AGAIN": SCREEN , 0
670 GOTO 625
675 LOCATE 7, 25: INPUT "", TEMP$
680 IF TEMPS <> "" THEN SCHOOL$ = TEMPS
685 LOCATE 8, 25: INPUT "", TEMPS
690 IF TEMPS <> "" THEN MUNIT$ = TEMPS
695 LOCATE 8, 71: INPUT "", TEMPS
700 IF TEMPS <> 0 THEN MU% = TEMPS
705 LOCATE 9, 25: INPUT "", TEMPS
710 IF TEMPS <> "" THEN SUNT$ = TEMPS
715 LOCATE 9, 74: INPUT "", TEMPS
720 IF TEMP% <> 0 THEN S" = TEMPS
725 FOR I = 7 TO 18
730 LOCATE 13+I%, 8: PRINT "
735 PRINT "
740 NEXT I
745 FLAG = 1: GOSUB 460
750 FOR I% = 1 TO MU%
755 LOCATE 13+I%, 10: INPUT "", TEMPS
760 IF TEMPS <> "" THEN WING$(I%, 1) = TEMPS
765 LOCATE 13+I%, 22: INPUT "", TEMPS
770 IF TEMPS <> "" THEN WING$(I%, 2) = TEMPS
775 LOCATE 13+I%, 34: INPUT "", TEMPS
780 IF TEMPS <> "" THEN WING$(I%, 3) = TEMPS
785 LOCATE 13+I%, 45: INPUT "", TEMPS
790 IF TEMPS <> "" THEN WING$(I%, 4) = TEMPS
795 LOCATE 13+I%, 58: INPUT "", TEMPS
800 IF TEMPS <> "" THEN WING$(I%, 5) = TEMPS
805 W2 = VAL(WINGS(I%, 2)); W3 = VAL(WINGS(I%, 3))
810 W4 = VAL(WINGS(I%, 4)); W5 = VAL(WINGS(I%, 5))
815 IF (W5 - W4 + 1) > W2 THEN GOTO 820 ELSE GOTO 855
820 LOCATE 19, 10: PRINT EMESS2$;
825 LOCATE 20, 10: PRINT EMESS3$;
830 LOCATE 13+I%, 67: INPUT "", TEMPS
835 LOCATE 19, 10: PRINT BLK$ + BLK$
840 LOCATE 20, 10: PRINT BLK$ + BLK$
845 IF TEMPS <> "" THEN WING$(I%, 6) = TEMPS
850    GOTO 865
855    WINGS(I%,6) = ""
860    LOCATE 13+I%,67; PRINT " ";
865    IF (W5 - W4 + 1) < W2 THEN LOCATE 18,12; PRINT EMFSS1$; : GOTO 765
870    LOCATE 18,12; PRINT BLK$ + BLK$ + " ";
875 NEXT I%
880 GOSUB 890
885 COTO 495
890 REM *** WRITE OUT SCHOOL DATA ***
895 OPEN "O",#1,"F:SMSSCHOOL.DAT"
900 PRINT #1, SCHOOL$; "; MUNIT$; "; SUNIT$; "; MU%; "; SU% 
905 FOR I = 1 TO MU%
910    PRINT #1, WINGS(I,1);": "; WING$(I,2);": "; WINGS(I,3);": "; WINGS(I,4);": "; WINGS(I,5);": "; WINGS(I,6)
915 NEXT I
920 CLOSE #1
925 REM ** END SCHOOL DATA OUTPUT ***
930 RETURN
935 REM *** RETRIEVE SCHOOL DATA ***
940 OPEN "I",#1,"F:SMSSCHOOL.DAT"
945 INPUT #1, SCHOOL$, MUNIT$, SUNIT$, MU%, SU%
950 FOR I = 1 TO MU%
960    INPUT #1, WINGS(I,1), WINGS(I,2), WINGS(I,3), WINGS(I,4), WINGS(I,5)
965 NEXT I
970 CLOSE #1; RETURN
975 REM *** END RETRIEVE SCHOOL DATA ***
980 REM *** SMSS2.BAS ***
985 REM ***********************************************
990 REM **
995 REM ** PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS)
1000 REM ** FILE NAME: SMSS2.BAS DATE: 27 NOV 1985 
1005 REM ** FUNCTION: MIX ASSIGNMENT RULES MENU
1010 REM ** COMPUTER: ZENITH 120 LANGUAGE: BASIC
1015 REM ** AUTHOR: KEN RITCHHART
1020 REM **
1025 REM ***********************************************
1030 BLK$ = " ";
1035 CLS
1040 GOSUB 3125
1045 GOSUB 1550
1050 SCREEN ,1: LOCATE 1,1
1055 PRINT " MIX ASSIGNMENT RULES MENU - SMSS2"
1060 PRINT " ";
1065 SCREEN ,0: LOCATE 4,10
1070 PRINT "1) PRIORITY GIVEN TO COMMUNICATION SKILLS: ";
1075 LOCATE 4,73: PRINT RULES(1): : LOCATE 5,10
1080 PRINT "2) PRIORITY GIVEN TO PPBS SKILLS: ";
1085 LOCATE 5,73: PRINT RULES(2): : LOCATE 6,10
1090 PRINT "3) PRIORITY GIVEN TO TACTICAL OPERATION SKILLS: ";
1095 LOCATE 6,73: PRINT RULES(3): : LOCATE 7,10
1100 PRINT "4) ARI STUDENTS DO NOT CHANGE SEMINARS: ";
1105 LOCATE 7,73: PRINT RULE$(4);: LOCATE 8,10
1110 PRINT "5) SOS STUDENTS DO NOT CHANGE SEMINARS: ";
1115 LOCATE 8,73: PRINT RULE$(5);: LOCATE 9,10
1120 PRINT "6) IO'S DO NOT CHANGE SEMINARS (OR 'X' TO DELETE): ";
1125 LOCATE 9,73: PRINT RULE$(6);: LOCATE 10,10
1130 PRINT "7) PRIORITY GIVEN TO STRATEGIC OPERATION SKILLS: ";
1135 LOCATE 10,73: PRINT RULE$(7);: LOCATE 11,10
1140 PRINT "8) STUDENTS MAY NOT HAVE BEEN ASSIGNED TO THIS SEMINAR BEFORE: ";
1145 LOCATE 11,73: PRINT RULE$(8);: LOCATE 12,10
1150 PRINT "9) MAX # OF STUDENTS PREVIOUSLY ASSIGNED WITH: ";
1155 LOCATE 12,73: PRINT RULE$(9);: LOCATE 13,10
1160 LOCATE 14,10
1165 PRINT "11) EVENLY DISTRIBUTE THE FOLLOWING STUDENTS: ";
1170 LOCATE 15,15: PRINT "ARMY";: LOCATE 15,35: PRINT RULE$(11);
1175 LOCATE 15,40: PRINT "NAVY";: LOCATE 15,55: PRINT RULE$(12);
1180 LOCATE 15,60: PRINT "RATED";: LOCATE 15,75: PRINT RULE$(13);
1185 LOCATE 16,15: PRINT "RES,NG,MC";: LOCATE 16,35: PRINT RULE$(14);
1190 LOCATE 16,40: PRINT "BY SEX";: LOCATE 16,55: PRINT RULE$(15);
1195 LOCATE 16,60: PRINT "BY RACE";: LOCATE 16,75: PRINT RULE$(16);
1200 LOCATE 17,15: PRINT "NON LINE";: LOCATE 17,35: PRINT RULE$(17);
1205 LOCATE 17,40: PRINT "BY RANK";: LOCATE 17,55: PRINT RULE$(18);
1210 LOCATE 17,60: PRINT "USAFA GRADS";: LOCATE 17,75: PRINT RULE$(19);
1215 LOCATE 18,15: PRINT "ED. LEVEL";: LOCATE 18,35: PRINT RULE$(20);
1220 LOCATE 18,60: PRINT "SINGLE/UNAC";: LOCATE 18,75: PRINT RULE$(22);
1225 SCREEN ,1
1230 LOCATE 19,10: PRINT "CODES: A) ALWAYS, P) PERFERRED, D) DONT CARE"
1235 SCREEN ,0
1240 LOCATE 23,10: PRINT BLKS;
1245 LOCATE 26,10: PRINT "ENTER C) TO CHANGE THE DEFAULTS";
1250 LOCATE 21,16: PRINT "X) TO EXIT BACK TO THE PREVIOUS MENU."
1255 LOCATE 23,10: PRINT "ENTER YOUR CHOICE: ";
1260 IF FLAG = 1 THEN FLAG = 0: RETURN
1265 INPUT CH$
1270 LOCATE 22,16: PRINT ";
1275 IF CH$ = "C" THEN GOTO 1305
1280 IF CH$ = "c" THEN GOTO 1305
1285 IF CH$ = "X" THEN RETURN
1290 IF CH$ = "x" THEN RETURN
1295 LOCATE 22,16: SCREEN ,1: PRINT "INCORRECT CHOICE - TRY AGAIN";: SCREEN ,0
1300 GOTO 1255
1305 REM CHANGE RULE VALUE
1310 SCREEN ,1
1315 FOR I = 1 TO 9
1320 LOCATE 3+I,73: INPUT ", TEMPS
1325 GOSUB 1410
1330 IF FLAG = 1 THEN GOTO 1320
1335 NEXT I
1340 I = 11
1345 FOR J = 15 TO 18
1350 FOR K = 35 TO 75 STEP 20
1355 IF I > 22 THEN GOTO 1385
1360 IF I = 10 OR I = 21 GOTO 1380
1365 LOCATE J, K: INPUT "", TEMP$
1370 GOSUB 1410
1375 IF FLAG = 1 THEN GOTO 1365
1380 I = I + 1
1385 NEXT K
1390 NEXT J
1400 SCREEN 0
1405 GOTO 1240
1410 REM *** PROCESS INPUT CODES ***
1415 IF FLAG = 1 THEN LOCATE 3, 10: SCREEN 0: PRINT BLKS + "": SCREEN 1
1420 IF TEMP$ = "" GOTO 1500
1425 IF I = 9 THEN GOTO 1490
1430 IF I = 10 THEN GOTO 1490
1435 IF TEMP$ = "A" THEN GOTO 1490
1440 IF TEMP$ = "a" THEN TEMP$ = "A": GOTO 1490
1445 IF TEMP$ = "p" THEN TEMP$ = "P": GOTO 1490
1450 IF TEMP$ = "d" THEN TEMP$ = "D": GOTO 1490
1455 IF TEMP$ = "D" THEN GOTO 1490
1460 IF TEMP$ = "D" THEN TEMP$ = "D": GOTO 1490
1465 IF TEMP$ = "x" AND I = 6 THEN GOTO 1490
1470 IF TEMP$ = "x" AND I = 6 THEN TEMP$ = "X": GOTO 1490
1475 LOCATE 3, 10: PRINT "ERROR - INCORRECT ENTRY TRY AGAIN"
1480 FLAG = 1
1485 GOTO 1500
1490 FLAG = 0
1495 RULES(I) = TEMPS
1500 RETURN
1505 REM *** END CODE PROCESSING ***
1510 REM *** WRITE OUT RULES DATA ***
1515 OPEN "O", #2, "F:SMRULE.DAT"
1520 FOR I = 1 TO 22
1525 PRINT #2, RULES(I); ", ";
1530 NEXT I
1535 CLOSE #2
1540 RETURN
1545 REM *** END RULES DATA OUTPUT ***
1550 REM *** RETRIEVE RULES DATA ***
1555 OPEN "I", #2, "F:SMRULE.DAT"
1560 FOR I = 1 TO 22
1565 INPUT #2, RULES(I)
1570 NEXT I
1575 CLOSE #2: RETURN
1580 REM *** END RETRIEVE RULES DATA ***
1585 REM *** SMSS3.BAS ***
1590 REM **********************************************************************
PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS)

