Craig Bina

Bina, C. R., A multi-user computer assisted instructional management system, *Library of Congress, TXU 52-217*, 129 pp., 1979.

A Multi-User Computer Assisted Instructional Management System:

Software Development and Implementation

Craig Binn

SUMMARY

The objective of this project was the development of a computer assisted instructional management system and its implementation on a Hewlett Packard 2000/Access Timeshared BASIC Level F computer system. This instructional package is potentially usable by twenty-six different schools of twenty-six teachers each; at present it is heavily used by six of the eight schools in high school district 214.

This project's major achievements lie in the area of software development. To implement this package, it was necessary to devise software to perform functions which HP2000/BASIC was neither equipped nor intended to perform. Prime examples of such software are the routines for expanding and contracting multiple-write-access "rubber-band" files, along with the priority I and II "locks" routines which permit the five-level inter-program communication necessary to the manipulation of these files. This project is of a dynamic nature; for the package still continues to grow in both complexity and in versatility of application.

Craig Bina

TABLE OF CONTENTS

INTRODUCTION

The purpose of this project was to create a computer assisted instructional management system which would, in a given subject area, generate unique sets of practice problems for individual students, store the correct answers to each practice sheet, permit the student to return after having worked the problems in order to check his answers, store the scores in a permanent record, and permit a teacher to access his/her permanent record of student scores. The original goal was to be able to generate unique sets of practice problems in each of twenty-four chemistry skills (each skill is referred to as a "fundamental" and is assigned a "fundamental number") for one teacher , who could potentially have nine classes, of which each class could potentially contain ninety-nine students. The management system, however, has proven to be quite dynamic. Throughout three years of development, including school days, weekends, and summers, it has grown to the point where it can now encompass twenty-six different schools, each one of which may possess twenty-six teachers with classes as outlined above. In addition, the management system now contains eleven "instructors' options" which facilitate its use by faculty members, and although the package presently contains thirty-three fundamentals with which to operate (twenty-five for chemistry, eight for general math), its potential applications are unlimited.

The continual expansion of this package has been possible only because of the cyclical nature of the management system: demonstrated by the fact that all users are continually routed through the programs MAIN (where it is confirmed that the inter-program common variables have been assigned) and CHMSTR (where the user is identified and then sent to the program which performs the function which he/she desires). The importance of the fact that all programs and program sequences return to MAIN upon the completion of their respective tasks cannot be overemphasized. If, at any time, a program is unable to return a user to MAIN, that user is promptly logged off of (disconnected from) the computer; this preserves package security. There are four distinct levels of package usage: that of student, instructor, administrator, and programmer. Only the programmer and, under certain conditions, an administrator are ever permitted to exit the MAIN-CHMSTR cycle in order to gain direct access to package software. This security is made complete with the inclusion of the program ERROR. This program monitors all other package programs and is immediately activated upon the occurrence of an error anywhere within the cycle of programs; it then analyzes and records the error and returns the user to MAIN.

In order to implement this system, it was necessary to develop software to perform functions which HP2000/BASIC (in which this instructional package is written) was not intended nor equipped to perform. The true achievements of this package are those functions which the user never sees. It is the development of such software as that for expanding and shrinking "rubber-band" files, along with the priority I and II "locks" routines which permit the five-level inter-program communication necessary to manipulating these files, which has made this package the success it has proven to be.

MATERIALS

The Instructional Package Management System was written and implemented on a Hewlett Packard 2000/Access Level F Timeshared BASIC system with a 48KB Input/Output Processor, 64KB System Processor, both 15MB and 50MB Disc Drive Systems, two 16-Port Multiplexers, and a 600 lpm line printer. Remote stations at the various schools in the district consist of Teletype Model 33 and 43 terminals and Teleray CRT terminals. Communication with the central system was accomplished through standard voice-communication telephone lines, thirty-two 110/300 baud Acoustic Modems, and thirty-two Data Access Arrangements.

NOTE ON MANNER OF REFERENCE

When referring to the Appendices in the sections which follow, the following format will be used: (A.BB), (A.BB:#) or (A.BB:#-#), where "A" is the number of the Appendix, "BB" is a particular section of an Appendix, "#" is a specific line number, and "#-#" is a given spread of line numbers. Thus (2.02) would refer the reader to Appendix 2, section 2, while (3.05:120) would refer to Appendix 3, section 5, line 120, and (3.02: 200-230) would refer to Appendix 3, section 2, lines 200 through 230.

METHODS OF IMPLEMENTATION

The purpose of this section is to give the reader a working knowledge of how the management system functions. It is intended to present an overview of the flow of programs which comprise the package and therefore makes extensive reference to Appendix 2: A Sample Execution. Details of the functioning of individual programs and routines are presented in the section entitled "Methods of Software Development."

When a user first logs onto (connects with) account W140 on the timeshare system (2.01), he/she is routed through the system library's HELLO program (2.02) into the program MAIN (2.03). This program, recognizing that the user has just logged on, prints the package banner (announcements), if any, and then routes the new user into the beginning of CHMSTR (2.04) which prints a brief welcome message. The program flow has now reached the point in CHMSTR known as the "return point" (2.05) at which the user is asked to enter some identifying information. At this point the user has several options. He/she may enter any of the following: ***, CENTRAL, !, STOP, SAME, or a teacher's last initial. "***" is used only by the programmer when he/she wishes to exit the management system in order to gain access to the package software (2.39). After entering "***" the user is asked for a security code (2.40). If the correct code is entered (note that the code does not print, this is because the program has temporarily inhibited the I/O Processor's echo function, thus putting the I/O Port in half-duplex mode), the user is granted access to the system (2.41); if not, the user is logged off.

"CENTRAL" is entered only by an administrator who wishes to generate the district requests which have been stored during the day. The user is asked for an administrative code; if it is incorrect, the user is logged off. If a correct code is entered, the program checks to see whether or not any requests are on file; if no requests are on file, the user is so informed and logged off. If requests are on file, the administrator is granted access to the package and is given instructions on how to generate the requests on the remote lineprinter using the program REMOTE.

"!" is entered by an instructor who wishes to designate him/herself a regular package user. After entering a usage code (if an incorrect code is entered, he/she is logged off), the instructor is asked for his/her last initial and the initial of his/her school. This initial pair ("BT" for example) is known as the "teacher/school." The instructor is then asked for his/her full name and a sequence of letters which is to become his/her instructors' options access code. After this information has been entered, the initialization program, SETUP, creates and formats all files necessary for the functioning of the teacher/school and then returns the user to MAIN. MAIN henceforth routes the user back to the return point in CHMSTR.

Entering "STOP" immediately logs the user off.

If a user enters a teacher's last initial, he/she is then asked for his/her school's last initial (2.06) and his/her own three initials (2.07).

If a user enters "SAME" (2.13), he/she is assigned the same teacher/school and three initials as the previous user; if there was no previous user, the current user is so informed and the query is repeated.

After the teacher/school and three initials of the user have been entered, CHMSTR searches through the appropriate file to establish the user's identity, records another usage in the package usage file, and prints the user's name (2.08). The program then checks the "cheatbox" file to see if the user has cheated. If the user is on file as having cheated, he/she is so informed and is forbidden further usage of the package; if the user is not on file, he/she is informed that everything is "all right" (2.09).

The user is now asked for a fundamental number (2.10). If the user is a student, he/she may enter zero or an integer from one to ninety-nine. If zero is entered (2.10), the user is routed into CATLOG (2.11), a catalog of fundamentals is printed, and the user is returned to MAIN (2.12).

If an integer from one to ninety-nine is entered (2.14), the user is asked to enter P for a practice, C for a check, or R for a request (2.15). If a P is entered (2.15), the user is routed into the appropriate fundamental program (in this case, FUND6), and a unique practice sheet is generated (2.16). The correct answers are assigned a sheet number (printed at the bottom of the sheet) and are stored in a file. The user is then routed into NOTICE (2.17) which prints the user's period number, a brief message, and sends the user back to MAIN (2.18).

If a C is entered (2.19), the user is routed into CHECK., This program asks the user to enter his/her integer sheet number (2.20).

After locating the correct answers, CHECK asks the student to enter his/her answers (2.21). The user is then given a score (2.22) and is routed into UPDATE where the score is recorded (2.24). Then the user is returned to MAIN (2.25).

If an R is entered (2.26), a request that a practice sheet be generated for the student by the instructor at a later time is stored in a file (2.27) and the user is returned to MAIN (2.28).

If the user is a teacher (2.29), he/she may perform two other functions, in addition to those usable by students. First, when requesting a catalog (2.35), the teacher may generate an entire list of all fundamentals, as may the student, but with a listing of the instructors' options (fundamentals usable only by teachers) appended to the end, or the teacher may request a listing of only the instructors' options (2.36). After CATLOG has finished the listing, the user is returned to MAIN (2.37).

Second, a teacher may request the use of one of the instructors' options. This is done merely by enter its fundamental number (2.30). If a student enters the number of an instructors' option (any number greater than ninety-nine), he/she is forbidden access to them. If an instructor request the use of one, he/she is asked to enter his/her access code (2.31). If the code is incorrect, access is not granted; if it is correct, the user is routed into the appropriate program (2.32). Upon completion of the instructors' option, the user is returned to MAIN (2.37). The functions of the various instructors' options are detailed in Appendix 1: The Instructors' Options.

METHODS OF SOFTWARE DEVELOPMENT

Part I: The I.D. Number

Every student and every teacher who use the management system are represented by an I.D. number. Each I.D. number contains sufficient information to preserve the uniqueness of each user. An I.D. number contains: period number, line in gradebook on which student appears, and the user's teacher/school. Since HP2000/BASIC allows for only six significant figures, it is necessary to encode this information. All I.D. numbers conform to the following format: A.BBCC x 10^{DD} where A, B, C, and D are decimal digits. A=period number (1-9, 0 for teachers).
BB=line on page for that period in teacher's gradebook on which student's name appears (01-99, 00 for teachers).
CC=ASCII numeric equivalent of school's initial (65-90).
DD=ASCII numeric equivalent of teacher's last initial minus sixty-four (01-26). It is necessary to subtract sixty-four from the exponent because the largest number which HP2000/BASIC can accommodate is 1.70141E+38. In order to decode an I.D. number it is necessary to perform a series of arithmetic functions upon it; these functions are exhibited in (3.12:1920-1980). Upon completion of this routine, L2=1 if the I.D. represents an instructor, L2=0 if the I.D. represents a student, T$=teacher's last initial, and T1$=school's initial. The purpose of (3.12:1940) is to eliminate errors in binary conversion through conversion to a character string. It should also be noted that, in HP2000/BASIC, LOG(X) is the natural logarithm function, not the common logarithm,

and that it is thus necessary to divide LOG(X) by 2.30259 (LOG(10)) in order to arrive at the common logarithm of X.

Part II: The Usage Routine

Each time a user is recognized by the package, whether it be through entering a teacher/school and three initials, through entering "SAME" or through remote generation of practice sheets, the "initial acceptance counter" for that day is incremented. This is accomplished by the routine that appears in (3.02:880-1180). This routine operates on the file USAGE; this file consists of two partitions of three records each. Each of the two partitions contains as its first item a number representing a year; the following 366 items are initial acceptance counters for each day in that year. In this way it is possible to keep track of total package usage over a two year period (this is necessary because the school year extends over parts of two calendar years). Partition one (p1) contains the previous year, partition two (p2) contains the current year. At the end of each year p1 is destroyed, p2 is moved into p1, and p2 is destroyed. An operative diagram follows.

Let ly=last year, let lly=two years ago, let oy=any other year.

```
What is in p1?
    p1=ly
        increment today in p2
    p1=oy
        clear p1
        clear p2
        increment today in p2
    p1=1ly
        what is in p2?
            p2=oy
                clear p1
                clear p2
                increment today in p2
            p2=1y
                move p2 into p1
                clear p2
                increment today in p2
        end
End
```

Part III: SOINIT: Inter-Program Common Variables

In transferring from one program to another throughout the package, it is necessary to communicate certain values to the new program. These values are presented in the COM statement in (3.06:10). I6=I.D. number of current user; N0$(80)=an eighty-character string which contains the name of the current user; F9=number of the fundamental currently being accessed; X6=number of current series being followed: 0=PSEUDO, 100=Normal, 200=MASLPR, 300=MASFND, 400=RNDFND; Y6=Used only by CHECK and UPDATE, it communicates the user's score, in other programs it is merely a placeholder; O0$(11)=an eleven character string which contains the name of the program currently running; E(3)=a three-item subscripted variable containing pertinent error data: E(1)=type of last error, E(2)=line of last error, E(3)=last file to have been accessed at time of last error.

This COM statement appears at the beginning of all package programs. The entire routine (3.06) appears in some form at the beginning of all fundamental programs. To build a fundamental it is merely necessary to fill in the REM statements, append the program body which contains the instructions for generating the questions, and then append SOTERM (3.07).

Part IV: SOTERM: Rubber-Band Files

Due to the fact that all HP2000/BASIC files are of a fixed length, it was necessary to create routines to simulate variable-length or "rubber-band" files. These files periodically destroy and re-create themselves at different lengths as they fill up or are emptied out. Another problem arises, however, in that a program cannot maintain control over a file which does not exist; thus it was necessary to create routines to accomplish that which HP2000/BASIC was not intended to do. Programs in the management system are in constant communication with each other about the states of their respective files. These programs communicate on five different levels: through COM variables, through the data contained in files, through determining whether or not a file exists, through determining whether or not a file is assigned with a write restriction (WR), and through determining whether or not a file is locked. Programs are allowed to respond to the actions of other programs through the use of the SYSTEM R,"PAU-#" statement. This statement produces a pause of # seconds in the System Processor; it does not, however, cause a pause in the I/O Processor, thus preventing a noticeable delay. This programmatic intercommunication takes place through what are known as locks routines. There are two priorities of locks routines, priority I having the ability to take files away from priority II. A sample priority II locks routine is (3.07:8000-8150). An operative diagram of this routine follows.

```
Let Attempt=1
01    Can file be assigned with WR?
          file in use by another priority II locks routine
02              close file
03              pause for 6 seconds
                jump to 01
          file does not currently exist
04              pause for 2 seconds
                increment Attempt
                     Is Attempt greater than 10?
                          yes
                               inform user that file is missing
                               stop
                          no
                               jump to 01
                     end
          file successfully assigned with WR
               attempt to lock file until successful
```

```
05          pause for 2 seconds
                 Can file be locked again?
06                    yes
                           jump to 02
07                    no
                           proceed to access file
                 end
     End
```

The pause in 03 allows the other priority II locks routine which is accessing the file to complete its task. The pause in 04 provides time for a priority I locks routine, if one happens to be accessing the file, to re-create the file. The pause in 05 allows time for a priority I locks routine to take the file away from the priority II locks routine. It does this by unlocking the file which the priority II locks routine had previously locked. If, in 06, the priority II locks routine is able to lock the file again, it realizes that a priority I locks routine has just told it to release its file. If, in 07, it is unable to lock the file again, it acknowledges that no priority I locks routine wishes to use the file and proceeds with its own accessing.

Now that SOTERM has "permission" to use the file, it proceeds to dump its data into the file. If it all fits, the user is returned to MAIN. If, however, the file becomes full, SOTERM must expand it. This entails a priority I locks routine as shown in (3.07:8480-9060). Here is an operative diagram.

Let Main be the data storage file, let L=length of Main

```
clear incompletely stored data from Main
01  Can Dummy be created at length=L+1?
        Dummy already exists
            purge Dummy
            jump to 01
        there is no space to create Dummy
            inform user that there is insufficient space
            stop
        Dummy successfully created
02          Can Buffer be created at length=L?
                Buffer already exists
                    purge Buffer
                    jump to 02
                there is no space to create Buffer
                    purge Dummy
                    inform user that there is insufficient space
                    stop
                Buffer successfully created
                    assign Buffer
                    copy Main into Buffer
03                  close Main
                        Can Main be purged?
                            no, file in use by another
                                assign Main
                                unlock until successful
04                              pause for 1 second
                                jump to 03
```

```
                                    yes, Main successfully purged
                                         purge Dummy
                                         create Main at length=L+1
                                         assign Main
                                         give Main private status
                                         give Main multiple-write status
05                                            Can Main be assigned with WR
                                                   no, in use by another
                                                        assign Main
                                                        unlock until succes
06                                                      pause 1 second
                                                        jump to 05
                                                   yes
                                                        copy Buffer to Main
                                                        close Buffer
                                                        purge Buffer
                                                        go back and try -
to dump data again
                                              end
                              end
               end
     End
```

The pauses in O4 and O6 allow the priority II locks routine which is trying to use the file time enough to realize that it must release its file to the priority I locks routine. Dummy determines whether the new Main will fit on the disc; Buffer stores Main's data while it is being expanded.

Part V: The Cheatbox

A simplified version of the checking process for fundamental practice sheets runs as follows: the correct answers are located, the student's answers are compared, a score is generated, the correct answers are erased, the score is stored. It seems that certain students discovered, early in the history of the package, that they could, in checking their practice sheets, type in all wrong answers, receive the correct answers from the computer, disconnect from the computer before it had erased the correct answers, log back on to the computer, and type in the correct answers which the computer had given them.

Two functions have been incorporated to prevent this. Immediately after locating a student's correct answers (2.20), CHECK stores the student's I.D. number in a file known as the "cheatbox" and then changes the sheet number in the file with the correct answers into a non-integer by adding .5 to it. The sheet cannot be accessed again because a user cannot enter a non-integer for a sheet number. The student's I.D. number remains in the cheatbox until he has completed checking his/her entire sheet; after completion, the I.D. number is removed (2.23). If the student disconnects from the computer at any time during the checking process, his/her I.D. remains in the cheatbox, and the student will not be allowed to use the package (for CHMSTR will not grant him/her access) until the instructor removes the I.D. from the cheatbox. Instructors are never put in the cheatbox; as they would be unable to remove themselves from it once they had been put in. The routine which puts users into the cheatbox is shown in (3.09:1510-1670). That which removes them is at (3.09:2420-2700).

Part VI: Input Comparison Modes

The first item in each data storage file for correct answers is a number which tells how many questions there are for that fundamental. There are various coding proceedures which may be performed upon this number which will affect how the user's answers and the correct answers will be compared by CHECK. If this number, known as Q9, is negative, this means that there are varying numbers of questions for each sheet for the fundamental. Therefore there will be a separate Q9 at the beginning of each set of correct answers. The routine which processes this is located at (3.09:1220-1500).

All alphabetic responses and answers are automatically upshifted from lower case, but all blank spaces will also be stripped from all answers (both those entered by the user and those on file) if Q9 has a .1 added to it. The routine for processing this is at (3.09:2810-2870).

All numeric responses are accurate to six significant figures, but all answers will be rounded to two significant figures if Q9 has a .2 added to it. Numeric responses will also be granted a plus or minus 1 error margin in the least significant figure if a .01 is added to Q9. The two significant figure routine is at (3.09:2710-2800), and the error margin routine is at (3.09:1880-1930).

Part VII: Output Formatting for Correct Answers

When CHECK prints out the correct answer to a numeric problem, it is desirable to have it print out in standard numeric format, not in exponential notation. Unfortunately, HP2000/BASIC automatically shifts into exponential notation whenever a number exceeds 999999 or becomes smaller than .000001. Thus, a number such as 20000000 would print as 2.E+07. In order to overcome this difficulty, it was necessary to develop the routine shown at (3.09:1890-1900,1940-2160). This routine operates as follows.

Let A=correct answer, let U$=" 0", let L=log(abs(A))=the number of digits to the left of the decimal point in A minus one.

```
    if A does not equal zero then
        let A equal the absolute value of A
        put A in U$ and move the decimal point to the extreme -
            left of the character string
    endif
    if A is less than zero then
        print a negative sign
    endif
    if L+1 is greater than zero then
        let characterposition=2
01      if there is a character in U$(characterposition) then
            print the character
        else
            print a zero
        endif
        increment characterposition
        if characterposition is less than or equal to L+2 then
            jump to 01
        endif
    endif
    if there remain any characters in U$ which have not been printed then
        print a decimal point
        if -(L+1) is greater than or equal to one then
            let zerocount =1
02          print a zero
            increment zerocount
            if zerocount is less than or equal to -(L+1) then
                jump to 02
            endif
        endif
    endif
stop
```

Part VIII: <u>Sheet Erasure and Score Recording</u>

The only concession which HP2000/BASIC makes for changing data items in the middle of a file is the UPDATE #F statement. With this statement it is only possible to change the value of a particular data item; its type cannot be changed, neither can its length, nor can it be erased completely. Attempting to PRINT #F in the middle of a file succeeds in changing data items, but it erases all the contents of the record directly after where the PRINT #F took place. Thus, in order to remove answer sets from the middle of files and to insert scores into the middle of files, it was necessary to devise the "double-file" technique. This process operates as follows, where * represent the file pointer. Suppose we have a file that looks like this: *item1,item2,item3,item4, item5,end-of-file. We wish to remove item2 and item3 from the file without disturbing the remaining contents. First, we assign the same file to two different locations in the user workspace (3.11:800-1020), location #1 and location #2. We now have a double-file which looks like this:

```
#1:  *item1,item2,item3,item4,item5,eof
#2:  *item1,item2,item3,item4,item5,eof
```

We now tab past the items to be removed in file #2 (3.11:1030-1220):

```
#1:  *item1,item2,item3,item4,item5,eof
#2:   item1,item2,item3,*item4,item5,eof
```

We now tab to the items to be removed in file #1 (3.11:470-500):

```
#1:  item1,*item2,item3,item4,item5,eof
#2:  item1,item2,item3,*item4,item5,eof
```

We now copy file #2 into file #1, starting from the current positions of their respective pointers (3.11:1270-1360):

```
#1:  item1,item4,item5,*eof
#2:  item1,item2,item3,item4,item5,*eof
```

We now, finally, close file #2 (3.11:560) and dump file #1 back onto the disc (3.11:570). We are left with:

```
#1:  *item1,item4,item5,eof
```

For inserting items into the middle of files, the process is just the reverse (3.11:1720-2060).

Now that UPDATE (3.11) has removed the practice sheet from the file, it must check to see if the remaining empty space amounts to a full record or more (3.11:520-560) ; it does this by attempting to read the next record. If an end-of-file condition occurs, there is no empty record; if the read is successful, the extra record must be removed. The shrinking routine for rubber-band files (3.11:1370-1710) operates in much the same manner as does the expansion routine mentioned earlier (in part IV). In fact, it uses the same priority I locks routine. The only differences are: 1)no Dummy is created since there is no need to check for adequate contiguous disc space since the new file will fit in the same space as did the old file, and 2)the Buffer is created at length=L-1 instead of L since the final record of Main is empty and does not need to be put into temporary storage.

DISCUSSION

The very fact that this package continues to be used and continues to grow indicates to me that it has been a success. This package is currently used by many teachers throughout district 214 as illustrated by the user list in (2.33). Please note that the teacher/schools BT and JT are merely used for demonstration and testing purposes and do not represent any regular classes. The continued popularity of this management system with both students and teachers is also exhibited by the usage report in (2.38).

In addition to the programs presented herein, there are also various maintenance programs used in testing, analyzing, and expanding the package. These may be seen on the catalog (3.26).

As stated before, this is not a static project; this is a dynamic system. It is constantly growing. At present I am planning to move the instructors' options from the 100 series of fundamentals to the 1000 series of fundamentals. This will allow 999 fundamental programs for students, instead of the current 99. I am also in the process of revising CATLOG so that all of the data which is now in PRINT statements will be stored in a file. This will allow any user to request a selective catalog when they ask for fundamental zero; so that instead of listing all (potentially 999) of the fundamentals available, a user may request a listing for a single subject area. In the realm of subject areas, it must be emphasized that this system is applicable to any area of study. The fact that it was first applied to chemistry holds no significance. In fact, additional math programs are in the planning stage (these in the area of algebra and trigonometry), and an entire series of English programs are well into the programming stage.

It is to its great flexibility that the Instructional Package Management System owes its popularity and success, and this flexibility is due to the fact that it overcomes the formidable limitations placed upon it by the creators of HP2000/BASIC through highly advanced software development.

APPENDIX I: The Instructors' Options

This appendix presents the functions of the various instructors' options listed in (2.36). Each section of this appendix is prefaced by the name of the program, the position of its listing in Appendix 3, and its fundamental number as designated in the instructors' options catalog (2.36).

1.01 MASFND (3.12) Fundamental #100

This program provides the teacher with the ability to generate a sequential series of practice sheets at one sitting, rather than typing in a student's teacher/school, three initials, and desired fundamental number, waiting for the sheet to print out, typing in the teacher/school, three initials, and fundamental number for the second sheet, etc. A "sequential series" of sheets consists of a given spread of fundamentals (perhaps fundamentals four through seven) for each of a given spread of students (perhaps students 1.01 through 4.99).

The program first requests the teacher to enter the beginning and ending fundamental numbers of the spread desired. He/she is then asked for the beginning and ending spread of I.D. numbers for the desired students. Note that these I.D. numbers are not the same as those discussed in "Methods of Software Development, Part I." The teacher is never made aware of the complexity of the system's identification procedures. To the teacher, an I.D. number consists merely of the A.BB part, where A is the period number of the student's class and BB is the number of the line in the teacher's gradebook on which the student's name appears.

After having been provided with the fundamental spread and the I.D. spread, MASFND locates the desired student in the class-list file and channels the user into the appropriate fundamental program. The fundamental program then generates a practice sheet for the desired student, stores the correct answers, and routes the teacher into NOTICE (3.08). NOTICE immediately recognizes that the user is in the midst of a MASFND series (designated series 300), and instead of printing a brief message and returning the user to MAIN, as it does for normal series 100 users, it routes the user back to a return point (3.12:740) in MASFND. MASFND now moves the paper in the printer to top-of-form. Since HP2000/BASIC does not provide any functions for paginating terminals, MASFND must calculate (3.12:990-1140) the proper number of blank lines to print using information supplied from DATA statements (3.12:850-980) and a spare COM variable (3.12:20) which was assigned a value by the previous fundamental program. MASFND now proceeds to the next fundamental number. When it has exhausted the entire spread of fundamentals for the current student, MASFND resets its fundamental counter and proceeds to the next student. When all fundamentals for all students have been generated, the program returns the user to MAIN. If, at any time during the fundamental processing, MASFND is unable to locate a student I.D. in the class-list file or is unable to locate a given fundamental program, it merely proceeds to the next practice sheet without interrupting the normal pagination. Each practice sheet which is successfully generated is recorded in the usage file as another set of accepted initials.

1.02 LDINTL (3.13) Fundamental #101

This program is used to manipulate a teacher's class-list file, which is named INTLts where "ts" represents the teacher/-school initial pair. LDINTL provides several options to the user (3.13:112-119). By responding to the program's request for a command, a teacher may perform any of the following functions.

"Help" merely prints instructions detailing the operation of the various functions; the user is then asked for another command.

"Reload" is used to erase and completely re-enter a teacher's class-list; because of the severity of this option, the user is allowed to abort its functioning by responding "no" to the program's query "are you sure that you want to do this?". If the teacher proceeds with the option, the class-list file is cleared, and the teacher is asked to enter his/her full name and three initials which are then stored in the file. The teacher is then asked for the number of the period which he/she now wishes to load. Following the teacher's response, LDINTL prints out a student I.D. number in the form A.01. The teacher is then asked to enter that student's three initials and full name; if no student is listed on that line in the teacher's gradebook, he/she merely enters ***. LDINTL then responds with a request for data on student A.02. This continues until A.99 is reached or the teacher types "quit." At that time the teacher is asked for a new command.

"List" is used to list a teacher's class-list file. After entering the numbers of the periods at which the teacher wishes the listing to begin and to end, the user is provided with a printout of the initials, I.D. numbers (in the form A.BB), and full names of all of the students within the requested spread. The teacher is then asked to enter another command.

"Add" is used to either add a new class period to the file or to add to an already existing class period. The teacher is first asked to enter the number of the class period desired. If it does not exist, the teacher proceeds to load it according to the proc e dure already outlined under the "Reload" command. If some students are already on file for that hour, the teacher has two options. He/she may either erase the already stored students and reload the entire period, in which case the period is erased and loading proceeds as under "Reload," or he/she may elect to preserve the already stored students and begin loading from the first available I.D. number, in this case loading takes place as under "Reload" with the exception that the loading does not start with A.01. After the teacher types "quit," he/she is asked to enter another command.

"Delete" is used to remove a student from the class-list. The teacher is asked to enter the I.D. number (in the form A.BB) of the student to be deleted. The initials of that student are then replaced by "***" and the name by "* *" in the class-list file. The purpose of changing the initials and name, as opposed to removing the student entirely, is to permit the teacher to enter data for a different student with the same I.D. number at a future data, should a new student appear on the same line of the teacher's gradebook. This insertion of a new student can be accomplished through use of the "Correct" command detailed below. After the teacher types "quit" for an I.D. number, he/she is asked for another command.

"Correct" is used to change the three initials and name of a student whose I.D. number has already been assigned. After entering the I.D. number (in the form A.BB) of the student to be changed, the teacher is asked to enter the new three initials and new full name. When the teacher types "quit" for an I.D. number, he/she is asked to enter a new command.

"Duplicates" performs a duplicate search on the class-list file. Since all users are identified by means of their teacher/-school and three initials, it is imperative that no two individuals in the same teacher/school class-list file possess the same three initials. This command informs the teacher of any such duplicates so that he/she can change the initials of one of the individuals using the "Correct" command detailed above. This is commonly done by changing the middle initial of one of the individuals to an "X." After the duplicate search is completed, the user is asked for another command.

"Stop" returns the user to MAIN.

1.03 OUTPUT (3.14) Fundamental #102

This program is used for accessing the scores which have been stored for checked practice sheets. It will print out the scores for each student for a maximum of twenty-four fundamentals (twenty-four scores are all that will fit on a sheet of paper). The teacher is first offered the option of listing scores for a contiguous spread of fundamentals, in which case he/she is asked to enter beginning and ending fundamental numbers, or scores for a group of selected fundamentals, in which case the teacher is asked to enter a list of up to twenty-four fundamental numbers, one at a time, terminating the list by typing a zero. This teacher is next offered the selection of the maximum-grades-only option. Normally OUTPUT lists all scores received by each student for the fundamentals desired; if this option is selected, however, OUTPUT will print only the highest score received by each student on each fundamental.

The teacher is now asked if he/she wishes to list only selected individual students. If not, the teacher is asked to enter the beginning and ending period numbers for the listing. The listing then proceeds: each student's I.D. number, three initials, and full name are printed first; then the scores are printed in evenly-spaced columns. Only those students are listed who have at least one score recorded for at least one of the desired fundamentals, if no score is on file for a particular fundamental, an " * " is printed.

