

AN ON-LINE SYSTEM FOR
MAINTAINING GRADES FOR
BASIC COMPUTER SCIENCE CLASSES

by

DARRELL E. MCINTOSH

B.I.E., Auburn University, 1971

A MASTER'S REPORT

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

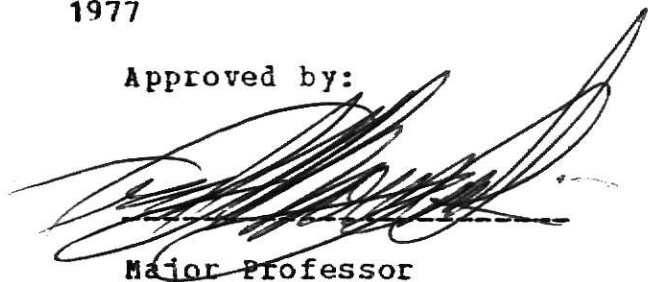
Department of Computer Science

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1977

Approved by:

A large, stylized handwritten signature in black ink, appearing to be written over a horizontal line.

Major Professor

TABLE OF CONTENTS

M32 C.2	List of Figures	3
I.	Introduction	4
1.1	Background	4
1.2	Objective	5
1.3	Integrated Database Management System	5
1.4	IDMS Definitions	6
II.	Data Base Design	8
2.1	Schema Description	8
2.2	Storage Requirements	11
2.3	Subschema Description	12
III.	Application Programs	18
3.1	General Design	18
3.2	Interactive Grader Program	18
3.3	Interactive Roster Program	19
IV.	Testing	21
4.1	Initial Testing	21
4.2	Further Testing	21
V.	Conclusion	23
	Bibliography	26
	Appendix A - CMS EXEC Procedures and Special Files	27
	Appendix B - IDMS Schema	34
	Appendix C - IDMS Device-Media Control	39
	Appendix D - IDMS Subschema	41
	Appendix E - Sample On-Line Session	43
	Appendix F - Sample Grade Reports	47
	Appendix G - User's Guide	54

LIST OF FIGURES

Figure

2.1	CALC Location Mode	14
2.2	VIA Location Mode	14
2.3	Schema Data Structures	15
2.4	Database Record Descriptions	16
2.5	Database Diagram Description	17

CHAPTER I

INTRODUCTION

1.1 BACKGROUND

In the past, grades for the Computer Science 200 lecture classes and corresponding laboratories have been maintained on the IBM S/370 using the University Computing Center's Roster program. Although this program provides a good way of maintaining the grades it was often costly and time-consuming for both the lecture and lab instructors. In addition, the grade reports were not always timely because of various due dates for assignments and because of the complexity involved in obtaining grade cards from each instructor and running them through the computer in a single batch.

The author of this report developed the concept of an on-line system that would allow the instructors to update grades and receive grade reports in a timely manner. After discussions with Elizabeth Unger, the current CS 200 instructor, it was decided to proceed with the design and implementation of such a system. It was also decided that a Database Management System (DBMS) should be used to maintain the grades in a direct-access file since its use would simplify design of the database and allow for easy future expansion.

The on-line system should provide protection for the information in the database. A password scheme was devised

that would allow lab instructors access to only their students' records while the lecture instructor could access all records in the database.

The on-line system runs on the IBM S/370 under the Conversational Monitoring System (CMS). The Integrated Database Management System (IDMS) developed by Cullinane Corporation was selected to maintain the database. IBM's PL/I was selected as the implementation language because of its structured programming facilities which are not readily available in other languages available on the S/370.

1.2. OBJECTIVE

This report will show the design and implementation of the on-line system for maintaining class grades. The report will include design of the database, design of a program to load and maintain the database, and design of a program to produce the grade reports. In addition, a User's Guide will be included to explain the use of the system and recommendations will be made for further enhancements to the system.

1.3. INTEGRATED DATABASE MANAGEMENT SYSTEM

Cullinane Corporation's Integrated Database Management System is available on the IBM S/370 under OS and provides facilities for PL/I or COBOL programs or any other language that has CALL capabilities.

To use IDMS for a single database application three separate steps must be completed: a description of the database must be developed (SCHEMA), a description of the

portions of the database to be used by the application programs must be developed (SUBSCHEMA), and a description of the physical file must be given (DMCL).