FILE NAME: SMSS3.BAS

FUNCTION: RUNNING THE MIX - MENU

COMPUTER: ZENITH 120

AUTHOR: KEN RITCHHART

DATE: 27 NOV 1985

FILE NAME: SMSS3.BAS
FUNCTION: RUNNING THE MIX - MENU - SMSS3

RUNNING THE MIX MENU - SMSS3

CLS
GOSUB 3125
SCREEN ,1: LOCATE 1,1
PRINT "RUNNING THE MIX MENU - SMSS3"
PRINT "
SCREEN ,0: LOCATE 4,10
PRINT "THIS MENU PROVIDES YOU WITH THE CAPABILITY TO RUN THE MIX PROGRAM"
LOCATE 5,10
PRINT "AND PRODUCE A STUDENT DISTRIBUTION BASED ON THE RULES THAT YOU"
LOCATE 6,10
PRINT "TURN ON OR OFF. IF YOU CHOOSE A STANDARD MIX (OPTION A, B, OR C)"
LOCATE 7,10
PRINT "- THE RULES WILL BE RESET TO THE STANDARD DEFAULTS FOR THAT MIX."
LOCATE 8,10
PRINT "YOU CAN THEN CUSTOMIZE OR CHANGE THE RULES AS YOU SEE FIT. YOU"
LOCATE 9,10
PRINT "SHOULD USE OPTION D) SPECIAL - IF YOU WANT TO DO SOMETHING LIKE"
LOCATE 10,10
PRINT "ADDING A MIX 4 OR USING THE EXISTING RULES WITHOUT resetting."
LOCATE 11,10: SCREEN ,1
PRINT "PLEASE NOTE: YOU MUST RUN THE DATA EXTRACTION PROGRAM UNDER THE"
LOCATE 12,10: SCREEN ,1
PRINT "CONDOR DBMS SMSS HELP MENU BEFORE EXECUTING THIS PROGRAM."
LOCATE 14,10: SCREEN ,0
PRINT "OPTIONS:"
LOCATE 16,15: PRINT "A) MIX 1"
LOCATE 17,15: PRINT "B) MIX 2"
LOCATE 18,15: PRINT "C) MIX 3"
LOCATE 19,15: PRINT "D) SPECIAL"
LOCATE 20,15: PRINT "X) RETURN TO MAIN MENU"
LOCATE 22,10: PRINT "ENTER YOUR CHOICE: "; INPUT CH$"; INPUT ",CH$
IF CH$ = "A" THEN GOTO 1845
IF CH$ = "a" THEN GOTO 1845
IF CH$ = "B" THEN GOTO 1885
IF CH$ = "b" THEN GOTO 1885
IF CH$ = "C" THEN GOTO 1930
IF CH$ = "c" THEN GOTO 1930
IF CH$ = "D" THEN GOTO 1990
IF CH$ = "d" THEN GOTO 1990
IF CH$ = "X" THEN RETURN
IF CH$ = "x" THEN RETURN
GOTO 1585
GOSUB 2035
1850 MIX = 1
1855 RULES$(10) = STR$(SU% + MU% + 1)
1860 GOSUB 1510
1865 CLS: GOSUB 3125
1870 SCREEN 1: LOCATE 3, 30: PRINT "MIX 1 RULES";: SCREEN 0
1875 GOSUB 1050
1880 GOTO 2020
1885 GOSUB 2035
1890 MIX = 2
1895 RULES$(1) = "D"
1900 RULES$(2) = "A"
1905 GOSUB 1510
1910 CLS: GOSUB 3125
1915 SCREEN 1: LOCATE 3, 30: PRINT "MIX 2 RULES";: SCREEN 0
1920 GOSUB 1050
1925 GOTO 2020
1930 GOSUB 2035
1935 MIX = 3
1940 RULES$(1) = "D"
1945 RULES$(3) = "A"
1950 RULES$(5) = "p"
1955 RULES$(6) = "X"
1960 RULES$(22) = "A"
1965 GOSUB 1510
1970 CLS: GOSUB 3125
1975 SCREEN 1: LOCATE 3, 30: PRINT "MIX 3 RULES";: SCREEN 0
1980 GOSUB 1050
1985 GOTO 2020
1990 LOCATE 19, 10: PRINT "WHICH MIX IS THIS RUN FOR? ": INPUT ", MIX$
1995 MIX = VAL(MIX$
2000 CLS: GOSUB 3125
2005 LOCATE 3, 30: SCREEN 1: PRINT "SPECIAL RUN FOR MIX "; MIX; : SCREEN 0
2010 GOSUB 1050
2015 GOTO 2020
2020 REM *** MIXING STUDENT ROUTINE ***
2025 CHAIN "SMSSMIXI"
2030 RETURN
2035 REM *** SET DEFAULT RULES ***
2040 RULES$(1) = "A": RULES$(2) = "D": RULES$(3) = "D": RULES$(4) = "A"
2045 RULES$(5) = "A": RULES$(6) = "A": RULES$(7) = "A": RULES$(8) = "A"
2050 RULES$(9) = "1": RULES$(10) = STR$(SU%): RULES$(11) = "A": RULES$(12) = "A"
2055 RULES$(13) = "A": RULES$(14) = "D": RULES$(15) = "A": RULES$(16) = "A"
2060 RULES$(17) = "A": RULES$(18) = "D": RULES$(19) = "D": RULES$(20) = "D"
2065 RULES$(21) = "D": RULES$(22) = "A"
2070 RETURN
2075 REM
2080porn: ***************** ************
2090 REM **
2095 REM ** PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS)
2100 REM ** FILE NAME: SMSS4.BAS DATE: 13 DEC 1985
2105 REM ** FUNCTION: REVIEW MIX STATISTICAL DISTRIBUTION
2110 REM ** COMPUTER: ZENITH 120 LANGUAGE: BASIC
2115 REM ** AUTHOR: KEN RITCHHART
2120 REM **
2115 REM **
2120 REM ************************************************************
2125 CLS
2130 GOSUB 3125
2135 SCREEN ,1: LOCATE 1,1
2140 PRINT " REVIEW MIX STATISTICS MENU - SMSS4"
2145 PRINT " 
2150 SCREEN ,0: LOCATE 4,10
2155 IF GDAT > 0 GOTO 2170
2160 GOSUB 2910
2165 GDAT = 1
2170 LOCATE 10,10: PRINT "STATISTICAL OPTIONS: ";
2175 LOCATE 12,15: PRINT "A) REVIEW ";SCHOOL$;" BY ";MUNIT$;
2180 LOCATE 13,15: PRINT "B) REVIEW ";MUNIT$;"S BY ";SUNIT$;
2185 LOCATE 14,15: PRINT ";X) RETURN TO PREVIOUS MENU"
2190 LOCATE 20,10: PRINT "ENTER YOUR CHOICE: ";: INPUT ",CH$ 
2195 IF CH$ = "A" THEN GOSUB 2295
2200 IF CH$ = "a" THEN GOSUB 2295
2205 IF CH$ = "B" THEN GOTO 2230
2210 IF CH$ = "b" THEN GOTO 2230
2215 IF CH$ = "X" THEN RETURN
2220 IF CH$ = "x" THEN RETURN
2225 GOTO 2075
2230 LOCATE 15,18: PRINT "WHICH ";MUNIT$;" DO YOU WISH TO REVIEW? ";
2235 FOR I = 1 TO MU$
2240 MDD = ((I/4) - INT(I/4)) * 4: IF MDD = 0 THEN MDD = 4
2245 LOCATE (15 + INT((I/4)+.9)),(5+(MDD*15)): PRINT WING$(I,1):
2250 NEXT I
2255 LOCATE 17+(MU$/4),18: PRINT "PLEASE TYPE IN THE ";MUNIT$;" YOU DESIRE ";
2260 INPUT ",CH$: WING = 0
2265 FOR I = 1 TO MU$
2270 IF CH$ = WING$(I,1) THEN WING = I
2275 NEXT I
2280 IF WING = 0 THEN PRINT "INCORRECT CHOICE - TRY AGAIN": GOTO 2255
2285 GOSUB 2530
2290 GOTO 2075
2295 REM DISPLAY SCHOOL STATISTICS
2300 CLS
2305 GOSUB 3125
2310 SCREEN ,1: LOCATE 1,30: PRINT "MIX STATISTICS - OVERALL SCHOOL LEVEL"
2315 LOCATE 3,5: PRINT "ATTRIBUTE"
2320 FOR I = 1 TO MU$
2325 LOCATE 3, 25+INT((55/MU%)*(I-1)): PRINT WING$(I,1):
2330 NEXT I
2335 SCREEN ,0
2340 FOR I = 1 TO 18
2345 LOCATE 3+i,5: PRINT STAT$(I)
2350 FOR J = 1 TO MU$
2355 LOCATE 3+i,24+INT((55/MU%)*(J-1)): PRINT STAT(I,J,0);
NEXT J
2365 NEXT I
2370 LOCATE 23,10: PRINT "WOULD YOU LIKE A HARDCOPY PRINT OF THIS? (Y/N) ";
2375 INPUT ", CH$.
2380 IF CH$ = "Y" THEN GOSUB 2400
2385 IF CH$ = "y" THEN GOSUB 2400
2390 RETURN
2395 REM END SCHOOL STATISTICAL DISPLAY
2400 REM *** PRINT SCHOOL STATISTICS
2405 OPEN "LPT1:" FOR OUTPUT AS #3
2410 PRINT #3, SPC(25); "MIX STATISTICS - OVERALL SCHOOL ";
2415 PRINT #3,
2420 PRINT #3, "ATTRIBUTE"; SPC(6);
2425 SP = INT(55/MU%)
2430 FOR I = 1 TO MU%
2435 PRINT #3, SPC(SP); WING$(I,1);
2440 NEXT I
2445 PRINT #3, : PRINT #3,
2450 FOR I = 1 TO 18
2455 PRINT #3, STAT$(I); SPC(16 - LEN(STAT$(I)));
2460 FOR J = 1 TO MU%
2465 IF STAT(I,J,0) > 99 THEN PRINT #3, SPC(SP-4);: GOTO 2480
2470 IF STAT(I,J,0) > 9 THEN PRINT #3, SPC(SP-3);: GOTO 2480
2475 PRINT #3, SPC(SP-2);
2480 PRINT #3, STAT(I,J,0);
2485 NEXT J
2490 PRINT #3,
2495 NEXT I
2500 FOR I = 1 TO 10
2505 PRINT #3,
2510 NEXT I
2515 CLOSE #3
2520 RETURN
2525 REM *** END PRINT WING STATISTICS ***
2530 REM DISPLAY WING STATISTICS
2535 CLS
2540 GOSUB 3125
2545 SCREEN,1: LOCATE 1,30: PRINT "MIX STATISTICS - FOR "; MUNIT$; " "; WING$(WING,1);
2550 LOCATE 3,5: PRINT "ATTRIBUTE";
2555 SU% = VAL(WING$(WING,2))
2560 I = 0
2565 FOR J = 1 TO SU%
2570 I = I + 1
2575 IF VAL(WING$(WING,4))+I-1 = VAL(WING$(WING,6)) THEN I = I + 1
2580 LOCATE 3, 25+INT((55/SU%)*(J-1)): PRINT VAL(WING$(WING,4))+I-1;
2585 NEXT J
2590 I = 0
2595 SCREEN,0
2600 FOR I = 1 TO 18
2605 LOCATE 3+I,5: PRINT STAT$(I)
2610 K = 0

68
FOR J = 1 TO SU%
    K = K + 1
    IF VAL(WING$(WING,4)) + K - 1 = VAL(WING$(WING,6)) THEN K = K + 1
    LOCATE 3 + I, 25 + INT((55/SU%)*(J - 1)): PRINT STAT(I, WING, K):
NEXT J

NEXT I
K = Ø
LOCATE 23, 10: PRINT "WOULD YOU LIKE A HARDCOPY PRINT OF THIS? (Y/N) ";
INPUT CH$.
IF CH$ = "Y" THEN GOSUB 2680
RETURN
REM END WING STATISTICAL DISPLASY
REM *** PRINT WING STATISTICS ***
OPEN "LPT1:" FOR OUTPUT AS #3
PRINT #3, SPC(30); "MIX STATISTICS - FOR "; MUNIT$; " "; WING$(WING, 1)
PRINT #3, SPC(30); "ATTRIBUTE": SPC(5);
SU% = VAL(WING$(WING, 2))
SP = INT(55/SU%)
I = Ø
FOR J = 1 TO SU%
    I = I + 1
    IF VAL(WING$(WING, 4)) + I - 1 = VAL(WING$(WING, 6)) THEN I = I + 1
    NUM = VAL(WING$(WING, 4)) + I - 1: NUM$ = STR$(NUM)
    PRINT #3, SPC(SP + 1 - LEN(NUM$)); NUM$;
NEXT J
FOR I = 1 TO 18
    PRINT #3, SPC(SP): PRINT #3, STAT$(I); SPC(15 - LEN(STAT$(I)));
    K = Ø
    FOR J = 1 TO SU%
        K = K + 1
        IF VAL(WING$(WING, 4)) + K - 1 = VAL(WING$(WING, 6)) THEN K = K + 1
        IF STAT(I, WING, K) > 99 THEN PRINT SPC(SP - 4): GOTO 2800
        IF STAT(I, WING, K) > 9 THEN PRINT #3, SPC(SP - 3): GOTO 2800
        PRINT #3, SPC(SP - 2);
        PRINT #3, STAT(I, WING, K);
NEXT J
PRINT #3, SPC(SP);
NEXT I
FOR I = 1 TO 10
    PRINT #3,
NEXT I
CLOSE #3
RETURN
REM *** END PRINT WING STATISTICS ***
REM *** INITIALIZE RULES ***
STAT$(1) = "COMM SKILLS": STAT$(2) = "PPBS SKILLS"
STAT$(3) = "TAC OPS SKILL": STAT$(4) = "STRAT OPS SKILL"
STAT$(5) = "ACQ/LOG SKILL": STAT$(6) = "PILOT"
STAT$(7) = "NAVIGATOR": STAT$(8) = "SINGLE/UNAC"
STAT$(9) = "USAFA GRADS": STAT$(10) = "ARMY"
STAT$(11) = "RES/NG/USN/USMC": STAT$(12) = "MINORITIES"
STAT$(13) = "FEMALES": STAT$(14) = "RANK - CAPT"
STAT$(15) = "88xx/89xx/9xxx": STAT$(16) = "SR ORG EXP"
STAT$(17) = "ARI/SOS": STAT$(18) = "NO MASTER ED"
FLE$ = "TRIAL1"