If the teacher selects the option of listing individual students, he/she is asked to enter the "code" for the desired student. The code consists of the first letter of the student's first name followed by the first <u>two</u> letters of his/her last name (for Craig Bina the code would be CBI). This code is entered rather than the student's three initials because it is far more likely that a teacher will remember a student's first and last names (or be able to locate them in his gradebook) than the student's three initials. Since there is no restriction on students having identical codes, it is quite possible that OUTPUT will locate a student who is not the one desired by the teacher. When OUTPUT locates a student with the appropriate code, it prints the student's I.D. number, three initials, and full name. The teacher is then asked if OUTPUT has found the correct student. If the reply is "no," the program continues its search until it finds a student with the appropriate code to whom the teacher responds with a "yes." If the reply is "yes," the student's scores are printed out as detailed in the preceding paragraph. After completion of the score printing, the teacher is asked for another code. This process continues until the teacher presses only the RETURN key instead of entering a code.

When all scores have been printed out for the requested spread of periods or, in the case of individual printouts, the teacher has pressed RETURN instead of entering a code, the user is returned to MAIN.

1.04 REMARK (3.15) Fundamental #103

This program allows teachers to leave and receive messages for and from one another. In REMARK's current state, teachers may leave messages for any teacher/school, but as noted in (3.15:110), by changing (3.15:100) from "U=1" to "U=0" one may prohibit teachers from leaving messages for any teacher/school other than that one which has been designated "the Authorities." The Authorities consist of those individuals who possess programmer-level access to the package and who are responsible for the maintenance of the package; the Authorities, of course, can always send messages to any teacher/school, regardless of the value of the variable "U."

REMARK first prints out all messages addressed to the teacher/school representing the current user; as each message is printed, it is erased from the message-storage file (MESSAQ). Each message is prefaced by the name and teacher/school of the teacher who left the message and the date on which the message was left. The user is then informed of which teacher/school currently represents the Authorities, and he/she is given to option of obtaining a list of the names and teacher/schools of all teachers who have been designated permanent users. The user is then given the opportunity to leave a message for the teacher/school of his/her choice. Messages may consist of any number of lines of up to 255 characters each. To terminate a message, the user merely presses the RETURN key instead of entering another message line. At the conclusion of a message, the teacher is given the opportunity to leave another message for another teacher/school. This process continues until the teacher declines to leave a message; at this time the user is returned to MAIN.

1.05 ERASE (3.16) Fundamental #104

This program is used to remove from the package's data storage files the correct answers for any practice sheets which will no longer be needed by the teacher's classes. The teacher is asked to enter beginning and ending fundamental numbers and beginning and ending I.D. numbers (in the form A.BB). Then ERASE removes all practice sheet answers for all I.D. numbers within the desired spread for the user's teacher/school from the data storage files for all of the fundamentals within the desired spread.

The teacher is then given the option of removing all data files of all types for his/her teacher/school from the management system. If he/she declines, he/she is returned to MAIN. If the user accepts, he/she is given a chance to abort his/her decision through the program's query of "are you sure that you want to do this?". If the teacher chooses to continue, ERASE destroys all data files connected with the user's teacher/school from the package, and it also removes evidence of the teacher/school's existence from all general data storage files. Thus, the teacher/school has been effectively removed from the system; all is as if the teacher/school had never existed, except for the immutable record of the number of usages stored in the usage record file. The teacher is subsequently logged off.

1.06 RNDFND (3.17) Fundamental #105

This program generates the fundamental requests which were stored in the file RNDMts (where "ts" represents the teacher/school initial pair) whenever a user typed "R" for a request in CHMSTR (see "Methods of Implementation," page 7, paragraph 1). RNDFND gives the teacher three options. He/she may select option 1, in which case only the student-stored requests are generated. He/she may select option 2, in which case the user may add to the student requests before generating them. Or he/she may select option 3, which clears all of the student requests, allows the teacher to enter his/her own requests, and then generates the newly-entered requests. Option 3 allows the user to abort it by asking "are you sure that you want to do this?".

A teacher enters a request by typing the three initials of of the student for whom the practice sheet is being generated followed by the number of the desired fundamental (thus a request for a fundamental seven practice sheet for Craig R. Bina would be entered by typing "CRB7"). When the teacher is finished entering all of his/her requests, he/she types "complete."

Practice sheet generation in RNDFND takes place in much the same manner as it does in MASFND (1.01), but NOTICE (3.08) recognizes the user as being in a number 400 series and returns the teacher to a return point (3.17:800) in RNDFND, instead of to MASFND. Pagination takes place in the same manner and each sheet which is generated is recorded as a set of accepted initials in the usage record file. After all requests have been processed, the request storage file is cleared and the user returned to MAIN.

1.07 LGLIZE (3.18) Fundamental #106

This program is used to "legalize" students who have been forbidden access to the package through the workings of the cheatbox system (see "Methods of Software Development," part V). It provides the teacher with four different functions.

The teacher is given the option of entering a beginning and an ending fundamental number. The program then lists the identities of and the correct answers for all incompletely checked practice sheets (those which have had .5 added to their sheet numbers) for the user's teacher/school.

Next, the teacher is given the option of listing all of the cheatbox occupants from his/her teacher/school along with the numbers of the fundamentals upon which they attempted to cheat.

Then the teacher is given the option of altering the status of incomplete practice sheets. If the sheet was incompletely checked due to an unintentional, accidental disconnection, the teacher may chose to "revive" the sheet; reviving a sheet returns the sheet number to integer status, thus making the sheet accessible in the future. The teacher may otherwise choose to "destroy" the sheet; this choice removes a sheet from the file, thus making it forever inaccessible. A status alteration request is made by entering the three initials of the student to whom the sheet belongs, followed by a letter signifying the alteration to be made (R for revive, D for destroy), followed by the fundamental number, followed by a comma, followed by the sheet number (thus, a request to revive sheet #2 for fundamental 16 for Craig R. Bina would be made by typing CRBR16,2). The teacher must enter "stop" for a request when he/she is done with all alterations.

Finally, the teacher is given the option of removing students from the cheatbox, thus allowing them to use the package once again. A request to remove a student from the cheatbox is made by typing the three initials of the student followed by the number of the fundamental on which the student attempted to cheat (thus, to remove Craig R. Bina from the cheatbox for fundamental number 5, the teacher would type CRB5). The teacher must enter "stop" for a request when he/she is finished removing students from the cheatbox.

After the teacher has been given the option to perform all four of these functions, he/she is returned to MAIN.

1.08 REPORT (3.19) Fundamental #107

This program is used to view the usage record file which records the total number of initial sets which were accepted each day over a two-year span. The user first enters a beginning report date (month,date,year), followed by an ending report date. REPORT then prints out each date which falls within the desired spread, followed by the usage count for that day. When all days within the given spread have been listed, a total usage count for the entire desired time period is printed. The teacher is then returned to MAIN.

1.09 INSERT (3.20) Fundamental #108

Shoulda teacher, for any reason, find it necessary to grade a fundamental practice sheet by hand, he/she may use this program to record the score in the score storage file (RECRts, where "ts" represents the user's teacher/school initial pair). The teacher is first asked to enter the I.D. number (in the form A.BB) of the student whose score is to be stored. He/she then enters the number of the fundamental for which the score was received. Finally, the teacher is asked to enter the actual score received by the student, as a percentage. INSERT then routes the user into UPDATE (3.11) which records the score and returns the user to INSERT. The teacher may now enter another I.D. number. This process continues until the teacher enters -1 for an I.D. number, at which time he/she is returned to MAIN.

1.10 MASLPR (3.21) Fundamental #109

This program is used to request sequential fundamental generation at a remote loaction. The method of entering requests for MASLPR is exactly the same as that for MASFND (1.01). The teacher is asked to enter beginning and ending fundamental numbers, and beginning and ending I.D. numbers. This request is then stored in the file RQUEST. After each request, the user is asked if he/she would like to file another request. If the response is "yes," the process is repeated; if it is "no," the user is returned to MAIN.

Later in the evening, Central Administration uses the program REMOTE (3.25) to generate the requests on the district's 600 line-per-minute lineprinter; the printouts are distributed to the various schools via intradistrict mail and placed in the appropriate teachers' mailboxes the next morning. REMOTE's functioning is almost exactly identical to that of MASFND (1.01). Notable exceptions are: REMOTE prints a header sheet at the start of each request which instructs Central where to send the printout; The pagination and print format are directed toward a lineprinter

ASCII-file device (which does have provisions for a top-of-form function); a routine is interwoven with the normal functioning of REMOTE which provides for detection of an out-of-paper condition on the lineprinter (HP2000/BASIC provides no such function); the requests stored in RQUEST are sequentially erased as each one is completed; NOTICE (3.08) recognizes that the user is in a number 200 series (the MASLPR series) and returns the user to the return point (3.25:810) in REMOTE, rather than to MAIN. After all requests have been generated, the administrator is logged off.

1.11 CHGACC (3.22) Fundamental #110

This program allows a teacher to alter his instructors' options access code (stored in the file ACCESS), should he/she feel the need to increase his/her security. Since the teacher needs to know the old access code in order to gain access to this instructors' option, the program merely asks the user for the new access code, which is then substituted for the old one. The user is then returned to MAIN.

1.12 PSEUDO (3.24) Maintenance Program

PSEUDO is not truly an instructors' option; it is a program used to simulate being routed from another package program. It accompiishes this by initializing the first four COM variables (I.D. number, name, fundamental number, series number); the values of the first three are entered by the user; the fourth is set to zero. It is a maintenance program, and as such it is usable only by the programmer, but it performs a special function when used in conjunction with LGLIZE (1.07). When LGLIZE is activated in the normal package sequence, the fundamental number is set equal to 106. In the program itself, a teacher may only list and alter those incomplete practice sheets and those cheatbox occupants who are of his/her teacher/school. If the programmer uses PSEUDO to set the fundamental number to zero, however, and has PSEUDO route him/her into LGLIZE, LGLIZE will list and alter all incomplete practice sheets and cheatbox occupants. The user is assigned a teacher school of "$$," and each time an alteration function is performed, LGLIZE will ask the user to enter the initial pair of the teacher/school upon which it is to operate.

APPENDIX II: A Sample Execution

(Printing enclosed in □ is user input.)

```
2.01  PLEASE LOG IN
      HEL-W140,RIKU,2
2.02  Logon D214 Access Basic  MON,Dec 10,1979
      Port #31  17:58:13  284 Records Remain
2.03


2.04

      *****************WELCOME TO THE INSTRUCTIONAL PACKAGE*****************

      PRESS <RETURN> AFTER ALL RESPONSES.

2.05  PLEASE TYPE YOUR TEACHER'S LAST INITIAL, 'STOP' TO LOG OFF, OR 'SAME'
      TO RETAIN THE TEACHER/SCHOOL & 3 INITIALS OF THE PREVIOUS USER. > B
2.06  WHAT IS YOUR SCHOOL'S INITIAL? T
2.07  TO BEGIN WORK ON FUNDAMENTALS, TYPE YOUR INITIALS IN THE FORM: FML
      :> AAB
2.08  HELLO ALPHA BETA, WELCOME TO THE INSTRUCTIONAL PACKAGE PROGRAMS.
2.09  ALL RIGHT ALPHA,
2.10  WHICH FUNDAMENTAL NUMBER DO YOU WANT?0
      YOU HAVE JUST SELECTED THE OPTION OF GETTING A CATALOG OF
      THE DIFFERENT FUNDAMENTALS.
2.11

         CATALOG OF INSTRUCTIONAL PROGRAMS

      FUND. NO.              TITLE

         0                CATALOG OF FUNDAMENTALS

      CHEMISTRY PROGRAMS

         1                ARITHMETIC
         2                SIGNIFICANT FIGURES
         3                GRAPHING
         4                SCIENTIFIC NOTATION
```

Craig Bina

```
     5                    DIATOMIC ELEMENTS
     6                    WEIGHTS AND MOLES
     7                    ANALYZING PROBLEMS
     8                    RATIO AND PROPORTION
     9                    WRITING IONIC FORMULAS
    10                    WRITING EQUATIONS
    11                    BALANCING EQUATIONS
    12                    CHANGING PARTNERS
    13                    CROSSING THE BRIDGE
    14                    MOLAR VOLUME
    15                    CALORIES
    16                    SOLUBILITY
    17                    DISSOLVING EQUATIONS FOR IONIC COMPOUNDS
    18                    NET IONIC EQUATIONS
    19                    PREPARING SOLUTIONS
    20                    PERIODIC TABLE
    21                    PROTONS, NEUTRONS, AND ELECTRONS
    22                    FORMULAS FROM THE PERIODIC TABLE
    23                    PROPERTIES OF COMPOUNDS
    24                    IONS FROM ATOMS
    25                    TYPES OF CHEMICAL REACTIONS

  MATH PROGRAMS

    50                    ONE-DIGIT MULTIPLICATION
    51                    TWO DIGIT BY ONE DIGIT MULTIPLICATION
    52                    TWO DIGIT BY TWO DIGIT MULTIPLICATION
    53                    MULTIPLYING WITH DECIMALS
    54                    DIVISION WITH SINGLE DIGIT DIVISOR
    55                    DIVISION WITH TWO DIGIT DIVISOR
    56                    DIVISION WITH THREE DIGIT DIVISORS (TOUGH ONES)
    57                    DIVIDING WITH DECIMALS

2.12
2.13  PLEASE TYPE YOUR TEACHER'S LAST INITIAL, 'STOP' TO LOG OFF, OR 'SAME'
      TO RETAIN THE TEACHER/SCHOOL & 3 INITIALS OF THE PREVIOUS USER. > SAME
      HELLO ALPHA BETA, WELCOME TO THE INSTRUCTIONAL PACKAGE PROGRAMS.
      ALL RIGHT ALPHA,
2.14  WHICH FUNDAMENTAL NUMBER DO YOU WANT?6
2.15  TYPE  P  TO PRACTICE,  C  TO CHECK, OR  R  TO REQUEST A FUNDAMENTAL.
      WHAT IS YOUR CHOICE?P

2.16  FUNDAMENTAL #6:  WEIGHTS AND MOLES

        1)   WHAT IS THE MOLAR MASS OF BERKELIUM TO THE NEAREST GRAM?

      CALCULATE THE ANSWERS TO #2-5 TO 2 SIGNIFICANT FIGURES.
```

Craig Bina

```
     2)   HOW MANY MOLES OF MERCURY ARE IN 460 GRAMS OF Hg ?

     3)   HOW MANY GRAMS OF TERBIUM ARE IN 76 MOLES OF Tb ?

     4)   HOW MANY MOLES OF Sn(OH)2 ARE IN 250 GRAMS OF Sn(OH)2 ?

     5)   HOW MANY GRAMS OF NO ARE IN 240 MOLES OF NO ?

     THIS IS SHEET # 2     FOR ALPHA BETA.

2.17                         PERIOD 1

     IF ANOTHER STUDENT WISHES TO USE THIS TERMINAL, PLEASE LEAVE IT AS IS
     AND WORK THESE PROBLEMS ELSEWHERE.  RETURN LATER TO CHECK YOUR ANSWERS.
2.18
     PLEASE TYPE YOUR TEACHER'S LAST INITIAL, 'STOP' TO LOG OFF, OR 'SAME'
     TO RETAIN THE TEACHER/SCHOOL & 3 INITIALS OF THE PREVIOUS USER. >SAME
     HELLO ALPHA BETA, WELCOME TO THE INSTRUCTIONAL PACKAGE PROGRAMS.
     ALL RIGHT ALPHA,
     WHICH FUNDAMENTAL NUMBER DO YOU WANT?6
2.19 TYPE  P  TO PRACTICE,  C  TO CHECK, OR  R  TO REQUEST A FUNDAMENTAL.
     WHAT IS YOUR CHOICE?C
2.20 NOW, ALPHA, WHAT IS YOUR SHEET NUMBER?2

2.21 RESPONSE TO QUESTION #  1 -> 247
     CORRECT
     RESPONSE TO QUESTION #  2 -> 1
     THE CORRECT ANSWER IS:  2.3
     RESPONSE TO QUESTION #  3 -> 1
     THE CORRECT ANSWER IS:  12000
     RESPONSE TO QUESTION #  4 -> 1
     THE CORRECT ANSWER IS:  1.6
     RESPONSE TO QUESTION #  5 -> 1
     THE CORRECT ANSWER IS:  7200
2.22 YOU HAD   4 WRONG OUT OF   5 PROBLEMS.
     THIS IS A SCORE OF  20.0% RIGHT.
2.23
2.24 PLEASE STAND BY WHILE YOUR SCORE IS RECORDED.

2.25 PLEASE TYPE YOUR TEACHER'S LAST INITIAL, 'STOP' TO LOG OFF, OR 'SAME'
     TO RETAIN THE TEACHER/SCHOOL & 3 INITIALS OF THE PREVIOUS USER. >SAME
     HELLO ALPHA BETA, WELCOME TO THE INSTRUCTIONAL PACKAGE PROGRAMS.
     ALL RIGHT ALPHA,
     WHICH FUNDAMENTAL NUMBER DO YOU WANT?6
2.26 TYPE  P  TO PRACTICE,  C  TO CHECK, OR  R  TO REQUEST A FUNDAMENTAL.
     WHAT IS YOUR CHOICE?R
2.27 YOUR REQUEST HAS BEEN FILED.
```

Craig Bina

41

```
2.28
2.29 PLEASE TYPE YOUR TEACHER'S LAST INITIAL, "STOP" TO LOG OFF, OR "SAME"
     TO RETAIN THE TEACHER/SCHOOL & 3 INITIALS OF THE PREVIOUS USER. >B
     WHAT IS YOUR SCHOOL'S INITIAL?T
     TO BEGIN WORK ON FUNDAMENTALS, TYPE YOUR INITIALS IN THE FORM: FML
     :>CRB
     HELLO CRAIG BINA, WELCOME TO THE INSTRUCTIONAL PACKAGE PROGRAMS.
     ALL RIGHT CRAIG,
2.30 WHICH FUNDAMENTAL NUMBER DO YOU WANT?103
2.31 YOU HAVE JUST SELECTED AN INSTRUCTOR'S OPTION.

     PLEASE ENTER YOUR ACCESS CODE.


2.32 INTERSCHOOL INSTRUCTOR MESSAGES AND COMMENTS

     MESSAGES FOR TEACHER/SCHOOL: BT


     END-OF-MESSAGES

     THE AUTHORITIES ARE CURRENTLY AT TEACHER/SCHOOL "BW"

2.33 WOULD YOU LIKE A LIST OF USERS?YES

     T/S        NAME

     BT         CRAIG BINA
     BW         DON BAUDER
     DF         MIKE DUELLMAN
     GW         GEORGIA GAY
     IV         SUE ILLINGWORTH
     JE         JAKE TURBAN
     KW         CAROL KALP
     SF         LESLIE STEARNES
     TE         MARILYN THACKER
     WW         REINE WASEMILLER
     ZW         BILL ZORBA
     AR         CARL ANDERSON
     MW         DAN MAJKA
     LR         JIM LEVINS
     SR         JOHN STITT
     GH         JAMES GORDON
     JT         JOHN JONES
     WC         GAIL WEATHERS
     SF         DOMINIC SPARACINO
```

Craig Bina

42

```
     WE          SUSAN WOOD

     DO YOU WISH TO SEND A MESSAGE?NO
     END OF MESSAGE MANIPULATION.
2.34
     PLEASE TYPE YOUR TEACHER'S LAST INITIAL, "STOP" TO LOG OFF, OR "SAME"
     TO RETAIN THE TEACHER/SCHOOL & 3 INITIALS OF THE PREVIOUS USER. > SAME
     HELLO CRAIG BINA, WELCOME TO THE INSTRUCTIONAL PACKAGE PROGRAMS.
     ALL RIGHT CRAIG,
2.35 WHICH FUNDAMENTAL NUMBER DO YOU WANT?0
     YOU HAVE JUST SELECTED THE OPTION OF GETTING A CATALOG OF
     THE DIFFERENT FUNDAMENTALS.

2.36 WOULD YOU LIKE A LISTING OF ONLY THE INSTRUCTORS' OPTIONS?YES


               CATALOG OF INSTRUCTIONAL PROGRAMS


     FUND. NO.                       TITLE

     INSTRUCTORS' OPTIONS

     *100                 SEQUENTIAL PRACTICE SHEET GENERATION
     *101                 CLASS LIST CREATION AND ALTERATION
     *102                 FUNDAMENTAL PRACTICE GRADEBOOK
     *103                 INSTRUCTOR MESSAGES AND COMMENTS
     *104                 ERASURE OF UNUSED FILE INFORMATION
     *105                 NON-SEQUENTIAL PRACTICE SHEET GENERATION
     *106                 LEGALIZATION OF STUDENT CHEATERS
     *107                 INSTRUCTIONAL PACKAGE USAGE REPORT
     *108                 FILING OF HAND-SCORED PRACTICE SHEETS
     *109                 REQUEST FOR GENERATION OF SEQUENTIAL PRACTICE
                             SHEETS BY COMPUTER SERVICES
     *110                 INSTRUCTORS' OPTIONS ACCESS CODE ALTERATION

          * DENOTES INSTRUCTORS-ONLY FUNDAMENTALS


2.37
     PLEASE TYPE YOUR TEACHER'S LAST INITIAL, "STOP" TO LOG OFF, OR "SAME"
     TO RETAIN THE TEACHER/SCHOOL & 3 INITIALS OF THE PREVIOUS USER. > SAME
     HELLO CRAIG BINA, WELCOME TO THE INSTRUCTIONAL PACKAGE PROGRAMS.
     ALL RIGHT CRAIG,
     WHICH FUNDAMENTAL NUMBER DO YOU WANT?107
     YOU HAVE JUST SELECTED AN INSTRUCTOR'S OPTION.

     PLEASE ENTER YOUR ACCESS CODE
```

Craig Bina

```
PLEASE ENTER YOUR ACCESS CODE:

2-38  INSTRUCTIONAL PACKAGE USAGE REPORT

  ENTER INITIAL REPORT DATE (EXAMPLE:  7,4,61)?9,1,79
  ENTER TERMINAL REPORT DATE (EXAMPLE:  8,8,62)?12,11,79


  SEP  1          0
  SEP  2          0
  SEP  3          0
  SEP  4          0
  SEP  5       1048
  SEP  6          0
  SEP  7          0
  SEP  8          0
  SEP  9          0
  SEP 10         42
  SEP 11          0
  SEP 12        178
  SEP 13        220
  SEP 14          0
  SEP 15          0
  SEP 16          0
  SEP 17         36
  SEP 18         49
  SEP 19        148
  SEP 20        561
  SEP 21        542
  SEP 22         66
  SEP 23          0
  SEP 24         59
  SEP 25        162
  SEP 26         29
  SEP 27         65
  SEP 28         71
  SEP 29          0
  SEP 30          0
  OCT  1        712
  OCT  2        825
  OCT  3         64
  OCT  4        234
  OCT  5        987
  OCT  6         41
  OCT  7         90
  OCT  8          1
  OCT  9        184
  OCT 10        104
```

Craig Bina

44

OCT 11	76
OCT 12	0
OCT 13	0
OCT 14	0
OCT 15	15
OCT 16	194
OCT 17	192
OCT 18	141
OCT 19	99
OCT 20	0
OCT 21	0
OCT 22	107
OCT 23	179
OCT 24	148
OCT 25	85
OCT 26	250
OCT 27	0
OCT 28	0
OCT 29	13
OCT 30	20
OCT 31	1
NOV 1	179
NOV 2	512
NOV 3	0
NOV 4	0
NOV 5	47
NOV 6	65
NOV 7	102
NOV 8	350
NOV 9	320
NOV 10	0
NOV 11	2
NOV 12	327
NOV 13	101
NOV 14	257
NOV 15	83
NOV 16	123
NOV 17	0
NOV 18	0
NOV 19	53
NOV 20	79
NOV 21	71
NOV 22	0
NOV 23	0
NOV 24	0
NOV 25	0
NOV 26	30
NOV 27	537
NOV 28	85
NOV 29	52
NOV 30	1026

Craig Bina

```
   DEC  1          0
   DEC  2          0
   DEC  3        399
   DEC  4         57
   DEC  5         47
   DEC  6        289
   DEC  7        119
   DEC  8          0
   DEC  9          0
   DEC 10        184
   DEC 11          0



   TOTAL NUMBER OF INITIALS ENTERED IN THIS TIME PERIOD IS 13534



2.39
   PLEASE TYPE YOUR TEACHER'S LAST INITIAL, "STOP" TO LOG OFF, OR "SAME"
   TO RETAIN THE TEACHER/SCHOOL & 3 INITIALS OF THE PREVIOUS USER. > ***
2.40 CODE *>

2.41 DONE
```

Craig Bina

APPENDIX III: Program Listings

3.01
MAIN

```
10   COM A,A$[80],F,X,Y,P$[11],E[3]
20   B=BRK(0)
30   P$="MAIN"
40   S=1
50   IF  ERROR  THEN 80
60   S=A
70   S=0
80   IF  ERROR  THEN 240
90   REM FOLLOWING VALUE IS DESTINATION LINE NUMBER FOR CHMSTR
100   L=140
110   IF S THEN 150
120   PRINT
130   CHAIN R,"CHMSTR",L
140   GOTO 200
150   PRINT LIN(3)
160   REM  PACKAGE BANNER FOLLOWS THIS LINE
170   REM  END OF PACKAGE BANNER
180   PRINT LIN(3)
190   CHAIN R,"CHMSTR"
200   PRINT "PROGRAM: CHMSTR  IS MISSING, PLEASE INFORM INSTRUCTOR."
210   PRINT "NO ACCOUNT ACCESS IS PERMITTED."
220   SYSTEM R,"BYE"
230   GOTO 220
240   REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
250   FOR I=0 TO 2
260   E[I+1]=SYS(I)
270   NEXT I
280   CHAIN R,"ERROR"
290   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
300   PRINT "PLEASE INFORM YOUR INSTRUCTOR."
310   GOTO 210
320   END
```

Craig Bina

3.02
CHMSTR

```
10   COM A,NO$[80],F,D1,D2,OO$[11],E[3]
20   OO$="CHMSTR"
30   IF  ERROR  THEN 2950
40   Y=BRK(0)
50   REM**********************NEVER RENUMBER THIS PROGRAM***************
70   DIM A$[255],B$[255],C$[3],L$[20],Q$[15]
80   DIM Y[367]
90   FILES *,*
100   PRINT
110   PRINT "*******************WELCOME TO THE INSTRUCTIONAL PACKAGE*****************"'10'10
120   A=-1
130   PRINT "PRESS <RETURN> AFTER ALL RESPONSES."'10
140   REM CHAIN HERE FOR A RETURN!
150   Y=BRK(0)
160   OO$="CHMSTR"
170   D1=D2=100
180   IF  ERROR  THEN 2950
190   PRINT "PLEASE TYPE YOUR TEACHER'S LAST INITIAL, ";
200   PRINT '34"STOP"'34" TO LOG OFF, OR "'34"SAME"'34
210   PRINT "TO RETAIN THE TEACHER/SCHOOL & 3 INITIALS OF THE PREVIOUS USER. > ";
220   LINPUT B$
230   IF B$="***" THEN 1950
240   IF B$="STOP" THEN 1540
250   IF B$="CENTRAL" THEN 2070
260   IF B$#"SAME" THEN 500
270   IF A <= 0 THEN 480
280   L=INT(LOG(A)/2.30259)
290   CONVERT A/10^(L-1) TO Q$
300   CONVERT Q$ TO Q
310   Q=L+5* NOT INT(100*(Q-INT(10*Q)/10))
320   Q$=CHR$(Q+64)
330   Q=A/10^Q
340   Q$[2]=CHR$(INT(10000*(Q-INT(100*Q)/100)+.9))
350   A$="INIL"
360   A$[5]=Q$
370   ASSIGN A$,1,R
380   IF R>2 THEN 710
390   IF  END #1 THEN 460
400   CONVERT A TO A$
410   READ #1:C$,A,NO$
420   CONVERT A TO B$
430   IF B$#A$ THEN 410
440   ASSIGN *,1
450   GOTO 880
460   PRINT "YOUR I.D. NUMBER IS NO LONGER ON FILE."
470   GOTO 190
480   PRINT "NO PREVIOUS USER"
490   GOTO 190
500   IF B$#"!" THEN 610
510   PRINT "YOU HAVE JUST SELECTED THE INITIALIZATION OPTION.  THIS OPTION"
520   PRINT "INSERTS A USER INTO THE PACKAGE BY CREATING ALL NECESSARY"
530   PRINT "TEACHER/SCHOOL FILES.  PLEASE ENTER YOUR USAGE CODE: ";
540   LINPUT A$
550   IF A$#------code-------- THEN 580
560   CHAIN R,"SETUP"
570   GOTO 2890
580   PRINT "YOU HAVE ENTERED AN UNACCEPTABLE USAGE CODE, NO ACCESS PERMITTED."
590   PRINT
600   GOTO 1550
```

Craig Bina

```
610   IF LEN(B$)#1 OR NUM(B$)<65 OR NUM(B$)>90 THEN 190
620   A$="INTL"
630   A$[5]=B$
640   PRINT "WHAT IS YOUR SCHOOL'S INITIAL? ";
650   LINPUT B$
660   IF LEN(B$)#1 OR NUM(B$)<65 OR NUM(B$)>90 THEN 640
670   A$[6]=B$
680   ASSIGN A$,1,R
690   Q$=A$[5]
700   IF R<3 THEN 730
710   PRINT "UNACCEPTABLE, TEACHER/SCHOOL "Q$" IS NOT ON FILE."
720   GOTO 190
730   X=0
740   X=X+1
750   IF X<3 THEN 780
760   PRINT "PERHAPS YOU ENTERED THE WRONG TEACHER/SCHOOL INITIALS."
770   GOTO 190
780   PRINT "TO BEGIN WORK ON FUNDAMENTALS, TYPE YOUR INITIALS IN THE FORM: FML"
790   PRINT ":> ";
800   LINPUT B$
810   IF LEN(B$)#3 THEN 740
820   IF B$="***" THEN 1950
830   IF  END #1 THEN 1210
840   READ #1,1
850   READ #1;C$,A,NO$
860   IF B$#C$ THEN 850
870   ASSIGN *,1
880   ASSIGN "USAGE",1,R
890   IF R THEN 1180
900   LOCK #1,R
910   IF R THEN 900
920   Y6=TIM(3)-1+100* NOT TIM(3)
930   Z6=Y6-1+100* NOT Y6
940   W6=TIM(2) MIN 366
950   IF  END #1 THEN 990
960   MAT  READ #1;Y
970   X6=(Y[1]=Z6)+2*(Y[1]=Y6)
980   GOTO X6 OF 1050,1120
990   IF  END #1 THEN 1180
1000  MAT Y=ZER
1010  Y[1]=Y6
1020  READ #1,1
1030  MAT  PRINT #1;Y, END
1040  GOTO 1060
1050  READ #1,4
1060  MAT  READ #1;Y
1070  GOTO (Y[1]=Y6)+1 OF 990,1020
1080  MAT Y=ZER
1090  Y[1]=TIM(3)
1100  READ #1,4
1110  MAT  PRINT #1;Y, END
1120  READ #1,4
1130  ADVANCE #1;W6,R
1140  READ #1:X6
1150  READ #1,4
1160  ADVANCE #1;W6,R
1170  UPDATE #1;X6+1
1180  ASSIGN *,1,R
1190  PRINT "HELLO "NO$", WELCOME TO THE INSTRUCTIONAL PACKAGE PROGRAMS."
1200  GOTO 1230
```