The Data Description Language (DDL) is used to describe the database in the SCHEMA, SUBSCHEMA and DMCL. The Data Manipulation Language (DML) is used in application programs to invoke the IDMS routines that manipulate the database. DML statements can be used any place in the application programs where input or output of information in the database is desired.

1.4. IDMS DEFINITIONS

The SCHEMA uses the DDL language to define the logical description of the database. It describes in detail each of the records in the database and establishes set relationships between the records.

The SUBSCHEMA uses DDL statements to describe to IDMS the portions of the database to be made available to an application program. It consists of record and set descriptions as indicated in the SCHEMA and is really the application programmer's view of the database.

The DEVICE-MEDIA CONTROL LANGUAGE is used to describe the physical layout of the database including buffering and paging. The logical database described in the SCHEMA is mapped into this physical layout by the IDMS software.

A DATA-ITEM is the smallest unit of named data.

A RECORD is a named collection of data-items. The record description specifies the data items contained within

the record. More than one record may exist within the database. An occurrence of a record is an actual collection of data as described in the record description. There may be many occurrences of a particular record.

A SET is a named collection of occurrences of a record. There may be many occurrences of a set. Each set consists of an owner record and one or more occurrences of a record that is not the same as the owner.

An AREA is a named logical sub-division of the physical storage space in the database. An area may contain occurrences of one or more records. The use of multiple areas allows the database to be distributed over several physical files.

A FILE is a named collection of physical records. In an IDMS application a file may contain one or more areas.

A DATABASE is a collection of occurrences of record types including the relationships between records.

CHAPTER II

DATA BASE DESIGN

This chapter describes in detail the design of the database used to implement the on-line grading system.

2.1. SCHEMA DESCRIPTION

The Schema is used to describe to IDMS information about each of the records that exists within the database. This information includes the record name, data items within the record and relationships between the various records, commonly known as set relations.

In this implementation two location modes, CALC and VIA, are used to store records within the database. The CALC option involves using a hashing function provided by IDMS to determine a unique key based upon a data item in the record which is then used to store the record within the database. Records stored as CALC can be accessed directly using the appropriate CALC key. Records stored in the database with the VIA option are physically placed as close as possible to the owner of a set in which the record is to be a member. The VIA option greatly reduces the number of physical accesses required to retrieve a record since when the owner record is read by IDMS the members of the sets it owns that are stored VIA will probably also be read into memory in the same page. Figures 2.1 and 2.2 illustrate the CALC and VIA concepts.

Three records and five set relations are used in this

implementation. Figure 2.3 shows the records and sets used. Figure 2.4 explains the various notations used in the database diagram. Figure 2.5 lists each record in the database and the data-items within the records. The actual schema is included as Appendix B to this report.

The first record used is the MASTER-LINE record. There is only one occurrence of this record in the database and it contains information common to all line numbers in the database such as semester and year, possible points for each lecture grade, the break points to be used in computing letter grades for students in lecture and the password for the MASTER mode.

The second record used in the database is the LINE-NO record. There is one occurrence of this record for each line number being maintained. This record contains information that is to be used only for a particular line number such as the instructor's name, type of class (e.g., PL/I BUSINESS), possible points for each lab grade, break points for computing lab letter grades and the password for the line number.

The third record is the STUDENT record. There is one occurrence of this record for each student in the data base. The STUDENT record contains information about the student such as the student's name, student number, an optional eight-character code used for posting grades, lab and lecture assignment and exam grades, and a field for storing a grade of WDP or WDF for lab and lecture.

Occurrences of sets are implemented in IDMS as linked-lists with a link to the next entry in the set always present and with optional links available to the previous entry in the set and to the owner of the set.

The first set of the five used in the database is the LINE-SET which consists of all occurrences of the LINE-NO record. The owner of the set is the single occurrence of the MASTER-LINE record. This set facilitates certain modifications of the database by the interactive GRADER program and allows easier creation of reports by the interactive ROSTER program. Duplicate line numbers are not allowed.

The second set is the STUD-SET which contains all occurrences of the STUDENT record within a line number. Each LINE-NO record in the database is the owner of one of these sets. Each set is in ascending order by student name with duplicate names not allowed within a line number. This set simplifies updating of the database and production of reports.