REM GET STATISTICAL DATA
OPFN "I", #4, "F:X" + FLE$ + ".DAT"
FOR I = 0 TO 18
    FOR J = 0 TO 5
        FOR K = 0 TO 12
            IF EOF (4) THEN GOTO 3010
            INPUT #4, STAT(I, J, K)
        NEXT K
    NEXT J
NEXT I
FOR I = 0 TO 5
    FOR J = 0 TO 12
        FOR K = 0 TO 15
            FOR L = 0 TO 3
                IF EOF (4) THEN GOTO 3010
                INPUT #4, SEMI(I, J, K, L)
            NEXT L
        NEXT K
    NEXT J
NEXT I
CLOSE # 4
RETURN

REM END STATISTICAL RETRIEVAL
REM OUTPUT STATS TO DISK
OPEN "O", #4, "F:X" + FLE$ + ".DAT"
FOR I = 1 TO 18
    FOR J = 0 TO 5
        FOR K = 0 TO 12
            PRINT #4, STAT(I, J, K)
        NEXT K
    NEXT J
NEXT I
FOR I = 0 TO 5
    FOR J = 0 TO 12
        FOR K = 0 TO 15
            FOR L = 0 TO 3
                PRINT #4, SEMI(I, J, K, L)
            NEXT L
        NEXT K
    NEXT J
NEXT I
RETURN
3115 CLOSE # 4
3120 RETURN
3125 REM *** CREATE BOX FOR SCREEN ***
3130 SCREEN 1
3135 LOCATE 1,1: PRINT " ";
3140 PRINT " ";
3145 FOR I% = 1 TO 24
3150 LOCATE I%,1: PRINT " ";: LOCATE I%,80: PRINT " ";
3155 NEXT I%
3160 LOCATE 24,1: PRINT " ";
3165 PRINT " ";: LOCATE 1,1
3170 SCREEN 0: RETURN
3175 REM *** END BOX ***
3180 REM
3185 REM *******************************************************************
3190 REM **
3195 REM ** PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS)
3200 REM ** FILE NAME: SMSSBAS.TXT DATE: 2 JAN 1986
3205 REM ** FUNCTION: DOCUMENTATION ON VARIABLES
3210 REM ** COMPUTER: ZENITH 120 LANGUAGE: BASIC
3215 REM ** AUTHOR: KEN RITCHHART
3220 REM **
3225 REM *******************************************************************
3230 REM ***** SYSTEM VARIABLE DOCUMENTATION *****
3235 REM SCHOOL$ - NAME OF THE SCHOOL
3240 REM MUNIT$ - NAME OF SCHOOL'S MAJOR UNITS (IE WINGS)
3245 REM SUNIT$ - NAME OF SCHOOL'S MINOR UNITS (IE SEMINAR)
3250 REM MU% - NUMBER OF MAJOR UNITS IN THE SCHOOL (IE 4)
3255 REM SU% - NUMBER OF MINOR UNITS IN THE SCHOOL (IE 44)
3260 REM WINGS$(8,6) - CHARACTERISTICS OF THE MAJOR UNITS
3265 REM WINGS$(I,1) - WING NAME
3270 REM WINGS$(I,2) - NUMBER OF SUBUNITS IN THE WING - W2
3275 REM WINGS$(I,3) - WING SENORITY (1 IS THE HIGHEST) - W3
3280 REM WINGS$(I,4) - FIRST SEMINAR IN THE WING - W4
3285 REM WINGS$(I,5) - LAST SEMINAR IN THE WING - W5
3290 REM WINGS$(I,6) - MISSING SUBUNITS (SEMINARS) IN THE WING
3295 REM RULES$(25) - NAME OF THE RULE USED FOR MIXING STUDENTS
3300 REM RULES$(1) - COMM SKILLS RULES$(2) - PPBS SKILLS
3305 REM RULES$(3) - OPERATIONAL SKILLS RULES$(4) - AR'S DONT MOVE
3310 REM RULES$(5) - SOS DON'T MOVE RULES$(6) - IO'S DONT MOVE
3315 REM RULES$(7) - RES/NG/USN/USMC RULES$(8) - DIFFRENT SEMINAR
3320 REM RULES$(9) - MAX # PREV STUDENTS RULES$(10) - ACQUISITION/LOG
3325 REM RULES$(11) - ARMY RULES$(12) - NAVY
3330 REM RULES$(13) - RATED RULES$(14) - COMP
3335 REM RULES$(15) - SEX RULES$(16) - RACE
3340 REM RULES$(17) - NON LINE RULES$(18) - RANK (CAPT)
3345 REM RULES$(19) - USAFA GRAD RULES$(20) - ED. LEVEL
3350 REM RULES$(21) - SENIOR PME RULES$(22) - SINGLE/UNACC
3355 REM VALUE OF RULES$(I) - A) ALWAYS, D) DON'T CARE, P) PERFERED
3360 REM CH$ - CHOICE

71
3365 REM TEMP$ - TEMPORARY INPUT VALUE
3370 REM STAT$(18) - NAME OF STATISTICAL ATTRIBUTE AFTER MIX FL:
3375 REM
3380 REM STAT$(1) = "COMM SKILLS": STAT$(2) = "PPBS SKILLS"
3385 REM STAT$(3) = "OPS SKILLS": STAT$(4) = "ED. NO MASTERS"
3390 REM STAT$(5) = "SR. ORG. EXP.": STAT$(6) = "PILOT"
3395 REM STAT$(7) = "NAVIGATOR": STAT$(8) = "SINGLE/UNAC"
3400 REM STAT$(9) = "USAFA GRADS": STAT$(10) = "ARMY"
3405 REM STAT$(11) = "RES/NG/USN/USMC": STAT$(12) = "MINORITIES"
3410 REM STAT$(13) = "FEMALES": STAT$(14) = "RANK - CAPT"
3415 REM STAT$(15) = "88xx/89xx/9xxx": STAT$(16) = "ACO/LOG"
3420 REM STAT$(17) = "ARI/SOS": STAT$(18) = "TOP PERFORMER"
3425 REM INPUT DATA FILE "F:BASICIF" & OUTPUT FILE "F:BASICOT"
3430 REM
3435 REM SNAME [NAME]
3440 REM SSN [SSAN]
3445 REM STN [ST.NO]
3450 REM DOR [DOR]
3455 REM MIX1 [MIX.1]
3460 REM MIX2 [MIX.2]
3465 REM MIX3 [MIX.3]
3470 REM MIXX [MIX.X]
3475 REM AWING [WING]
3480 REM USAF DERIVE FROM [COMP]
3485 REM NOED DERIVE FROM [ED.LEVEL] NO MASTERS DEGREE
3490 REM HORG DERIVE FROM [H.ORG] MAJCOM/SOA/USAF/DOD
3495 REM PILOT DERIVE FROM [AFRO] PILOT OR SR. PILOT
3500 REM NAV DERIVE FROM [AFRO] NAV OR SR. NAV
3505 REM SING DERIVE FROM [MAR.ST] U, D, S
3510 REM AFA DERIVE FROM [COMP] AF ACAD
3515 REM NAVY DERIVE FROM [COMP]
3520 REM ARMY DERIVE FROM [COMP]
3525 REM RES DERIVE FROM [COMP] USMC,ANG,AFRES
3530 REM RACE DERIVE FROM [RACE] BLACK OR OTHER
3535 REM FEM DERIVE FROM [SEX] F
3540 REM RNK DERIVE FROM [RANK] 03
3545 REM NOLINE DERIVE FROM [PAFSC] 88xx,89xx,9xxx
3550 REM SRPME DERIVE FROM [PMO.3]
3555 REM TOPPER ****************
3560 REM COMM ****************
3565 REM OPS ****************
3570 REM PPR$ ****************
3575 REM SL1 ****************
3580 REM SL2 ****************
3585 REM SL3 ****************
3590 REM SLX ****************
3595 REM ASL1 ****************
3600 REM ASL2 ****************
3605 REM ASL3 ****************
3610 REM SOS ****************
3615 REM ARI ****************

72
3615 REM SRO ***********
3620 REM CC ***********
3625 REM IO ***********
3630 REM PAFSC [PAFSC]
**PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS)**

**FILE NAME:** SMSSMIXI.BAS  **DATE:**  2 FEB 1986 PM  

**FUNCTION:** PROCESS INPUT DATA FOR SMSSMIX

**COMPUTER:** ZENITH 120  **LANGUAGE:** BASIC

**AUTHOR:** KEN RITCHHART

---

```basic
4000 REM
4005 REM ********************************************
4010 REM **
4015 REM ** PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS) **
4020 REM ** FILE NAME: SMSSMIXI.BAS DATE: 2 FEB 1986 PM **
4025 REM ** FUNCTION: PROCESS INPUT DATA FOR SMSSMIX **
4030 REM ** COMPUTER: ZENITH 120 LANGUAGE: BASIC **
4035 REM ** AUTHOR: KEN RITCHHART **
4040 REM **
4045 REM ********************************************
4050 DIM TEMPIN$(44), ALOT(18,5,12)
4055 DIM WING$(5,6), STAT(18,5,12), SEMI(5,12,15,3), RULE$(22)
4060 COMMON MIX, SFLAG, WING$(), STAT$(), SEMI(), RULE$(), FLES, TRONS, SU$, MU$
4065 CLS: PRINT "RUNNING SMSS MIX INPUT PROCESSING ": PRINT
4070 GOSUB 4500
4075 REM ********** PROCESS INPUT DATA **********
4080 OPEN "I", #3, "F:BASICIF."
4085 OPEN "R", #4, "F:SMSTNT1.DAT"
4090 OPEN "R", #5, "F:SMSTNT2.DAT"
4095 FIELD #4, 27 AS SNAME$, 9 AS SSN$, 4 AS STN$, 6 AS DOR$, 2 AS MIX1$, 2 AS MIX2$, 2 AS MIX3$, 2 AS MIXX$, 1 AS AWING$, 1 AS USAF$, 1 AS NOED$, 1 AS HORG$, 1 AS PILOT$, AS NAV$, 1 AS SING$, 1 AS AFA$, 1 AS NAVY$, 1 AS ARMY$, 1 AS RES$, 1 AS RACE$, 1 AS PI M$, 1 AS RNK$
4100 FIELD #5, 1 AS NOLINE$, 1 AS SRPME$, 1 AS TOPER$, 1 AS COMM$, 1 AS TOPS$, 1 AS S$, 1 AS PPS$, 1 AS PPBS$, 1 AS AQLOG$, 1 AS SL$, 1 AS SL1$, 1 AS SL2$, 1 AS SL3$, 1 AS SLX$, 1 AS ASL$, 1 AS ASL2$, 1 AS ASL3$, 1 AS ASLX$, 1 AS SOS$, 1 AS ARI$, 1 AS SRO$, 1 AS CCS$, 1 AS IOS$
4105 KO = 1: PRINT: PRINT "READING ";
4110 FOR KS = 1 TO 600
4115 IF KS/10 >= KO THEN PRINT ".":; KO = KO + 1
4120 STNDT = SI
4125 FOR J = 1 TO 44
4130 IF EOF (3) THEN GOTO 4455
4135 INPUT #3, TEMPIN$(J)
4140 NEXT J
4145 LSET SNAME$ = TEMPIN$(1)
4150 LSET SSN$ = TEMPIN$(2)
4155 LSET STN$ = TEMPIN$(3)
4160 LSET DOR$ = TEMPIN$(4)
4165 LSET MIX1$ = TEMPIN$(5)
4170 LSET MIX2$ = TEMPIN$(6)
4175 LSET MIX3$ = TEMPIN$(7)
4180 LSET MIXX$ = TEMPIN$(8)
4185 LSET AWING$ = TEMPIN$(9)
4190 LSET USAF$ = TEMPIN$(10)
4195 LSET NOED$ = TEMPIN$(11)
4200 LSET HORG$ = TEMPIN$(12)
4205 LSET PILOT$ = TEMPIN$(13)
4210 LSET NAV$ = TEMPIN$(14)
4215 LSET SING$ = TEMPIN$(15)
4220 LSET AFA$ = TEMPIN$(16)
4225 LSET NAVY$ = TEMPIN$(17)
4230 LSET ARMY$ = TEMPIN$(18)
```
LSET RES$ = TEMPIN$(19)
LSET RACE$ = TEMPIN$(20)
LSET FEM$ = TEMPIN$(21)
LSET RNK$ = TEMPIN$(22)
LSET NOLINE$ = TEMPIN$(23)
LSET SRPME$ = TEMPIN$(24)
LSET TOPPER$ = TEMPIN$(25)
LSET COMM$ = TEMPIN$(26)
LSET SOPS$ = TEMPIN$(27)
LSET TOPS$ = TEMPIN$(28)
LSET PPBS$ = TEMPIN$(29)
LSET ACQLOG$ = TEMPIN$(30)
LSET SL1$ = TEMPIN$(31)
LSET SL2$ = TEMPIN$(32)
LSET SL3$ = TEMPIN$(33)
LSET SLX$ = TEMPIN$(34)
LSET ASL1$ = TEMPIN$(35)
LSET ASL2$ = TEMPIN$(36)
LSET ASL3$ = TEMPIN$(37)
LSET ASLX$ = TEMPIN$(38)
LSET SOS$ = TEMPIN$(39)
LSET ARI$ = TEMPIN$(40)
LSET SRO$ = TEMPIN$(41)
LSET CC$ = TEMPIN$(42)
LSET IO$ = TEMPIN$(43)
LSET PAFC$ = TEMPIN$(44)