Craig Bina

```
1210  PRINT B$" IS NOT LISTED IN TEACHER/SCHOOL FILE "Q$"."
1220  GOTO 740
1230  ASSIGN "CHTBOX",2,R
1240  IF R THEN 1380
1250  LOCK #2,R
1260  IF R THEN 1250
1270  IF  END #2 THEN 1380
1280  CONVERT A TO B$
1290  READ #2;N1,N2
1300  CONVERT N1 TO L$
1310  IF L$#B$ THEN 1290
1320  PRINT
1330  PRINT "I AM SORRY, "NO$", BUT MY FILES INDICATE THAT YOU HAVE ATTEMPTED TO"
1340  PRINT "CHEAT WHILE CHECKING A PRACTICE SHEET FOR FUNDAMENTAL NUMBER"N2"."
1350  PRINT "PLEASE INFORM YOUR INSTRUCTOR.  YOU WILL NOT BE GRANTED ACCESS TO THE"
1360  PRINT "PACKAGE UNTIL YOU HAVE DONE SO AND HE HAS CLEARED YOU FOR FURTHER WORK."'10
1370  GOTO 190
1380  UNLOCK #2
1390  ASSIGN *,2
1400  PRINT "ALL RIGHT "NO$[1,POS(NO$," ")-1]","
1410  PRINT "WHICH FUNDAMENTAL NUMBER DO YOU WANT";
1420  INPUT F
1430  IF F=0 THEN 2310
1440  IF F>99 AND F=INT(F) THEN 2370
1450  IF F<1 OR F>99 OR F#INT(F) THEN 1410
1460  PRINT "TYPE  P  TO PRACTICE,  C  TO CHECK, OR  R  TO REQUEST A FUNDAMENTAL."
1470  PRINT "WHAT IS YOUR CHOICE";
1480  INPUT B$
1490  IF B$="P" THEN 1670
1500  IF B$="C" THEN 1570
1510  IF B$="R" THEN 1730
1520  IF B$="CR04" THEN 1950
1530  GOTO 1460
1540  PRINT "THANK YOU FOR USING THE INSTRUCTIONAL PACKAGE."LIN(10)
1550  SYSTEM R,"BYE"
1560  GOTO 1550
1570  REM CHECKING
1580  IF F#3 THEN 1610
1590  CHAIN R,"AUXCK3"
1600  GOTO 1620
1610  CHAIN R,"CHECK"
1620  PRINT "SORRY, THE PROGRAM IS UNACCESSABLE.  PLEASE INFORM YOUR TEACHER."
1630  CHAIN R,"MAIN"
1640  PRINT "THE PROGRAM:  MAIN  IS MISSING."
1650  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
1660  GOTO 1550
1670  REM PRACTICING
1680  B$="FUND"
1690  CONVERT F TO B$[5]
1700  PRINT
1710  CHAIN R,B$
1720  GOTO 1620
1730  REM REQUESTING
1740  B$="RNDM"
1750  B$[5]=Q$
1760  ASSIGN B$,1,R
1770  IF  NOT R THEN 1800
1780  PRINT "FILE:  "B$"  IS CURRENTLY UNAVAILABLE, PLEASE INFORM YOUR INSTRUCTOR."
1790  GOTO 190
1800  L$=C$
```

Craig Bina

```
1810  CONVERT F TO L$[4]
1820  LOCK #1,R
1830  IF R THEN 1820
1840  ADVANCE #1:32767,R
1850  IF  NOT R THEN 1840
1860  IF  END #1 THEN 1920
1870  PRINT #1:L$, END
1880  UNLOCK #1
1890  ASSIGN *,1
1900  PRINT "YOUR REQUEST HAS BEEN FILED."'10'10
1910  GOTO 190
1920  PRINT "FILE:  "B$"  IS TOO SMALL TO STORE YOUR REQUEST.  PLEASE INFORM"
1930  PRINT "YOUR INSTRUCTOR."
1940  GOTO 190
1950  SYSTEM R,"ECH-OFF"
1960  FOR B=1 TO 2
1970  PRINT '13;
1980  PRINT "CODE *> ";
1990  LINPUT B$
2000  IF B$= -code-- THEN 2040
2010  NEXT B
2020  PRINT "NO SECURITY CLEARANCE"'7'7'7'7'7'7
2030  GOTO 1550
2040  SYSTEM R,"ECH-ON"
2050  PRINT
2060  STOP
2070  SYSTEM R,"ECH-OFF"
2080  PRINT "ENTER ADMINISTRATIVE SECURITY ACCESS CODE > ";
2090  LINPUT B$
2100  IF B$= --code--- THEN 2130
2110  PRINT '13'10"NO ACCESS CLEARANCE"'7'7'7'10
2120  GOTO 1550
2130  PRINT '13'10"CLEARED FOR ACCESS"'10
2140  ASSIGN "RQUEST",1,R
2150  IF  NOT R THEN 2200
2160  PRINT "NO ACCESS.  FILE:  RQUEST  IS MISSING."
2170  GOTO 1550
2180  PRINT "NO ACCESS.  NO REQUESTS ARE CURRENTLY ON FILE."
2190  GOTO 1550
2200  LOCK #1
2210  IF  END #1 THEN 2180
2220  READ #1:B$
2230  IF B$[1,1]#"+" THEN 2220
2240  ASSIGN *,1
2250  PRINT "AFTER SYSTEM PRINTS "'34"DONE"'34" YOU MUST"
2260  PRINT "READY THE LINE PRINTER AND TYPE THE FOLLOWING:"
2270  PRINT '10"EXE*OUT=LPROUT*REMOTE"'10
2280  PRINT "IF YOU ENCOUNTER ANY DIFFICULTIES, IMMEDIATELY TYPE:"
2290  PRINT '10"BYE"'13'10'10"AND NOTIFY THE AUTHORITIES AT WHS."
2300  GOTO 2040
2310  REM CATALOG
2320  PRINT "YOU HAVE JUST SELECTED THE OPTION OF GETTING A CATALOG OF"
2330  PRINT "THE DIFFERENT FUNDAMENTALS."'10
2340  CHAIN R,"CATLOG"
2350  PRINT "SORRY, THE CATALOG IS CURRENTLY OUT OF COMMISSION."'10
2360  GOTO 140
2370  REM SPECIALS
2380  PRINT "YOU HAVE JUST SELECTED AN INSTRUCTOR'S OPTION."
2390  CONVERT A/10^(INT(LOG(A)/2.30259)-1) TO A$
2400  CONVERT A$ TO X
```

Craig Bina

```
2410   X=INT(100*(X-INT(10*X)/10))
2420   IF X THEN 2570
2430   ASSIGN "ACCESS",2,R,"CODES"
2440   IF R>2 THEN 2910
2450   LOCK #2
2460   IF  END #2 THEN 2930
2470   READ #2;A$
2480   IF A$[1,2]#Q$ THEN 2470
2490   ASSIGN *,2
2500   PRINT '10"PLEASE ENTER YOUR ACCESS CODE.";
2510   SYSTEM R,"ECH-OFF"
2520   LINPUT B$
2530   SYSTEM R,"ECH-ON"
2540   PRINT
2550   B$=UPS$(B$)
2560   IF B$=A$[3] THEN 2590
2570   PRINT "YOU HAVE NOT BEEN AUTHORIZED TO USE THE INSTRUCTOR-LEVEL SYSTEM."
2580   GOTO 190
2590   GOTO F-99 OF 2630,2650,2670,2690
2600   GOTO F-103 OF 2710,2730,2750,2770,2790,2810,2830
2610   PRINT "NO SUCH OPTION CURRENTLY EXISTS."
2620   GOTO 140
2630   CHAIN R,"MASFND"
2640   GOTO 2840
2650   CHAIN R,"LDINTL"
2660   GOTO 2840
2670   CHAIN R,"OUTPUT"
2680   GOTO 2840
2690   CHAIN R,"REMARK"
2700   GOTO 2840
2710   CHAIN R,"ERASE"
2720   GOTO 2840
2730   CHAIN R,"RNDFND"
2740   GOTO 2840
2750   CHAIN R,"LGLIZE"
2760   GOTO 2840
2770   CHAIN R,"REPORT"
2780   GOTO 2840
2790   CHAIN R,"INSERT"
2800   GOTO 2840
2810   CHAIN R,"MASLPR"
2820   GOTO 2840
2830   CHAIN R,"CHGACC"
2840   PRINT "I AM SORRY, THE REQUESTED PROGRAM, #";
2850   CONVERT F TO B$
2860   PRINT B$", IS MISSING."
2870   PRINT "PLEASE INFORM THE APPROPRIATE AUTHORITIES."
2880   GOTO 140
2890   PRINT "SORRY, THE INITIALIZATION PROGRAM IS CURRENTLY OUT OF COMMISSION,"
2900   GOTO 2870
2910   PRINT "THE AUTHORITIES HAVE NOT YET ENABLED THE INSTRUCTORS' OPTIONS."
2920   GOTO 190
2930   PRINT "TEACHER/SCHOOL "Q$" DOES NOT HAVE INSTRUCTORS' OPTIONS CAPABILITY."
2940   GOTO 190
2950   REM  ERROR TRAP ROUTINE
2960   FOR I=0 TO 2
2970   E[I+1]=SYS(I)
2980   NEXT I
2990   CHAIN R,"ERROR"
3000   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
```

Craig Bina

```
3010  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
3020  GOTO 1630
3030  END
```

Craig Bina

3.03

SETUP

```
10   COM A,A0$[80],B,C,D,00$[11],E[3]
20   O=BRK(0)
30   00$="SETUP"
40   IF  ERROR  THEN 950
50   FILES *
60   A=0
70   PRINT
80   DIM A$[80],B$[6],C$[10],E$[80]
90   B$="INTL"
100   PRINT "WHAT IS YOUR LAST INITIAL";
110   INPUT A$
120   IF LEN(A$)#1 OR NUM(A$)>90 OR NUM(A$)<65 THEN 100
130   B$[5]=A$
140   E=NUM(A$)-64
150   PRINT "WHAT IS YOUR SCHOOL'S INITIAL";
160   INPUT A$
170   IF LEN(A$)#1 OR NUM(A$)>90 OR NUM(A$)<65 THEN 150
180   B$[6]=A$
190   P=NUM(A$)/10000
200   GOSUB 760
210   ASSIGN B$,1,R
220   B$[1,4]="RECR"
230   GOSUB 760
240   B$[1,4]="RNDM"
250   GOSUB 760
260   PRINT "PLEASE ENTER YOUR THREE INITIALS"
270   PRINT "->";
280   LINPUT C$
290   IF LEN(C$)#3 THEN 260
300   IF C$="***" THEN 260
310   PRINT "PLEASE ENTER YOUR FIRST NAME"
320   PRINT "==>";
330   LINPUT A$
340   IF POS(A$," ") THEN 320
350   IF  NOT LEN(A$) THEN 310
360   Y1=LEN(A$)+1
370   A$[Y1]=" "
380   PRINT "PLEASE ENTER YOUR LAST NAME"
390   PRINT "===>";
400   LINPUT A$[Y1+1]
410   IF LEN(A$)=Y1 THEN 380
420   IF POS(A$[Y1+1]," ") THEN 380
430   A0$=A$
440   PRINT #1;C$,P*10^E,A$, END
450   ASSIGN "USERS",1,R,"*"
460   IF R THEN 920
470   LOCK #1
480   ADVANCE #1;32767,R
490   IF  NOT R THEN 480
500   A$[3]=A$
510   A$[1,2]=B$[5]
520   IF  END #1 THEN 920
530   PRINT #1;A$, END
540   ASSIGN *,1,R
550   ASSIGN "ACCESS",1,R,"CODES"
560   IF R THEN 870
570   A$=B$[5]
580   PRINT "PLEASE ENTER A SEQUENCE OF FROM ONE TO TEN LETTERS."
590   PRINT "THIS WILL BECOME YOUR INSTRUCTORS' OPTIONS ACCESS CODE."'13'10"> ";
```

Craig Bina

```
600   SYSTEM R,"ECH-OFF"
610   LINPUT A$[3]
620   A$=UPS$(A$)
630   SYSTEM R,"ECH-ON"
640   PRINT '13;
650   LOCK #1
660   ADVANCE #1;32767,R
670   IF  NOT R THEN 660
680   IF  END #1 THEN 900
690   PRINT #1:A$, END
700   ASSIGN *,1,R
710   PRINT "SET-UP HAS BEEN COMPLETED."
720   CHAIN R,"MAIN"
730   PRINT "THE PROGRAM: MAIN  IS MISSING, PLEASE INFORM THE APPROPRIATE AUTHORITIES."
740   SYSTEM R,"BYE"
750   GOTO 740
760   REM  FILE CREATE ROUTINE
770   CREATE R,B$,1
780   IF  NOT R THEN 810
790   PRINT "UNABLE TO CREATE ONE RECORD FILE:  "B$
800   GOTO 860
810   C$="MWA-"
820   C$[5]=B$
830   SYSTEM R,C$
840   C$[1,3]="PRI"
850   SYSTEM R,C$
860   RETURN
870   PRINT "DUE TO ABSENCE OF THE FILE:  ACCESS  I AM UNABLE TO ASSIGN YOU AN"
880   PRINT "INSTRUCTORS' OPTIONS ACCESS CODE.  PLEASE INFORM THE AUTHORITIES."
890   GOTO 700
900   PRINT "DUE TO SIZE OF THE FILE:  ACCESS  I AM UNABLE TO RECORD YOUR"
910   GOTO 880
920   PRINT "I AM UNABLE TO RECORD YOUR NAME IN THE FILE:  USERS:"
930   PRINT "PLEASE INFORM THE AUTHORITIES."
940   GOTO 540
950   REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
960   FOR I=0 TO 2
970   E[I+1]=SYS(I)
980   NEXT I
990   CHAIN R,"ERROR"
1000   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
1010   PRINT "PLEASE INFORM YOUR INSTRUCTOR."
1020   GOTO 720
1030   END
```

Craig Bina

55

3.04

```
CATLOG

10   COM A6,N1$[80],F9,X6,Y6,O0$[11],E[3]
20   O0$="CATLOG"
30   IF  ERROR  THEN 770
40   Q=BRK(0)
50   DIM V$[80],N0$[255]
60   Q=A6/10^(INT(LOG(A6)/2.30259)-1)
70   CONVERT Q TO V$
80   CONVERT V$ TO Q
90   Q=INT(100*(Q-INT(10*Q)/10))
100   N0$="N"
110   IF Q THEN 140
120   PRINT "WOULD YOU LIKE A LISTING OF ONLY THE INSTRUCTORS' OPTIONS";
130   INPUT N0$
140   PRINT LIN(2)
150   PRINT "         CATALOG OF INSTRUCTIONAL PROGRAMS"
160   PRINT LIN(2)
170   PRINT "FUND. NO.                         TITLE"
180   IF N0$[1,1]="Y" THEN 560
190   REM         GENERAL OPTIONS
200   PRINT '10"     0                    CATALOG OF FUNDAMENTALS"
210   PRINT '10"CHEMISTRY PROGRAMS"'10
220   PRINT "    1                  ARITHMETIC "
230   PRINT "    2                  SIGNIFICANT FIGURES"
240   PRINT "    3                  GRAPHING"
250   PRINT "    4                  SCIENTIFIC NOTATION"
260   PRINT "    5                  DIATOMIC ELEMENTS"
270   PRINT "    6                  WEIGHTS AND MOLES"
280   PRINT "    7                  ANALYZING PROBLEMS"
290   PRINT "    8                  RATIO AND PROPORTION"
300   PRINT "    9                  WRITING IONIC FORMULAS"
310   PRINT "   10                  WRITING EQUATIONS"
320   PRINT "   11                  BALANCING EQUATIONS"
330   PRINT "   12                  CHANGING PARTNERS"
340   PRINT "   13                  CROSSING THE BRIDGE"
350   PRINT "   14                  MOLAR VOLUME"
360   PRINT "   15                  CALORIES"
370   PRINT "   16                  SOLUBILITY"
380   PRINT "   17                  DISSOLVING EQUATIONS FOR IONIC COMPOUNDS"
390   PRINT "   18                  NET IONIC EQUATIONS"
400   PRINT "   19                  PREPARING SOLUTIONS"
410   PRINT "   20                  PERIODIC TABLE"
420   PRINT "   21                  PROTONS, NEUTRONS, AND ELECTRONS"
430   PRINT "   22                  FORMULAS FROM THE PERIODIC TABLE"
440   PRINT "   23                  PROPERTIES OF COMPOUNDS"
450   PRINT "   24                  IONS FROM ATOMS"
460   PRINT "   25                  TYPES OF CHEMICAL REACTIONS"
470   PRINT '10"MATH PROGRAMS"'10
480   PRINT "   50                  ONE-DIGIT MULTIPLICATION"
490   PRINT "   51                  TWO DIGIT BY ONE DIGIT MULTIPLICATION"
500   PRINT "   52                  TWO DIGIT BY TWO DIGIT MULTIPLICATION"
510   PRINT "   53                  MULTIPLYING WITH DECIMALS"
520   PRINT "   54                  DIVISION WITH SINGLE DIGIT DIVISOR"
530   PRINT "   55                  DIVISION WITH TWO DIGIT DIVISOR"
540   PRINT "   56                  DIVISION WITH THREE DIGIT DIVISORS (TOUGH ONES)"
550   PRINT "   57                  DIVIDING WITH DECIMALS"
560   REM      INSTRUCTORS' OPTIONS
570   IF Q THEN 720
580   PRINT '10"INSTRUCTORS' OPTIONS"'10
590   PRINT "*100                   SEQUENTIAL PRACTICE SHEET GENERATION"
```

Craig Bina

56

```
600  PRINT "*101                 CLASS LIST CREATION AND ALTERATION"
610  PRINT "*102                 FUNDAMENTAL PRACTICE GRADEBOOK"
620  PRINT "*103                 INSTRUCTOR MESSAGES AND COMMENTS"
630  PRINT "*104                 ERASURE OF UNUSED FILE INFORMATION"
640  PRINT "*105                 NON-SEQUENTIAL PRACTICE SHEET GENERATION"
650  PRINT "*106                 LEGALIZATION OF STUDENT CHEATERS"
660  PRINT "*107                 INSTRUCTIONAL PACKAGE USAGE REPORT"
670  PRINT "*108                 FILING OF HAND-SCORED PRACTICE SHEETS"
680  PRINT "*109                 REQUEST FOR GENERATION OF SEQUENTIAL PRACTICE"
690  PRINT "                         SHEETS BY COMPUTER SERVICES"
700  PRINT "*110                 INSTRUCTORS' OPTIONS ACCESS CODE ALTERATION"
710  PRINT '10"     * DENOTES INSTRUCTORS-ONLY FUNDAMENTALS"
720  PRINT LIN(2)
730  CHAIN R,"MAIN"
740  PRINT "THE PROGRAM: MAIN  IS MISSING, PLEASE INFORM YOUR INSTRUCTOR."
750  SYSTEM R,"BYE"
760  GOTO 750
770  REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
780  FOR I=0 TO 2
790  E[I+1]=SYS(I)
800  NEXT I
810  CHAIN R,"ERROR"
820  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
830  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
840  GOTO 730
850  END
```

Craig Bina

```
3.05
FUND6

10   COM A6,N0$[80],F9,D1,D2,00$[11],E[3]
20   COM C6
30   Y=BRK(0)
40   00$="FUND6"
50   IF  ERROR  THEN 3400
60   C6=0
70   FILES *,*,*,*,*
80   F4=1
90   F5=5
100   Q9=5.21
110   DATA "PRDTBL","CVCMPD","IONSFL"
120   DIM B$[80],C$[72],A[5],D$[72],E$[80],F$[80]
130   DIM R$[80],S$[72]
140   DIM B0$[72],C0$[72],P[10]
150   DIM A$[80],V$[255]
160   DIM W$[255],X$[255]
170   DIM C1$[5]
180   DIM A1$[5]
190   W$=",2,3,4,5,6,11,13,15,17,18,19,20,21,24,25,26,27,34,35,37,40,"
200   W$[LEN(W$)+1]="43,44,46,48,51,52,"
210   X$=",3,4,5,6,8,10,11,12,13,14,28,29,30,31,32,34,49,53,54,55,56,61,"
220   X$[LEN(X$)+1]="62,63,64,65,"
230   FOR I=2 TO 4
240   READ B$
250   ASSIGN B$,I,R
260   IF R>2 THEN 290
270   NEXT I
280   GOTO 310
290   PRINT "THERE IS NO DATA SOURCE FILE, "B$" , FOR FUNDAMENTAL #6."
300   GOTO 2670
310   PRINT "FUNDAMENTAL #6:  WEIGHTS AND MOLES"LIN(1)
320   GOSUB 1580
330   PRINT "1)   WHAT IS THE MOLAR MASS OF ";
340   IF INT(RND(1)+.5) THEN 370
350   PRINT B$" TO THE NEAREST GRAM?"
360   GOTO 380
370   PRINT B$" TO THE NEAREST GRAM?"
380   A[1]=W
390   PRINT
400   PRINT LIN(1)"CALCULATE THE ANSWERS TO #2-5 TO 2 SIGNIFICANT FIGURES."LIN(1)
410   D$="2"
420   E$="MOLES"
430   F$="GRAMS"
440   GOSUB 1580
450   GOSUB 810
460   N=N/W
470   GOSUB 2150
480   A[2]=N
490   D$=F$
500   E$=F$
510   F$=D$
520   D$="3"
530   GOSUB 1580
540   GOSUB 810
550   N=N*W
560   GOSUB 2150
570   A[3]=N
580   V=INT(RND(1)+.5)
590   E$="MOLES"
```

Craig Bina

58

```
600   F$="GRAMS"
610   D$="4"
620   GOSUB V+1 OF 1070,1640
630   GOSUB 810
640   N=N/W
650   GOSUB 2150
660   A[4]=N
670   D$=E$
680   E$=F$
690   F$=D$
700   D$="5"
710   GOSUB V+1 OF 1640,1070
720   GOSUB 810
730   N=N*W
740   GOSUB 2150
750   A[5]=N
760   GOTO 2250
770   PRINT "ERROR: POSITIVE ION FOUND INSTEAD OF ANION."
780   GOTO 980
790   PRINT "NEGATIVE ION FOUND INSTEAD OF CATION."
800   GOTO 980
810   REM OUTPUT ROUTINE FOR QUESTIONS 2, 3, 4, AND 5
820   S=INT(RND(1)+.5)
830   PRINT D$")   HOW MANY "E$" OF ";
840   D1$=D$
850   IF D$="4" THEN 900
860   IF D$="5" THEN 900
870   IF S THEN 900
880   PRINT B$;
890   GOTO 910
900   PRINT C$;
910   PRINT " ARE IN ";
920   N=100*INT(6*RND(1))
930   N=N+10*INT(9*RND(1)+1)+(N=0)*(9*RND(1)+1)
940   GOSUB 2150
950   CONVERT N TO D$
960   PRINT D$" "F$" OF ";
970   IF D1$="4" THEN 1000
980   IF D1$="5" THEN 1000
990   IF S THEN 1020
1000   PRINT C$" ?"
1010   GOTO 1040
1020   PRINT B$" ?"
1030   GOTO 1040
1040   PRINT
1050   C6=C6+((35+LEN(B$)+LEN(C$)+LEN(D$)+LEN(E$)+LEN(F$))>72)
1060   RETURN
1070   REM IONIC CMPD
1080   C1=INT(66*RND(1)+1)
1090   C1$=","
1100   CONVERT C1 TO C1$[2]
1110   C1$[LEN(C1$)+1]=","
1120   IF POS(X$,C1$) THEN 1080
1130   A1=INT(53*RND(1)+1)
1140   A1$=","
1150   CONVERT A1 TO A1$[2]
1160   A1$[LEN(A1$)+1]=","
1170   IF POS(W$,A1$) THEN 1130
1180   READ #4,1
1190   ADVANCE #4;3*(A1-1),R
```

Craig Bina

```
1200  READ #4;B0$,C0$,C0
1210  IF C0 >= 0 THEN 770
1220  C0=-C0
1230  ADVANCE #4;3*(53-A1+C1-1),R
1240  READ #4;B$,C$,C
1250  IF C <= 0 THEN 790
1260  B$[LEN(B$)+1]=" "
1270  B$[LEN(B$)+1]=B0$
1280  IF C=C0 THEN 1350
1290  IF C=1 OR C0=1 THEN 1360
1300  T=(C MAX C0)/(C MIN C0)
1310  IF T#INT(T) THEN 1360
1320  C=C/T
1330  C0=C0/T
1340  GOTO 1360
1350  C=C0=1
1360  IF C0=1 THEN 1480
1370  GOSUB 1390
1380  GOTO 1480
1390  REM SUBSCRIPT ROUTINE W/PARENS.
1400  IF LEN(C$)>2 THEN 1430
1410  IF LEN(C$)=1 THEN 1460
1420  IF NUM(C$[2])>91 THEN 1460
1430  C$[2]=C$
1440  C$[1,1]="("
1450  C$[LEN(C$)+1]=")"
1460  CONVERT C0 TO C$[LEN(C$)+1]
1470  RETURN
1480  IF C=1 THEN 1550
1490  B0$=C$
1500  C$=C0$
1510  C0=C
1520  GOSUB 1390
1530  C0$=C$
1540  C$=B0$
1550  C$[LEN(C$)+1]=C0$
1560  GOSUB 1710
1570  RETURN
1580  REM ELEMENT
1590  X=INT(103*RND(1)+1)
1600  READ #2,1
1610  ADVANCE #2;(X-1)*3,R
1620  READ #2;B$,C$,W
1630  RETURN
1640  REM COVALENT COMPOUND
1650  X=INT(106*RND(1)+1)
1660  READ #3,1
1670  ADVANCE #3;2*(X-1),R
1680  READ #3;B$,C$
1690  GOSUB 1710
1700  RETURN
1710  REM MOLAR-MASS CALCULATOR
1720  REM INPUT C$=FORMULA; OUTPUT W=MOLAR MASS
1730  MAT P=ZER
1740  F=1
1750  FOR I=1 TO LEN(C$)
1760  J=K=I
1770  M=1
1780  IF I=LEN(C$) THEN 1970
1790  IF C$[I,I]#"(" THEN 1830
```

Craig Bina

```
1800  F=F+1
1810  IF F>10 THEN 2090
1820  GOTO 2020
1830  IF C$[I,I]#")" THEN 1920
1840  IF I=LEN(C$) THEN 2050
1850  CONVERT C$[I+1,I+1] TO W,2050
1860  F=F-1
1870  IF F=0 THEN 2070
1880  P[F]=P[F]+W*P[F+1]
1890  P[F+1]=0
1900  I=I+1
1910  GOTO 2020
1920  IF NUM(C$[I+1])<92 THEN 1950
1930  K=I=I+1
1940  IF I=LEN(C$) THEN 1970
1950  CONVERT C$[I+1,I+1] TO M,1970
1960  I=I+1
1970  READ #2,1
1980  IF  END #2 THEN 2110
1990  READ #2:R$,S$,T0
2000  IF S$#C$[J,K] THEN 1990
2010  P[F]=P[F]+M*T0
2020  NEXT I
2030  W=P[1]
2040  RETURN
2050  PRINT "NO SUBSCRIPT AFTER RIGHT PARENTHESIS: "C$
2060  GOTO 1850
2070  PRINT "EXTRA RIGHT PARENTHESIS: "C$
2080  GOTO 1880
2090  PRINT "MORE THAN TEN LEFT PARENTHESES/LEVELS NOT ALLOWED: "C$
2100  GOTO 1820
2110  PRINT "NON-EXISTENT ELEMENT, "C$[J,K]", REFERENCED: "C$
2120  REM CAUSE A 9 ERROR
2130  X=79
2140  REM  END-OF-SUBPROGRAM
2150  REM 2-SIG FIG ROUNDING ROUTINE
2160  REM SUBR.  (IN N) USE E,L,T (OUT N)
2170  T=SGN(N)
2180  IF  NOT T THEN 2230
2190  N=ABS(N)
2200  E=INT(LOG(N)/2.30259)
2210  N=N/10^E
2220  N=INT(10*N+.5)*10^(E-1)
2230  N=N*T
2240  RETURN
2250  REM STANDARD ENDING ROUTINE FOR FUNDAMENTAL PROGRAMS
2260  FOR X8=1 TO 10
2270  ASSIGN "MAIN6",F4,R, WR
2280  IF R<6 THEN 2320
2290  ASSIGN *,F4
2300  SYSTEM R,"PAU-6"
2310  GOTO 2270
2320  IF R=0 THEN 2370
2330  SYSTEM R,"PAU-2"
2340  NEXT X8
2350  PRINT "THERE IS NO DATA STORAGE FILE:  MAIN6  FOR FUNDAMENTAL #6."
2360  GOTO 2660
2370  LOCK #F4
2380  SYSTEM R,"PAU-2"
2390  LOCK #F4,R
```

Craig Bina

19

```
2400  IF R=0 THEN 2290
2410  IF  END #F4 THEN 2440
2430  REM STANDARD ENDING ROUTINE FOR FUNDAMENTAL PROGRAMS
2440  IF  END #F4 THEN 2470
2450  READ #F4,1:Q6
2460  IF Q6=Q9 THEN 2480
2470  PRINT #F4,1;Q9, END
2480  S6=0
2490  Q9=INT(ABS(Q9))
2500  IF  END #F4 THEN 2560
2510  READ #F4:I6,J6
2520  IF I6#A6 THEN 2540
2530  S6=S6 MAX INT(J6)
2540  ADVANCE #F4;Q9,R
2550  GOTO 2510
2560  PRINT "THIS IS SHEET #"S6+1"FOR "N0$"."LIN(1)
2570  X7=REC(F4)
2580  Y7=ITM(F4)
2590  IF  END #F4 THEN 2720
2600  PRINT #F4;A6,S6+1
2610  MAT  PRINT #F4;A
2620  IF  END #F4 THEN 2640
2630  PRINT #F4; END
2640  CHAIN R,"NOTICE"
2650  PRINT "THE PROGRAM: NOTICE  IS MISSING,"
2660  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
2670  CHAIN R,"MAIN"
2680  PRINT "THE PROGRAM: MAIN  IS MISSING,"
2690  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
2700  SYSTEM R,"BYE"
2710  GOTO 2700
2720  REM ERASE THE MISTAKES
2730  READ #F4,X7
2740  ADVANCE #F4;Y7,R
2750  IF  END #F4 THEN 2770
2760  PRINT #F4; END
2770  REM  ROUTINE FOR MAIN# RUBBER-BAND FILES
2780  X7=REC(F4)
2790  A$="DUMY"
2800  CONVERT F9 TO A$[5]
2810  CREATE R,A$,X7+1
2820  IF R THEN 3370
2830  B$="BEER"
2840  CONVERT F9 TO B$[5]
2850  CREATE R,B$,X7
2860  IF R THEN 3350
2870  READ #F4,1
2880  ASSIGN B$,F5,R
2890  GOTO TYP(F4)-1 OF 2930,2960
2900  READ #F4:Y7
2910  PRINT #F5:Y7
2920  GOTO 2890
2930  READ #F4:V$
2940  PRINT #F5;V$
2950  GOTO 2890
2960  REM  PURGE--RE-CREATE
2970  IF  END #F5 THEN 2990
2980  PRINT #F5; END
2990  C$="MAIN"
3000  CONVERT F9 TO C$[5]
```