The third set in the database is the CODE-SET. Each LINE-NO record in the database is the owner of a CODE-SET which contains all occurrences of the STUDENT record within the line number that contain the optional eight-character code. The set is in ascending order by code with duplicates not allowed. It is used to produce a grade report for posting using the optional code.

The NAME-SET is the fourth set in the database. It is

owned by the single MASTER-LINE record and contains all occurrences of the STUDENT record in ascending order by name. This set is used to create a grade report containing all students in the database in alphabetical order.

The SSN-SET, the final set, contains all students in the database in ascending order by student number with duplicates allowed. This set is also owned by the MASTER-LINE record and is included for future interface with the Computing Center's Grader program and for a report of students listed by student number.

The five set relationships require some system overhead to maintain, however they should not reduce the speed of accessing the database significantly. The pointers for each set within the records would have to be modified only when a record is added to or deleted from the database. When a record is modified the pointers are not adjusted unless the control data for a set is changed. The elimination of sorts otherwise needed to produce reports justifies the maintenance of the pointers.

2.2. STORAGE REQUIREMENTS

In order to properly allocate direct access space, the space requirements must be calculated using the formulas specified in the IDMS documentation.¹ The occurrences of each record are estimated and the length is determined from

1. Data Definition Languages, Utilities and GCI Reference Guide, Release 3.1 (Boston: Cullinane Corporation, 1975), Appendix G.

the Schema description by computing the bytes required for each record type and adding four bytes for each pointer in the record. A page size of 3156 bytes is used since it is large enough to contain a number of records and fit an IBM 3330 disk track without wasting space. For purposes of computing the space required the following values were used:

<u>record</u>	<u>occurrences</u>	<u>total length (bytes)</u>
MASTER-LINE	1	68
LINE-NO	15	92
STUDENT	700	92

Database Total

MASTER-LINE	68
LINE-NO	1,380
STUDENT	64,400

Pages for Data

$$N = (68 + 1,380 + 64,400) / 3,156 \\ = 20.86 = 21 \text{ pages}$$

Pages for Indexing

$$I = (8 (1 + 15 + 700) + 32 * 21) / 3156 \\ = 2.03 = 3 \text{ pages}$$

Pages for Space Management

$$S = 2(21 + 3) / (3156 - 32) \\ = .02 = 1 \text{ page}$$

Total Pages Required

$$A = 21 + 3 + 1 \\ = 25$$

2.3. SUBSCHEMA DESCRIPTION

The Subschema specifies that information in the database that is to be made available to an application program and the access rights that programs are allowed.

In this application only one Subschema was used. Since both the interactive GRADER program and the interactive ROSTER program require access to all values in the database the Subschema includes all data items specified in the Schema. The GRADER program must update the database while the ROSTER program need only retrieve values from the database. Only one subschema is used, however, since the extra disk space required to store two subschema, one for update and one for retrieval, would not be justified. Both programs are, therefore, allowed update access to the database. Appendix D contains the actual Subschema used in this implementation.