IF IO$ = "Y" AND RULE$ = "X" GOTO 4450

REM ***** RECORD SCHOOL OVERALL STATISTICS **************

JJ = 0
KK = 0
GOSUB 4770

IF AWING$ = " " GOTO 4435

REM *** POST WING STATISTICS *****

FOR J = 1 TO MU$

IF AWING$ <> WINGS$(J, 1) GOTO 4430

JJ = J
KK = 0

GOSUB 4770

GOTO 4435

NEXT J

SI = SI + 1
PUT #4, SI
PUT #5, SI

NEXT KS

STDNT = SI

PRINT "PROCESSED IN "; KS-1; " STUDENTS INTO SMSS"
PRINT STDNT; " STUDENTS KEPT FOR FURTHER PROCESSING"

STAT(0, 0, 0) = STDNT

GOSUB 4665

REM END OF DATA INPUT

GOSUB 4920

GOTO 5045

'OUTPUT STATS TO SMSS
PRINT: PRINT
REM ***** INITIALIZE STAT & GET RULES
PRINT "INITIALIZING ";
FOR I = 0 TO 18
    PRINT ".";
FOR J = 0 TO 5
    FOR K = 0 TO 12
        STAT(I,J,K) = 0
    NEXT K
NEXT J
NEXT I
FOR I = 0 TO 5
    FOR J = 0 TO 12
        FOR K = 0 TO 15
            FOR L = 0 TO 3
                SEMI(I,J,K,L) = 0
            NEXT L
        NEXT K
    NEXT J
NEXT I
REM * RETRIEVE SCHOOL DATA **
OPEN "I", #1, "F:SMSCHOOL.DAT"
INPUT #1, SCHOOL$, MUNIT$, SUNIT$, MU$, SU$
FOR I = 1 TO MU$
    INPUT #1, WING$(I,1),WING$(I,2),WING$(I,3),WING$(I,4),WING$(I,5), WING$(I,6)
NEXT I
CLOSE #1
REM ***** RETRIEVE RULES DATA *****
OPEN "I", #1, "F:SMRULE.DAT"
FOR I = 1 TO 22
    INPUT #1, RULE$(I)
NEXT I
CLOSE #1
RETURN
REM **********PRINT OUT INPUT STATISTICS **********
PRINT 
PRINT "THERE WERE "; STAT(1,0,0); "STUDENTS WITH COMMUNICATION SKILLS"
PRINT "THERE WERE "; STAT(2,0,0); "STUDENTS WITH PPBS SKILLS"
PRINT "THERE WERE "; STAT(3,0,0); "STUDENTS WITH TAC OPS SKILLS"
PRINT "THERE WERE "; STAT(4,0,0); "STUDENTS WITH STRAT OPS SKILLS"
PRINT "THERE WERE "; STAT(5,0,0); "ACQUISITION / LOGISTICS STUDENTS"
PRINT "THERE WERE "; STAT(6,0,0); "STUDENT PILOTS"
PRINT "THERE WERE "; STAT(7,0,0); "STUDENT NAVIGATORS"
PRINT "THERE WERE "; STAT(8,0,0); "SINGLE STUDENTS"
PRINT "THERE WERE "; STAT(9,0,0); "STUDENT USAFA GRADS"
PRINT "THERE WERE "; STAT(10,0,0); "STUDENTS FROM THE ARMY"
PRINT "THERE WERE "; STAT(11,0,0); "STUDENT AFRES/ ANG/ USMC/ NAVY"
PRINT "THERE WERE "; STAT(12,0,0); "STUDENT MINORITIES"
PRINT "THERE WERE "; STAT(13,0,0); "STUDENT FEMALES"
PRINT "THERE WERE "; STAT(14,0,0); "CAPTAINS"
PRINT "THERE WERE "; STAT(15,0,0); "STUDENT 88XX, 89XX, 9XXX"
PRINT "THERE WERE "; STAT(16,0,0); "STUDENTS WITH SR ORG EXPERIENCE"
PRINT "THERE WERE "; STAT(17,0,0); "SOS / ARI STUDENTS"
PRINT "THERE WERE "; STAT(18,0,0); "WITH NO MASTERS ED"

RETURN

REM *************** RECORD STATISTICS **************************************
IF JJ <> 0 THEN STAT(0,JJ,0)=STAT(0,JJ,0) + 1
IF KK = 0 GOTO 4820
FOR MM = 1 TO 15
  IF SEMI(JJ,KK,MM,0) <> 0 THEN GOTO 4815
  SEMI(JJ,KK,MM,0)=SI: SEMI(JJ,KK,0,0)=SEMI(JJ,KK,0,0)
  SEMI(JJ,KK,MM,1)=VAL(MIX1$): SEMI(JJ,KK,MM,2)=VAL(MIX2$)
  SEMI(JJ,KK,MM,3)=VAL(MIX3$)
  GOTO 4820
NEXT MM

IF COMM$ = "Y" THEN STAT(1,JJ,KK) = STAT(1,JJ,KK) + 1
IF PPBS$ = "Y" THEN STAT(2,JJ,KK) = STAT(2,JJ,KK) + 1
IF TOP$ = "Y" THEN STAT(3,JJ,KK) = STAT(3,JJ,KK) + 1
IF SOPS$ = "Y" THEN STAT(4,JJ,KK) = STAT(4,JJ,KK) + 1
IF ACQLOG$ = "Y" THEN STAT(5,JJ,KK) = STAT(5,JJ,KK) + 1
IF PILOT$ = "Y" THEN STAT(6,JJ,KK) = STAT(6,JJ,KK) + 1
IF NAV$ = "Y" THEN STAT(7,JJ,KK) = STAT(7,JJ,KK) + 1
IF SING$ = "Y" THEN STAT(8,JJ,KK) = STAT(8,JJ,KK) + 1
IF APA$ = "Y" THEN STAT(9,JJ,KK) = STAT(9,JJ,KK) + 1
IF ARMY$ = "Y" THEN STAT(10,JJ,KK) = STAT(10,JJ,KK) + 1
IF RES$ = "Y" OR NAVY$ = "Y" THEN STAT(11,JJ,KK) = STAT(11,JJ,KK) + 1
IF RACE$ = "Y" THEN STAT(12,JJ,KK) = STAT(12,JJ,KK) + 1
IF FEM$ = "Y" THEN STAT(13,JJ,KK) = STAT(13,JJ,KK) + 1
IF RNK$ = "Y" THEN STAT(14,JJ,KK) = STAT(14,JJ,KK) + 1
IF NOLINE$ = "Y" THEN STAT(15,JJ,KK) = STAT(15,JJ,KK) + 1
IF HORG$ = "Y" THEN STAT(16,JJ,KK) = STAT(16,JJ,KK) + 1
IF ARS$ = "Y" OR SOS$ = "Y" THEN STAT(17,JJ,KK) = STAT(17,JJ,KK) + 1
IF NOED$ = "Y" THEN STAT(18,JJ,KK) = STAT(18,JJ,KK) + 1

RETURN

REM *************** END STATISTICS ******************************************
REM *************** OUTPUT STATISTICS ***************************************
OPEN "O", #1, "F:XTRIAL1.DAT"
FOR I = 0 TO 18
  PRINT ";
FOR J = 0 TO 5
  FOR K = 0 TO 12
    PRINT #1, STAT(I,J,K)
  NEXT K
NEXT J
NEXT I
FOR I = 0 TO 5
  PRINT ";
FOR J = 0 TO 12
  FOR K = 0 TO 15
    FOR L = 0 TO 3
5000 PRINT #1, SEMI(I,J,K,L)
5005 NEXT L
5010 NEXT K
5015 NEXT J
5020 NEXT I
5025 CLOSE
5030 RETURN
5035 REM ****************** DONE ********************************************
5040 SFLG = 1
5045 CHAIN "SMSSMIX"
5050 END
REM
REM ********************************************************************
**
PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS)
FILE NAME: SMSSMIX.BAS DATE: 4 FEB 1986 AM
FUNCTION: PROGRAM TO MIX THE STUDENTS
COMPUTER: ZENITH 120 LANGUAGE: BASIC
AUTHOR: KEN RITCHHART
**