Craig Bina

```
3010  ASSIGN *,F4
3020  PURGE R,C$
3030  IF R=0 THEN 3080
3040  ASSIGN C$,F4,R
3050  UNLOCK #F4
3060  SYSTEM R,"PAU-1"
3070  GOTO 3010
3080  PURGE R,A$
3090  CREATE R,C$,X7+1
3100  ASSIGN C$,F4,R
3110  C$[5]=C$
3120  C$[1,4]="MWA-"
3130  SYSTEM R,C$
3140  C$[1,3]="PRI"
3150  SYSTEM R,C$
3160  ASSIGN C$[5],F4,R, WR
3170  IF R=0 THEN 3220
3180  ASSIGN C$[5],F4,R
3190  UNLOCK #F4
3200  SYSTEM R,"PAU-1"
3210  GOTO 3160
3220  READ #F5,1
3230  IF  END #F4 THEN 2770
3240  GOTO TYP(F5)-1 OF 3280,3310
3250  READ #F5;Y7
3260  PRINT #F4;Y7
3270  GOTO 3240
3280  READ #F5:V$
3290  PRINT #F4:V$
3300  GOTO 3240
3310  ASSIGN *,F5
3320  PRINT #F4; END
3330  PURGE R,B$
3340  GOTO 2570
3350  PURGE R,A$
3370  PRINT "THE MAIN-FILE FOR FUNDAMENTAL"F9"IS FULL AND THERE IS INSUFFICIENT"
3380  PRINT "MEMORY SPACE TO EXPAND IT.  PLEASE INFORM YOUR INSTRUCTOR OF THIS."
3390  GOTO 2670
3400  REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
3410  FOR I=0 TO 2
3420  E[I+1]=SYS(I)
3430  NEXT I
3440  CHAIN R,"ERROR"
3450  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
3460  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
3470  GOTO 2670
3480  END
```

Craig Bina

```
3.06
SOINIT

10   COM I6,N0$[80],F9,X6,Y6,00$[11],E[3]
20   REM**************************************   00$="FUND##"
30   O=BRK(0)
40   IF  ERROR  THEN 9070
50   FILES *,*
60   REM**************************************  FILES *,ETC. FOR DATA FILES ONLY
70   DIM M0$[10],D0$[6],B0$[6],V0$[255]
80   REM**************************************  DIM ADDITIONAL VARIABLES
90   F4=1
100   F5=2
110   REM*************************************  Q9= # OF QUESTIONS W/CODING
120   M0$="MAIN"
130   D0$="DUMY"
140   B0$="BFFR"
150   CONVERT F9 TO M0$[5]
160   D0$[5]=M0$[5]
170   B0$[5]=M0$[5]
180   REM*************************  ASSIGN DATA FILES FROM #3 UNWARD
190   REM*************************  PRINT "FUNDAMENTAL ##:  FUNDAMENTAL NAME"LIN(1)
```

Craig Bina

3.07

SOTERM

```
8000  REM  SOTERM  STANDARD ENDING ROUTINE FOR FUNDAMENTAL PROGRAMS
8010  FOR X8=1 TO 10
8020  ASSIGN M0$,F4,R, WR
8030  IF R<6 THEN 8070
8040  ASSIGN *,F4
8050  SYSTEM R,"PAU-6"
8060  GOTO 8020
8070  IF R=0 THEN 8120
8080  SYSTEM R,"PAU-2"
8090  NEXT X8
8100  PRINT "THERE IS NO DATA STORAGE FILE:  "M0$"  FOR FUNDAMENTAL #"F9"."
8110  GOTO 8420
8120  LOCK #F4
8130  SYSTEM R,"PAU-2"
8140  LOCK #F4,R
8150  IF R=0 THEN 8040
8160  IF  END #F4 THEN 8190
8170  READ #F4,1;Q8
8180  IF Q8=Q9 THEN 8200
8190  PRINT #F4,1;Q9, END
8200  S9=0
8210  Q8=Q9 >= 0
8220  Q9=INT(ABS(Q9))
8230  IF  END #F4 THEN 8340
8240  IF Q8 THEN 8260
8250  READ #F4,1
8260  IF Q8 THEN 8290
8270  READ #F4;Q9
8280  Q9=INT(ABS(Q9))
8290  READ #F4;I9,J9
8300  IF I9#I6 THEN 8320
8310  S9=S9 MAX INT(J9)
8320  ADVANCE #F4;Q9,R
8330  GOTO 8260
8340  PRINT "THIS IS SHEET #"S9+1"FOR "N0$"."LIN(1)
8350  X9=REC(F4)
8360  Y9=ITM(F4)
8370  IF  END #F4 THEN 8480
8380  PRINT #F4;I6,S9+1
8390  REM*******************************PRINT #F4;"ANSWERS",END
8400  CHAIN R,"NOTICE"
8410  PRINT "THE PROGRAM:  NOTICE  IS MISSING."
8420  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
8430  CHAIN R,"MAIN"
8440  PRINT "THE PROGRAM:  MAIN  IS MISSING."
8450  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
8460  SYSTEM R,"BYE"
8470  GOTO 8460
8480  X8=REC(F4)
8490  READ #F4,X9
8500  ADVANCE #F4;Y9,R
8510  PRINT #F4; END
8520  CREATE R,D0$,X8+1
8530  IF R>1 THEN 8900
8540  IF R=0 THEN 8570
8550  PURGE R,D0$
8560  GOTO 8520
8570  CREATE R,B0$,X8
8580  IF R>1 THEN 8890
```

Craig Bina

```
8590  IF R=0 THEN 8620
8600  PURGE R,B0$
8610  GOTO 8570
8620  ASSIGN B0$,F5,R
8630  F8=F4
8640  T8=F5
8650  GOSUB 8960
8660  PURGE R,M0$
8670  IF R=0 THEN 8710
8680  GOSUB 8920
8690  ASSIGN *,F4
8700  GOTO 8660
8710  PURGE R,D0$
8720  CREATE R,M0$,X8+1
8730  ASSIGN M0$,F4,R
8740  M0$[5]=M0$
8750  M0$[1,4]="PRI-"
8760  SYSTEM R,M0$
8770  M0$[1,3]="MNA"
8780  SYSTEM R,M0$
8790  M0$=M0$[5]
8800  ASSIGN M0$,F4,R, WR
8810  IF R=0 THEN 8840
8820  GOSUB 8920
8830  GOTO 8800
8840  F8=F5
8850  T8=F4
8860  GOSUB 8960
8870  PURGE R,B0$
8880  GOTO 8350
8890  PURGE R,D0$
8900  PRINT "THE FILE:  "M0$"  IS FULL AND THERE IS INSUFFICIENT SPACE TO EXPAND IT."
8910  GOTO 8420
8920  ASSIGN M0$,F4,R
8930  UNLOCK #F4
8940  SYSTEM R,"PAU-1"
8950  RETURN
8960  READ #F8,1
8970  GOTO TYP(F8)-1 OF 9010,9040
8980  READ #F8:Y8
8990  PRINT #T8;Y8
9000  GOTO 8970
9010  READ #F8:V0$
9020  PRINT #T8;V0$
9030  GOTO 8970
9040  PRINT #T8; END
9050  ASSIGN *,F8
9060  RETURN
9070  REM  S0ERR  STANDARD ERROR TRAP ROUTINE
9080  FOR I=0 TO 2
9090  E[I+1]=SYS(I)
9100  NEXT I
9110  CHAIN R,"ERROR"
9120  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
9130  GOTO 8420
9140  END
```

Craig Bina

3.08

```
NOTICE

10   COM A,A$[80],B,C,D,C$[11],E[3]
20   COM Q8
30   DIM E1$[80]
40   C$="NOTICE"
50   IF  ERROR  THEN 400
60   REM K IS THE CHAIN-DESTINATION-LINE IN  MASFND .
70   K=740
80   REM K2 IS THE CHAIN-DESTINATION-LINE IN  RNDFND .
90   K2=800
100  REM K3 IS THE CHAIN-DESTINATION-LINE IN  REMOTE .
110  K3=810
120  PRINT TAB(25);
130  L=INT(LOG(A)/2.30259)
140  Q=A/10^(L-1)
150  CONVERT Q TO E1$
160  CONVERT E1$ TO Q
170  L=L+3* NOT INT(100*(Q-INT(10*Q)/10))
180  Q=INT(A/10^L)
190  PRINT "PERIOD"Q;LIN(1)
200  IF C=200 THEN 290
210  IF C=300 THEN 320
220  IF C=400 THEN 370
230  PRINT "IF ANOTHER STUDENT WISHES TO USE THIS TERMINAL, PLEASE LEAVE IT AS IS"
240  PRINT "AND WORK THESE PROBLEMS ELSEWHERE.  RETURN LATER TO CHECK YOUR ANSWERS."
250  CHAIN R,"MAIN"
260  PRINT "THE PROGRAM: MAIN  IS MISSING, PLEASE INFORM YOUR INSTRUCTOR."
270  SYSTEM R,"BYE"
280  GOTO 270
290  CHAIN R,"REMOTE",K3
300  PRINT CTL(1)"PROGRAM:  REMOTE  IS MISSING.  JOB TERMINATED PREMATURELY."
310  GOTO 480
320  CHAIN R,"MASFND",K
330  PRINT "PROGRAM:  MASFND  IS MISSING, PLEASE INFORM YOUR INSTRUCTOR."
340  PRINT LIN(2)
350  A=0
360  GOTO 250
370  CHAIN R,"RNDFND",K2
380  PRINT "PROGRAM: RNDFND  IS MISSING, PLEASE INFORM YOUR INSTRUCTOR."
390  GOTO 340
400  FOR I=0 TO 2
410  E[I+1]=SYS(I)
420  NEXT I
430  CHAIN R,"ERROR"
440  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
450  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
460  CHAIN R,"MAIN"
470  PRINT "THE PROGRAM: MAIN  IS ALSO MISSING, PLEASE INFORM YOUR INSTRUCTOR OF THIS."
480  SYSTEM R,"BYE"
490  GOTO 480
500  END
```

Craig Bina

3.09

CHECK

```
10    COM I6,NO$[80],F9,X7,Y7,OO$[11],E[3]
20    OO$="CHECK"
30    IF  ERROR  THEN 2880
40    O=BRK(0)
50    FILES *,*,*
60    DIM A$[255],F$[6],U$[255],V$[255],W$[255],X$[15],Y$[15]
70    DIM N[5]
80    IMAGE #,""
90    IMAGE /""
100    IMAGE #,"RESPONSE TO QUESTION #",3DX,"-> "
110    IMAGE /"YOU HAD ",3DX,"WRONG OUT OF ",3DX,"PROBLEMS."
120    IMAGE "THIS IS A SCORE OF ",3D.D,"% RIGHT."
130    IMAGE "I AM SORRY, BUT THE REQUESTED ANSWERS ARE NOT ON FILE."
140    IMAGE "ILLEGAL FORMAT"/
150    IMAGE "CORRECT"
160    IMAGE #,"THE CORRECT ANSWER IS:  "
170    IMAGE "THE FILE:  ",6A2X,"IS MISSING."
180    IMAGE "I AM SORRY, BUT THE REQUESTED ANSWERS ARE NO LONGER AVAILABLE."
190    IMAGE "PREMATURE TERMINATION OF CHECKING PROCEDURE"
200    IMAGE "I AM SORRY, BUT I HAVE INADVERTENTLY RECORDED THAT YOU HAVE ATTEMPTED TO CHEAT"
210    IMAGE "AND AM CURRENTLY UNABLE TO ERASE THIS INCORRECT INFORMATION."
220    IMAGE "THE PROGRAM:  UPDATE  IS MISSING."
230    IMAGE "THE PROGRAM:  MAIN  IS MISSING."
240    IMAGE "PLEASE INFORM YOUR INSTRUCTOR."
250    G=X=0
260    F$="MAIN"
270    CONVERT F9 TO F$[5]
280    CONVERT I6 TO X$
290    PRINT "NOW, "NO$[1,POS(NO$," ")-1]", WHAT IS YOUR SHEET NUMBER";
300    INPUT S
310    IF S<1 OR S#INT(S) THEN 290
320    GOSUB 870
330    IF R THEN 820
340    GOSUB 1060
350    IF Q<0 THEN 820
360    GOSUB 1220
370    ASSIGN *,1
380    IF Q >= 0 THEN 410
390    PRINT  USING 130
400    GOTO 820
410    B=INT(LOG(I6)/2.30259)
420    B=I6/10^(B-1)
430    B= NOT INT(100*(B-INT(10*B)/10))
440    IF B THEN 480
450    PRINT  USING 80
460    PRINT  USING 90
470    GOSUB 1510
480    GOSUB 870
490    GOSUB 1220
500    GOSUB 2350
510    FOR I=1 TO Q
520    GOSUB 870
530    IF  NOT R THEN 580
540    PRINT  USING 190
550    IF B THEN 570
560    GOSUB 2420
570    GOTO 820
580    GOSUB 1220
590    IF Q >= 0 THEN 620
```

Craig Bina

```
600 PRINT  USING 180
610 GOTO 540
620 GOSUB 1680
630 GOTO 650
640 PRINT  USING 140
650 PRINT  USING 100;I
660 LINPUT U$
670 U$=UPS$(U$)
680 IF  NOT LEN(U$) THEN 640
690 IF T THEN 710
700 CONVERT U$ TO Q,640
710 GOSUB T+1 OF 1780,2190
720 G=G+C
730 NEXT I
740 Y7=G/Q*100
750 PRINT  USING 110;Q-G,Q
760 PRINT  USING 120;Y7
770 IF B THEN 790
780 GOSUB 2420
790 CHAIN R,"UPDATE"
800 PRINT  USING 220
810 PRINT  USING 240
820 CHAIN R,"MAIN"
830 PRINT  USING 230
840 PRINT  USING 240
850 SYSTEM R,"BYE"
860 GOTO 850
870 REM  LOCKS ROUTINE:  IN (F$) OUT (R)
880 FOR J=1 TO 10
890 ASSIGN F$,1,R, WR
900 IF R<6 THEN 940
910 ASSIGN *,1
920 SYSTEM R,"PAU-6"
930 GOTO 890
940 IF R=0 THEN 1010
950 SYSTEM R9,"PAU-2"
960 NEXT J
970 ASSIGN *,1
980 PRINT  USING 170;F$
990 PRINT  USING 240
1000 RETURN
1010 LOCK #1
1020 SYSTEM R,"PAU-2"
1030 LOCK #1,R9
1040 IF R9=0 THEN 910
1050 RETURN
1060 REM  ANSWER TYPES ROUTINE:  OUT (Q,Q0,N1,MAT N[2])
1070 IF  END #1 THEN 1190
1080 READ #1,1;Q
1090 Q0=Q >= 0
1100 CONVERT ABS(Q) TO U$
1110 N1=POS(U$,".")
1120 MAT N=ZER[2]
1130 IF  NOT N1 OR POS(U$,"E") THEN 1170
1140 FOR J=1 TO 2 MIN LEN(U$)-N1
1150 CONVERT U$[N1+J,N1+J] TO N[J]
1160 NEXT J
1170 Q=INT(ABS(Q))
1180 RETURN
1190 PRINT  USING 130
```

Craig Bina

```
1200   Q=-1
1210   RETURN
1220   REM  SHEET SEARCH ROUTINE:  IN (Q0,X$,S) OUT (Q,X,Y)
1230   IF  END #1 THEN 1340
1240   IF  NOT X THEN 1340
1250   READ #1,X
1260   ADVANCE #1;Y,R
1270   IF TYP(1)#1 THEN 1340
1280   READ #1;W
1290   IF TYP(1)#1 THEN 1340
1300   READ #1;Z
1310   IF Z#S THEN 1340
1320   CONVERT W TO Y$
1330   IF Y$=X$ THEN 1480
1340   READ #1,1
1350   ADVANCE #1;Q0,R
1360   IF  END #1 THEN 1490
1370   IF Q0 THEN 1400
1380   READ #1;Q
1390   Q=INT(ABS(Q))
1400   X=REC(1)
1410   Y=ITM(1)
1420   READ #1;W,Z
1430   IF Z=S THEN 1460
1440   ADVANCE #1;Q,R
1450   GOTO 1370
1460   CONVERT W TO Y$
1470   IF Y$#Y$ THEN 1440
1480   RETURN
1490   Q=-1
1500   RETURN
1510   REM  CHEATBOX-IN ROUTINE:  IN (I6,F9) OUT (B)
1520   ASSIGN "CHTBOX",1,B
1530   IF B THEN 1610
1540   LOCK #1
1550   ADVANCE #1;32767,R
1560   IF  NOT R THEN 1550
1570   X0=REC(1)
1580   Y0=ITM(1)
1590   IF  END #1 THEN 1630
1600   PRINT #1;I6,F9, END
1610   ASSIGN *,1
1620   RETURN
1630   READ #1,X0
1640   ADVANCE #1;Y0,R
1650   PRINT #1; END
1660   B=1
1670   RETURN
1680   REM  ANSWER RETRIEVAL ROUTINE:  IN (I) OUT (A OR A$,T)
1690   ADVANCE #1;I-1,R
1700   T=TYP(1)-1
1710   IF T THEN 1740
1720   READ #1;A
1730   GOTO 1760
1740   READ #1;A$
1750   A$=UPS$(A$)
1760   ASSIGN *,1
1770   RETURN
1780   REM  NUMERIC COMPARISON/OUTPUT ROUTINE:  IN (A,D,MAT N[2]) OUT (C)
1790   K=A
```

Craig Bina

```
1800  IF N[1]#2 THEN 1870
1810  V=A
1820  GOSUB 2710
1830  A=V
1840  V=D
1850  GOSUB 2710
1860  D=V
1870  M=0
1880  IF  NOT A THEN 1920
1890  F4=LOG(ABS(A))/2.30259
1900  F4=INT(F4+.001*SGN(F4))
1910  M=(N[2]=1)*10^(F4-1)
1920  C=D >= A-M AND D <= A+M
1930  IF C THEN 2170
1940  PRINT  USING 160
1950  A=K
1960  F1=F2=SGN(A)
1970  U$=" 0"
1980  IF  NOT F2 THEN 2020
1990  A=ABS(A)
2000  F1=F4
2010  CONVERT A/10^(F1+1) TO U$
2020  IF F2+1 THEN 2040
2030  PRINT "-";
2040  FOR F3=2 TO F1+2
2050  IF F3 <= LEN(U$) THEN 2070
2060  U$[F3]="0"
2070  PRINT U$[F3,F3];
2080  NEXT F3
2090  IF F3>LEN(U$) THEN 2150
2100  PRINT ".";
2110  FOR F5=1 TO -1-F1
2120  PRINT "0";
2130  NEXT F5
2140  PRINT U$[F3];
2150  PRINT
2160  RETURN
2170  PRINT  USING 150
2180  RETURN
2190  REM  STRING COMPARISON/OUTPUT ROUTINE:  IN (A$,U$,MAT N[2]) OUT (C)
2200  V$=A$
2210  IF N[1]#1 THEN 2270
2220  W$=U$
2230  GOSUB 2810
2240  U$=V$
2250  W$=A$
2260  GOSUB 2810
2270  IF U$=V$ THEN 2320
2280  PRINT  USING 160
2290  PRINT A$
2300  C=0
2310  RETURN
2320  PRINT  USING 150
2330  C=1
2340  RETURN
2350  REM  SHEET NUMBER FRACTIONALIZATION ROUTINE:  IN (X,Y,S) OUT (S)
2360  READ #1,X
2370  ADVANCE #1;Y+1,R
2380  E[1]=S=S+.5
2390  UPDATE #1;S
```

Craig Bina

```
2400  ASSIGN *,1
2410  RETURN
2420  REM  CHEATBOX-OUT ROUTINE:  IN (X$,F9)
2430  ASSIGN "CHTBOX",1,R
2440  IF  NOT R THEN 2490
2450  PRINT  USING 200
2460  PRINT  USING 210
2470  PRINT  USING 240
2480  GOTO 2670
2490  LOCK #1
2500  IF  END #1 THEN 2670
2510  X=REC(1)
2520  Y=ITM(1)
2530  READ #1;I,F
2540  IF F#F9 THEN 2510
2550  CONVERT I TO Y$
2560  IF Y$#X$ THEN 2510
2570  READ #1,X
2580  ADVANCE #1;Y,R
2590  ASSIGN "CHTBOX",2,R
2600  IF R>2 THEN 2450
2610  READ #2,X
2620  ADVANCE #2;Y+2,R
2630  IF  END #2 THEN 2670
2640  READ #2;I,F
2650  PRINT #1;I,F
2660  GOTO 2640
2670  PRINT #1; END
2680  ASSIGN *,2
2690  ASSIGN *,1
2700  RETURN
2710  REM  TWO SIG. FIG. ROUTINE:  IN (V) OUT (V)
2720  T8=SGN(V)
2730  IF  NOT T8 THEN 2790
2740  V=ABS(V)
2750  L8=LOG(V)/2.30259
2760  E8=INT(L8)
2770  V=V/10^E8
2780  V=INT(10*V+.5)*10^(E8-1)
2790  V=V*T8
2800  RETURN
2810  REM  BLANK STRIPPING ROUTINE:  IN (W$) OUT (V$)
2820  V$=""
2830  FOR I8=1 TO LEN(W$)
2840  IF NUM(W$[I8])=32 THEN 2860
2850  V$[LEN(V$)+1]=W$[I8,I8]
2860  NEXT I8
2870  RETURN
2880  REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
2890  FOR I=0 TO 2
2900  E[I+1]=SYS(I)
2910  NEXT I
2920  CHAIN R,"ERROR"
2930  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
2940  GOTO 810
2950  END
```

Craig Bina

3.10

AUXCK3

```
10   COM A,N$[80],B1,X1,X2,O0$[11],E[3]
20   O0$="AUXCK3"
30   IF  ERROR  THEN 1700
40   Y=BRK(0)
50   FILES *,*
60   ASSIGN "MAIN3",2,R
70   ASSIGN "MAIN3",1,R1
80   IF R=0 THEN 170
90   PRINT "SORRY, FILE: MAIN3 IS NON-EXISTENT."
100   GOTO 1760
110   PRINT #2,1: END
120   ASSIGN *,1
130   PRINT "NO ANSWERS HAVE BEEN STORED FOR FUNDAMENTAL #3."
140   GOTO 1760
150   PRINT "SORRY, THE REQUESTED ANSWERS HAVE NOT BEEN STORED IN FILE: MAIN3 ."
160   GOTO 1760
170   PRINT "OKAY, "N$", WHAT IS YOUR SHEET NUMBER";
180   INPUT B
190   IF B<1 OR B#INT(B) THEN 170
200   DIM A[100],X$[50],Y$[50],L[4]
210   DIM P$[150],A$[150],Q$[150],G[19,26]
220   DIM X1$[50],Y1$[50]
230   PRINT "PLEASE ANSWER THE FOLLOWING QUESTIONS CONCERNING FUNDAMENTAL"
240   PRINT "THREE; GRAPHING.  IF NO UNITS ARE SPECIFIED FOR AN AXIS, THEN"
250   PRINT "TYPE "'34"NO UNITS"'34"."
260   CONVERT A TO Q$
270   PRINT
280   W=0
290   IF  END #1 THEN 130
300   READ #1:F
310   IF F#S THEN 110
320   IF  END #1 THEN 150
330   X1=REC(1)
340   X2=ITM(1)
350   READ #1:F,S,X$,Y$
360   CONVERT F TO P$
370   MAT  READ #1:L
380   IF P$#Q$ THEN 400
390   IF S=B THEN 420
400   ADVANCE #1;INT((L[2]-L[1])/L[3])+1,R
410   GOTO 330
420   READ #1,X1
421   ADVANCE #1;X2+1,R
422   UPDATE #1:.5
423   ADVANCE #1;4,R
429   X$=UPS$(X$)
430   Q$=X$
440   GOSUB 890
450   X1$=Q$
460   Y$=UPS$(Y$)
470   Q$=Y$
480   GOSUB 890
490   Y1$=Q$
500   PRINT "1) WHAT IS THE LABEL ON THE X-AXIS";
510   GOSUB 850
520   IF Q$=X1$[1,POS(X1$,"(")-1] THEN 560
530   PRINT "THE CORRECT ANSWER IS "X$[1,POS(X$,"(")-2]
540   W=1
550   GOTO 570
```

Craig Bina

```
560   PRINT "CORRECT!"
570   PRINT "2) WHAT IS THE LABEL ON THE Y-AXIS";
580   GOSUB 850
590   IF Q$=Y1$[1,POS(Y1$,"(")-1] THEN 630
600   PRINT "THE CORRECT ANSWER IS "Y$[1,POS(Y$,"(")-2]
610   W=W+1
620   GOTO 640
630   PRINT "CORRECT!"
640   PRINT "3) WHAT ARE THE UNITS ON THE X-AXIS";
650   GOSUB 850
660   IF Q$=X1$[POS(X1$,"(")+1,LEN(X1$)-1] THEN 700
670   PRINT "THE CORRECT ANSWER IS "X$[POS(X$,"(")+1,LEN(X$)-1]
680   W=W+1
690   GOTO 710
700   PRINT "CORRECT!"
710   PRINT "4) WHAT ARE THE UNITS ON THE Y-AXIS";
720   GOSUB 850
730   IF Q$=Y1$[POS(Y1$,"(")+1,LEN(Y1$)-1] THEN 770
740   PRINT "THE CORRECT ANSWER IS "Y$[POS(Y$,"(")+1,LEN(Y$)-1]
750   W=W+1
760   GOTO 780
770   PRINT "CORRECT!"
780   PRINT "ENTER YOUR GRAPH'S TITLE, DO NOT ABBREVIATE THE WORD "'34"VERSUS"'34"."
790   GOSUB 850
800   A$=Q$
810   P$=Y$[1,POS(Y$,"(")-2]
820   P$[LEN(P$)+1]=" VERSUS "
830   P$[LEN(P$)+1]=X$[1,POS(X$,"(")-2]
840   GOTO 970
850   INPUT Q$
860   Q$=UPS$(Q$)
870   GOSUB 890
880   RETURN
890   J=1
900   FOR I=1 TO LEN(Q$)
910   IF Q$[I,I]=" " THEN 940
920   Q$[J,J]=Q$[I,I]
930   J=J+1
940   NEXT I
950   Q$=Q$[1,J-1+(J=1)]
960   RETURN
970   Q$=P$
980   GOSUB 890
990   IF A$=Q$ THEN 1040
1000   PRINT "THE CORRECT ANSWER IS:"
1010   PRINT P$
1020   W=W+1
1030   GOTO 1050
1040   PRINT "CORRECT!"
1050   S=(5-W)/5*100
1060   IF SYS(4)#3 THEN 1090
1070   PRINT "PRESS <RETURN> FOR A PRINTOUT OF YOUR GRAPH."
1080   LINPUT P$
1090   PRINT "YOUR GRAPH SHOULD LOOK LIKE THIS:"'10
1100   MAT G=CON
1110   MAT G=(32)*G
1120   FOR I=1 TO 26
1130   IF I>18 THEN 1150
1140   G[I,1]=NUM("I")
1150   G[19,I]=NUM("-")
```

Craig Bina

```
1160  NEYT I
1170  G[19,1]=NUM("0")
1180  MAT A=ZER[INT((L[2]-L[1])/L[3])+1]
1190  MAT  READ #1;A
1200  Y1=-1
1210  FOR I=1 TO INT((L[2]-L[1])/L[3])+1
1220  A[I]=A[I]/L[4]
1230  Y1=Y1 MAX A[I]
1240  NEXT I
1250  X3=L[2]
1260  I1=(X3/25) MAX (Y1/36)
1270  I=0
1280  FOR J=L[1] TO L[2] STEP L[3]
1290  I=I+1
1300  X=INT(J/I1+.5)
1310  Y=INT(A[I]/(2*I1)+.5)
1320  G[19-Y,X+1]=NUM("X")
1330  NEXT J
1340  FOR I=1 TO 19
1350  FOR J=26 TO 2 STEP -1
1360  IF G[I,J]#32 THEN 1390
1370  G[I,J]=127
1380  NEXT J
1390  NEXT I
1400  FOR X=1 TO 19
1410  FOR Y=1 TO 26
1420  IF G[X,Y]=127 THEN 1450
1430  PRINT CHR$(G[X,Y]);
1440  NEXT Y
1450  PRINT
1460  NEXT X
1470  GOSUB 1540
1480  PRINT LIN(2)"YOU HAD"W"WRONG OUT OF 5 PROBLEMS;"
1490  PRINT "THIS YIELDS A SCORE OF"S"PERCENT RIGHT.";10
1500  CHAIN R,"UPDATE"
1510  PRINT "THE PROGRAM: UPDATE  IS MISSING,"
1520  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
1530  GOTO 1760
1540  REM CLEARING ROUTINE
1550  READ #2,X1
1560  ADVANCE #2;X2+1,R
1570  UPDATE #2;-S
1580  ADVANCE #2;5,R
1590  GOTO TYP(1) OF 1600,1630,1660
1600  READ #1;X
1610  PRINT #2:X
1620  GOTO 1590
1630  READ #1;P$
1640  PRINT #2:P$
1650  GOTO 1590
1660  ASSIGN *,1
1670  PRINT #2; END
1680  ASSIGN *,2
1690  RETURN
1700  FOR I=0 TO 2
1710  E[I+1]=SYS(I)
1720  NEXT I
1730  CHAIN R,"ERROR"
1740  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
1750  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
```

Craig Bina

```
1760  CHAIN R,"MAIN"
1770  PRINT "THE PROGRAM: MAIN  IS ALSO MISSING, PLEASE INFORM YOUR INSTRUCTOR OF THIS."
1780  SYSTEM R,"BYE"
1790  GOTO 1780
1800  END
```