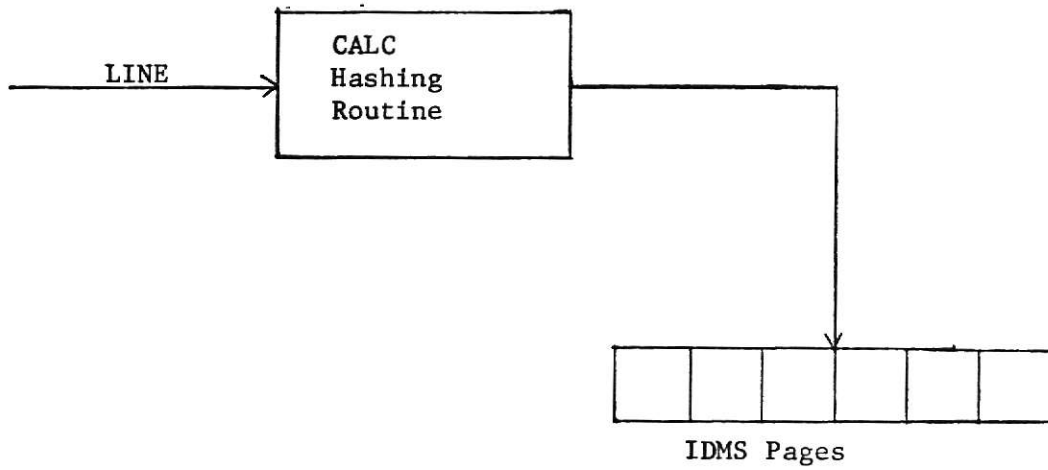


Figure 2.1 CALC Location Mode

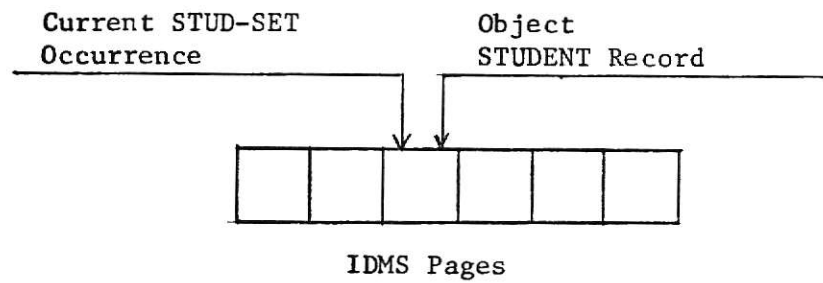


Figure 2.2 VIA Location Mode

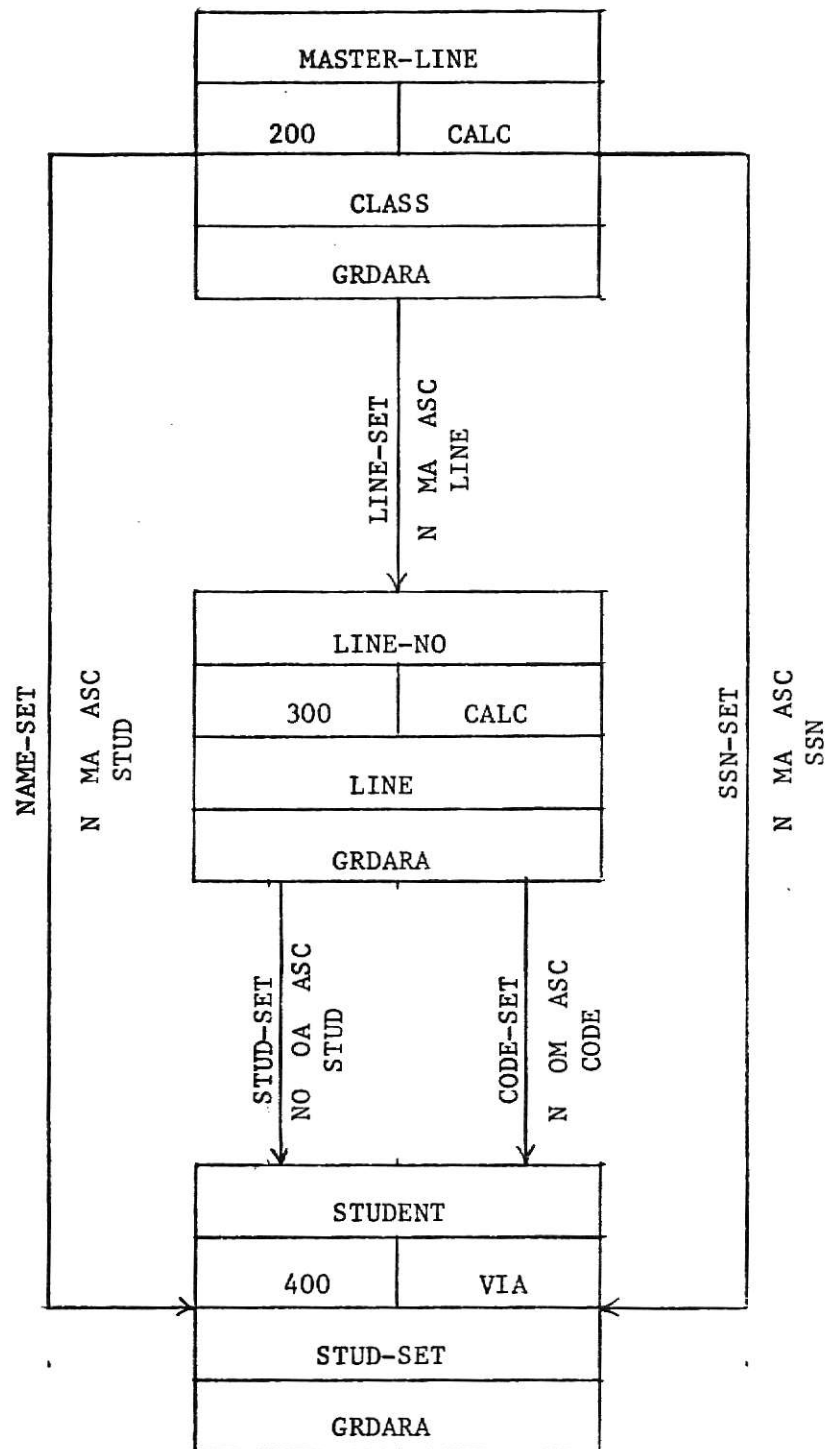


Figure 2.3 Schema Data Structures

MASTER-LINE Record

CLASS	(Semester and year)
PASS	(Password for Master mode)
POSSIBLES	(Possible points for each lecture field)
BREAK-LECT	(Breakpoints for lecture letter grades)

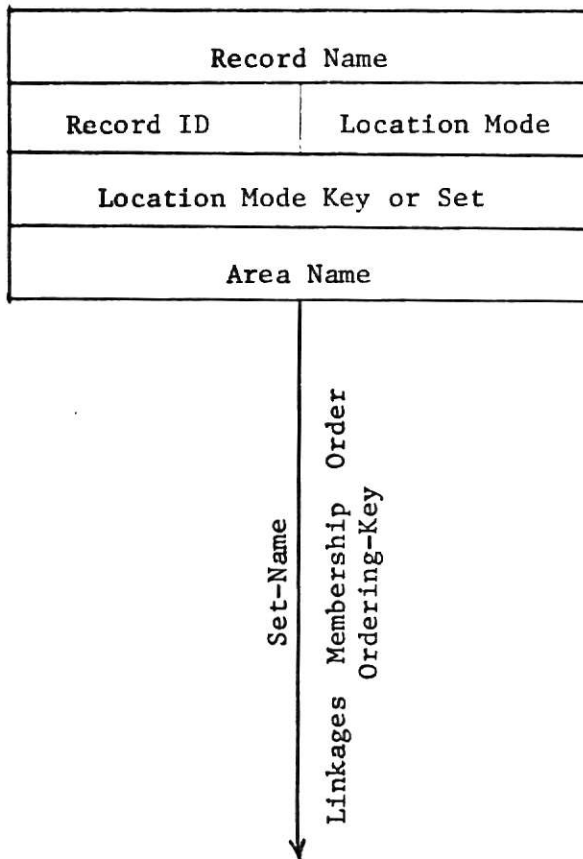
LINE-NO Record

LINE	(Line number)
PASSWORD	(Password for line number)
INSTRUCT	(Instructor's name)
CLASS-TYPE	(Language and section type)
POSS	(Possible points for each lab field)
BREAK-LAB	(Breakpoints for lab letter grades)

STUDENT Record

STUD	(Student's name)
SSN	(Student's ID number)
CODE	(Optional code for posting grades)
GRADES	(Lab and lecture grades)
GRADE-LECT	(Lecture grade of WDP, WDF, or null)
GRADE-LAB	(Lecture grade of WDP, WDF, or null)

Figure 2.4 Database Record Descriptions



Linkages

N - Next
P - Prior
O - Owner

Set Membership

Mx - Mandatory
Ox - Optional
xA - Automatic
xM - Manual

Order

ASC - Ascending
DES - Descending
FIRST
LAST
PRIOR
NEXT

Figure 2.5 Database Diagram Description

CHAPTER III

APPLICATION PROGRAMS

This chapter describes the application programs used in this implementation.

3.1. GENERAL DESIGN

All programs in this implementation are designed to be interactive and to be run under the Conversational Monitoring System (CMS) on the IBM System/370. When a decision is to be made by the user or data is to be entered the user is prompted for a response.