DIM RULE$(22), WING$(5, 6), STAT(18, 5, 12), SL(12), ASL(12)
DIM SEMI(5, 12, 15, 3), ALOT(18, 5, 12)
COMMON MIX, SFLG, WING$( ), STAT( ), SEMI( ), RULE$( ), FLES$, TRONS, MU%, SUB
RULES$(0) = "Z"
CLS: PRINT TAB(25); "SMSS MIXING ROUTINE": PRINT: PRINT
REM ******** PROCESS STUDENTS
GOSUB 5820
GOSUB 6455
GOSUB 6975
GOSUB 7425
GOSUB 7495
RL$=" ARMY": RL = 1: IR = 10: GOSUB 8115 'ASSIGN ARMY
RL$=" COMM SKILL": RL = 1: IR = 1: GOSUB 8115 'ASSIGN CRITICAL SKILL
RL$=" PPBS SKILL": RL = 2: IR = 2: GOSUB 8115 'ASSIGN CRITICAL SKILL
RL$=" ACQ/LOG SKILL": RL = 10: IR = 5: GOSUB 8115 'ASSIGN CRITICAL SKILL
RL$=" TACOPS SKILL": RL = 3: IR = 3: GOSUB 8115 'ASSIGN CRITICAL SKILL
RL$=" STRATOPS SKILL": RL = 3: IR = 4: GOSUB 8115 'ASSIGN CRITICAL SKILL
RL$=" USN,MC,NG,RES": RL = 7: IR = 11: GOSUB 8115 'ASSIGN NAVY, RES
RL$=" MINORITIES ": RL = 16: IR = 12: GOSUB 8115 'ASSIGN MINORITIES
RL$=" FEMALES ": RL = 15: IR = 13: GOSUB 8115 'ASSIGN FEMALES
RL$=" NON LINE": RL = 17: IR = 15: GOSUB 8115 'ASSIGN NON LINE
RL$=" SINGLES ": RL = 22: IR = 8: GOSUB 8115 'ASSIGN SINGLES
RL$=" PILOTS ": RL = 13: IR = 6: GOSUB 8115 'ASSIGN PILOTS
RL$=" NAVIGATORS ": RL = 13: IR = 7: GOSUB 8115 'ASSIGN NAVIGATORS
RL$=" NO MASTERS ": RL = 18: IR = 18: GOSUB 8115 'ASSIGN NO EDUCATION
RL$=" USAFA GRAD ": RL = 19: IR = 9: GOSUB 8115 'ASSIGN USAFA GRADS
RL$=" CAPTAINS ": RL = 18: IR = 14: GOSUB 8115 'ASSIGN CAPTAINS
RL$=" SR ORG EXP": RL = 21: IR = 16: GOSUB 8115 'ASSIGN SR ORG EXP
RL$=" ALL OTHERS": RL = 0: IR = 0: GOSUB 8115 'ASSIGN REMAINING
GOSUB 6350
GOTO 8305
REM SELECT SENIOR STAFF
IF MIX <> 1 GOTO 5810
CC = 0
FOR I = 1 TO 200
GET #4, I
GET #5, I
IF IOS$ = "Y" THEN GOTO 5810
IF NOLINE$ = "Y" THEN GOTO 5810
IF USAF$ <> "Y" GOTO 5810
IF CC > 0 THEN GOTO 5780
LSET CC$ = "Y"
FOR K = 1 TO MU%
IF VAL(WINGS(K,3)) = 1 THEN LSET AWING$ = WINGS(K,1)
NEXT K
GOTO 5795
LSET SRO$ = "Y"
IF CC > MU% THEN GOTO 5815
LSET AWING$ = WINGS(CC,1)
CC = CC + 1
PUT #4,1
PUT #5,1
NEXT I
REM ASSIGN PEOPLE TO WINGS
REM ********** RETRIEVE STAT & RULES
OPEN "O", #2, "F: SMSERROR.DAT"
OPEN "I", #1, "F:XTRIAL1.DAT"
OPEN "R", #4, "F: SMSTDNT1.DAT"
OPEN "R", #5, "F: SMSTDNT2.DAT"
FIELD #4, 27 AS SNAME$, 9 AS SSN$, 6 AS DOR$, 2 AS MIX1$, 2 AS MIX2$, 2 AS MIX3$, 2 AS MIX4$, 1 AS AWING$, 1 AS USAF$, 1 AS MXED$, 1 AS MORG$, 1 AS PILOTS$, 1 AS NAV$, 1 AS SING$, 1 AS AFAS$, 1 AS NAVY$, 1 AS ARMY$, 1 AS RES$, 1 AS RACF$, 1 AS F1$, 1 AS RNK$
FIELD #5, 1 AS NOLINE$, 1 AS SRPME$, 1 AS TOPER$, 1 AS COMM$, 1 AS TOPS$, 1 AS S01$, 1 AS PBB$, 1 AS ACQLOG$, 1 AS SL1$, 1 AS SL2$, 1 AS SL3$, 1 AS ASL$, 1 AS APL2$, 1 AS ASL3$, 1 AS ASLX$, 1 AS SOS$, 1 AS ARIS$, 1 AS SRO$, 1 AS CC$, 1 AS IO$
6 AS PAPSC$
XT = TIME
XT = XT - (INT(XT/100)*100)
FOR I = 1 TO XT: X = RND: NEXT I
X = X * 10000: RANDOMIZE X
REM -----------------------------------------------
PRINT TAB(10); "RETRIEVING STATISTICS & RULES ";
FOR I = 0 TO 18
PRINT ".";
FOR J = 0 TO 5
FOR K = 0 TO 12
INPUT #1, STAT(I,J,K)
NEXT K
NEXT J
NEXT I
FOR I = 0 TO 5: PRINT ":";
FOR J = 0 TO 12
FOR K = 0 TO 15
FOR L = 0 TO 3
INPUT #1, SEMI(I,J,K,L)
NEXT L
NEXT K
NEXT J
NEXT I
CLOSE #1
REM * RETRIEVE SCHOOL DATA **
OPEN "I", #1, "F: SMSCHOOL.DAT"
INPUT #1, SCHOOL$, MUNIT$, SUNIT$, MU%, SU%
INPUT #1, WINGS(I,1),WINGS(I,2),WINGS(I,3),WINGS(I,4),WINGS(I,5),
WINGS(I,6)

PRINT "W";
NEXT I
CLOSE #1
REM ******* RETRIEVE RULES DATA *******
OPEN "I", #1, "F:SMRULE.DAT"
FOR I = 1 TO 22
  INPUT #1, RULE$(I)
NEXT I
PRINT "R";
CLOSE #1
STDNT = STAT(0,0,0)
RETURN
REM ******************** REMOVING STUDENTS FROM SEMINAR ASSIGNMENT ***************
IF JJ = 0 THEN STAT(0,JJ,0)=STAT(0,JJ,0) - 1
FOR MM = SS TO SEMI(JJ,KK,0,0)
  SEMI(JJ,KK,MM,0) = SEMI(JJ,KK,MM+1,0)
NEXT MM
SEMI(JJ,KK,0,0) = SEMI(JJ,KK,0,0) - 1
STAT(0,JJ,KK) = STAT(0,JJ,KK) - 1
IF COMM$ = "Y" THEN STAT(1,JJ,KK) = STAT(1,JJ,KK) - 1
IF PPBS$ = "Y" THEN STAT(2,JJ,KK) = STAT(2,JJ,KK) - 1
IF TOPS$ = "Y" THEN STAT(3,JJ,KK) = STAT(3,JJ,KK) - 1
IF SOPS$ = "Y" THEN STAT(4,JJ,KK) = STAT(4,JJ,KK) - 1
IF ACOLOG$ = "Y" THEN STAT(5,JJ,KK) = STAT(5,JJ,KK) - 1
IF PILOT$ = "Y" THEN STAT(6,JJ,KK) = STAT(6,JJ,KK) - 1
IF NAV$ = "Y" THEN STAT(7,JJ,KK) = STAT(7,JJ,KK) - 1
IF SING$ = "Y" THEN STAT(8,JJ,KK) = STAT(8,JJ,KK) - 1
IF APA$ = "Y" THEN STAT(9,JJ,KK) = STAT(9,JJ,KK) - 1
IF ARMY$ = "Y" THEN STAT(10,JJ,KK) = STAT(10,JJ,KK) - 1
IF RESS$ = "Y" OR NAVY$ = "Y" THEN STAT(11,JJ,KK) = STAT(11,JJ,KK)
IF RACE$ = "Y" THEN STAT(12,JJ,KK) = STAT(12,JJ,KK) - 1
IF FEMS$ = "Y" THEN STAT(13,JJ,KK) = STAT(13,JJ,KK) - 1
IF RNK$ = "Y" THEN STAT(14,JJ,KK) = STAT(14,JJ,KK) - 1
IF NOLINE$ = "Y" THEN STAT(15,JJ,KK) = STAT(15,JJ,KK) - 1
IF HORG$ = "Y" THEN STAT(16,JJ,KK) = STAT(16,JJ,KK) - 1
IF ARI$ = "Y" OR SOS$ = "Y" THEN STAT(17,JJ,KK) = STAT(17,JJ,KK)
IF NOEDS$ = "Y" THEN STAT(18,JJ,KK) = STAT(18,JJ,KK) - 1
RETURN
REM **************************** RECORD STATISTICS *******************************
IF JJ <> 0 THEN STAT(0,JJ,0)=STAT(0,JJ,0) + 1
IF KK = 0 GOTO 6245
FOR MM = 1 TO 15
  IF SEMI(JJ,KK,MM,0) <> 0 THEN GOTO 6240
  SEMI(JJ,KK,MM,0)=SI: SEMI(JJ,KK,0,0)=SEMI(JJ,KK,0,0)+1
  SEMI(JJ,KK,MM,1)=VAL(MIX1$): SEMI(JJ,KK,MM,2)=VAL(MIX2$)
  SEMI(JJ,KK,MM,3)=VAL(MIX3$)
GOTO 6245
NEXT MM
STAT(0,JJ,KK) = STAT(0,JJ,KK) + 1
IF COMM$ = "Y" THEN STAT(1,JJ,KK) = STAT(1,JJ,KK) + 1
IF PPBS$ = "Y" THEN STAT(2,JJ,KK) = STAT(2,JJ,KK) + 1
IF TOPS$ = "Y" THEN STAT(3,JJ,KK) = STAT(3,JJ,KK) + 1
IF SOPS$ = "Y" THEN STAT(4, JJ, KK) = STAT(4, JJ, KK) + 1
IF ACLOG$ = "Y" THEN STAT(5, JJ, KK) = STAT(5, JJ, KK) + 1
IF PILOT$ = "Y" THEN STAT(6, JJ, KK) = STAT(6, JJ, KK) + 1
IF NAV$ = "Y" THEN STAT(7, JJ, KK) = STAT(7, JJ, KK) + 1
IF SING$ = "Y" THEN STAT(8, JJ, KK) = STAT(8, JJ, KK) + 1
IF AFA$ = "Y" THEN STAT(9, JJ, KK) = STAT(9, JJ, KK) + 1
IF ARMY$ = "Y" THEN STAT(10, JJ, KK) = STAT(10, JJ, KK) + 1
IF RES$ = "Y" OR NAVY$ = "Y" THEN STAT(11, JJ, KK) = STAT(11, JJ, KK) + 1
IF RACE$ = "Y" OR NAVY$ = "Y" THEN STAT(11, JJ, KK) = STAT(11, JJ, KK) + 1
IF FEM$ = "Y" THEN STAT(12, JJ, KK) = STAT(12, JJ, KK) + 1
IF RNK$ = "Y" THEN STAT(14, JJ, KK) = STAT(14, JJ, KK) + 1
IF NOLINE$ = "Y" THEN STAT(15, JJ, KK) = STAT(15, JJ, KK) + 1
IF HORG$ = "Y" THEN STAT(16, JJ, KK) = STAT(16, JJ, KK) + 1
IF API$ = "Y" THEN STAT(17, JJ, KK) = STAT(17, JJ, KK) + 1
IF NOED$ = "Y" THEN STAT(18, JJ, KK) = STAT(18, JJ, KK) + 1
RETURN
REM ************************ END STATISTICS *******************************
REM ************************ OUTPUT STATISTICS *******************************
PRINT TAB(10); "OUTPUT STATISTICS & SEMINAR ASSIGNMENTS"
OPEN "O", #1, "F:XTRIAL1.DAT"
FOR I = 0 TO 18
FOR J = 0 TO 5
FOR K = 0 TO 15
PRINT #1, STAT(I, J, K)
NEXT K
NEXT J
NEXT I
CLOSE #1
RETURN
REM ****** RANDOMALLOCATE CHARACTERISTICS TO EACH SEMINAR ******
PRINT TAB(10); "ALLOCATE SLOTS TO SEMINARS"
FOR J = 1 TO MU%
PRINT TAB(10); J;
GOSUB 6645
FOR I = 0 TO 18
PRINT ";";
TA = STAT(I, J, 0) / NS
IF TA < 1 THEN TA = 1
FOR K = 1 TO NS
TX = K
IF K + FS - 1 = MS THEN TX = LS - FS + 1
IF I = 11 AND ALOT(10, J, TX) > 1 AND STAT(I, J, 0) / NS < 1 THEN GOTO 6545
ALOT(I, J, TX) = INT(TA)
NEXT K
IF INT(((TA-INT(TA)) * NS) + .5) = 0 THEN GOTO 6560
6535 FOR L = 1 TO INT(((TA-INT(TA))*NS)+.5) 'GIVE OUT REMAINING
6540 GOSUB 6580 'ASSIGN TSEM
6545 IF ALOT(I,J,PS) > TA THEN GOTO 6540
6550 ALOT(I,J,PS) = ALOT(I,J,PS) + 1
6555 NEXT L
6560 NEXT I
6565 PRINT
6570 NEXT J
6575 RETURN
6580 REM ******* RANDOMLY ASSIGN TEMPORARILY TO SEMINAR TSEM *******
6585 IF TRONS > 1 THEN PRINT " RND A "
6590 RS = RND
6595 FOR LL = 1 TO NS
6600 IF RS > LL/NS GOTO 6625
6605 PS = LL
6610 TSEM = FS + LL - 1
6615 IF TSEM = MS THEN TSEM = LS: PS = NS+1
6620 GOTO 6630
6625 NEXT LL
6630 KK = PS
6635 JJ = J
6640 RETURN
6645 REM ******* GET SEMINAR DATA FOR WING *******
6650 FS = VAL(WING$(J,4)) 'FIRST SEMINAR IN WING
6655 LS = VAL(WING$(J,5)) 'LAST SEMINAR IN WING
6660 MS = VAL(WING$(J,6)) 'MISSING SEMINAR IN WING
6665 NS = VAL(WING$(J,2)) '# SEMINARS IN WING
6670 TS = STAT(0,J,0) 'TOTAL STUDENTS IN WING
6675 RETURN
6680 REM ******* FIND CORRECT WING PERSON IS ALREADY ASSIGNED TO *******
6685 FOR J = 1 TO MU%
6690 IF AWING$ <> WING$(J,1) GOTO 6705
6695 GOSUB 6645
6700 GOTO 6710
6705 NEXT J
6710 JJ = J
6715 RETURN
6720 REM ******* FIND OFFSET FOR ASSIGNED STUDENTS *******
6725 IF MIX = 1 THEN TSEM = VAL(MIX1$)
6730 IF MIX = 2 THEN TSEM = VAL(MIX2$)
6735 IF MIX = 3 THEN TSEM = VAL(MIX3$)
6740 IF MIX >=4 THEN TSEM = VAL(MIXX$)
6745 IF TSEM = 0 THEN GOTO 6760
6750 KK = TSEM - FS + 1
6755 IF TSEM = MS THEN GOSUB 6580
6760 RETURN
6765 REM ******* CHECK ON PREVIOUSLY ASSIGNED STUDENTS *******
6770 PCF = 0 'RULE CHANGE FLAG
6775 PC = 0 'RETRY COUNT
6780 CN = 0 'PREV ASSIGNED WITH
6785 IF RULES$(8) = "D" GOTO 6810