Craig Bina

3.11

UPDATE

```
10    COM I6,N0$[80],F9,X6,Y6,00$[11],E[3]
20    00$="UPDATE"
30    IF  ERROR  THEN 2770
40    O=BRK(0)
50    FILES *,*,*,*
60    DIM A$[255],B0$[6],D0$[6],M$[6],R$[6],S0$[6],X$[15],Y$[15],Z$[6]
70    DIM I0$[6],B1$[6]
80    IMAGE /"PLEASE STAND BY WHILE YOUR SCORE IS RECORDED."/
90    IMAGE /"THE FILE:  ",6A2X,"IS MISSING."
100    IMAGE"PLEASE STAND BY WHILE THE ANSWER STORAGE FILE IS SHRUNK TO ",5DX,"RECORDS."
110    IMAGE "PLEASE STAND BY WHILE THE SCORE STORAGE FILE IS EXPANDED."
120    IMAGE/"THE FILE:  ",6A2X,"IS FULL AND THERE IS INSUFFICIENT ROOM TO EXPAND IT."
130    IMAGE"CONSEQUENTLY, YOUR SCORE HAS NOT BEEN RECORDED."
140    IMAGE /"THE PROGRAM:  INSERT  IS MISSING."
150    IMAGE /"THE PROGRAM:  MAIN  IS MISSING."
160    IMAGE "PLEASE INFORM YOUR INSTRUCTOR."
170    IMAGE "PLEASE INFORM THE AUTHORITIES."
180    B0$="BRUP"
190    D0$="DYUP"
200    S0$="SDUP"
210    R$="RFCR"
220    CONVERT I6 TO X$
230    L=INT(LOG(I6)/2.30259)
240    CONVERT I6/10^(L-1) TO A$
250    CONVERT A$ TO Q
260    L=L+3* NOT INT(100*(Q-INT(10*Q)/10))
270    Q=I6/10^L
280    H=INT(100*Q)/100
290    Q=INT(10000*(Q-INT(100*Q)/100)+.9)
300    M$="MAIN"
310    B1$="BFFR"
320    R$[5]=CHR$(L+64)
330    R$[6]=CHR$(Q)
340    B0$[5]=R$[5]
350    D0$[5]=R$[5]
360    S0$[5]=R$[5]
370    Y6=1000*F9+Y6
380    PRINT  USING 80
390    IF  NOT E[1] THEN 580
400    CONVERT F9 TO M$[5]
410    B1$[5]=M$[5]
420    Z$=M$
430    GOSUB 800
440    IF R THEN 580
450    GOSUB 1030
460    IF Q<0 THEN 560
470    READ #1,X
480    ADVANCE #1;Y,R
490    F4=2
500    T4=1
510    GOSUB 1270
520    M=REC(1)
530    IF  END #1 THEN 560
540    READ #1,M+1
550    GOSUB 1370
560    ASSIGN *,2
570    ASSIGN *,1
580    Z$=R$
590    GOSUB 800
```

Craig Bina

```
600  IF  NOT R THEN 630
610  PRINT  USING 130
620  GOTO 660
630  GOSUB 1720
640  ASSIGN *,2
650  ASSIGN *,1
660  IF E[1] THEN 710
670  I6=-1
680  CHAIN R,"INSERT"
690  PRINT  USING 140
700  PRINT  USING 170
710  CHAIN R,"MAIN"
720  PRINT  USING 150
730  PRINT  USING 160
740  SYSTEM R,"BYE"
750  GOTO 740
760  F4=2
770  T4=1
780  GOSUB 1240
790  GOTO 640
800  REM  LOCKS ROUTINE:  IN (Z$) OUT (R)
810  FOR I=1 TO 10
820  ASSIGN Z$,1,R, WR
830  IF R<6 THEN 870
840  ASSIGN *,1
850  SYSTEM R,"PAU-6"
860  GOTO 820
870  IF R=0 THEN 960
880  SYSTEM R9,"PAU-2"
890  NEXT I
900  ASSIGN *,1
910  ASSIGN *,2
920  PRINT  USING 90;Z$
930  PRINT  USING 160
940  R=1
950  RETURN
960  LOCK #1
970  SYSTEM R,"PAU-2"
980  LOCK #1,R9
990  IF R9=0 THEN 840
1000  ASSIGN Z$,2,R9
1010  IF R9>2 THEN 900
1020  RETURN
1030  REM  SHEET LOCATION ROUTINE:  IN (X$,E[1]) OUT (Q,X,Y)
1040  IF  END #2 THEN 1210
1050  READ #2,1;Q
1060  Q0=Q >=  0
1070  Q=INT(ABS(Q))
1080  IF Q0 THEN 1100
1090  READ #2,1
1100  X=REC(2)
1110  Y=ITM(2)
1120  IF Q0 THEN 1150
1130  READ #2;Q
1140  Q=INT(ABS(Q))
1150  READ #2;W,Z
1160  ADVANCE #2;Q,R
1170  IF Z#E[1] THEN 1100
1180  CONVERT W TO Y$
1190  IF Y$#X$ THEN 1100
```

Craig Bina

```
1200  RETURN
1210  Q=-1
1220  RETURN
1230  REM  COPY ROUTINE:  IN (F4,T4)
1240  REM   REWIND DESTINATION
1250  READ #F4,1
1260  READ #T4,1
1270  REM   RANDOM DESTINATION
1280  GOTO TYP(F4)-1 OF 1320,1350
1290  READ #F4;N
1300  PRINT #T4;N
1310  GOTO 1280
1320  READ #F4;A$
1330  PRINT #T4;A$
1340  GOTO 1280
1350  PRINT #T4; END
1360  RETURN
1370  REM  SHRINKING ROUTINE:  IN (M)
1380  PRINT  USING 100;M
1390  CREATE R,B1$,M
1400  IF R THEN 1700
1410  ASSIGN B1$,2,R
1420  IF R THEN 1690
1430  F4=1
1440  T4=2
1450  GOSUB 1240
1460  ASSIGN *,1
1470  PURGE R,M$
1480  IF R=0 THEN 1530
1490  ASSIGN M$,1,R
1500  UNLOCK #1
1510  SYSTEM R,"PAU-1"
1520  GOTO 1460
1530  CREATE R,M$,M
1540  ASSIGN M$,1,R
1550  A$="MWA-"
1560  A$[5]=M$
1570  SYSTEM R,A$
1580  A$[1,3]="PRI"
1590  SYSTEM R,A$
1600  ASSIGN M$,1,R, WR
1610  IF R=0 THEN 1660
1620  ASSIGN M$,1,R
1630  UNLOCK #1
1640  SYSTEM R,"PAU-1"
1650  GOTO 1600
1660  F4=2
1670  T4=1
1680  GOSUB 1240
1690  ASSIGN *,2
1700  PURGE R,B1$
1710  RETURN
1720  REM  SCORE INSERTION ROUTINE:  IN (H,Y6,F9)
1730  IF  END #1 THEN 2070
1740  IF  END #2 THEN 1810
1750  X=REC(2)
1760  Y=ITM(2)
1770  READ #2;P
1780  IF P>0 THEN 1750
1790  IF -P<H THEN 1750
```

Craig Bina

```
1800  IF -P=H THEN 1860
1810  READ #1,X
1820  ADVANCE #1;Y,R
1830  C=1
1840  PRINT #1;-H,Y6, END
1850  GOTO 1960
1860  IF  END #2 THEN 1920
1870  X=REC(2)
1880  Y=ITM(2)
1890  READ #2;P
1900  IF P <= 0 THEN 1920
1910  IF INT(P/1000)<F9 THEN 1870
1920  READ #1,X
1930  ADVANCE #1;Y,R
1940  C=2
1950  PRINT #1;Y6, END
1960  IF  END #2 THEN 2050
1970  C=3
1980  READ #2,X
1990  ADVANCE #2;Y,R
2000  READ #2;P
2010  X=REC(1)
2020  Y=ITM(1)
2030  PRINT #1;P
2040  GOTO 2000
2050  PRINT #1; END
2060  RETURN
2070  REM  EXPANSION CONTROLLER ROUTINE:  IN (C)
2080  PRINT  USING 110
2090  GOSUB 2140
2100  IF E THEN 760
2110  GOTO C OF 1840,1950
2120  IF  END #2 THEN 2050
2130  GOTO 2030
2140  REM  EXPANSION ROUTINE:  IN (X,Y,R$,B0$,D0$,S0$) OUT (E)
2150  X7=X
2160  Y7=Y
2170  I0$=R$
2180  E=1
2190  READ #1,X7
2200  ADVANCE #1;Y7,R
2210  PRINT #1: END
2220  CREATE R,B0$,X7
2230  IF R THEN 2720
2240  CREATE R,S0$,X7
2250  IF R THEN 2710
2260  CREATE R,D0$,X7+1
2270  IF R THEN 2700
2280  ASSIGN B0$,3,R
2290  F4=1
2300  T4=3
2310  GOSUB 1240
2320  ASSIGN S0$,3,R
2330  X3=REC(2)
2340  Y3=ITM(2)
2350  F4=2
2360  GOSUB 1240
2370  ASSIGN *,1
2380  ASSIGN *,2
2390  PURGE R,I0$
```

Craig Bina

```
2400  IF R=0 THEN 2450
2410  ASSIGN IO$,1,R
2420  UNLOCK #1
2430  SYSTEM R,"PAU-1"
2440  GOTO 2370
2450  PURGE R,DO$
2460  CREATE R,IO$,X7+1
2470  ASSIGN IO$,1,R
2480  A$="MWA-"
2490  A$[5]=IO$
2500  SYSTEM R,A$
2510  A$[1,3]="PRI"
2520  SYSTEM R,A$
2530  ASSIGN IO$,1,R, WR
2540  IF R=0 THEN 2590
2550  ASSIGN IO$,1,R
2560  UNLOCK #1
2570  SYSTEM R,"PAU-1"
2580  GOTO 2530
2590  ASSIGN IO$,2,R
2600  F4=3
2610  T4=2
2620  GOSUB 1240
2630  ASSIGN BO$,3,R
2640  T4=1
2650  GOSUB 1240
2660  ASSIGN *,3
2670  READ #2,X3
2680  ADVANCE #2;Y3,R
2690  E=0
2700  PURGE R,SO$
2710  PURGE R,BO$
2720  IF E THEN 2740
2730  RETURN
2740  PRINT  USING 120;IO$
2750  PRINT  USING 130
2760  RETURN
2770  REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
2780  FOR I=0 TO 2
2790  E[I+1]=SYS(I)
2800  NEXT I
2810  CHAIN R,"ERROR"
2820  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
2830  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
2840  GOTO 660
2850  END
```

Craig Bina

3.12

MASFND

```
10   COM I,N$[80],F,X,Y,00$[11],E[3]
20   COM L
30   REM**************************NEVER RENUMBER THIS PROGRAM**************************
40   00$="MASFND"
50   0=BRK(0)
60   IF  ERROR  THEN 2000
70   FILES *
80   DIM A$[72],B$[80],K$[50]
85   DIM Y[367]
90   X=300
100   Y=I
110   IF SYS(4)#3 THEN 160
120   PRINT "THIS PROGRAM SHOULD NOT BE RUN ON A CRT UNIT."
130   PRINT "DO YOU WANT TO CONTINUE";
140   INPUT A$
150   IF A$[1,1]="N" THEN 1730
160   PRINT '10'10"GENERATION OF MASSIVE QUANTITIES OF PRACTICE SHEETS."'10'10
170   GOSUB 1920
180   PRINT "PRACTICE SHEETS FOR FUNDAMENTALS X THROUGH Y (X,Y)."
190   INPUT A,B
200   IF A <= 0 OR INT(A)#A OR B>99 OR INT(B)#B OR A>B THEN 180
210   A$="CAT-FUND"
220   FOR N=A TO B
230   CONVERT N TO A$[9]
240   SYSTEM B$,A$
250   IF A$[5]=B$[1,POS(B$," ")-1] THEN 270
260   PRINT "PROGRAM: "A$[5]"  IS MISSING."
270   NEXT N
280   F=E[1]=A
290   E[2]=B
300   PRINT "FOR I.D. NUMBER X TO I.D. NUMBER Y (X,Y)."
310   INPUT A,B
320   IF A<0 OR B>9.99 OR A>B THEN 300
340   PRINT LIN(7)
350   A=(A+.0001*NUM(T1$))*10^(NUM(T$)-64)
360   B=(B+.0001*NUM(T1$))*10^(NUM(T$)-64)
370   CONVERT B TO A$
380   CONVERT A$ TO E[3]
390   CONVERT A TO A$
400   CONVERT A$ TO I
410   PRINT "SET TO TOP-OF-FORM AND PRESS <RETURN>"
420   ENTER 255,R,B$
430   IF R<0 THEN 420
440   B$="MES-AUTOMATIC FUNDAMENTAL GENERATION IN PROGRESS:"
450   SYSTEM R,B$
460   IF R THEN 450
470   B$[5]="PLEASE DO NOT SEND ANY MESSAGES TO W140 ON PORT #"'1
480   ENTER #07
490   CONVERT Q7 TO B$[POS(B$,'1)]
500   SYSTEM R,B$
510   IF R THEN 500
520   REM  NAME FINDER:
530   A$="INTL"
540   GOSUB 1920
550   A$[5]=T$
560   A$[6]=T1$
570   ASSIGN A$,1,R
580   IF R<3 THEN 610
590   PRINT "YOUR INITIAL FILE, "A$", IS MISSING."
```

Craig Bina

82

```
600   GOTO 1730
610   IF  END #1 THEN 1510
620   READ #1,1
630   CONVERT I TO K$
640   CONVERT K$ TO I
650   READ #1;A$,Q,N$
660   CONVERT Q TO B$
670   CONVERT B$ TO Q
680   IF Q<I THEN 650
690   IF Q>I THEN 1510
700   IF A$="***" THEN 650
710   I=Q
720   F=F-1
730   GOTO 1150
740   REM  CHAIN HERE FOR A RETURN
750   00$="MASFND"
760   0=BRK(0)
770   IF  ERROR  THEN 2000
780   REM  K6 IS THE NUMBER OF LINES PER PAGE FOR DESIGNATED DEVICE
790   REM MINUS 5 WHICH IS USED FOR HEADER SPACE
800   K6=50-5
810   RESTORE 850
820   N1=0
830   REM ..... MOVING DOWN THE PAGE????  THEN
840   REM ........... INCREASE THE ABSOLUTE VALUE IN THE DATA STATEMENT
850   DATA 22,46,-13,22
860   DATA -19,18,-14,-12
870   DATA 29,-16,25,20,-20
880   DATA 20
890   DATA 28
900   DATA 20,20,20,17,-19,21
910   DATA 21,19,26
920   DATA 41,0,0,0,0
930   DATA 0,0,0,0,0
940   DATA 0,0,0,0,0
950   DATA 0,0,0,0,0
960   DATA 0,0,0,0,0
970   DATA 54,54,54,54
980   DATA 54,54,54,54
990   FOR X2=1 TO F
1000   IF TYP(0)=3 THEN 1060
1010   READ N
1020   NEXT X2
1030   REM   ACCOUNT FOR TWO FREE LINES DONATED BY "NOTICE"
1040   N=N+2*SGN(N)
1050   GOTO 1070
1060   N=0
1070   IF N >= 0 THEN 1090
1080   N=-N+L
1090   IF N <= K6 THEN 1130
1100   N=N-K6
1110   N1=5
1120   GOTO 1090
1130   PRINT LIN(K6-N+N1)
1140   PRINT LIN(5)
1150   F=F+1
1160   IF F>E[2] THEN 1580
1170   A$="FUND#"
1180   CONVERT F TO A$[5]
1190   ASSIGN "USAGE",1,R
```

Craig Bina

83

```
1200   IF R THEN 1480
1210   LOCK #1
1220   Y6=TIM(3)-1+100* NOT TIM(3)
1230   Z6=Y6-1+100* NOT Y6
1240   W6=TIM(2) MIN 366
1250   IF  END #1 THEN 1290
1260   MAT  READ #1;Y
1270   X6=(Y[1]=Z6)+2*(Y[1]=Y6)
1280   GOTO X6 OF 1350,1420
1290   IF  END #1 THEN 1480
1300   MAT Y=ZER
1310   Y[1]=Y6
1320   READ #1,1
1330   MAT  PRINT #1;Y, END
1340   GOTO 1380
1350   READ #1,4
1360   MAT  READ #1;Y
1370   GOTO (Y[1]=Y6)+1 OF 1290,1320
1380   MAT Y=ZER
1390   Y[1]=TIM(3)
1400   READ #1,4
1410   MAT  PRINT #1;Y, END
1420   READ #1,4
1430   ADVANCE #1;W6,R
1440   READ #1;X6
1450   READ #1,4
1460   ADVANCE #1;W6,R
1470   UPDATE #1;X6+1
1480   ASSIGN *,1,R
1490   CHAIN R,A$
1500   GOTO 1150
1510   REM NO I.D.
1520   GOSUB 1920
1540   IF  INT(Q/L)=INT(I/L) THEN 1560
1550   I=(INT(I/L)+1+NUM(T1$)/10000)*L
1560   I=I+.01*L
1570   GOTO 1600
1580   F=E[1]
1590   GOSUB 1920
1593   IF L2 THEN 1550
1595   I=I+.01*L
1600   CONVERT I TO A$
1610   CONVERT A$ TO I
1620   IF I <= E[3] THEN 520
1630   PRINT LIN(5)
1640   B$="MES-FUNDAMENTAL PROCESSING COMPLETED.  THANK YOU FOR YOUR COOPERATION."
1650   SYSTEM R,B$
1660   IF R THEN 1650
1670   B$[5]="MESSAGES NOW ACCEPTED BY W140 ON PORT #"'1
1680   ENTER #Q7
1690   CONVERT Q7 TO B$[POS(B$,'1)]
1700   SYSTEM R,B$
1710   IF R THEN 1700
1720   PRINT "PROCESSING COMPLETED"
1730   I=Y
1750   GOSUB 1920
1760   A$="INTL"
1770   A$[5]=T$
1780   A$[6]=T1$
1790   ASSIGN A$,1,R
```

Craig Bina

```
1800  IF  NOT R THEN 1830
1810  I=-1
1820  GOTO 1880
1830  IF  END #1 THEN 1810
1835  CONVERT I TO A$
1840  READ #1;B$,11,N$
1860  CONVERT I1 TO B$
1870  IF A$#B$ THEN 1840
1880  CHAIN R,"MAIN"
1890  PRINT "THE PROGRAM:  MAIN  IS MISSING.  ACCOUNT IS CLOSED."
1900  SYSTEM R,"BYE"
1910  GOTO 1900
1920  L=INT(LOG(I)/2.30259)
1930  CONVERT I/10^(L-1) TO B$
1940  CONVERT B$ TO Q2
1950  L2= NOT INT(100*(Q2-INT(10*Q2)/10))
1954  L=L+3*L2
1960  T$=CHR$(L+64)
1965  L=10^L
1970  I2=I/L
1980  T1$=CHR$(INT(10000*(I2-INT(I2*100)/100)+.9))
1990  RETURN
2000  REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
2010  FOR I=0 TO 2
2020  E[I+1]=SYS(I)
2030  NEXT I
2040  CHAIN R,"ERROR"
2050  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM:  ERROR  IS MISSING."
2060  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
2070  GOTO 1880
2080  END
```

Craig Bina

3.13

LDINTL

```
10   COM I6,N1$[80],F9,X6,Y6,O0$[11],E[3]
20   O0$="LDINTL"
30   O=BRK(0)
40   IF  ERROR  THEN 9000
50   FILES *,*,*
60   DIM I0$[6],B0$[6],D0$[6],S0$[6],C0$[24]
70   DIM A$[255],B$[255],C$[255],D$[20],I$[3],J$[3]
71   DEF FNH(X)=INT((X/B-A)*100)/100
72   IMAGE "STUDENTS",XD.DDX,"THROUGH",XD.DDX,"ARE ALREADY IN THE CLASS LIST FILE."
73   IMAGE #,XD.DD/"->",X
80   I0$="INTL"
90   B0$="BR01"
100   D0$="DY01"
110   S0$="SD01"
111   C0$="HELRELLISADDDELCORDUPSTO"
112   DATA "HELP - GIVES INSTRUCTIONS"
113   DATA "RELOAD - RELOADS ENTIRE CLASS LIST FILE"
114   DATA "LIST - LISTS CLASS LIST FILE"
115   DATA "ADD - ADDS STUDENTS TO CLASS LIST FILE"
116   DATA "DELETE - DELETES STUDENTS FROM CLASS LIST FILE"
117   DATA "CORRECT - CHANGES NAMES AND INITIALS FOR A GIVEN I.D. NUMBER"
118   DATA "DUPLICATES - SEARCHES CLASS LIST FILE FOR DUPLICATE INITIALS"
119   DATA "STOP - TERMINATES CLASS LIST MANIPULATION"
120   PRINT '10'10"CLASS LIST CREATION AND ALTERATION"'10'10
130   L=INT(LOG(I6)/2.30259)
140   CONVERT I6/10^(L-1) TO A$
150   CONVERT A$ TO Q
160   L=L+3* NOT INT(100*(Q-INT(10*Q)/10))
170   I0$[5]=CHR$(L+64)
180   Q=I6/10^L
181   I0$[6]=CHR$(INT(10000*(Q-INT(100*Q)/100)+.9))
182   B0$[5]=I0$[5]
183   D0$[5]=I0$[5]
184   S0$[5]=I0$[5]
191   A=NUM(I0$[6])/10000
192   B=10^(NUM(I0$[5])-64)
200   ASSIGN I0$,1,R
210   IF  NOT R THEN 240
220   PRINT "FILE:  "I0$"  IS MISSING."
230   GOTO 2010
240   ASSIGN I0$,2,R
250   IF R>2 THEN 220
260   PRINT "TYPE  HELP  FOR INSTRUCTIONS."'10
270   PRINT '10"COMMAND > ";
280   LINPUT A$
290   IF  NOT LEN(A$) THEN 270
300   IF LEN(A$)<3 THEN 410
310   C=(POS(C0$,A$[1,3])-1)/3+1
320   IF C#INT(C) THEN 410
330   GOSUB 430
340   PRINT
350   GOSUB C OF 490,560,650,840,1300,1314,1640,1970
360   F4=1
370   T4=2
380   GOSUB 2700
390   READ #1,1
400   GOTO 270
410   PRINT '10"ILLEGAL RESPONSE"
420   GOTO 270
```

Craig Bina

```
430  RESTORE 112
440  FOR I4=1 TO C
450  READ A$
460  NEXT I4
470  PRINT A$
480  RETURN
490  REM  HELP
500  PRINT '10"IT IS ONLY NECESSARY TO TYPE THE FIRST"
510  PRINT "THREE LETTERS OF A COMMAND, NOT THE ENTIRE WORD."'10
520  FOR C=1 TO 8
530  GOSUB 430
540  NEXT C
550  RETURN
560  REM  RELOAD
570  PRINT "ARE YOU SURE THAT YOU WISH TO DO THIS";
580  INPUT A$
590  IF A$[1,1]="Y" THEN 620
600  PRINT "RELOADING ABORTED"
610  RETURN
620  I3=.01
621  GOSUB 2040
630  PRINT '10"RELOADING COMPLETED"
640  RETURN
650  REM  LIST
660  PRINT "WHAT ARE THE FIRST AND LAST HOURS TO BE LISTED (EXAMPLE:  0,9)";
670  INPUT F,L
680  F=INT(F) MIN 9 MAX 0
690  L=INT(L) MIN 9 MAX F
700  PRINT "PRESS <INTRPT> TO INTERRUPT LISTING"
701  I=SYS(3)
710  PRINT LIN(3)"REFERENCE CHECKLIST:  HOURS"F"TO"L'10
711  PRINT "INITIALS","I.D. #","FULL NAME"
720  IF  END #1 THEN 820
730  READ #1;I$,D,A$
740  D=FNH(D)
750  H=INT(D)
760  IF H<F THEN 730
770  IF H>L THEN 820
780  PRINT I$,
790  PRINT  USING "#,XD.DD";D
800  PRINT " ",A$
810  IF  NOT SYS(3) THEN 730
811  PRINT '10"INTERRUPT DETECTED"'10
812  RETURN
820  PRINT '10"LISTING COMPLETED"LIN(3)
830  RETURN
840  REM  ADD
850  PRINT "WHICH HOUR IS TO BE ADDED OR ADDED TO";
860  INPUT H
870  IF H#INT(H) OR H<1 OR H>9 THEN 850
880  IF  END #1 THEN 950
890  X5=REC(1)
900  Y5=ITM(1)
910  READ #1;I$,D,A$
920  H1=INT(FNH(D))
930  IF H1<H THEN 890
940  IF H1=H THEN 1100
950  PRINT '10"BEGINNING HOUR #"H
951  I3=.01
960  READ #1,X5
```

Craig Bina

```
970   ADVANCE #1;Y5,R
980   GOSUB 2040
990   READ #2,X5
1000   ADVANCE #2;Y5,R
1010   IF TYP(2)=3 THEN 1089
1020   PRINT "PLEASE STAND BY WHILE THE HOURS GREATER THAN"H
1030   PRINT "ARE MOVED DOWN IN THE CLASS LIST FILE."
1031   IF  END #1 THEN 1071
1040   IF  END #2 THEN 1089
1050   READ #2;I$,D,A$
1051   X7=REC(1)
1052   Y7=ITM(1)
1060   PRINT #1;I$,D,A$, END
1070   GOTO 1050
1071   GOSUB 2820
1072   IF E THEN 1076
1073   IF  END #1 THEN 1071
1074   IF  END #2 THEN 1080
1075   GOTO 1051
1076   PRINT "YOU NOW HAVE TWO OPTIONS:"
1077   PRINT "1) RETURN THE CLASS LIST FILE TO THE STATE IT WAS IN"
1078   PRINT "    BEFORE YOU TYPED  ADD  AND THUS IGNORE ALL OF"
1079   PRINT "    THE ADDITIONS YOU HAVE JUST ENTERED, OR"
1080   PRINT "2) RETAIN YOUR ADDITIONS AND DROP ALL STUDENTS"
1081   PRINT "    AFTER AND INCLUDING #"D"."
1082   PRINT "PLEASE TYPE 1 OR 2.  WHAT IS YOUR CHOICE";
1083   INPUT T4
1084   IF T4#1 AND T4#2 THEN 1082
1085   F4=3-T4
1086   GOSUB 2700
1087   PRINT "OPTION"T4"ACCOMPLISHED"
1089   PRINT '10"ADDITIONS COMPLETED"
1090   RETURN
1100   D=D1=FNH(D)
1101   IF  END #1 THEN 1170
1110   D2=D
1120   X6=REC(1)
1130   Y6=ITM(1)
1140   READ #1;I$,D,A$
1150   D=FNH(D)
1160   IF INT(D)=H THEN 1110
1170   PRINT  USING 72;D1,D2
1180   PRINT "YOU NOW HAVE TWO OPTIONS:"
1190   PRINT "1) ADD TO THEM, OR"
1200   PRINT "2) ERASE AND RE-ENTER THEM."
1210   PRINT "PLEASE TYPE 1 OR 2.  WHAT IS YOUR CHOICE";
1220   INPUT I
1230   IF I=2 THEN 950
1240   IF I#1 THEN 1210
1250   X5=X6
1260   Y5=Y6
1270   I3=D2-H+.01
1280   PRINT '10"ADDING TO HOUR"H", BEGINNING AT #"I3+H
1290   GOTO 960
1300   REM  DELETE
1310   PRINT "ENTER THE I.D. NUMBER OF THE STUDENT TO BE DELETED AFTER EACH "'34">"'34"."
1311   GOSUB 1319
1312   PRINT '10"DELETIONS COMPLETED"
1313   RETURN
1314   REM  CORRECT
```

Craig Bina

```
1315  PRINT "ENTER THE I.D. NUMBER OF THE STUDENT TO BE CORRECTED AFTER EACH "'34">"'34"."
1316  GOSUB 1319
1317  PRINT '10"CORRECTIONS COMPLETED"
1318  RETURN
1319  REM  ---CHANGE
1320  PRINT "TYPE  QUIT  FOR AN I.D. NUMBER TO END DELETION."'10
1321  J$="***"
1322  C$="*  *"
1330  PRINT "> ";
1340  LINPUT A$
1350  IF A$="QUIT" THEN 1630
1360  CONVERT A$ TO I,1590
1365  IF I <= 0 OR I>9.99 THEN 1590
1370  CONVERT I TO A$
1380  READ #2,1
1390  IF  END #2 THEN 1610
1400  X5=REC(2)
1410  Y5=ITM(2)
1420  READ #2;I$,D,B$
1430  CONVERT FNH(D) TO D$
1440  IF D$#A$ THEN 1400
1441  PRINT  USING "#,XD.DD,2X";FNH(D)
1442  PRINT I$"  "B$"   ";
1443  IF C#6 THEN 1459
1444  PRINT '13'10"NEW INITIALS > ";
1445  LINPUT C$
1446  IF LEN(C$)#3 THEN 1444
1447  J$=UPS$(C$)
1448  PRINT "NEW NAME > ";
1449  LINPUT C$
1450  C$=UPS$(C$)
1451  P=POS(C$," ")
1452  IF P<2 OR P=LEN(C$) THEN 1448
1453  IF POS(C$[P+1]," ") THEN 1448
1459  READ #1,X5
1460  ADVANCE #1;Y5,R
1461  IF  END #1 THEN 1621
1462  S=0
1463  X7=REC(1)
1464  Y7=ITM(1)
1470  PRINT #1;J$,D,C$, END
1480  S=1
1500  IF  END #2 THEN 1540
1510  READ #2;I$,D,A$
1511  X7=REC(1)
1512  Y7=ITM(1)
1520  PRINT #1;I$,D,A$, END
1530  GOTO 1510
1540  IF C=6 THEN 1543
1541  PRINT "DELETED"'10
1542  GOTO 1550
1543  PRINT "CORRECTED"'10
1550  F4=1
1560  T4=2
1570  GOSUB 2700
1580  GOTO 1330
1590  PRINT '10"UNACCEPTABLE I.D. NUMBER"'10
1600  GOTO 1330
1610  PRINT "I.D. NUMBER NOT FOUND IN CLASS LIST FILE"'10
1620  GOTO 1330
```

Craig Bina

```
1621  GOSUB 2820
1622  IF E THEN 1627
1623  IF  END #1 THEN 1621
1624  IF  END #2 THEN 1540
1625  IF S THEN 1511
1626  GOTO 1463
1627  PRINT "CORRECTION NOT MADE"
1630  RETURN
1640  REM  DUPLICATES
1650  PRINT "THE FOLLOWING SETS OF STUDENTS HAVE"
1660  PRINT "BEEN RECORDED IN THE CLASS LIST FILE WITH IDENTICAL"
1670  PRINT "SETS OF INITIALS.  PLEASE REMEDY THE SITUATION"
1680  PRINT "THROUGH USE OF THE  CORRECT  COMMAND."'10
1690  T=V=0
1700  IF  END #1 THEN 1900
1710  IF  END #2 THEN 1720
1720  READ #1;I$,D,A$
1721  IF I$="***" THEN 1720
1722  IF  NOT V THEN 1730
1723  PRINT
1730  V=0
1740  READ #2,REC(1)
1741  ADVANCE #2;ITM(1),R
1742  X5=REC(2)
1743  Y5=ITM(2)
1750  READ #2;J$,D1,B$
1760  IF J$="***" THEN 1742
1770  IF J$#I$ THEN 1742
1780  IF T THEN 1800
1790  PRINT '10"D U P L I C A T E S"'10
1791  T=1
1800  IF V THEN 1830
1810  PRINT  USING "#,XD.DD,2X";FNH(D)
1820  PRINT I$"   "A$
1821  V=1
1830  PRINT  USING "#,XD.DD,2X";FNH(D1)
1840  PRINT J$"   "B$
1850  READ #2,X5
1860  ADVANCE #2;Y5,R
1870  UPDATE #2;"***"
1880  ADVANCE #2;2,R
1890  GOTO 1742
1900  F4=1
1910  T4=2
1920  GOSUB 2700
1930  IF T THEN 1950
1940  PRINT "NO DUPLICATES FOUND"
1950  PRINT LIN(3)"DUPLICATE SEARCH COMPLETED"
1960  RETURN
1970  REM  STOP
1980  PRINT '10"ALL CLASS LIST MANIPULATION TERMINATED"'10
1990  CHAIN R,"MAIN"
2000  PRINT "PROGRAM:  MAIN  IS MISSING."
2010  PRINT "PLEASE INFORM THE AUTHORITIES."
2020  SYSTEM R,"BYE"
2030  GOTO 2020
2040  REM  --ENTER
2050  P=C#2
2060  PRINT "DO YOU WANT INSTRUCTIONS";
2070  INPUT A$
```

Craig Bina

```
2080  IF A$[1,1]#"Y" THEN 2160
2090  PRINT "ENTER THE STUDENT'S THREE INITIALS (FML) AFTER A "'34"->"'34
2100  PRINT "AND HIS/HER FULL NAME (FIRST NAME FIRST WITH"
2110  PRINT "NO CUMMAS AND ONLY ONE BLANK SPACE WHICH MUST"
2120  PRINT "APPEAR ONLY BETWEEN THE FIRST NAME AND THE LAST)"
2130  PRINT "AFTER A "'34"=>"'34".  A SAMPLE SEQUENCE WOULD BE:"
2140  PRINT '10" 2.03"'13'10"-> CRB"'13'10"=> CRAIG BINA"'10
2150  PRINT "TO LEAVE AN I.D. NUMBER UNASSIGNED, TYPE  ***  FOR INITIALS."
2160  IF P THEN 2240
2170  PRINT "INSTRUCTOR, PLEASE ENTER YOUR OWN INITIALS AND NAME."
2180  IF A$[1,1]#"Y" THEN 2200
2190  PRINT "FOLLOW THE ABOVE FORMAT."
2191  D=0
2200  GOSUB 2350
2210  PRINT "WHICH HOUR ARE YOU ENTERING FIRST";
2220  INPUT H
2230  IF H<1 OR H>9 OR H#INT(H) THEN 2210
2240  PRINT "AFTER YOU HAVE FINISHED WITH HOUR #"H
2250  PRINT "TYPE  QUIT  FOR INITIALS.  TO ADD TO THIS HOUR OR TO"
2260  PRINT "ENTER OR ADD TO A DIFFERENT HOUR, USE THE  ADD  COMMAND AFTER"
2261  PRINT "TERMINATING THE RELOADING PROCESS BY TYPING  QUIT ."'10
2270  IF  END #1 THEN 2620
2271  Q=0
2280  FOR I=I3 TO .99 STEP .01
2281  D=H+I
2282  E=0
2290  GOSUB 2350
2295  IF E THEN 2340
2300  IF Q THEN 2330
2310  NEXT I
2320  PRINT '10"END-OF-HOUR"'13'10"-> QUIT"
2330  PRINT
2340  RETURN
2350  PRINT  USING 73;D
2360  LINPUT A$
2370  A$=UPS$(A$)
2380  IF  NOT P THEN 2420
2390  IF A$#"QUIT" THEN 2420
2400  Q=1
2410  RETURN
2420  IF LEN(A$)=3 THEN 2450
2430  PRINT "THREE (3) INITIALS PLEASE"
2440  GOTO 2350
2450  I$=A$
2460  IF I$#"***" THEN 2500
2470  PRINT "SKIPPED"
2480  A$="*  *"
2490  GOTO 2580
2500  PRINT "=> ";
2510  LINPUT A$
2520  A$=UPS$(A$)
2530  P=POS(A$," ")
2540  IF P>1 AND P<LEN(A$) THEN 2570
2550  PRINT '10"BAD NAME FORMAT"
2560  GOTO 2500
2570  IF POS(A$[P+1]," ") THEN 2550
2580  X7=REC(1)
2590  Y7=ITM(1)
2600  PRINT #1;I$,(D+A)*B,A$, END
2610  RETURN
```

Craig Bina

```
2620  GOSUB 2820
2630  IF E THEN 2660
2640  IF  END #1 THEN 2620
2650  GOTO 2580
2660  PRINT "THE LAST STUDENT ENTERED, #";
2670  PRINT  USING "#,XD.DD";D
2680  PRINT ", WAS IGNORED."
2690  RETURN
2700  REM  --COPY
2710  READ #F4,1
2720  READ #T4,1
2730  GOTO TYP(F4)-1 OF 2770,2800
2740  READ #F4;N
2750  PRINT #T4;N
2760  GOTO 2730
2770  READ #F4;B$
2780  PRINT #T4;B$
2790  GOTO 2730
2800  PRINT #T4; END
2810  RETURN
2820  REM  --EXPAND
2830  PRINT "PLEASE STAND BY WHILE THE CLASS LIST FILE IS EXPANDED."
2840  E=1
2850  READ #1,X7
2860  ADVANCE #1;Y7,R
2870  PRINT #1; END
2880  CREATE R,B0$,X7
2890  IF R THEN 3240
2900  CREATE R,S0$,X7
2910  IF R THEN 3230
2920  CREATE R,D0$,X7+1
2930  IF R THEN 3220
2940  ASSIGN B0$,3,R
2950  F4=1
2960  T4=3
2970  GOSUB 2700
2980  ASSIGN S0$,3,R
2981  X3=REC(2)
2982  Y3=ITM(2)
2990  F4=2
3000  GOSUB 2700
3010  ASSIGN *,1
3020  ASSIGN *,2
3030  PURGE R,I0$
3040  PURGE R,D0$
3050  CREATE R,I0$,X7+1
3051  ASSIGN I0$,1,R
3060  B$="MWA-"
3070  B$[5]=I0$
3080  SYSTEM R,B$
3090  B$[1,3]="PRI"
3100  SYSTEM R,B$
3110  ASSIGN I0$,1,R
3120  ASSIGN I0$,2,R
3130  F4=3
3140  T4=2
3150  GOSUB 2700
3160  ASSIGN B0$,3,R
3170  T4=1
3180  GOSUB 2700
```

Craig Bina

```
3181   ASSIGN *;3
3190   READ #2,X3
3200   ADVANCE #2;Y3,R
3210   E=0
3220   PURGE R,SOS
3230   PURGE R,BOS
3240   IF E THEN 3270
3250   PRINT "EXPANSION COMPLETED"
3260   RETURN
3270   PRINT "INSUFFICIENT ROOM TO EXPAND FILE"
3280   RETURN
9000   REM  ERROR TRAP DESTINATION
9010   FOR I=0 TO 2
9020   E[I+1]=SYS(I)
9030   NEXT I
9040   CHAIN R,"ERROR"
9050   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM:  ERROR  IS MISSING."
9060   GOTO 2010
9999   END
```

Craig Bina

3.14

```
OUTPUT

10   CUM Q7,A0$[80],F9,X6,Y6,OO$[11],E[3]
20   OU$="OUTPUT"
30   IF  ERROR  THEN 2270
40   Y=BRK(0)
50   FILES *,*
60   DIM A$[72],B$[72],Z$[15]
70   DIM C$[20]
80   DIM P[25,99],C[99],Q[24]
90   DIM M[12],D$[10]
100   S=1
110   A$="INTL"
180   PRINT '10'10"FUNDAMENTAL PRACTICE GRADEBOOK"'10'10
185   L=INT(LOG(Q7)/2.30259)+3
190   C5=10^L
200   L1=Q7/C5
210   C6=L1-INT(L1*100)/100
220   A$[5]=CHR$(L+64)
230   A$[6]=CHR$(INT(C6*10000+.9))
240   ASSIGN A$,2,R
250   IF R<3 THEN 280
260   PRINT "SORRY, FILE: "A$" DOES NOT EXIST."
270   GOTO 2230
280   A$[1,4]="RECR"
290   ASSIGN A$,1,R
300   IF R THEN 260
310   PRINT "PRESS <BREAK> OR <INTERRUPT> TO STOP THE LISTING."'10
320   Y=SYS(3)
330   DATA 31,0,31,30,31,30,31,31,30,31,30,31
340   MAT  READ M
350   M[2]=28+(TIM(3)/4=INT(TIM(3)/4))
360   D=TIM(2)
370   FOR M=1 TO 12
380   IF D <= M[M] THEN 420
390   D=D-M[M]
400   NEXT M
410   D=99
420   CONVERT M TO D$
430   D$[LEN(D$)+1]="/"
440   CONVERT D TO D$[LEN(D$)+1]
450   D$[LEN(D$)+1]="/"
460   CONVERT TIM(3) TO D$[LEN(D$)+1]
470   N2=0
480   PRINT "THE SCORES OF A MAXIMUM OF 24 DIFFERENT FUNDAMENTALS MAY BE PRINTED OUT."
490   PRINT "TYPE 1 FOR A CONTIGUOUS SPREAD OF FUNDAMENTALS, TYPE 2 FOR A SELECTION."
500   PRINT "YOUR CHOICE";
510   INPUT K2
520   GOTO K2 OF 540,620
530   GOTO 490
540   PRINT "WHAT ARE THE BEGINNING AND ENDING FUNDAMENTAL NUMBERS (B,E)";
550   INPUT B2,E2
560   IF B2>E2 OR E2-B2>23 OR B2#INT(B2) OR E2#INT(E2) OR B2<1 OR E2<1 OR B2>99 OR E2>99 THEN 540
570   N2=E2-B2+1
580   FOR I=0 TO N2-1
590   Q[I+1]=B2+I
600   NEXT I
610   GOTO 720
620   PRINT "ENTER SELECTED FUNDAMENTAL NUMBERS; TYPE A ZERO TO STOP."
630   FOR I=1 TO 24
640   PRINT "#"I;
```

Craig Bina

```
650   INPUT F2
660   IF F2=0 THEN 710
670   IF F2#INT(F2) OR F2<1 OR F2>99 THEN 640
680   N2=I
690   Q[I]=F2
700   NEXT I
710   IF  NOT N2 THEN 490
720   PRINT "DO YOU WANT THE MAXIMUM-GRADES-ONLY OPTION";
730   INPUT A$
740   L7=NUM(A$)#NUM("Y")
750   PRINT
760   S3=INT(72/N2)-3
770   PRINT "DO YOU WANT TO LIST INDIVIDUALS";
780   INPUT A$
790   IF A$[1,1]="Y" THEN 1790
795   I3=1
800   PRINT "STARTING HOUR";
810   INPUT H1
820   K1=0
830   PRINT "ENDING HOUR";
840   INPUT H2
850   PRINT "SET TO TOP OF FORM AND PRESS <RETURN>."
860   PPINT LIN(5)
870   LINPUT A$
880   H9=SYS(3)
890   H9=-1
900   GOTO 990
910   PRINT "PERCENT CORRECT ON THE FOLLOWING FUNDAMENTALS:"
920   PRINT
930   FOR X=1 TO N2
940   PRINT SPA(S3);
950   PRINT  USING "#,X2D";Q[X]
960   NEXT X
970   PRINT LIN(1)
980   RETURN
990   IF  END #1 THEN 1400
1000   READ #1;I
1010   I=-I
1020   IF INT(I)>H2 THEN 1770
1030   I3=INT(I)<H1
1040   IF I3 THEN 1270
1050   READ #2,1
1060   IF  END #2 THEN 1130
1070   READ #2;C$,X3,A$
1080   X3=X3/C5-C6
1090   CONVERT X3 TO B$
1100   CONVERT B$ TO X3
1110   IF X3=I THEN 1150
1120   IF X3<I THEN 1070
1130   A$="UN-FILED"
1140   C$="???"
1150   IF INT(I)=H9 THEN 1190
1160   H9=INT(I)
1170   PRINT LIN(3);D$,"PERIOD"H9
1180   GOSUB 910
1190   PRINT  USING 1200;I
1200   IMAGE #,"#",2D.DDX
1210   PRINT C$"      "A$
1220   MAT C=ZER
1230   S=0
```

Craig Bina

```
1240    MAT P=CON
1250    MAT P=(-1)*P
1260    K=0
1270    READ #1;I
1280    IF I <= 0 THEN 1400
1290    IF 13 THEN 1270
1300    J=INT(I/1000)
1310    I=I-J*1000
1320    C[J]=C[J]+1
1330    K=K MAX C[J]
1340    IF C[J]<26 THEN 1380
1350    PRINT "OVERFLOW ERROR IN NO."I"OF"J
1360    PRINT SPA(3*J);
1370    GOTO 1270
1380    P[C[J],J]=I
1390    GOTO 1270
1400    IF I3 THEN 1010
1410    IF S THEN 2220
1420    IF L7 THEN 1500
1430    FOR M=1 TO 99
1440    FOR L=1 TO 25
1450    IF P[L,M]=-1 THEN 1480
1460    P[1,M]=P[1,M] MAX P[L,M]
1470    NEXT L
1480    NEXT M
1490    P[2,1]=-5
1500    FOR L=1 TO K
1510    R5=-1
1520    FOR R=1 TO N2
1530    R5=R5 MAX P[L,Q[R]]
1540    NEXT R
1550    IF R5=-1 THEN 1670
1560    IF P[L,1]=-5 THEN 1680
1570    FOR M2=1 TO N2
1580    M=Q[M2]
1590    IF P[L,M]=-1 THEN 1630
1600    PRINT SPA(S3);
1610    PRINT  USING "#,3D";INT(P[L,M]+.5)
1620    GOTO 1640
1630    PRINT SPA(S3)" * ";
1640    NEXT M2
1650    PRINT
1660    IF SYS(3) THEN 1730
1670    NEXT L
1680    IF K1 THEN 1890
1690    PRINT
1700    IF TYP(1)#3 THEN 1010
1710    PRINT '10"END OF DATA";LIN(5)
1720    GOTO 2230
1730    PRINT "INTERRUPT DETECTED"
1740    IF K1 THEN 1890
1750    PRINT LIN(4)
1760    GOTO 2230
1770    IF S THEN 2220
1780    GOTO 1710
1790    REM   INDIVIDUALS...........
1800    PRINT "INSTRUCTIONS";
1810    INPUT A$
1820    IF A$[1,1]="N" THEN 1880
1830    PRINT "ENTER CODES AS 3 ALPHABETICAL CHARACTERS IN THE FORM: XYZ"
```

Craig Bina

96

```
1840  PRINT "WHERE X IS THE FIRST LETTER OF THE STUDENT'S  P R O P E R"
1850  PRINT "FIRST NAME, AND YZ ARE THE FIRST 2 LETTERS OF THE STUDENT'S"
1860  PRINT "LAST NAME.  EXAMPLE:  DBA  FOR MR. DONALD BAUDER."
1870  PRINT
1880  PRINT "TYPE <RETURN> FOR A CODE WHEN YOU ARE FINISHED."'10
1890  PRINT "CODE: ";
1900  LINPUT A$
1910  IF NUM(A$)=0 THEN 2230
1920  IF LEN(A$)#3 THEN 1890
1930  READ #2,1
1940  IF  END #2 THEN 1970
1950  READ #2;B$,A,B$
1960  GOTO 1990
1970  PRINT "NO SUCH STUDENT CURRENTLY IN EXISTENCE."
1980  GOTO 1890
1990  Z$=B$[1,1]
2000  B=POS(B$," ")+1
2010  Z$[2]=B$[B,B+1]
2020  IF Z$#A$ THEN 1950
2030  K1=1
2040  A=A/C5-C6
2050  CONVERT A TO A0$
2060  CONVERT A0$ TO A
2070  PRINT "#"A" "B$
2080  PRINT "IS THIS THE STUDENT YOU WANT?";
2090  INPUT Z$
2100  IF Z$[1,1]="N" THEN 1950
2110  PRINT A$" #"A;B$
2120  READ #1,1
2130  IF  END #1 THEN 2200
2140  READ #1;I
2150  IF -I#A THEN 2140
2160  GOSUB 910
2170  IF  END #1 THEN 1410
2180  I3=0
2190  GOTO 1220
2200  PRINT "NO FUNDAMENTALS USED"
2210  GOTO 1890
2220  PRINT "NO SCORES HAVE BEEN RECORDED FOR THE SPECIFIED HOUR(S)."
2230  CHAIN R,"MAIN"
2240  PRINT "THE PROGRAM: MAIN  IS MISSING."
2250  SYSTEM R,"BYE"
2260  GOTO 2250
2270  REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
2280  FOR I=0 TO 2
2290  E[I+1]=SYS(I)
2300  NEXT I
2310  CHAIN R,"ERROR"
2320  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
2330  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
2340  GOTO 2230
2350  END
```

Craig Bina

3.15

REMARK

```
10   COM A,A1$[80],B,C,D,00$[11],E[3]
20   0=BRK(0)
30   00$="REMARK"
40   IF  ERROR  THEN 2240
50   FILES *,*,*
60   DIM B$[255],C$[2],T$[2],M[12],V$[255]
70   DIM A$[80]
80   A$=A1$
90   M=0
100   U=1
110   REM TO REVOKE "TALK PROGRAM" STATUS, CHANGE PREVIOUS LINE TO U=0.
120   PRINT '10'10"INTERSCHOOL INSTRUCTOR MESSAGES AND COMMENTS"'10'10
130   FOR X=1 TO 5
140   ASSIGN "MESSAQ",1,R1,"MOLE"
150   SYSTEM R,"PAU-1"
160   IF  NOT R1 THEN 200
170   NEXT X
180   PRINT "FILE:  MESSAQ  IS MISSING, PLEASE INFORM THE APPROPRIATE AUTHORITIES."
190   GOTO 1480
200   LOCK #1
210   SYSTEM R,"PAU-5"
220   LOCK #1,R
230   IF R THEN 270
240   ASSIGN *,1
250   SYSTEM R,"PAU-3"
260   GOTO 120
270   L=INT(LOG(A)/2.30259+3)
280   C$=CHR$(L+64)
290   A6=A/10^L
300   P=INT(10000*(A6-(INT(A6*100)/100))+.9)
310   C$[2]=CHR$(P)
320   IF TYP(1)#2 THEN 360
330   READ #1;B$
340   IF LEN(B$)#2 THEN 360
350   IF NUM(B$)<91 AND NUM(B$)>64 AND NUM(B$[2])<91 AND NUM(B$[2])>64 THEN 430
360   IF  END #1 THEN 400
370   FOR X=1 TO 32767
380   PRINT #1,X
390   NEXT X
400   PRINT #1,1: END
410   PRINT "SORRY, THE AUTHORITIES HAVE NOT YET ENABLED THE MESSAGE FUNCTION."
420   GOTO 1480
430   T$=B$
440   PRINT "MESSAGES FOR TEACHER/SCHOOL: "C$;'10'10'10
450   IF  END #1 THEN 650
460   X5=REC(1)
470   Y5=ITM(1)
480   READ #1;X,B$
490   IF B$[1,2]=C$ THEN 530
500   ADVANCE #1;1,R
510   IF TYP(1)=2 THEN 500
520   GOTO 460
530   Y=POS(B$,"-")
540   PRINT "SENT BY TEACHER/SCHOOL "'34;B$[3,4];'34" ON "B$[5,Y-1]
550   PRINT "FROM: "B$[Y+1];'10
560   READ #1;B$
570   FOR I=1 TO LEN(B$)
580   PRINT B$[I,I];
590   NEXT I
```

Craig Rina

```
600   PRINT
610   IF  TYP(1)=2  THEN 560
620   PRINT  '10'10
630   GOSUB 1530
640   GOTO 450
650   PRINT  '10"END-OF-MESSAGES"'10
660   DATA 31,0,31,30,31,30,31,31,30,31,30,31
670   MAT  READ M
680   M[2]=28+(TIM(3)/4=INT(TIM(3)/4))
690   IF  C$#T$  THEN 730
700   U=1
710   A$="THE AUTHORITIES"
720   GOTO 740
730   PRINT  "THE AUTHORITIES ARE CURRENTLY AT TEACHER/SCHOOL "'34T$'34'10
740   PRINT  "WOULD YOU LIKE A LIST OF USERS";
750   INPUT B$
760   IF  B$[1,1]#"Y"  THEN 890
770   ASSIGN "USERS",3,R,"*"
780   IF R THEN 360
790   LOCK #3,R
800   IF R THEN 790
810   PRINT  '10'10"T/S"TAB(10)"NAME"'10
820   IF   END #3 THEN 870
830   READ #3;B$
840   PRINT B$[1,2];TAB(10);B$[3]
850   GOTO 830
860   PRINT  "UNABLE TO ACCESS FILE:  USERS"
870   ASSIGN *,3,R
880   PRINT
890   PRINT  "DO YOU WISH TO SEND A";
900   IF   NOT (U AND M) THEN 920
910   PRINT  "NOTHER";
920   PRINT  " MESSAGE";
930   INPUT B$
940   IF B$[1,1]="N"  THEN 1470
950   B$=T$
960   IF   NOT U THEN 1010
970   PRINT "SEND MESSAGE TO WHICH TEACHER/SCHOOL";
980   INPUT B$
990   IF LEN(B$)#2 THEN 970
1000   IF NUM(B$)>90 OR NUM(B$)<65 OR NUM(B$[2])>90 OR NUM(B$[2])<65 THEN 970
1010   B$[3]=C$
1020   Y=TIM(2)
1030   Z=TIM(3)
1040   FOR I=1  TO 12
1050   IF  Y  <=  M[I]  THEN 1100
1060   Y=Y-M[I]
1070   NEXT  I
1080   I=1
1090   Z=Z+1
1100   CONVERT  I  TO  B$[5]
1110   B$[LEN(B$)+1]="/"
1120   CONVERT  Y  TO  B$[LEN(B$)+1]
1130   B$[LEN(B$)+1]="/"
1140   CONVERT  Z  TO  B$[LEN(B$)+1]
1150   PRINT  "TO  "'34;B$[1,2];'34" ON "B$[5]" FROM "A$" AT "'34;B$[3,4];'34"."'10
1160   B$[LEN(B$)+1]="-"
1170   B$[LEN(B$)+1]=A$
1180   ADVANCE #1;32767,R
1190   IF   NOT R THEN 1180
```

Craig Bina

```
1200  IF  END #1 THEN 1340
1210  PRINT #1;5,B$
1220  PRINT "BEGIN ENTERING YOUR MESSAGE.  LINES MAY BE UP TO 255 CHARACTERS LONG."
1230  PRINT "PRESS <RETURN> AT THE END OF EACH LINE.  PRESS <RETURN> AT THE BEGINNING"
1240  PRINT "OF A LINE TO COMPLETE YOUR MESSAGE."
1250  Q=0
1260  PRINT "> ";
1270  LINPUT B$
1280  IF  NOT LEN(B$) THEN 1410
1290  Q=1
1300  X7=REC(1)
1310  Y7=ITM(1)
1320  PRINT #1;UPS$(B$)
1330  GOTO 1260
1340  PRINT "PLEASE STAND BY WHILE THE FILE IS EXPANDED."
1350  GOSUB 1830
1360  IF  END #1 THEN 1340
1370  IF  NOT E THEN 1320
1380  PRINT "THE FILE: MESSAQ  IS FULL, AND THERE IS NOT ENOUGH ROOM TO EXPAND IT."
1390  PRINT "THE LAST LINE ENTERED HAS BEEN DISREGARDED."
1400  GOTO 1470
1410  IF  END #1 THEN 1450
1420  IF Q THEN 1440
1430  PRINT #1;" "
1440  PRINT #1; END
1450  M=1
1460  IF U THEN 890
1470  PRINT "END OF MESSAGE MANIPULATION."
1480  CHAIN R,"MAIN"
1490  PRINT "THE PROGRAM:  MAIN  IS MISSING,"
1500  PRINT "PLEASE INFORM THE APPROPRIATE AUTHORITIES."
1510  SYSTEM R,"BYE"
1520  GOTO 1510
1530  REM    ERASING ROUTINE
1540  ASSIGN "MESSAQ",2,R,"MOLE"
1550  READ #2,REC(1)
1560  ADVANCE #2;ITM(1),R
1570  READ #1,X5
1580  ADVANCE #1;Y5,R
1590  F=2
1600  GOSUB 1710
1610  F=1
1620  GOSUB 1670
1630  ASSIGN *,2
1640  READ #1,X5
1650  ADVANCE #1;Y5,R
1660  RETURN
1670  REM  COPY FROM BEGINNING MARK
1680  T=3-F
1690  READ #T,1
1700  READ #F,1
1710  REM COPY FROM RANDOM POINT
1720  T=3-F
1730  GOTO TYP(F)-1 OF 1770,1800
1740  READ #F;Z9
1750  PRINT #T;Z9
1760  GOTO 1730
1770  READ #F;V$
1780  PRINT #T;V$
1790  GOTO 1730
```

Craig Bina

100

```
1800   IF  END #T THEN 1820
1810   PRINT #T; END
1820   RETURN
1830   REM   E X P A N D
1840   E=0
1850   X8=REC(1)
1860   READ #1,X7
1870   ADVANCE #1;Y7,R
1880   IF  END #1 THEN 1900
1890   PRINT #1; END
1900   CREATE R,"DMY103",X8+1
1910   IF R THEN 2040
1920   CREATE R,"BFR103",X8
1930   IF R THEN 2030
1940   ASSIGN "BFR103",2,R,"MOLE"
1950   F=1
1960   GOSUB 1670
1970   ASSIGN *,1
1980   PURGE R,"MESSAQ"
1990   IF  NOT R THEN 2060
2000   ASSIGN "MESSAQ",1,R
2010   UNLOCK #1,R
2020   GOTO 1970
2030   PURGE R,"DMY103"
2040   E=1
2050   RETURN
2060   REM  START CRITICAL
2070   PURGE R,"DMY103"
2080   CREATE R,"MESSAQ",X8+1
2090   ASSIGN "MESSAQ",1,R,"MOLE"
2100   LOCK #1,R
2110   IF  NOT R THEN 2150
2120   UNLOCK #1
2130   SYSTEM R,"PAU-6"
2140   GOTO 2100
2150   V$="MWA-MESSAQ"
2160   SYSTEM R,V$
2170   V$[1,3]="PRI"
2180   SYSTEM R,V$
2190   F=2
2200   GOSUB 1670
2210   ASSIGN *,2
2220   PURGE R,"BFR103"
2230   RETURN
2240   REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
2250   FOR I=0 TO 2
2260   E[I+1]=SYS(I)
2270   NEXT I
2280   CHAIN R,"ERROR"
2290   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
2300   PRINT "PLEASE INFORM YOUR INSTRUCTOR."
2310   GOTO 1480
2320   END
```

Craig Bina

3.16

```
ERASE

10   COM A,NO$[80],B,C,D,OO$[11],E[3]
20   O=BRK(0)
30   OO$="ERASE"
40   IF  ERROR  THEN 1690
50   FILES *,*
60   DIM A$[255],B$[255],C$[255],D$[255],W$[2],Z$[2]
70   DATA "INTL","RECR","RNDM"
80   Q=INT(LOG(A)/2.30259)
90   CONVERT A/10^(Q-1) TO A$
100   CONVERT A$ 10 L
110   L=Q+3* NOT INT(100*(Q-INT(10*Q)/10))
120   Z$=CHR$(L+64)
130   A1=A/10^L
140   Z$[2]=CHR$(INT(10000*(A1-INT(A1*100)/100)+.9))
150   PRINT '10'10"ERASURE OF OBSOLETE PRACTICE SHEET ANSWERS AND DATA FILES"'10'10
160   PRINT "DO YOU WANT TO REMOVE STORED ANSWERS FROM CERTAIN FUNDAMENTALS";
170   INPUT A$
180   IF A$[1,1]="N" THEN 550
190   PRINT "PLEASE ENTER STARTING FUNDAMENTAL # AND ENDING FUNDAMENTAL #."
200   PRINT "EX: 1,24"
210   INPUT X,Y
220   X=INT(X) MAX 1
230   Y=INT(Y) MIN 99
240   PRINT "ENTER STARTING I.D. NUMBER AND ENDING I.D. NUMBER";
250   INPUT M,N
260   M=INT(100*M)
270   N=INT(100*N)
280   IF M>N THEN 240
290   A$="MAIN"
300   FOR I=X TO Y
310   CONVERT I TO A$[5]
320   FOR X8=1 TO 10
330   ASSIGN A$,2,R, WR
340   IF R<6 THEN 390
350   ASSIGN *,1,R
360   ASSIGN *,2
370   SYSTEM R,"PAU-6"
380   GOTO 330
390   IF R=0 THEN 450
400   SYSTEM R,"PAU-2"
410   NEXT X8
420   PRINT "UNABLE TO LOCATE FILE: "A$
430   ASSIGN *,2,R
440   GOTO 540
450   ASSIGN A$,1,R
460   LOCK #1
470   SYSTEM R,"PAU-2"
480   LOCK #1,R
490   IF R=0 THEN 350
500   PRINT "ERASING FROM FILE: "A$
510   GOSUB 1230
520   ASSIGN *,2
530   ASSIGN *,1
540   NEXT I
550   PRINT "DO YOU WANT TO REMOVE YOUR CLASS-LIST FILES FROM THE SYSTEM";
560   INPUT A$
570   IF A$[1,1]="N" THEN 1200
580   PRINT "ARE YOU CERTAIN THAT YOU WISH TO DO SO";
590   INPUT A$
```

Craig Bina

```
600   IF A$[1,1]#"Y" THEN 1200
610   G=1
620   A$="****"
630   A$[5]=Z$
640   IF TYP(0)#2 THEN 760
650   READ B$
660   IF LEN(B$)#4 THEN 640
670   A$[1,4]=B$
680   PURGE R,A$
690   IF R=1 THEN 680
700   IF R THEN 730
710   PRINT "FILE:  "A$"  HAS BEEN SUCCESSFULLY PURGED."
720   GOTO 640
730   PRINT "FILE:  "A$"  IS INACCESSIBLE, PLEASE INFORM THE AUTHORITIES, R ="R
740   G=0
750   GOTO 640
760   ASSIGN "ACCESS",1,R,"CODES"
770   IF  NOT R THEN 810
780   PRINT "UNABLE TO ERASE YOUR INSTRUCTORS' OPTIONS ACCESS CODE."
790   PRINT "PLEASE INFORM THE AUTHORITIES."
800   GOTO 830
810   ASSIGN "ACCESS",2,R,"CODES"
820   GOSUB 910
830   ASSIGN "USERS",1,R,"*"
840   IF  NOT R THEN 880
850   PRINT "UNABLE TO ERASE YOUR NAME FROM THE USER LIST."
860   PRINT "PLEASE INFORM THE AUTHORITIES."
870   GOTO 1080
880   ASSIGN "USERS",2,R,"*"
890   GOSUB 910
900   GOTO 1080
910   REM  DATA REMOVAL ROUTINE
920   LOCK #2
930   IF  END #2 THEN 1070
940   A=REC(2)
950   B=ITM(2)
960   READ #2;A$
970   IF A$[1,2]#Z$ THEN 940
980   READ #1,A
990   ADVANCE #1;B,R
1000   IF  END #2 THEN 1040
1010   READ #2;A$
1020   PRINT #1;A$
1030   GOTO 1010
1040   PRINT #1; END
1050   ASSIGN *,1
1060   ASSIGN *,2
1070   RETURN
1080   REM  DEPARTING MESSAGE FOLLOWS
1090   PRINT '10"SINCE YOU HAVE OPTED TO REMOVE YOURSELF FROM THE SYSTEM,"
1100   PRINT "NO FURTHER ACCESS WILL BE POSSIBLE USING THE TEACHER/SCHOOL"
1110   PRINT "INITIALS "'34Z$'34" AND SO YOU WILL NOW BE LOGGED OFF."
1120   IF G THEN 1160
1130   PRINT "IF ANY OF THE ABOVE-LISTED FILES WERE DESIGNATED AS BEING"
1140   PRINT "INACCESSIBLE, PLEASE RECORD THE PRINTED VALUE OF "'34"R"'34" AND INFORM"
1150   PRINT "THE APPROPRIATE AUTHORITIES SO THAT THEY MAY BE REMOVED."
1160   PRINT "THANK YOU FOR USING THE INSTRUCTIONAL PACKAGE; WE WELCOME ANY"
1170   PRINT "COMMENTS, SUGGESTIONS, OR REQUESTS THAT YOU MIGHT HAVE."
1180   SYSTEM R,"BYE"
1190   GOTO 1180
```