83
6790 IF TSFM = VAL(MIX1$) GOTO 6860
6795 IF TSEM = VAL(MIX2$) GOTO 6860
6800 IF TSEM = VAL(MIX3$) GOTO 6860
6805 IF TSEM = VAL(MIXX$) GOTO 6860
6810 IF SEMI(JJ,KK,0,0) < 1 GOTO 6920 'NO ONE ASSIGNED YET
6815 FOR AB = 1 TO SEMI(JJ,KK,0,0)
6820 IF VAL(MIX1$) = 0 GOTO 6920
6825 IF VAL(MIX1$) = SEMI(JJ,KK,AB,1) THEN CN = CN + 1
6830 IF VAL(MIX2$) = 0 GOTO 6850
6835 IF VAL(MIX2$) = SEMI(JJ,KK,AB,2) THEN CN = CN + 1
6840 IF VAL(MIX3$) = 0 GOTO 6850
6845 IF VAL(MIX3$) = SEMI(JJ,KK,AB,3) THEN CN = CN + 1
6850 NEXT AB
6855 IF CN <= VAL(RULES(9)) GOTO 6920
6860 REM * CAN NOT DO - RETRY NEW SEMINAR *
6865 IF TRONS > 1 THEN PRINT "CHECK SHOWS ";CN:" PREV CLASSMATES";
6870 IF TRONS > 1 THEN PRINT " FOR ";SNAME$;SI:" IN SEMINAR ";TSEM
6875 SASN$ = "N"
6880 IF ARFLAG$ = "N" THEN GOTO 6970
6885 GOSUB 6580
6890 PC = PC + 1
6895 IF PC <= 20 GOTO 6780
6900 PRINT "*** ERROR - UNABLE TO ASSIGN ";SNAME$;" TO SEMINAR WITH 
6905 PRINT "LESS THAN ";RULES(9);" PREVIOUS CLASSMATES"
6910 RULES(9) = STR$(VAL(RULES(9)) + 1): PCF = PCF + 1
6915 GOTO 6775
6920 REM *********** ASSIGN TO SEMINAR ************
6925 IF PCF > 0 THEN PRINT #2,CHR$(34);SNAME$;" ";SI;" ASSIGNED TO SE 
6930 IF PCF > 0 THEN RULES(9) = STR$(VAL(RULES(9)) - PCF): PCF = 0
6935 IF TSEM < 10 THEN Z = 1 ELSE Z = 2
6940 IF MIX = 1 THEN LSET MIX1$ = MIDS(STR$(TSEM),Z,2)
6945 IF MIX = 2 THEN LSET MIX2$ = MIDS(STR$(TSEM),Z,2)
6950 IF MIX = 3 THEN RSET MIX3$ = MIDS(STR$(TSEM),Z,2)
6955 IF MIX >= 4 THEN RSET MIXX$ = MIDS(STR$(TSEM),Z,2)
6960 GOSUB 6195
6965 SASN$ = "Y"
6970 RETURN
6975 REM *********** ASSING SL & ASL TO EACH SEMINAR ***************
6980 PRINT TAB(10);"ASSIGNING SL & ASL"
6985 FOR J = 1 TO MU%
6990 GOSUB 6645
6995 SL(J) = NS
7000 ASL(J) = NS
7005 NEXT J
7010 ARFLAG$ = "N"
7015 FOR SSI = 1 TO STDNT
7020 SI = SSI
7025 GET #4,SI
7030 GET #5,SI
7035 IF IOS$ = "Y" GOTO 7190
7040 IF ARIS$ = "Y" GOTO 7190
7045 IF SOS$ = "Y" AND MIX < 3 THEN GOTO 7190
7050 IF MIX = 1 AND NOLINE$ = "Y" GOTO 7190
7055 PCF = PCF + 1
7060 IF PCF > 0 THEN RULES(9) = STR$(VAL(RULES(9)) - PCF): PCF = 0
7065 IF TSEM < 10 THEN Z = 1 ELSE Z = 2
7070 IF MIX = 1 THEN LSET MIX1$ = MIDS(STR$(TSEM),Z,2)
7075 IF MIX = 2 THEN LSET MIX2$ = MIDS(STR$(TSEM),Z,2)
7080 IF MIX = 3 THEN RSET MIX3$ = MIDS(STR$(TSEM),Z,2)
7085 IF MIX >= 4 THEN RSET MIXX$ = MIDS(STR$(TSEM),Z,2)
7090 GOSUB 6195
7095 SASN$ = "Y"
7097 RETURN
7097 RETURN
7055  IF NOLINES$ = "Y" AND PAFSC$ = "" THEN GOTO 7190
7060  IF CC$ = "Y" OR SRO$ = "Y" THEN GOTO 7190
7065  IF SL1$ = "Y" GOTO 7190
7070  IF SL2$ = "Y" GOTO 7190
7075  IF SL3$ = "Y" GOTO 7190
7080  IF MIX = 1 AND USAF$ <> "Y" GOTO 7115
7085  GOSUB 6680
7090  SSL = 0
7095  IF SL(J) < 1 GOTO 7115
7100  IF SL(J) <= 2 GOTO 7355
7105       X = 0
7110       GOTO 7285
7115  REM ASSIGN ASL'S
7120  IF ASL1$ = "Y" GOTO 7190
7125  IF ASL2$ = "Y" GOTO 7190
7130  IF ASL3$ = "Y" GOTO 7190
7135  GOSUB 6680
7140  SSL = 1
7145  IF ASL(J) >= 1 GOTO 7175
7150       FOR XI = 1 TO MU%
7155       IF ASL(XI) = 0 GOTO 7165
7160       GOTO 7190
7165       NEXT XI
7170       GOTO 7195
7175  IF ASL(J) <= 2 GOTO 7355
7180       X = 1
7185       GOTO 7285
7190  NEXT SSI
7195  REM - ALL SL'S AND ASL'S ASSIGNED
7200  ARFLAG$ = "Y"
7205  RETURN
7210  REM ********** UPDATE STUDENT RECORDS ***********************
7215  IF SL(J) < 1 GOTO 7245
7220  IF MIX = 1 THEN LSET SL1$ = "Y"
7225  IF MIX = 2 THEN LSET SL2$ = "Y"
7230  IF MIX = 3 THEN LSET SL3$ = "Y"
7235  IF MIX >=4 THEN LSET SLX$ = "Y"
7240  GOTO 7270
7245  REM UPDATE ASL
7250  IF MIX = 1 THEN LSET ASL1$ = "Y"
7255  IF MIX = 2 THEN LSET ASL2$ = "Y"
7260  IF MIX = 3 THEN LSET ASL3$ = "Y"
7265  IF MIX >=4 THEN LSET ASLX$ = "Y"
7270  PUT #4,SI
7275  PUT #5,SI
7280  RETURN
7285  REM ********** SL/ASL CHECK FOR SOMEONE ALREADY THERE ***********
7290  GOSUB 6590
7295  IF TRONS > 1 THEN PRINT "J,PS,TSEM ";J,PS,TSEM,SI
7300  IF SEMI(J,PS,0,0) > X GOTO 7290
7305       JJ = J
7310       KK = PS
85
GOSUB 6765 'CHECK ON CONSTRAINTS
IF SASN$ = "N" GOTO 7290 'TRY AGAIN
IF SL(J) > 0 THEN ASG$ = "SL" ELSE ASG$ = "ASL"
GOSUB 7210 'UPDATE RECORDS
IF TRONS > 0 THEN PRINT "ASSIGNING "; SNAME$; "; # = "; SI; "; AS 
IF TRONS > 0 THEN PRINT ASG$; " FOR SEMINAR = "; TSEM
IF SL(J) > 0 THEN SL(J) = SL(J) - 1 ELSE ASL(J) = ASL(J) - 1
GOTO 7190 'NEXT RECORD
FOR LY = 1 TO NS
LX = LY: IF MS = (FS + LX - 1) THEN LX = NS + 1
IF SEMI(J, LX, 0, 0) > X GOTO 7400
TSEM = FS + LX - 1
JJ = J
KK = LX
GOSUB 6765 'CHECK PREV CLASSMATES
IF SASN$ = "N" GOTO 7400
GOTO 7330
NEXT LY
REM - A SLOT IS LEFT BUT INDIVIDUAL CANNOT FILL IT
IF SSL = 0 THEN GOTO 7115 'TRY TO USE AS ASL
GOTO 7190

REM **************************************** END CHECK ****************************************

PRINT TAB(10); "POSTING IO'S"
IF RULE$(6) = "X" GOTO 7490 'NO IO'S THIS MIX
IF RULE$(6) = "D" GOTO 7490 'REASSIGN LIKE OTHERS
FOR SSI = 1 TO STDNT
SI = SSI
GET #4, SI
GET #5, SI
IF IOS < "Y" GOTO 7485 'NOT AN IO
GOSUB 6720 'FIND RIGHT SEMINAR
GOSUB 6195 'UPDATE STATS
IF TRONS > 0 THEN PRINT "ASSIGN IO "; SNAME$; " TO "; TSEM
NEXT SSI
RETURN

SA = 0
PRINT TAB(10); "ASSIGNING ARI/SOS STUDENTS"
IF RULE$(4) = "D" AND RULE$(5) = "D" GOTO 7685 'TREAT LIKE OTHERS
IF RULE$(4) = "D" OR RULE$(5) = "D" THEN SA = 1
ARI = STAT(17, 0, 0)
FOR SSI = 1 TO STDNT
SI = SSI
IF ARI < 1 GOTO 7685 'DONE
GET #4, SI
GET #5, SI
GOSUB 6680
IF ARI$ = "Y" OR SOS$ = "Y" THEN GOTO 7560 ELSE GOTO 7670
GOSUB 6720
IF TSEM > 0 THEN GOTO 7605
IF ARI$ = "Y" AND RULE$(4) = "D" THEN GOTO 7585