Craig Bina

```
1200  CHAIN R,"MAIN"
1210  PRINT "THE PROGRAM: MAIN  IS MISSING, PLEASE INFORM THE APPROPRIATE AUTHORITIES."
1220  GOTO 1180
1230  REM ID CHECKER AND REMOVAL SUBROUTINE
1240  IF  END #1 THEN 1660
1250  READ #1,1;B
1260  C=B >= 0
1270  D=INT(ABS(B))
1280  READ #2,1
1290  IF  NOT C THEN 1350
1300  PRINT #2;B
1310  GOTO 1350
1320  IF C THEN 1350
1330  READ #1;B
1340  D=INT(ABS(B))
1350  READ #1;E,S
1360  IF I#3 THEN 1390
1370  READ #1;C$,D$,I1,T,I2,Y1
1380  D=INT((T-I1)/I2)+1
1390  L=INT(LOG(E)/2.30259)
1400  CONVERT E/10^(L-1) TO B$
1410  CONVERT B$ TO Q
1420  L=L+3* NOT INT(100*(Q-INT(Q*10)/10))
1430  W$=CHR$(L+64)
1440  L=E/10^L
1450  Q=INT(10000*(L-INT(L*100)/100)+.9)
1460  W$[2]=CHR$(Q)
1470  IF W$#7$ THEN 1520
1480  L=INT(100*(L-Q/10000))
1490  IF L<M OR L>N THEN 1520
1500  ADVANCE #1;D,R
1510  GOTO 1320
1520  IF C THEN 1540
1530  PRINT #2;B
1540  PRINT #2;E,S
1550  IF I#3 THEN 1570
1560  PRINT #2;C$,D$,I1,T,I2,Y1
1570  FOR J=1 TO D
1580  GOTO TYP(1)-1 OF 1620,1660
1590  READ #1;N7
1600  PRINT #2;N7
1610  GOTO 1640
1620  READ #1;B$
1630  PRINT #2;B$
1640  NEXT J
1650  GOTO 1320
1660  IF  END #2 THEN 1680
1670  PRINT #2; END
1680  RETURN
1690  REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
1700  FOR I=0 TO 2
1710  E[I+1]=SYS(I)
1720  NEXT I
1730  CHAIN R,"ERROR"
1740  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
1750  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
1760  GOTO 1200
1770  END
```

Craig Bina

3.17

```
 RNDFND

 10   COM I6,N1$[80],F,X7,Y7,00$[11],E[3]
 20   COM L
 25   REM**************************NEVER RENUMBER THIS PROGRAM****************
 30   Y=BRK(0)
 40   00$="RNDFND"
 50   IF  ERROR  THEN 2070
 60   FILES *,*
 70   DIM A1$[15]
 80   DIM A$[80],B$[80]
 85   DIM Y[367]
 90   X7=400
 100   Y7=I6
 110  PRINT '10'10"NON-SEQUENTIAL PRACTICE SHEET GENERATION"'10'10
 120  F=R=0
 130  GOSUB 1960
 140  IF  NOT R THEN 200
 150  PRINT "FILE:  "A1$"  IS MISSING.  PLEASE INFORM YOUR INSTRUCTOR."
 160  CHAIN R,"MAIN"
 170  PRINT "PROGRAM: MAIN  IS MISSING.  PLEASE INFORM YOUR INSTRUCTOR."
 180  SYSTEM R,"BYE"
 190  GOTO 180
 200  PRINT "TYPE 1 TO GENERATE ONLY STUDENT REQUESTS, 2 TO ADD TO THE STUDENT"
 210  PRINT "REQUESTS, OR 3 TO ERASE THE STUDENT REQUESTS AND ENTER YOUR OWN."
 220  PRINT "WHAT IS YOUR CHOICE";
 230  INPUT C6
 240  LOCK #1,R
 250  IF R THEN 240
 260  GOTO C6 OF 680,290,290
 270  UNLOCK #1
 280  GOTO 200
 290  PRINT "DO YOU WANT INSTRUCTIONS";
 300  INPUT A$
 310  IF A$[1,1]="N" THEN 400
 320  PRINT "ENTER PRACTICE SHEETS TO BE GENERATED IN THE FOLLOWING FORMAT:  FML#"
 330  PRINT "WHERE F, M, L, AND # ARE AS FOLLOWS:"
 340  PRINT "F-STUDENT'S FIRST INITIAL"
 350  PRINT "M-STUDENT'S MIDDLE INITIAL"
 360  PRINT "L-STUDENT'S LAST INITIAL"
 370  PRINT "#-THE FUNDAMENTAL FOR WHICH THE PRACTICE SHEET IS TO BE GENERATED"
 380  PRINT "EXAMPLE:  TO GENERATE A FUNDAMENTAL 5 PRACTICE SHEET FOR STUDENT  CRB ,"
 390  PRINT "TYPE THE FOLLOWING;               CRB5    "
 400  PRINT "TYPE  COMPLETE  WHEN YOU ARE FINISHED."'10
 410  IF INT(C6)=3 THEN 460
 420  ADVANCE #1;256,R
 430  IF  NOT R THEN 420
 440  PRINT "ADDING TO STUDENT REQUESTS"
 450  GOTO 500
 460  PRINT "ERASING STUDENT REQUESTS AND ENTERING ONLY YOUR REQUESTS"
 470  PRINT "ARE YOU ABSOLUTELY SURE THAT YOU WANT TO DO THIS??";
 480  INPUT A$
 490  IF A$[1,1]#"Y" THEN 200
 500  IF  END #1 THEN 620
 510  PRINT "> ";
 520  LINPUT A$
 530  IF A$="COMPLETE" THEN 660
 540  IF LEN(A$)<4 OR LEN(A$)>5 THEN 640
 550  FOR I=1 TO 3
 560  IF NUM(A$[I])<65 OR NUM(A$[I])>90 THEN 640
 570  NEXT I
```

Craig Bina

```
580  CONVERT A$[4] TO I,640
590  IF I<1 THEN 640
600  PRINT #1;A$
610  GOTO 510
620  PRINT "FILE:  "A1$"  IS FULL, LAST ENTRY WAS IGNORED."
630  GOTO 670
640  PRINT "ILLEGAL FORMAT"
650  GOTO 510
660  PRINT #1; END
670  READ #1,1
680  PRINT "PRESS <RETURN> TO BEGIN PROCESSING"'10
690  ENTER 255,R,A$
700  IF R<0 THEN 690
710  A$="MES-AUTOMATIC FUNDAMENTAL GENERATION IN PROGRESS:"
720  SYSTEM R,A$
730  IF R THEN 720
740  A$[5]="PLEASE DO NOT SEND ANY MESSAGES TO W140 ON PORT #"'1
750  ENTER #07
760  CONVERT Q7 TO A$[POS(A$,'1)]
770  SYSTEM R,A$
780  IF R THEN 770
790  E[1]=0
800  REM CHAIN HERE FOR A RETURN!
810  O0$="RNDFND"
820  Y=BRK(0)
830  IF  ERROR  THEN 2070
840  REM  K6 IS THE NUMBER OF LINES PER PAGE FOR DESIGNATED DEVICE
850  REM MINUS 5 WHICH IS USED FOR HEADER SPACE
860  K6=50-5
870  RESTORE 910
880  N1=0
890  REM ..... MOVING DOWN THE PAGE????  THEN
900  REM .......... INCREASE THE ABSOLUTE VALUE IN THE DATA STATEMENT
910  DATA 22,46,-13,22
920  DATA -19,18,-14,-12
930  DATA 29,-16,25,20,-20
940  DATA 20
950  DATA 28
960  DATA 20,20,20,17,-19,21
970  DATA 21,19,26
980  DATA 41,0,0,0,0
990  DATA 0,0,0,0,0
1000  DATA 0,0,0,0,0
1010  DATA 0,0,0,0,0
1020  DATA 0,0,0,0,0
1030  DATA 54,54,54,54
1040  DATA 54,54,54,54
1050  FOR X2=1 TO F
1060  IF TYP(0)=3 THEN 1120
1070  READ N
1080  NEXT X2
1090  REM   ACCOUNT FOR TWO FREE LINES DONATED BY "NOTICE"
1100  N=N+2*SGN(N)
1110  GOTO 1130
1120  N=0
1130  IF N >= 0 THEN 1150
1140  N=-N+L
1150  IF N <= K6 THEN 1190
1160  N=N-K6
1170  N1=5
```

Craig Bina

106

```
1180  GOTO 1150
1190  PRINT LIN(K6-N+N1)
1200  PRINT LIN(5)
1210  GOSUB 1960
1220  IF R THEN 150
1230  LOCK #1,R
1240  IF R THEN 1230
1250  E[1]=E[1]+1
1260  READ #1,1
1270  IF  END #1 THEN 1810
1280  FOR I=1 TO E[1]
1290  V1=REC(1)
1300  V2=ITM(1)
1310  READ #1;A$
1320  NEXT I
1330  IF NUM(A$)=127 THEN 1250
1340  B$=A$
1350  B$[1,1]='127
1360  READ #1,V1
1370  ADVANCE #1;V2,R
1380  UPDATE #1;B$
1390  B$="INTL"
1400  B$[5]=A1$[5]
1410  ASSIGN B$,2,R
1420  IF R<3 THEN 1450
1430  PRINT "FILE: "B$"  IS INACCESSIBLE.  PLEASE INFORM YOUR INSTRUCTOR."
1440  GOTO 1830
1450  IF  END #2 THEN 1790
1460  READ #2;B$,I6,N1$
1470  IF B$#A$[1,3] THEN 1460
1480  B$="FUND"
1490  B$[5]=A$[4]
1500  CONVERT B$[5] TO F
1520  ASSIGN "USAGE",1,R
1530  IF R THEN 1730
1531  LOCK #1
1532  Y6=TIM(3)-1+100* NOT TIM(3)
1533  Z6=Y6-1+100* NOT Y6
1534  W6=TIM(2) MIN 366
1535  IF  END #1 THEN 1539
1536  MAT  READ #1;Y
1537  X6=(Y[1]=Z6)+2*(Y[1]=Y6)
1538  GOTO X6 OF 1545,1552
1539  IF  END #1 THEN 1730
1540  MAT Y=ZER
1541  Y[1]=Y6
1542  READ #1,1
1543  MAT  PRINT #1;Y, END
1544  GOTO 1548
1545  READ #1,4
1546  MAT  READ #1;Y
1547  GOTO (Y[1]=Y6)+1 OF 1539,1542
1548  MAT Y=ZER
1549  Y[1]=TIM(3)
1550  READ #1,4
1551  MAT  PRINT #1;Y, END
1552  READ #1,4
1553  ADVANCE #1;W6,R
1554  READ #1;X6
1555  READ #1,4
```

Craig Bina

107

```
1556  ADVANCE #1;W6,R
1557  UPDATE #1;X6+1
1730  ASSIGN *,1,R
1740  CHAIN R,B$
1750  PRINT "PROGRAM: "B$"  IS INACCESSIBLE.  PLEASE INFORM YOUR INSTRUCTOR."
1760  PRINT "INSTRUCTION: "A$"  HAS BEEN SKIPPED"
1770  PRINT LIN(K6-2);
1780  GOTO 1200
1790  PRINT "STUDENT: "A$[1,3]"  NOT FOUND."
1800  GOTO 1760
1810  PRINT LIN(10)"J O B   I S   F I N I S H E D "
1820  PRINT #1,1; END
1830  A$="MES-FUNDAMENTAL PROCESSING COMPLETED.  THANK YOU FOR YOU COOPERATION."
1840  SYSTEM R,A$
1850  IF R THEN 1840
1860  A$[5]="MESSAGES NOW ACCEPTED BY W140 ON PORT #"'1
1870  ENTER #Q7
1880  CONVERT Q7 TO A$[POS(A$,'1)]
1890  SYSTEM R,A$
1900  IF R THEN 1890
1910  UNLOCK #1
1920  PRINT LIN(3)
1921  I6=Y7
1922  GOSUB 1960
1923  A1$[1,4]="INTL"
1924  ASSIGN A1$,1,R
1925  IF  NOT R THEN 1928
1926  I6=-1
1927  GOTO 160
1928  IF  END #1 THEN 1926
1929  CONVERT I6 TO A$
1930  READ #1;B$,I1,N1$
1931  CONVERT I1 TO B$
1932  IF B$#A$ THEN 1930
1933  GOTO 160
1950  GOTO 160
1960  REM  TEACHER/SCHOOL ROUTINE
1970  L=INT(LOG(I6)/2.30259)
1980  CONVERT I6/10^(L-1) TO A1$
1990  CONVERT A1$ TO Q
2000  A1S="RNDM"
2010  Q=L+3* NOT INT(100*(Q-INT(10*Q)/10))
2020  A1$[5]=CHR$(Q+64)
2030  I2=I6/10^Q
2040  A1$[6]=CHR$(INT(10000*(I2-INT(I2*100)/100)+.9))
2050  ASSIGN A1$,1,R
2060  RETURN
2070  REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
2080  FOR I=0 TO 2
2090  E[I+1]=SYS(I)
2100  NEXT I
2110  CHAIN R,"ERROR"
2120  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
2130  PRINT "PLEASE INFORM YOUR INSTRUCTOR."
2140  GOTO 160
2150  END
```

Craig Bina

3.18

LGLIZE

```
10    COM I7,N0$[80],F9,D0,D1,O0$[11],E[3]
20    O0$="LGLIZE"
30    O=BRK(0)
40    IF  ERROR  THEN 3300
50    FILES *,*
60    DIM A$[255],B$[6],D$[6],I$[15],N$[80],T$[15],V$[15],Z$[2]
70    PRINT '10'10"LOCATION AND LEGALIZATION OF STUDENT CHEATERS"'10'10
80    L=INT(LOG(I7)/2.30259)
90    T=I7/10^(L-1)
100    CONVERT T TO V$
110    CONVERT V$ TO T
120    L=L+3* NOT INT(100*(T-INT(10*T)/10))
130    J=I7/10^L
140    M=INT(10000*(J-INT(J*100)/100)+.9)
150    Z$=CHR$(L+64)
160    Z$[2]=CHR$(M)
170    D$="MAIN"
180    B$="INTL"
190    IF F9 THEN 410
200    Z$="$$"
210    PRINT "DO YOU WANT AN EXPLANATION OF THE FUNCTIONS OF THIS PROGRAM";
220    INPUT A$
230    IF A$[1,1]="N" THEN 410
240    PRINT "WHEN A STUDENT DISCONNECTS FROM THE COMPUTER WHILE IN THE MIDST OF"
250    PRINT "CHECKING A FUNDAMENTAL PRACTICE SHEET, HIS/HER INITIALS ARE RECORDED IN"
260    PRINT "A FILE KNOWN AS "'34"THE CHEATBOX."'34"  HENCEFORTH HE/SHE IS PROHIBITED"
270    PRINT "FROM USING THE INSTRUCTIONAL PACKAGE UNTIL "'34"LEGALIZED"'34" BY"
280    PRINT "HIS/HER INSTRUCTOR.  AT THE SAME TIME, THE STUDENT'S PRACTICE SHEET IS"
290    PRINT "RENDERED TEMPORARILY INACCESSIBLE; THIS IS WHAT IS KNOWN AS AN"
300    PRINT "INCOMPLETELY CHECKED FUNDAMENTAL PRACTICE SHEET, OR AN "'34"INCOMPLETE"
310    PRINT "PRACTICE SHEET."'34
320    PRINT "PART ONE OF THIS PROGRAM REVEALS THE EXISTENCE OF INCOMPLETE PRACTICE"
330    PRINT "SHEETS, ALONG WITH LISTING ALL ANSWERS TO THE PRACTICE PROBLEMS."
340    PRINT "PART TWO LISTS CURRENT OCCUPANTS OF THE CHEATBOX, TO WHOM ALL"
350    PRINT "ACCESS IS FORBIDDEN."
360    PRINT "PART THREE ALTERS THE STATUS OF INCOMPLETE PRACTICE SHEETS."
370    PRINT "THE "'34"REVIVE"'34" OPERATION RETURNS THE SHEET TO NORMAL STATUS."
380    PRINT "THE "'34"DESTROY"'34" OPERATION PERMANENTLY ERASES THE SHEET."
390    PRINT "PART FOUR REMOVES USERS FROM THE CHEATBOX, THUS PERMITTING THEM"
400    PRINT "FURTHER ACCESS IN THE FUTURE."
410    PRINT '10"DO YOU WISH TO LIST INCOMPLETE PRACTICE SHEETS";
420    INPUT A$
430    IF A$[1,1]="N" THEN 990
440    PRINT "START SEARCH AT WHICH FUNDAMENTAL NUMBER";
450    INPUT A
460    PRINT "END SEARCH AT WHAT FUNDAMENTAL NUMBER";
470    INPUT B
480    PRINT LIN(5)"INCOMPLETE PRACTICE SHEETS FOR TEACHER/SCHOOL "Z$":"'10'10
490    PRINT
500    FOR F=INT(A MIN 99 MAX 1) TO INT(B MIN 99 MAX 1)
510    D7=0
520    PRINT "--FUNDAMENTAL"F"      ";
530    CONVERT F TO D$[5]
540    R8=0
550    GOSUB 3120
560    IF R8 THEN 960
570    B=0
580    GOSUB 2680
590    B=1
```

Craig Bina

```
600  IF S<0 THEN 940
610  READ #1,X
620  ADVANCE #1;Y+2+ NOT Q1,R
630  GOSUB 2420
640  IF  NOT K THEN 730
650  IF K<0 THEN 920
660  PRINT LIN( NOT D7)
670  PRINT A$;TAB(25);I$"    PERIOD"INT(J)"SHEET #"S;
680  IF F9 THEN 700
690  PRINT "T/S = "B$[5];
700  PRINT '13'10'10"ANSWERS:"
710  D7=1
720  GOTO 760
730  A$="NAME UNKNOWN"
740  I$="???"
750  GOTO 660
760  G=1
770  FOR K=1 TO Q
780  GOTO TYP(1)-1 OF 830
790  READ #1;X
800  PRINT X;
810  G=0
820  GOTO 880
830  READ #1;A$
840  IF G THEN 860
850  PRINT
860  G=1
870  PRINT A$
880  NEXT K
890  PRINT LIN( NOT G);
900  ASSIGN *,2
910  GOTO 580
920  ADVANCE #1;Q,R
930  GOTO 900
940  IF D7 THEN 960
950  PRINT "OKAY"
960  PRINT
970  NEXT F
980  PRINT LIN(5)
990  PRINT "DO YOU WISH TO LIST THE OCCUPANTS OF THE CHEATBOX";
1000  INPUT A$
1010  IF A$[1,1]="N" THEN 1250
1020  ASSIGN "CHTBOX",1,R
1030  IF R<3 THEN 1060
1040  PRINT '10"SORRY, THE FILE:  CHTBOX  IS MISSING."'10
1050  GOTO 1250
1060  LOCK #1
1070  IF  END #1 THEN 1220
1080  PRINT LIN(5)"CURRENT CHEATBOX OCCUPANTS FOR TEACHER/SCHOOL "Z$":"'10'10
1090  D8=0
1100  READ #1;J,F
1110  GOSUB 2420
1120  IF K<0 THEN 1100
1130  IF K THEN 1160
1140  A$="NAME UNKNOWN"
1150  I$="???"
1160  PRINT A$;TAB(25);I$"    PERIOD"INT(J)"FUNDAMENTAL #"F;
1170  IF F9 THEN 1190
1180  PRINT "T/S = "B$[5];
1190  PRINT
```

Craig Bina

110

```
1200  D8=1
1210  GOTO 1100
1220  IF D8 THEN 1240
1230  PRINT "N O N E"
1240  PRINT LIN(3)
1250  PRINT "DO YOU WISH TO ALTER THE STATUS OF INCOMPLETE PRACTICE SHEETS";
1260  INPUT A$
1270  IF A$[1,1]="N" THEN 1810
1280  PRINT "      DO YOU WANT INSTRUCTIONS";
1290  INPUT A$
1300  IF A$[1,1]="N" THEN 1410
1310  PRINT "ENTER REQUESTS IN THE FOLLOWING FORMAT:  FMLA#,%"
1320  PRINT "WHERE F, M, L, #, AND % ARE AS FOLLOWS:"
1330  PRINT "F-STUDENT'S FIRST INITIAL"
1340  PRINT "M-STUDENT'S MIDDLE INITIAL"
1350  PRINT "L-STUDENT'S LAST INITIAL"
1360  PRINT "A-ALTERATION TO BE PERFORMED (R=REVIVE, D=DESTROY)"
1370  PRINT "#-FUNDAMENTAL NUMBER (1-99)"
1380  PRINT "%-SHEET NUMBER (1-INFINITY)"
1390  PRINT "EXAMPLE:  TO REVIVE PRACTICE SHEET #11 FOR FUNDAMENTAL #9"
1400  PRINT "FOR STUDENT "'34"CRB"'34" TYPE:  CRBR9,11"
1410  PRINT "TYPE  STOP  WHEN YOU ARE FINISHED WITH ALL ALTERATIONS."
1420  PRINT "> ";
1430  LINPUT A$
1440  IF A$="STOP" THEN 1810
1450  IF LEN(A$)<7 THEN 1740
1460  T$=A$[1,3]
1470  C=POS(A$,",")
1480  IF C<6 OR C>LEN(A$)-1 THEN 1740
1490  V$="RD"
1500  A=POS(V$,A$[4,4])
1510  IF  NOT A THEN 1740
1520  CONVERT A$[5,C-1] TO F,1740
1530  CONVERT A$[C+1] TO S2,1740
1540  GOSUB 2930
1550  CONVERT F TO D$[5]
1560  IF I=0 THEN 1420
1570  IF I<0 THEN 1760
1580  R8=0
1590  GOSUB 3120
1600  IF R8 THEN 1420
1610  CONVERT I TO V$
1620  B=0
1630  GOSUB 2680
1640  B=1
1650  IF S<0 THEN 1780
1660  IF S#S2 THEN 1630
1670  CONVERT I TO I$
1680  IF I$#V$ THEN 1630
1690  READ #1,X
1700  ADVANCE #1;Y+2-Q1,R
1710  UPDATE #1;(A#2)*INT(S)
1720  ASSIGN *,1
1730  GOTO 1420
1740  PRINT "INVALID FORMAT"
1750  GOTO 1420
1760  PRINT "STUDENT:  "T$"  DOES NOT EXIST."
1770  GOTO 1420
1780  PRINT "STUDENT:  "T$"  DOES NOT HAVE AN INCOMPLETE SHEET #"S2
1790  PRINT "FOR FUNDAMENTAL #"F"."
```

Craig Bina

```
1800  GOTO 1420
1810  PRINT "DO YOU WISH TO REMOVE USERS FROM THE CHEATBOX";
1820  INPUT A$
1830  IF A$[1,1]="N" THEN 2370
1840  PRINT "      DO YOU WANT INSTRUCTIONS";
1850  INPUT A$
1860  IF A$[1,1]="N" THEN 1950
1870  PRINT "ENTER REQUESTS IN THE FOLLOWING FORMAT:  FML#"
1880  PRINT "WHERE F, M, L, AND # ARE AS FOLLOWS:"
1890  PRINT "F-STUDENT'S FIRST INITIAL"
1900  PRINT "M-STUDENT'S MIDDLE INITIAL"
1910  PRINT "L-STUDENT'S LAST INITIAL"
1920  PRINT "#-FUNDAMENTAL NUMBER (1-99)"
1930  PRINT "EXAMPLE:  TO REMOVE STUDENT "'34"CRB"'34" FROM THE CHEATBOX"
1940  PRINT "FOR FUNDAMENTAL #12 TYPE:  CRB12"
1950  PRINT "TYPE  STOP  WHEN YOU ARE FINISHED WITH ALL ALTERATIONS."
1960  PRINT "> ";
1970  LINPUT A$
1980  IF A$="STOP" THEN 2370
1990  IF LEN(A$)<4 THEN 2310
2000  T$=A$[1,3]
2010  CONVERT A$[4] TO F,2310
2020  GOSUB 2930
2030  IF I=0 THEN 1960
2040  IF I<0 THEN 2330
2050  ASSIGN "CHTBOX",1,R
2060  IF R THEN 2080
2070  IF  NOT R THEN 2100
2080  PRINT '10"SORRY, THE FILE:  CHTBOX  IS MISSING."'10
2090  GOTO 2370
2100  LOCK #1
2110  ASSIGN "CHTBOX",2,R
2120  IF R THEN 2080
2130  IF  END #1 THEN 2350
2140  CONVERT I TO V$
2150  X=REC(1)
2160  Y=ITM(1)
2170  READ #1;A,B
2180  IF B#F THEN 2150
2190  CONVERT A TO I$
2200  IF I$#V$ THEN 2150
2210  READ #2,X
2220  ADVANCE #2;Y,R
2230  IF  END #1 THEN 2270
2240  READ #1;A,B
2250  PRINT #2;A,B
2260  GOTO 2240
2270  PRINT #2; END
2280  ASSIGN *,2
2290  ASSIGN *,1
2300  GOTO 1960
2310  PRINT "INVALID FORMAT"
2320  GOTO 1960
2330  PRINT "STUDENT:  "T$"  DOES NOT EXIST."
2340  GOTO 1960
2350  PRINT "STUDENT:  "T$"  IS NOT IN THE CHEATBOX FOR FUNDAMENTAL #"F"."
2360  GOTO 1960
2370  PRINT LIN(5)
2380  CHAIN R,"MAIN"
2390  PRINT "THE PROGRAM:  MAIN  IS MISSING, PLEASE INFORM YOUR INSTRUCTOR."
```

Craig Bina

```
2400  SYSTEM R,"BYE"
2410  GOTO 2400
2420  REM  I.D. NUMBER IDENTIFICATION  ROUTINE
2430  K=1
2440  L=INT(LOG(I)/2.30259)
2450  T=I/10^(L-1)
2460  CONVERT T TO V$
2470  CONVERT V$ TO T
2480  L=L+3* NOT INT(100*(T-INT(10*T)/10))
2490  J=I/10^L
2500  M=INT(10000*(J-INT(J*100)/100)+.9)
2510  B$[5]=CHR$(L+64)
2520  B$[6]=CHR$(M)
2530  IF  NOT F9 THEN 2570
2540  IF B$[5]=Z$ THEN 2570
2550  K=-1
2560  GOTO 2670
2570  ASSIGN B$,2,R
2580  IF R<3 THEN 2620
2590  PRINT '10"SORRY, FILE:  "B$"  IS MISSING."'10
2600  K=0
2610  GOTO 2670
2620  CONVERT I TO T$
2630  IF  END #2 THEN 2600
2640  READ #2;I$,K,A$
2650  CONVERT K TO V$
2660  IF V$#T$ THEN 2640
2670  RETURN
2680  REM  PRACTICE SHEET DATA SEARCH ROUTINE
2690  IF  END #1 THEN 2890
2700  IF B THEN 2760
2710  READ #1,1;Q
2720  Q1=Q >= 0
2730  Q=INT(ABS(Q))
2740  IF Q1 THEN 2760
2750  READ #1,1
2760  X=REC(1)
2770  Y=ITM(1)
2780  IF Q1 THEN 2810
2790  READ #1;Q
2800  Q=INT(ABS(Q))
2810  READ #1;I,S
2820  IF F#3 THEN 2860
2830  ADVANCE #1;2,R
2840  READ #1;I1,T,I2,Y1
2850  Q=INT((T-I1)/I2)+1
2860  ADVANCE #1;Q,R
2870  IF S#INT(S) THEN 2910
2880  GOTO 2760
2890  S=-1
2900  GOTO 2920
2910  S=INT(ABS(S))
2920  RETURN
2930  REM  THREE INITIALS IDENTIFICATION ROUTINE
2940  IF F9 THEN 2990
2950  PRINT "WHAT IS TEACHER/SCHOOL";
2960  INPUT A$
2970  IF LEN(A$)#2 THEN 2950
2980  Z$=A$
2990  B$[5]=Z$
```

Craig Bina

```
3000   ASSIGN B$,2,R
3010   IF R<3 THEN 3070
3020   PRINT '10"SORRY, FILE:  "B$"  IS MISSING."'10
3030   I=0
3040   GOTO 3100
3050   I=-1
3060   GOTO 3100
3070   IF  END #2 THEN 3050
3080   READ #2;I$,I,N$
3090   IF I$#T$ THEN 3080
3100   ASSIGN *,2
3110   RETURN
3120   REM  LOCKS ROUTINE
3130   FOR X8=1 TO 10
3140   ASSIGN D$,1,R, WR
3150   IF R<6 THEN 3190
3160   ASSIGN *,1
3170   SYSTEM R,"PAU-6"
3180   GOTO 3140
3190   IF R=0 THEN 3250
3200   SYSTEM R,"PAU-2"
3210   NEXT X8
3220   PRINT "FILE:  "D$"  IS MISSING."
3230   R8=1
3240   GOTO 3290
3250   LOCK #1
3260   SYSTEM R,"PAU-2"
3270   LOCK #1,R
3280   IF R=0 THEN 3160
3290   RETURN
3300   REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
3310   FOR I=0 TO 2
3320   E[I+1]=SYS(I)
3330   NEXT I
3340   CHAIN R,"ERROR"
3350   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
3360   PRINT "PLEASE INFORM YOUR INSTRUCTOR."
3370   GOTO 2380
3380   END
```