86
7575 IF SOS$ = "Y" AND RULE$(5) = "D" THEN GOTO 7585
7580 GOTO 7615
7585 IF SA = 1 GOTO 7670
7590 GOSUB 6580
7595 IF STAT(17, JJ, KK) >= ALOT(17, JJ, KK) THEN GOTO 7590
7600 GOTO 7645
7605 IF TRONS > 0 THEN PRINT SNAME$; " ALREADY ASSIGNED TO "; TSEM
7610 GOTO 7670
7615 IF MIX > 1 THEN TSEM = VAL(MIX1$): GOTO 7640
7620 FOR I = 1 TO NS
7625 IF STAT(17, J, I) = 0 AND STAT(0, J, I) < 4 THEN GOTO 7635
7630 NEXT I
7635 TSEM = FS + I - 1
7640 GOSUB 6750
7645 GOSUB 6935
7650 GOSUB 7270
7655 ARI = ARI - 1
7660 IF TRONS > 0 THEN PRINT "ASSIGN "; SNAME$; " TO "; TSEM
7665 IF ARI < 1 GOTO 7675
7670 NEXT SSI
7675 PRINT 7680 IF SA = 1 THEN SA = 0: GOTO 7525
7685 RETURN
7690 REM "************ SKILLS SELECTION SUBROUTINE ******************"
7695 EX = 0: LC = 0: LCC = 0: NCC = 0
7700 GOSUB 6720
7705 CQ = 0
7710 IF TSEM > 0 GOTO 8105
7715 GOSUB 6680
7720 GOSUB 6580
7725 IF STAT(0, JJ, KK) >= ALOT(0, JJ, KK) GOTO 7720 "ALREADY FILLED"
7730 CQ = CQ + 1
7735 IF CQ > 30 THEN CQ = 0: GOTO 7805
7740 IF STAT(IR, JJ, KK) >= ALOT(IR, JJ, KK) + EX GOTO 7720 "ALREADY FILLED"
7745 GOSUB 6765
7750 IF SASN$ = "Y" GOTO 7775
7755 CP = CP + 1
7760 IF SWC = 1 GOTO 8065
7765 IF CP > 10 GOTO 7805
7770 GOTO 7720
7775 GOSUB 7270
7780 IF TRONS > 0 THEN PRINT SNAME$; SI; " IR="; IR; "IP="; "ASSIGNED TO SEM "; TSEM
7785 IF TRONS > 1 THEN PRINT "J="; JJ; " KK="; KK; " STAT="; STAT(IR, JJ, KK);
7790 IF TRONS > 1 THEN PRINT " ALOT="; ALOT(IR, JJ, KK); " CP="; CP
7795 IF SWC = 1 GOTO 7805
7800 GOTO 8105
7805 REM - TRY EXCHANGING STUDENTS ALREADY ASSIGNED
7810 IF SWC = 1 GOTO 8030
7815 IF SWC > 1 GOTO 7950
7820 FOR III = 1 TO NS
7825 SS = SEMI(JJ, III, 0, 0) - LCC
IF SS <= 2 GOTO 7950  
'ONLY SL/ASL ASSIGNED
SR = GEMI(JJ,III,SS,0)  
'GET LAST ASSIGNED STU
GET #4,SR
GET #5,SR
1: IR = 1 AND COMM$ = "Y" THEN GOTO 7985
IF IR = 2 AND PPBS$ = "Y" THEN GOTO 7985
IF IR = 3 AND TOPS$ = "Y" THEN GOTO 7985
IF IR = 4 AND SOP$ = "Y" THEN GOTO 7985
IF IR = 5 AND ACQLOG$ = "Y" THEN GOTO 7985
IF IR = 6 AND PILOT$ = "Y" THEN GOTO 7985
IF IR = 7 AND NAV$ = "Y" THEN GOTO 7985
IF IR = 8 AND SING$ = "Y" THEN GOTO 7985
IF IR = 9 AND APA$ = "Y" THEN GOTO 7985
IF IR = 10 AND ARMY$ = "Y" THEN GOTO 7985
IF IR = 11 AND RES$ = "Y" THEN GOTO 7985
IF IR = 11 AND NAVY$ = "Y" THEN GOTO 7985
IF IR = 12 AND RACE$ = "Y" THEN GOTO 7985
IF IR = 13 AND FEM$ = "Y" THEN GOTO 7985
IF IR = 14 AND RNK$ = "Y" THEN GOTO 7985
IF IR = 15 AND NOLINE$ = "Y" THEN GOTO 7985
IF IR = 16 AND HORG$ = "Y" THEN GOTO 7985
IF IR = 17 AND AR$ = "Y" THEN GOTO 7985
IF IR = 18 AND NOED$ = "Y" THEN GOTO 7985
IF IR = 0 GOTO 7985
NEXT III

LC = LC + 1: IF LC <= 1 GOTO 7980
IF IR = 0 OR IR = 6 OR IR = 7 THEN LCC = LC + 1 ELSE LC = LC + 1
IF NCC > 1 THEN RULES(9) = STR$(VAL(RULES(9)) + 1): PCF = PCF + 1: LC = 0: NCC = 0
IF LC > 2 THEN EX = EX + 1: LCC = LCC - 2: NCC = NCC + 1
IF LC < 0 THEN LCC = 0
GOTO 7820