Craig Bina

3.19

REPORT

```
10   COM I7,N0$[80],F9,D0,D1,00$[11],E[3]
20   00$="REPORT"
30   0=BRK(0)
40   IF  ERROR  THEN 650
50   PRINT '10'10"INSTRUCTIONAL PACKAGE USAGE REPORT"'10'10
60   FILES USAGE
70   DIM M[12],A$[36],B$[3]
80   DATA 31,0,31,30,31,30,31,31,30,31,30,31
90   DEF FNL(X)=28+(INT(X/4)=X/4)
100   IMAGE 3AX,DD,7X6D
110   IMAGE /"DATA AVAILABLE ONLY FOR YEARS",XDDX,"AND",XDD/
120   MAT  READ M
130   AS="JANFEBMARAPRMAYJUNJULAUGSEPOCTNOVDEC"
140   IF  END #1 THEN 600
150   READ #1;Y
160   Y1=(Y+1)*(Y#99)
170   PRINT "ENTER INITIAL REPORT DATE (EXAMPLE:  7,4,61)";
180   INPUT M,D,V
190   IF M#INT(M) OR M<1 OR M>12 OR D#INT(D) OR D<1 OR D>31 THEN 170
200   IF V=Y OR V=Y1 THEN 230
210   PRINT  USING 110;Y,Y1
220   GOTO 170
230   PRINT "ENTER TERMINAL REPORT DATE (EXAMPLE:  8,8,62)";
240   INPUT M2,D2,V2
250   IF M2#INT(M2) OR M2<1 OR M2>12 OR D2#INT(D2) OR D2<1 OR D2>31 THEN 230
260   IF V2=Y OR V2=Y1 THEN 290
270   PRINT  USING 110;Y,Y1
280   GOTO 230
290   IF V2+100* NOT V2<V OR V2=V AND (M2<M OR M2=M AND D2<D) THEN 170
300   M[2]=FNL(V)
310   D=D MIN M[M]
320   M[2]=FNL(V2)
330   D2=D2 MIN M[M2]
340   A=0
350   M[2]=FNL(V)
360   READ #1,1+3*(V=Y1)
370   ADVANCE #1;1,R
380   FOR I=1 TO M-1
390   ADVANCE #1;M[I],R
400   NEXT I
410   ADVANCE #1;D-1,R
420   PRINT LIN(2)
425   M3=M2+12*(V#V2)
430   FOR I1=M TO M3
440   I=I1-12*(I1>12)
450   IF I1#13 THEN 490
460   M[2]=FNL(V2)
470   READ #1,4
480   ADVANCE #1;1,R
490   BS=A$[3*I-2,3*I]
500   FOR J=(1-D)*(I#M)+D TO (M[I]-D2)*(I1#M3)+D2
510   READ #1;X
520   PRINT  USING 100;B$,J,X
530   A=A+X
540   NEXT J
550   NEXT I1
560   PRINT LIN(5)
570   PRINT "TOTAL NUMBER OF INITIALS ENTERED IN THIS TIME PERIOD IS";A
580   PRINT LIN(5)
```

Craig Bina

```
590   GOTO 610
600   PRINT "PREMATURE END-OF-FILE ENCOUNTERED"
610   CHAIN R,"MAIN"
620   PRINT "THE PROGRAM:  MAIN  IS MISSING.  PLEASE INFORM YOUR INSTRUCTOR."
630   SYSTEM R,"BYE"
640   GOTO 630
650   REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
660   FOR I=0 TO 2
670   E[I+1]=SYS(I)
680   NEXT I
690   CHAIN R,"ERROR"
700   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
710   PRINT "PLEASE INFORM YOUR INSTRUCTOR."
720   GOTO 610
730   END
```

Craig Bina

3.20

INSERT

```
10   COM I6,Q0$[80],F9,X6,Y6,O0$[11],E[3]
20   O0$="INSERT"
30   IF  ERROR  THEN 430
40   O=BRK(0)
50   DIM A$[15]
60   E[1]=0
70   IF I6=-1 THEN 200
80   PRINT '10'10"INSERTION OF HAND-SCORED PRACTICE SHEET GRADES"'10'10
90   PRINT "ENTER I.D. NUMBER OF STUDENT WHOSE SCORE IS"
100   PRINT "TO BE INSERTED AFTER EACH "'34"?"'34" IN THE FORM:  D.DD"
110   PRINT "WHERE EACH  D  IS A DECIMAL DIGIT."
120   PRINT "EXAMPLE:  I.D.?2.03"
130   PRINT "TYPE  -1  FOR AN I.D. NUMBER WHEN YOU HAVE FINISHED ENTERING SCORES."'10
140   L=INT(LOG(I6)/2.30259)
150   CONVERT I6/10^(L-1) TO A$
160   CONVERT A$ TO Q
170   E[2]=10^(L+3* NOT INT(100*(Q-INT(10*Q)/10))-2)
180   Q=I6/E[2]/100
190   E[3]=INT(10000*(Q-INT(100*Q)/100)+.9)/100
200   PRINT
210   PRINT "I.D.";
220   INPUT I6
230   IF I6=-1 THEN 410
240   IF I6<0 OR I6>9.99 THEN 210
250   I6=(INT(I6*100+.5)+E[3])*E[2]
260   PRINT "FUNDAMENTAL NUMBER";
270   INPUT F9
280   IF F9<1 OR F9>99 OR F9#INT(F9) THEN 260
290   PRINT "SCORE RECEIVED (PERCENTAGE)";
300   INPUT Y6
310   IF Y6<0 OR Y6>100 THEN 290
320   CHAIN R,"UPDATE"
330   PRINT "THE PROGRAM:  UPDATE  IS MISSING."
340   PRINT "PLEASE INFORM THE AUTHORITIES."
350   I6=E[3]*E[2]
360   CHAIN R,"MAIN"
370   PRINT "THE PROGRAM:  MAIN  IS MISSING."
380   PRINT "PLEASE INFORM THE AUTHORITIES."
390   SYSTEM R,"BYE"
400   GOTO 390
410   PRINT '10"SCORE INSERTION COMPLETED"
420   GOTO 350
430   REM  ERROR TRAP DESTINATION
440   FOR I=0 TO 2
450   E[I+1]=SYS(I)
460   NEXT I
470   CHAIN R,"ERROR"
480   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM:  ERROR  IS MISSING."
490   GOTO 380
500   END
```

Craig Bina

117

3.21

MASLPR

```
10   COM I9,N1$[80],F9,X9,Y9,00$[11],E[3]
20   00$="MASLPR"
30   Q=BRK(0)
40   IF  ERROR  THEN 480
50   FILES *
60   DIM A$[80],B$[80]
70   PRINT '10'10"FILING OF REQUESTS FOR REMOTE FUNDAMENTAL GENERATION"'10'10
80   L=INT(LOG(I9)/2.30259)
90   CONVERT I9/10^(L-1) TO A$
100   CONVERT A$ TO Q
110   Q=L+3* NOT INT(100*(Q-INT(10*Q)/10))
120   A$="+"
130   A$[2]=CHR$(Q+64)
140   Q=I9/10^Q
150   Q=INT(10000*(Q-INT(100*Q)/100)+.9)
160   A$[3]=CHR$(Q)
170   ASSIGN "RQUEST",1,R
180   IF R THEN 430
190   IF  END #1 THEN 450
200   PRINT "PRACTICE SHEETS FOR FUNDAMENTALS X THROUGH Y (X,Y)."
210   INPUT A,B
220   IF A <= 0 OR A>B OR A#INT(A) OR B#INT(B) THEN 200
230   CONVERT A TO A$[4,6]
240   CONVERT B TO A$[7,9]
250   PRINT "FOR I.D. NUMBERS X THROUGH Y (X,Y)."
260   INPUT A,B
270   IF A<0 OR B>9.99 OR A>B THEN 250
280   CONVERT A TO A$[10,13]
290   CONVERT B TO A$[14,17]
300   LOCK #1
310   ADVANCE #1:32767,R
320   IF  NOT R THEN 310
330   PRINT #1;A$, END
340   UNLOCK #1
350   PRINT "DO YOU WISH TO FILE ANOTHER REQUEST";
360   INPUT B$
370   IF B$[1,1]="Y" THEN 200
380   CHAIN R,"MAIN"
390   PRINT "PROGRAM:  MAIN  IS MISSING."
400   PRINT "PLEASE INFORM THE AUTHORITIES."
410   SYSTEM R,"BYE"
420   GOTO 410
430   PRINT "FILE:  RQUEST  IS MISSING."
440   GOTO 380
450   PRINT "FILE:  RQUEST  IS CURRENTLY FULL."
460   PRINT "UNABLE TO STORE LAST REQUEST ENTERED."
470   GOTO 380
480   REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
490   FOR I=0 TO 2
500   E[I+1]=SYS(I)
510   NEXT I
520   CHAIN R,"ERROR"
530   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
540   GOTO 400
550   END
```

Craig Bina

3.22

CHGACC

```
10   COM I6,A0$[80],F9,X6,Y6,00$[11],E[3]
20   00$="CHGACC"
30   0=BRK(0)
40   IF  ERROR  THEN 500
50   FILES *,*
60   DIM A$[80],B$[80],I$[2]
70   PRINT '10'10"ALTERATION OF INSTRUCTORS' OPTIONS ACCESS CODE BY TEACHER"'10'10
80   L=INT(LOG(I6)/2.30259)
90   CONVERT I6/10^(L-1) TO A$
100   CONVERT A$ TO Q
110   Q=L+3* NOT INT(100*(Q-INT(10*Q)/10))
120   I$=CHR$(Q+64)
130   Q=I6/10^Q
140   I$[2]=CHR$(INT(10000*(Q-INT(100*Q)/100)+.9))
150   ASSIGN "ACCESS",1,R,"CODES"
160   IF  NOT R THEN 190
170   PRINT "THE AUTHORITIES HAVE NOT YET ENABLED THE INSTRUCTORS' OPTIONS."
180   GOTO 450
190   ASSIGN "ACCESS",2,R,"CODES"
200   A$=I$
210   PRINT "PLEASE ENTER ONE TO TEN LETTERS; THIS WILL BE YOUR NEW"
220   PRINT "INSTRUCTORS' OPTIONS ACCESS CODE.  > ";
230   SYSTEM R,"ECH-OFF"
240   LINPUT A$[3]
250   SYSTEM R,"ECH-ON"
260   PRINT '13;
270   A$=UPS$(A$)
280   LOCK #2
290   IF  END #2 THEN 410
300   A=REC(2)
310   B=ITM(2)
320   READ #2;B$
330   IF B$[1,2]#I$ THEN 300
340   READ #1,A
350   ADVANCE #1;B,R
360   PRINT #1;A$
370   IF  END #2 THEN 430
380   READ #2;B$
390   PRINT #1;B$
400   GOTO 380
410   PRINT "TEACHER/SCHOOL "Z$" DOES NOT HAVE THE INSTRUCTORS' OPTIONS CAPABILITY."
420   GOTO 450
430   ASSIGN *,1
440   PRINT "NEW ACCESS CODE HAS BEEN RECORDED"'10
450   CHAIN R,"MAIN"
460   PRINT "THE PROGRAM:  MAIN  IS MISSING."
470   PRINT "PLEASE INFORM THE AUTHORITIES."
480   SYSTEM R,"BYE"
490   GOTO 480
500   REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
510   FOR I=0 TO 2
520   E[I+1]=SYS(I)
530   NEXT I
540   CHAIN R,"ERROR"
550   PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
560   GOTO 470
570   END
```

Craig Bina

3.23

ERROR

```
10   COM I,N$[80],F,X,Y,P$[11],E[3]
20   0=BRK(0)
30   IF  ERROR  THEN 340
40   FILES *
50   PRINT LIN(2)
60   PRINT "FATAL EXECUTION ERROR OF TYPE"E[1]"HAS OCCURRED IN PROGRAM: "P$
70   PRINT "IN LINE NUMBER"E[2]", LAST FILE ACCESSED WAS #"E[3]"; SERIES #"X"."
80   PRINT "PROGRAM ABORTED.  PLEASE INFORM YOUR INSTRUCTOR OF THE PRECEDING."
90   PRINT
100   ASSIGN "ERRFIL",1,R
110   IF  NOT R THEN 140
120   PRINT "THE FILE:  ERRFIL  IS MISSING, PLEASE INFORM YOUR INSTRUCTOR."
130   GOTO 240
140   LOCK #1
150   ADVANCE #1;32767,R
160   IF  NOT R THEN 150
170   IF  END #1 THEN 210
180   MAT  PRINT #1;E
190   PRINT #1;I,P$,TIM(0),TIM(1),TIM(2),X,F, END
200   GOTO 240
210   READ #1,1
220   IF  END #1 THEN 240
230   GOTO 180
240   IF  ERROR  THEN 280
250   IF X#200 THEN 280
260   PRINT "P L E A S E   I N F O R M   T H E   A U T H O R I T I E S"
270   GOTO 320
280   IF  ERROR  THEN 340
290   I=-1
300   CHAIN R,"MAIN"
310   PRINT "THE PROGRAM: MAIN  IS MISSING, PLEASE INFORM YOUR INSTRUCTOR."
320   SYSTEM R,"BYE"
330   GOTO 320
340   PRINT "AN ERROR HAS OCCURRED IN THE PROGRAM:  ERROR"
350   PRINT "INSTRUCTIONAL PACKAGE TERMINATION FOLLOWS."
360   GOTO 320
370   END
```

Craig Bina

3.24

```
PSEUDO

10   COM A,A$[80],B,C
20   DIM B$[11]
30   C=0
40   PRINT "WHAT SHALL YOUR I.D. NUMBER BE";
50   INPUT A
60   PRINT "WHAT SHALL YOUR NAME BE";
70   INPUT A$
80   IF POS(A$," ")<2 THEN 60
90   PRINT "WHAT SHALL THE FUNDAMENTAL NUMBER EQUAL";
100   INPUT B
110   PRINT "WHAT IS DESTINATION CHAIN";
120   INPUT B$
130   CHAIN B$
140   END
```

Craig Bina

3.25

REMOTE

```
10   COM I,N$[80],F,X,Y,00$[11],E[3]
20   00$="REMOTE"
30   0=BRK(0)
40   IF  ERROR  THEN 1660
50   X=200
60   P=R=0
70   REM ************************** NEVER RENUMBER THIS PROGRAM *****************************
80   FILES *
90   DIM A$[72],B$[80],K$[50]
100   DIM R$[80],C$[80],Y[367]
110   PRINT CTL(1)
120   ASSIGN "RQUEST",1,R
130   IF  NOT R THEN 160
140   PRINT "FILE:  RQUEST  IS MISSING."
150   GOTO 1490
160   LOCK #1
170   IF  END #1 THEN 1470
180   V1=REC(1)
190   V2=ITM(1)
200   READ #1;R$
210   IF R$[1,1]#"+" THEN 180
220   T$=R$[2,2]
230   T1$=R$[3,3]
240   CONVERT R$[4,6] TO E[1]
250   F=E[1]
260   CONVERT R$[7,9] TO E[2]
270   CONVERT R$[10,13] TO A
280   CONVERT R$[14,17] TO B
290   R$[1,1]="-"
300   READ #1,V1
310   ADVANCE #1;V2,R
320   UPDATE #1;R$
330   ASSIGN *,1
340   A$="INTL"
350   A$[5]=R$[2,3]
360   ASSIGN A$,1,R
370   READ #1:A$,R,A$
380   A$=A$[POS(A$," ")+1]
390   P=1
400   PRINT CTL(1)"* * * * * * * * * * * * * * * * * * * * *"
410   FOR I9=1 TO 40
420   PRINT "S C H O O L :     "T1$"  - - - - -   T E A C H E R :     "A$
430   NEXT I9
440   P=2
450   PRINT CTL(1);
460   A=(A+.0001*NUM(T1$))*10^(NUM(T$)-64)
470   B=(B+.0001*NUM(T1$))*10^(NUM(T$)-64)
480   CONVERT B TO A$
490   CONVERT A$ TU E[3]
500   CONVERT A TO A$
510   CONVERT A$ TO I
520   B$="MES-AUTOMATIC FUNDAMENTAL GENERATION IN PROGRESS:"
530   SYSTEM R,B$
540   IF R THEN 530
550   B$[5]="PLEASE DO NOT SEND ANY MESSAGES TO W140 ON PORT #"'1
560   ENTER #Q7
570   CONVERT Q7 TO B$[POS(B$,'1)]
580   SYSTEM R,B$
590   IF R THEN 580
```

Craig Bina

```
600   REM  NAME FINDER:
610   A$="INTL"
620   GOSUB 1560
630   A$[5]=T$
640   A$[6]=T1$
650   ASSIGN A$,1,R
660   IF R<3 THEN 700
670   P=3
680   PRINT CTL(1)"FILE:  "A$"  IS MISSING."
690   GOTO 1460
700   IF  END #1 THEN 1250
710   READ #1,1
720   CONVERT I TO K$
730   CONVERT K$ TO I
740   READ #1;A$,Q,N$
750   CONVERT Q TO B$
760   CONVERT B$ TO Q
770   IF Q<I THEN 740
780   IF Q>I THEN 1250
790   IF A$="***" THEN 740
800   GOTO 890
810   REM  CHAIN HERE FOR A RETURN
820   O0$="REMOTE"
830   0=BRK(0)
840   IF  ERROR  THEN 1660
850   P=4
860   PRINT CTL(1);
870   PRINT LIN(5)
880   F=F+1
890   IF F>E[2] THEN 1310
900   A$="FUND#"
910   CONVERT F TO A$[5]
920   ASSIGN "USAGE",1,R
930   IF R THEN 1210
940   LOCK #1
950   Y6=TIM(3)-1+100* NOT TIM(3)
960   Z6=Y6-1+100* NOT Y6
970   W6=TIM(2) MIN 366
980   IF  END #1 THEN 1020
990   MAT  READ #1;Y
1000   X6=(Y[1]=Z6)+2*(Y[1]=Y6)
1010   GOTO X6 OF 1080,1150
1020   IF  END #1 THEN 1210
1030   MAT Y=ZER
1040   Y[1]=Y6
1050   READ #1,1
1060   MAT  PRINT #1;Y, END
1070   GOTO 1110
1080   READ #1,4
1090   MAT  READ #1;Y
1100   GOTO (Y[1]=Y6)+1 OF 1020,1050
1110   MAT Y=7ER
1120   Y[1]=TIM(3)
1130   READ #1,4
1140   MAT  PRINT #1;Y, END
1150   READ #1,4
1160   ADVANCE #1;W6,R
1170   READ #1;X6
1180   READ #1,4
1190   ADVANCE #1;W6,R
```

Craig Bina

```
1200  UPDATE #1;X6+1
1210  ASSIGN *,1,R
1220  CHAIN R,A$
1230  PRINT "PROGRAM:  "A$"  IS MISSING."
1240  GOTO 850
1250  REM NO I.D.
1260  GOSUB 1560
1270  IF INT(Q/L)=INT(I/L) THEN 1290
1280  I=(INT(I/L)+1+NUM(T1$)/10000)*L
1290  I=I+.01*L
1300  GOTO 1350
1310  F=E[1]
1320  GOSUB 1560
1330  IF L2 THEN 1280
1340  I=I+.01*L
1350  CONVERT I TO A$
1360  CONVERT A$ TO I
1370  IF I <= E[3] THEN 600
1380  B$="MES-FUNDAMENTAL PROCESSING COMPLETED.  THANK YOU FOR YOUR COOPERATION."
1390  SYSTEM R,B$
1400  IF R THEN 1390
1410  B$[5]="MESSAGES NOW ACCEPTED BY W140 ON PORT #"'1
1420  ENTER #Q7
1430  CONVERT Q7 TO B$[POS(B$,'1)]
1440  SYSTEM R,B$
1450  IF R THEN 1440
1460  GOTO 110
1470  PRINT #1,1; END
1480  ASSIGN *,1
1490  P=5
1500  PRINT CTL(1)
1510  FOR I9=1 TO 40
1520  PRINT "P R O C E S S I N G   C O M P L E T E D"
1530  NEXT I9
1540  SYSTEM R,"BYE"
1550  GOTO 1540
1560  L=INT(LOG(I)/2.30259)
1570  CONVERT I/10^(L-1) TO B$
1580  CONVERT B$ TO Q2
1590  L2= NOT INT(100*(Q2-INT(10*Q2)/10))
1600  L=L+3*L2
1610  T$=CHR$(L+64)
1620  L=10^L
1630  I2=I/L
1640  T1$=CHR$(INT(10000*(I2-INT(I2*100)/100)+.9))
1650  RETURN
1660  REM ERROR TRAP DESTINATION FOR IF-ERROR STATEMENT
1670  IF R=-500 THEN 1740
1680  W8=SYS(0)
1690  IF W8<300 THEN 1780
1700  C$="MES-*** OUT OF PAPER ON LINEPRINTER; PLEASE RELOAD ***"
1710  SYSTEM R,C$
1720  IF R THEN 1710
1730  R=-500
1740  PRINT CTL(1)
1750  R=0
1760  GOTO P OF 400,450,680,860,1500,1820
1770  GOTO 110
1780  E[1]=W8
1790  E[2]=SYS(1)
```

Craig Bina

124

```
1800  E[3]=SYS(2)
1810  P=6
1820  PRINT CTL(1)"E   R   R   O   R"
1830  CHAIN R,"ERROR"
1840  PRINT "AN ERROR HAS OCCURRED AND THE PROGRAM: ERROR  IS MISSING."
1850  PRINT "PLEASE INFORM THE AUTHORITIES."
1860  GOTO 1540
1870  END
```

Craig Bina

```
[REPAIR] AL     5    6.3   REPORT  C      4     [REVEAL]           3
RNDFND  C      11          RNDMAR  M      1      RNDMBT  M         1
RNDMBW  M       2          RNDMDF  M      1      RNDMGH  M         1
RNDMGW  M       1          RNDMIV  M      1      RNDMJE  M         1
RNDMJT  M       1          RNDMKW  M      1      RNDMLR  M         1
RNDMMW  M       1          RNDMSE  M      1      RNDMSF  M         1
RNDMSR  M       1          RNDMTE  M      1      RNDMWC  M         1
RNDMWE  M       1          RNDMWW  M      1      RNDMZA  M         1
RQUEST  M       2          ~~SOFNR        1~~    ~~SOIO            1~~
~~SOINIT        2~~        ~~SOLUCK       1~~    ~~SONTL           1~~
~~SOSEAR        1~~        ~~SOSANK       2~~    ~~[illegible]     4~~
~~SOUBGE        1~~        ~~SAVE1        1~~    ~~SAVE12          1~~
~~SAVE2         1~~        ~~SAVEM        7~~    ~~SAVEN           2~~
SETUP   C       6          SOLFIL  M      7      STOR3   M         4
~~TEST~~  AL    JT         UPDATE  C     11     [UPFIX]            1
USAGE   M       6          USERS   M      3      WORD78  M        60
```

127

Craig Bina

ACKNOWLEDGEMENTS

Without the assistance of my chemistry teacher and project sponsor: Mr. Donald Bauder, , this project would never have been begun, much less completed. His continual support, suggestions, and guidance have been invaluable, and his incessant cries of "Is it possible to...?" have served as a constant inspiration. This man's devotion to teaching knows no bounds.

My appreciation must also extend to all of the faculty, students, and administrators of high school district 214. Their constant comments and suggestions and their perseverance in the face of crises are what have allowed this management system to operate on the district-wide scale which it has now achieved.

SOURCES OF INFORMATION

2000/Access BASIC Pocket Guide, Hewlett Packard part #22687-90003, March 1976.

HP20854A Timeshared BASIC/2000 Level F Reference Manual, Hewlett Packard part #02000-90073, October 1974.

O'Connor, Paul; Davis, Joseph, Jr.; Haenisch, Edward; MacNab, W. Keith; McClellan, A. L., Chemistry--Experiments and Principles, Lexington, Massachusetts, D. C. Heath and Company, 1977.

Toon, Ernest, and Ellis, George, Foundations of Chemistry, New York, New York, Holt, Rinehart and Winston, Incorporated, 1973.

The four abovementioned texts have served as valuable reference works in developing and testing this package.

CERTIFICATE OF COPYRIGHT REGISTRATION

FORM TX
UNITED STATES COPYRIGHT OFFICE

This certificate, issued under the seal of the Copyright Office in accordance with the provisions of section 410(a) of title 17, United States Code, attests that copyright registration has been made for the work identified below. The information in this certificate has been made a part of the Copyright Office records.

David L. Ladd

REGISTER OF COPYRIGHTS
United States of America

REGISTRATION NUMBER
TXu 52-217
TX TXU

EFFECTIVE DATE OF REGISTRATION
9 15 80
(Month) (Day) (Year)

DO NOT WRITE ABOVE THIS LINE. IF YOU NEED MORE SPACE, USE CONTINUATION SHEET (FORM TX/CON)

1 Title

TITLE OF THIS WORK: A Multi-User Computer Assisted Instructional Management System: Software Development and Implementation

If a periodical or serial give: Vol....... No....... Issue Date.......

PREVIOUS OR ALTERNATIVE TITLES: The Instructional Package

PUBLICATION AS A CONTRIBUTION: (If this work was published as a contribution to a periodical, serial, or collection, give information about the collective work in which the contribution appeared.)

Title of Collective Work: .... Vol....... No....... Date....... Pages.......

2 Author(s)

IMPORTANT: Under the law, the "author" of a "work made for hire" is generally the employer, not the employee (see instructions). If any part of this work was "made for hire" check "Yes" in the space provided, give the employer (or other person for whom the work was prepared) as "Author" of that part, and leave the space for dates blank.

1

NAME OF AUTHOR: CRAIG RICHARD BINA

Was this author's contribution to the work a "work made for hire"? Yes...... No. X

DATES OF BIRTH AND DEATH: Born 1962 (Year) Died....... (Year)

AUTHOR'S NATIONALITY OR DOMICILE: Citizen of U.S.A. (Name of Country) or Domiciled in .... (Name of Country)

AUTHOR OF: (Briefly describe nature of this author's contribution)
Entire Text.

WAS THIS AUTHOR'S CONTRIBUTION TO THE WORK:
Anonymous? Yes...... No. X
Pseudonymous? Yes...... No. X
If the answer to either of these questions is "Yes," see detailed instructions attached.

2

NAME OF AUTHOR:

Was this author's contribution to the work a "work made for hire"? Yes...... No......

DATES OF BIRTH AND DEATH: Born....... (Year) Died....... (Year)

AUTHOR'S NATIONALITY OR DOMICILE: Citizen of .... (Name of Country) or Domiciled in .... (Name of Country)

AUTHOR OF: (Briefly describe nature of this author's contribution)

WAS THIS AUTHOR'S CONTRIBUTION TO THE WORK:
Anonymous? Yes...... No......
Pseudonymous? Yes...... No......
If the answer to either of these questions is "Yes," see detailed instructions attached.

3

NAME OF AUTHOR:

Was this author's contribution to the work a "work made for hire"? Yes...... No......

DATES OF BIRTH AND DEATH: Born....... (Year) Died....... (Year)

AUTHOR'S NATIONALITY OR DOMICILE: Citizen of .... (Name of Country) or Domiciled in .... (Name of Country)

AUTHOR OF: (Briefly describe nature of this author's contribution)

WAS THIS AUTHOR'S CONTRIBUTION TO THE WORK:
Anonymous? Yes...... No......
Pseudonymous? Yes...... No......
If the answer to either of these questions is "Yes," see detailed instructions attached.

3 Creation and Publication

YEAR IN WHICH CREATION OF THIS WORK WAS COMPLETED:
Year 1979
(This information must be given in all cases.)

DATE AND NATION OF FIRST PUBLICATION:
Date....... (Month) (Day) (Year)
Nation....... (Name of Country)
(Complete this block ONLY if this work has been published.)

4 Claimant(s)

NAME(S) AND ADDRESS(ES) OF COPYRIGHT CLAIMANT(S):
Craig R. Bina
[redacted]

TRANSFER: (If the copyright claimant(s) named here in space 4 are different from the author(s) named in space 2, give a brief statement of how the claimant(s) obtained ownership of the copyright.)

- *Complete all applicable spaces (numbers 5-11) on the reverse side of this page*

DO NOT WRITE HERE

TXu 52-217	EXAMINED BY: CHECKED BY:	APPLICATION RECEIVED:	FOR COPYRIGHT OFFICE USE ONLY
	CORRESPONDENCE: □ Yes	DEPOSIT RECEIVED:	
	DEPOSIT ACCOUNT FUNDS USED: □	REMITTANCE NUMBER AND DATE:	

DO NOT WRITE ABOVE THIS LINE. IF YOU NEED ADDITIONAL SPACE, USE CONTINUATION SHEET (FORM TX/CON)

PREVIOUS REGISTRATION:

5 **Previous Registration**

- Has registration for this work, or for an earlier version of this work, already been made in the Copyright Office? Yes........ No .X....
- If your answer is "Yes," why is another registration being sought? (Check appropriate box)
  - □ This is the first published edition of a work previously registered in unpublished form.
  - □ This is the first application submitted by this author as copyright claimant.
  - □ This is a changed version of the work, as shown by line 6 of this application.
- If your answer is "Yes," give: Previous Registration Number .... Year of Registration ....

COMPILATION OR DERIVATIVE WORK: (See instructions)

6 **Compilation or Derivative Work**

PREEXISTING MATERIAL: (Identify any preexisting work or works that this work is based on or incorporates.)

MATERIAL ADDED TO THIS WORK: (Give a brief, general statement of the material that has been added to this work and in which copyright is claimed.)

MANUFACTURERS AND LOCATIONS: (If this is a published work consisting preponderantly of nondramatic literary material in English, the law may require that the copies be manufactured in the United States or Canada for full protection. If so, the names of the manufacturers who performed certain processes, and the places where these processes were performed *must* be given. See instructions for details.)

7 **Manufacturing**

NAMES OF MANUFACTURERS	PLACES OF MANUFACTURE

REPRODUCTION FOR USE OF BLIND OR PHYSICALLY-HANDICAPPED PERSONS: (See instructions)

8 **License For Handicapped**

- Signature of this form at space 10, and a check in one of the boxes here in space 8, constitutes a non-exclusive grant of permission to the Library of Congress to reproduce and distribute solely for the blind and physically handicapped and under the conditions and limitations prescribed by the regulations of the Copyright Office: (1) copies of the work identified in space 1 of this application in Braille (or similar tactile symbols); or (2) phonorecords embodying a fixation of a reading of that work; or (3) both.

a ☒ Copies and phonorecords   b □ Copies Only   c □ Phonorecords Only

DEPOSIT ACCOUNT: (If the registration fee is to be charged to a Deposit Account established in the Copyright Office, give name and number of Account.)

Name: ....
Account Number: ....

CORRESPONDENCE: (Give name and address to which correspondence about this application should be sent.)

Name: Craig Bina
Address: [redacted] (Apt.)
[redacted]
(City) (State) (ZIP)

9 **Fee and Correspondence**

CERTIFICATION: * I, the undersigned, hereby certify that I am the: (Check one)

☒ author □ other copyright claimant □ owner of exclusive right(s) □ authorized agent of: ....
(Name of author or other copyright claimant, or owner of exclusive right(s))

of the work identified in this application and that the statements made by me in this application are correct to the best of my knowledge.

Handwritten signature: (X) ....

Typed or printed name... Craig R. Bina .... Date 9-4-1980...

10 **Certification (Application must be signed)**

CRAIG BINA
(Name)
[redacted]
(Number, Street and Apartment Number)
[redacted]
(City) (State) (ZIP code)

MAIL CERTIFICATE TO

(Certificate will be mailed in window envelope)

11 **Address For Return of Certificate**

*17 U.S.C. § 506(e): Any person who knowingly makes a false representation of a material fact in the application for copyright registration provided for by section 409, or in any written statement filed in connection with the application, shall be fined not more than $2,500.

U.S. GOVERNMENT PRINTING OFFICE: 1978-261-022/16