REM ------ SWAP SI & RS ------
IF TRNS > 0 THEN PRINT "EXCHANGING ";SI;" FOR ";SR
PS = III: KK = III: JJ = J
TH = SI: SI = SR
GET #4,SI: GET #5,SI
SWC = 1: TSEM = FS + III - 1
GOSUB 6065
SI = TH: GET #4,SI: GET #5,SI
GOTO 7745
REM ------- NOW FIND A PLACE FOR THE GUY YOU REPLACED ------
SWC = 2
SI = SR
CP = 0
GET #4,SI
GET #5,SI
GOTO 7720
REM ------- DID NOT FIT TRY SOMEONE ELSE -------
TH = SI: SI = SR
GET #4,SI: GET #5,SI
KK = III
8085 IF TRONS > 0 THEN PRINT "DIDNT WORK - REINSERT "; SNAME$; SR
8090 GOSUB 6195
8095 SI = TH: GET #4, SI: GET #5, SI
8100 GOTO 7950
8105 SWC = 0
8110 RETURN
8115 REM ******************** ASSIGN REMAINING STUDENTS ********************
8120 IF RULE$(RL) = "D" THEN GOTO 8295
8125 NP = STAT(IR, 0, 0)
8130 IF TRONS > 0 THEN PRINT "ASSIGNING REMAINING "; RL$
8135 ARFLAG$ = "N"
8140 FOR SSI = 1 TO STDNT
8145 SI = SSI
8150 GET #4, SI
8155 GET #5, SI
8160 IF IR = 0 THEN GOTO 8255
8165 IF IR = 1 AND COMM$ <> "Y" THEN GOTO 8290
8170 IF IR = 2 AND PPBS$ <> "Y" THEN GOTO 8290
8175 IF IR = 3 AND TOPS$ <> "Y" THEN GOTO 8290
8180 IF IR = 4 AND SOPS$ <> "Y" THEN GOTO 8290
8185 IF IR = 5 AND ACOLOG$ <> "Y" THEN GOTO 8290
8190 IF IR = 6 AND PILOT$ <> "Y" THEN GOTO 8290
8195 IF IR = 7 AND NAV$ <> "Y" THEN GOTO 8290
8200 IF IR = 8 AND SING$ <> "Y" THEN GOTO 8290
8205 IF IR = 9 AND AFA$ <> "Y" THEN GOTO 8290
8210 IF IR = 10 AND ARMY$ <> "Y" THEN GOTO 8290
8215 IF IR = 11 AND NAVY$ = "Y" THEN GOTO 8255
8220 IF IR = 11 AND RES$ <> "Y" THEN GOTO 8290
8225 IF IR = 12 AND RACE$ <> "Y" THEN GOTO 8290
8230 IF IR = 13 AND FEM$ <> "Y" THEN GOTO 8290
8235 IF IR = 14 AND RNK$ <> "Y" THEN GOTO 8290
8240 IF IR = 15 AND NOED$ <> "Y" THEN GOTO 8290
8245 IF IR = 16 AND HORG$ <> "Y" THEN GOTO 8290
8250 IF IR = 18 AND NOED$ <> "Y" THEN GOTO 8290
8255 GOSUB 6680 'FIND WING DATA
8260 GOSUB 6720
8265 IF TSEM > 0 THEN PRINT SNAME$; "ALREADY ASSIGNED"; TSEM: GOTO 828
8270 IF RULE$(RL) = "A" THEN EX = 0 ELSE EX = 1
8275 IF RULE$(RL) = "Z" THEN EX = 3
8280 GOSUB 7700 'ASSIGN SKILLS ROUTINE
8285 NP = NP - 1: IF NP < 1 GOTO 8295
8290 NEXT SSI
8295 RETURN
8300 ARFLAG$ = "Y"
8305 REM ******************** FINISHED ********************
8310 CLOSE #2
8315 OPEN "I", #2, "F:SMSERROR.DAT"
8320 FOR I=1 TO 200
8325 IF EOF(2) GOTO 8345
8330 INPUT #2, TMP$
8335 PRINT TMP$
8340 NEXT I
8345 SFLG = 1
8350 CLOSE
8355 CHAIN "SMSSMIXO"
8360 END
REM *********************************************************************
DIM RULE$(22), WING$(5,6), STAT(18,5,12), SL(12), ASL(12)
DIM SEMI(5,12,15,3), ALOT(18,5,12)
COMMON MIX, SFLG, WING$(, STAT(), SEMI(), RULE$(, FLS$, TRONS, MU%, SU%
CLS: PRINT TAB(25);"SMSS OUTPUTING MIX TO CONDOR": PRINT: PRINT
REM ******** PROCESS STUDENTS **********
STDNT = STAT(0,0,0)
OPEN "0", #1, "F:ASICOT"
OPEN "R", #4, "F:SMSTDNT1.DAT"
OPEN "R", #5, "F:SMSTDNT2.DAT"
FIELD #4, 27 AS SNAME$,9 AS SSN$,4 AS STN$,6 AS DOR$,2 AS MIX1$,2 AS MIX2$,2 AS MIX3$,2 AS MIXX$,1 AS AWING$,1 AS USAF$,1 AS NOEDS$,1 AS HORG$,1 AS PILOT$,AS NAV$,1 AS SING$,1 AS AFA$,1 AS NAVY$,1 AS ARMY$,1 AS RES$,1 AS RACE$,1 AS FMS,1 AS RNK$
FIELD #5,1 AS NOLINE$,1 AS SRPME$,1 AS TOPER$,1 AS COMM$,1 AS TOPS$,1 AS SPS$,1 AS PPBS$,1 AS ACQLOG$,1 AS SL1$,1 AS SL2$,1 AS SL3$,1 AS SLX$,1 AS ASL1$,AS ASL2$,1 AS ASL3$,1 AS ASLX$,1 AS SOS$,1 AS ARI$,1 AS SRO$,1 AS CC$,1 AS IO$6 AS PAFCSC$
KO = 1: PRINT " OUTPUTING ";
FOR SSI = 1 TO STDNT
       IF SSI/10 >= KO THEN PRINT ";"; KO = KO + 1
SI = SSI
GET #4,SI: GET #5,SI
OTTREC$ = CHR$(34)+SNAME$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+SSN$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+STN$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+DOR$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+MIX1$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+MIX2$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+MIX3$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+MIXX$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+AWING$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+USA$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+NOEDS$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+HORG$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+PILOT$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+NAV$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+AFA$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+NAVY$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+ARMY$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+RES$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+RACE$+CHR$(34)+","
OTTREC$ = OTTREC$+CHR$(34)+FEM$+CHR$(34)+"","
8735  OTREC$ = OTREC$+CHR$(34)+RNKS+CHR$(34)+""," 
8740  OTREC$ = OTREC$+CHR$(34)+NOLINES+CHR$(34)+""," 
8745  OTREC$ = OTREC$+CHR$(34)+SRPME$+CHR$(34)+""," 
8750  OTREC$ = OTREC$+CHR$(34)+TOPPERS+CHR$(34)+""," 
8755  OTREC$ = OTREC$+CHR$(34)+COMM$+CHR$(34)+""," 
8760  OTREC$ = OTREC$+CHR$(34)+SOPPS+CHR$(34)+""," 
8765  OTPECS = OTREC$+CHR$(34)+TOPPS+CHR$(34)+""," 
8770  OTREC$ = OTREC$+CHR$(34)+PPB$+CHR$(34)+""," 
8775  OTREC$ = OTREC$+CHR$(34)+ACQLOGS+CHR$(34)+""," 
8780  OTREC$ = OTREC$+CHR$(34)+SL1$+CHR$(34)+""," 
8785  OTREC$ = OTREC$+CHR$(34)+SL2$+CHR$(34)+""," 
8790  OTREC$ = OTREC$+CHR$(34)+SL3$+CHR$(34)+""," 
8795  OTREC$ = OTREC$+CHR$(34)+SLX$+CHR$(34)+""," 
8800  OTREC$ = OTREC$+CHR$(34)+ASL1$+CHR$(34)+""," 
8805  OTREC$ = OTREC$+CHR$(34)+ASL2$+CHR$(34)+""," 
8810  OTREC$ = OTREC$+CHR$(34)+ASL3$+CHR$(34)+""," 
8815  OTREC$ = OTREC$+CHR$(34)+ASLX$+CHR$(34)+""," 
8820  OTREC$ = OTREC$+CHR$(34)+SOS$+CHR$(34)+""," 
8825  OTREC$ = OTREC$+CHR$(34)+ARI$+CHR$(34)+""," 
8830  OTREC$ = OTREC$+CHR$(34)+SOS$+CHR$(34)+""," 
8835  OTREC$ = OTREC$+CHR$(34)+CSS$+CHR$(34)+""," 
8840  OTREC$ = OTREC$+CHR$(34)+IOS$+CHR$(34)+""," 
8845  OTREC$ = OTREC$+CHR$(34)+PAFSC$+CHR$(34) 
8850  PRINT #1,OTREC$ 
8855  NEXT SSI 
8860  PRINT #1,CHR$(26) 
8865  CLOSE 
8870  CHAIN "SMSS" 
8875  END
REM
8905 REM *******************************************************
8910 REM **
8915 REM ** PROGRAM NAME: STUDENT MIX SOFTWARE SYSTEM (SMSS)
8920 REM ** FILE NAME: SMSSUPDT.RAS  DATE:  18 FEB 1986 AM
8925 REM ** FUNCTION: UPDATE THE MANUAL CHANGES TO STAT & SEMI ARRAYS
8930 REM ** COMPUTER: ZENITH 120  LANGUAGE: BASIC
8935 REM ** AUTHOR: KEN RITCHHART
8940 REM **
8945 REM *******************************************************
8950 DIM TEMPINS(44), ALOT(18,5,12)
8955 DIM WING$(5,6), STAT(18,5,12), SEMI(5,12,15,3), RULE$(22)
8960 COMMON MIX, SFLAG, WING$(,), STAT(,), SEMI(,), RULE(,), FLES, TRONS, SU, MU
8965 CLS: PRINT "  REGISTERING MANUAL CHANGES TO SEMINARS": PRIN
8970 GOSUB 9430
8975 REM **************************** PROCESS INPUT DATA ****************************
8980 OPEN "I", #3, "F:\BASICIF."
8985 OPEN "R", #4, "F:\SMSTDNT1.DAT"
8990 OPEN "R", #5, "F:\SMSTDNT2.DAT"
8995 FIELD #4, 27 AS SNAME$,9 AS SSN$,4 AS STN$,6 AS DOR$,2 AS MIX1$,2 AS MI
9000 FIELD #5,1 AS NOLINE$,1 AS SRPME$,1 AS TOPER$,1 AS COMM$,1 AS TOPS$,1 AS P
9005 AS NAV$,1 AS SING$,1 AS AFA$,1 AS NAVYS,1 AS ARMY$,1 AS RES$,1 AS RES$,1 AS R
9010 AS RNK$1 AS PAFSC?
9015 KO = 1: PRINT: PRINT "READING ";
9020 FOR KS = 1 TO 600
9025 STDNT = SI
9030 FOR J = 1 TO 44
9035 IF EOF (3) THEN GOTO 9390
9040 INPUT #3, TEMPINS(J)
9045 NEXT J
9050 LSET SNAME$ = TEMPINS(1)
9055 LSET SSN$ = TEMPINS(2)
9060 LSET STN$ = TEMPINS(3)
9065 LSET DOR$ = TEMPINS(4)
9070 LSET MIX1$ = TEMPINS(5)
9075 LSET MIX2$ = TEMPINS(6)
9080 LSET MIX3$ = TEMPINS(7)
9085 LSET MIXX$ = TEMPINS(8)
9090 LSET AWING$ = TEMPINS(9)
9095 LSET USAF$ = TEMPINS(10)
9100 LSET NOED$ = TEMPINS(11)
9105 LSET HORG$ = TEMPINS(12)
9110 LSET PILOTS$ = TEMPINS(13)
9115 LSET NAVS = TEMPINS(14)
9120 LSET SING$ = TEMPINS(15)
9125 LSET AFA$ = TEMPINS(16)
9130 LSET NAVYS = TEMPINS(17)
9135 LSET ARMY$ = TEMPINS(18)
9135 LSET RES$ = TEMPIN$(19)
9140 LSET RACE$ = TEMPIN$(20)
9145 LSET FEM$ = TEMPIN$(21)
9150 LSET RNK$ = TEMPIN$(22)
9155 LSET NOLINE$ = TEMPIN$(23)
9160 LSET SKPMES$ = TEMPIN$(24)
9165 LSET TOPPER$ = TEMPIN$(25)
9170 LSET COMM$ = TEMPIN$(26)
9175 LSET SOPSS$ = TEMPIN$(27)
9180 LSET TOPSS$ = TEMPIN$(28)
9185 LSET PPBS$ = TEMPIN$(29)
9190 LSET ACOLOG$ = TEMPIN$(30)
9195 LSET SL1$ = TEMPIN$(31)
9200 LSET SL2$ = TEMPIN$(32)
9205 LSET SL3$ = TEMPIN$(33)
9210 LSET SLX$ = TEMPIN$(34)
9215 LSET ASL1$ = TEMPIN$(35)
9220 LSET ASL2$ = TEMPIN$(36)
9225 LSET ASL3$ = TEMPIN$(37)
9230 LSET ASLX$ = TEMPIN$(38)
9235 LSET SOS$ = TEMPIN$(39)
9240 LSET ARI$ = TEMPIN$(40)
9245 LSET SRO$ = TEMPIN$(41)
9250 LSET CC$ = TEMPIN$(42)
9255 LSET IOS$ = TEMPIN$(43)
9260 LSET PAFSC$ = TEMPIN$(44)
9265 IF IOS$ = "Y" AND RULES(6) = "X" GOTO 9385
9270 REM ***** RECORD SCHOOL OVERALL STATISTICS ***************
9275 JJ = 0
9280 KK = 0
9285 GOSUB 9595
9290 IF AWING$ = " " GOTO 9335
9295 REM *** POST WING STATISTICS *****
9300 FOR J = 1 TO MUI
9305 IF AWING$ <> WING$(J,1) GOTO 9330
9310 JJ = J
9315 KK = 0
9320 GOSUB 9595
9325 GOTO 9335
9330 NEXT J
9335 SI = SI + 1
9340 IF KS > 1 GOTO 9365
9345 IF VAL(MIX1$) > 0 THEN MIX = 1
9350 IF VAL(MIX2$) > 0 THEN MIX = 2
9355 IF VAL(MIX3$) > 0 THEN MIX = 3
9360 IF VAL(MIXX$) > 0 THEN MIX = 4
9365 GOSUB 9885
9370 GOSUB 9595
9375 PUT #4, SI
9380 PUT #5, SI
9385 NEXT KS
9390 STDNT = SI
9395 PRINT: PRINT "PROCESSED IN ";KS-1;" STUDENTS INTO SMSS"
9400 PRINT STDNT; " STUDENT RECORDS & STATISTICS UPDATED FROM MANUAL CHANGES"
9405 STAT(0,0,0) = STDNT
9410 REM END OF DATA INPUT
9415 GOSUB 9750
9420 GOTO 9865
9425 PRINT: PRINT
9430 REM ******* INITIALIZE STAT & GET RULES
9435 PRINT "INITIALIZING ";
9440 FOR I = 0 TO 18
9445 PRINT "."
9450 FOR J = 0 TO 5
9455 FOR K = 0 TO 12
9460 STAT(I,J,K) = 0
9465 NEXT K
9470 NEXT J
9475 NEXT I
9480 FOR I = 0 TO 12
9485 FOR J = 0 TO 15
9490 FOR K = 0 TO 3
9495 SEMI(I,J,K,L) = 0
9500 NEXT L
9505 NEXT K
9510 NEXT J
9515 NEXT I
9520 NEXT I
9525 REM * RETRIEVE SCHOOL DATA **
9530 OPEN "I", #1, "F:SMSCHOOL.DAT"
9535 INPUT #1, SCHOOL$, MUNIT$, SUNIT$, MU%, SU%
9540 FOR I = 1 TO MUI
9545 INPUT #1, WING$(I,1),WING$(I,2),WING$(I,3),WING$(I,4),WING$(I,5),
9550 WING$(I,6)
9555 NEXT I
9559 CLOSE #1
9560 REM  **** RETRIEVE RULES DATA *****
9565 OPEN "I", #1, "F:SMRULE.DAT"
9570 FOR I = 1 TO 22
9575 INPUT #1, RULE$(I)
9580 NEXT I
9584 CLOSE #1
9590 RETURN
9595 REM ************** RECORD STATISTICS *********************
9600 IF JJ <> 0 THEN STAT(0,JJ,0)=STAT(0,JJ,0) + 1
9605 IF KK = 0 GOTO 9650
9610 FOR MM = 1 TO 15
9615 IF SEMI(JJ,KK,MM,0) <> 0 THEN GOTO 9645
9620 SEMI(JJ,KK,MM,0)=SI: SEMI(JJ,KK,0,0)=SEMI(JJ,KK,0,0)+1
9625 SEMI(JJ,KK,MM,1)=VAL(MIX1$): SEMI(JJ,KK,MM,2)=VAL(MIX2$)
9630 SEMI(JJ,KK,MM,3)=VAL(MIX3$)
9635 GOTO 9650
9640 NEXT MM
9645 STAT(0,JJ,KK) = STAT(0,JJ,KK) + 1
IF COMM$ = "Y" THEN STAT(1, JJ, KK) = STAT(1, JJ, KK) + 1
IF PPBS$ = "Y" THEN STAT(2, JJ, KK) = STAT(2, JJ, KK) + 1
IF TOPS$ = "Y" THEN STAT(3, JJ, KK) = STAT(3, JJ, KK) + 1
IF SOPS$ = "Y" THEN STAT(4, JJ, KK) = STAT(4, JJ, KK) + 1
IF ACOLOG$ = "Y" THEN STAT(5, JJ, KK) = STAT(5, JJ, KK) + 1
IF PILOT$ = "Y" THEN STAT(6, JJ, KK) = STAT(6, JJ, KK) + 1
IF NAV$ = "Y" THEN STAT(7, JJ, KK) = STAT(7, JJ, KK) + 1
IF SING$ = "Y" THEN STAT(8, JJ, KK) = STAT(8, JJ, KK) + 1
IF AFA$ = "Y" THEN STAT(9, JJ, KK) = STAT(9, JJ, KK) + 1
IF ARMY$ = "Y" THEN STAT(10, JJ, KK) = STAT(10, JJ, KK) + 1
IF RES$ = "Y" THEN STAT(11, JJ, KK) = STAT(11, JJ, KK) + 1
IF RACE$ = "Y" THEN STAT(12, JJ, KK) = STAT(12, JJ, KK) + 1
IF FEM$ = "Y" THEN STAT(13, JJ, KK) = STAT(13, JJ, KK) + 1
IF RNK$ = "Y" THEN STAT(14, JJ, KK) = STAT(14, JJ, KK) + 1
IF NOLINE$ = "Y" THEN STAT(15, JJ, KK) = STAT(15, JJ, KK) + 1
IF HORG$ = "Y" THEN STAT(16, JJ, KK) = STAT(16, JJ, KK) + 1
IF ARI$ = "Y" OR SOS$ = "Y" THEN STAT(17, JJ, KK) = STAT(17, JJ, KK) + 1
IF NOED$ = "Y" THEN STAT(18, JJ, KK) = STAT(18, JJ, KK) + 1

RETURN

OPEN "0", #1, "F:XTRIAL1.DAT"
FOR I = 0 TO 18
    PRINT ".";
    FOR J = 0 TO 5
        FOR K = 0 TO 12
            PRINT #1, STAT(I, J, K)
        NEXT K
    NEXT J
NEXT I

FOR I = 0 TO 5
    PRINT ";";
    FOR J = 0 TO 12
        FOR K = 0 TO 15
            FOR L = 0 TO 3
                PRINT #1, SEMI(I, J, K, L)
            NEXT L
        NEXT K
    NEXT J
NEXT I
CLOSE

RETURN

REM *************** DONE ***************

IF MIX = 1 THEN TSEM = VAL(MIX1$)
IF MIX = 2 THEN TSEM = VAL(MIX2$)
IF MIX = 3 THEN TSEM = VAL(MIX3$)
9905      IF MIX >=4 THEN TSEM = VAL(MIXX$)
9910      IF TSEM = 0 THEN GOTO 9970
9915      FOR J = 1 TO MU8
9920          IF AWING$ <> WING$(J,1) GOTO 9955
9925              FS = VAL(WING$(J,4))
9930              LS = VAL(WING$(J,5))
9935              MS = VAL(WING$(J,6))
9940              NS = VAL(WING$(J,2))
9945              TS = STAT(0,J,0)
9950          GOTO 9960
9955      NEXT J
9960      JJ = J
9965      KK = TSEM - FS + 1
9970      RETURN
9975      END

'FIRST SEMINAR IN WING'
'LAST SEMINAR IN WING'
'MISSING SEMINAR IN WING'
'# SEMINARS IN WING'
'TOTAL STUDENTS IN WING'