All programs run in one of two modes, MASTER or USER. When each program begins execution, the user is asked to enter a password. If the password entered matches the password in the MASTER-LINE record, the program runs in the MASTER mode, otherwise it runs in the USER mode. A person in the MASTER mode has access to all records in the database. A person in the USER mode has access only to those line numbers whose password in the LINE-NO record matches the user's password. This password protection system allows teaching assistants to access only their students' records while the lecture instructor may make any needed changes to any records in the database.

3.2. INTERACTIVE GRADER PROGRAM

The interactive GRADER program is used to add records to the database, modify records and to query the database.

The program runs in either the MASTER or USER mode. A person in the MASTER mode has access to all records in the database and to a powerful set of commands used to maintain the database. A person in the USER mode is provided a restricted, less powerful subset of commands. Commands available to a USER include adding a new line number, adding student records to a line number, changing values within a LINE-NO or STUDENT record and querying the database for values in the LINE-NO or STUDENT record. A person in the MASTER mode has all the USER commands available to him or her plus the ability to move students from one line number to another, to delete a student from the database and to modify the MASTER-LINE record. In addition, in Master mode a grade score for all students in the database can be incremented or decremented using a single CHANGE command. A complete list of all the available commands is given in the User's Guide with a description of each.

Since the STUDENT records are stored VIA the LINE-SET, the LINE-NO record that is the owner of the set must be made current in order to retrieve the STUDENT record. The Update command is used to make a LINE-NO record current so that all following commands up to the next End command will reference data for that line number. Certain commands, such as the Inquire command, allow the user to specify the line number so that it may be made current before the STUDENT record is referenced.

3.3. INTERACTIVE ROSTER PROGRAM

The interactive ROSTER program produces two reports. When in the USER mode a listing is produced for individual line numbers alphabetically by student name and optionally by the eight-character code for posting grades. In the MASTER mode the report may be by line number as in the USER mode or the report may contain all students in the data base arranged alphabetically. In either mode the option is also given of computing and printing letter grades for the students. Additional information on the interactive ROSTER program may be found in the User's Guide.

CHAPTER IV

TESTING

This chapter describes the testing procedures used for evaluating the application programs.

4.1. INITIAL TESTING

After the Schema, DMCL, Subschema and application programs were successfully compiled all commands in the system were checked using sample data for four classes. The initial testing revealed only a few major problems which were easily corrected.

4.2. FURTHER TESTING

Once the initial testing was completed the system was put into actual use by the author during Fall semester 1976. Grades for three classes taught by the author consisting of approximately 100 students were maintained in the database. No significant problems were found and no data was lost during this phase of the testing. The disk space used for the database was reduced for the testing since only a portion of all the CS 200 classes were being used for testing.

Since the programs were tested using actual classes all commands were thoroughly tested including loading the database, adding and deleting students, updating grades and changing breakpoints. Grade reports were frequently generated, both by student name and by the optional code.

The optional code field was used during the test period

to allow posting of grades. Student reaction to this was favorable in that they were able to verify their grades in the database and to compare their standing in the class with others. The only problem that arose was a difficulty in keeping the lecture exam grades updated to reflect grade adjustments. Normally the lecture exam grades would be maintained by the lecture instructor so that this problem would be minimized.

The testing of the system indicates that the database design and application program designs are sound and workable. The minimal number of errors found can be attributed, in part, to the structured design of the overall system and to structured programming techniques used in writing the application programs.

CHAPTER V

CONCLUSION

Several additional areas require some further review and discussion. One of these areas is the integrity of the data. Since the system runs under CMS the database must first be retrieved from the File Manager space. The user then updates the latest version of the database as it existed in the File Manager. To save the database it must then be written back into the File Manager space, therefore, if serious problems occurred during maintenance of the database the user can replace his copy with the previous copy of the database in the File Manager and then update it or he can sign off without saving his copy of the database thus preventing an erroneous copy from being saved.

Other areas of interest are security and protection. Two levels of passwords provide this protection. If the CMS account number is password protected the user must know this password. In addition the application programs require that a password be entered before actual processing of the database. The password determines whether the entire database or only a portion of it will be made available to the user. If passwords are changed frequently the possibility of unauthorized persons accessing the database would be lessened. An additional problem in this area is that if a person knows the account number and account password he has the capability of erasing the database from

the File Manager space. One solution to this would be to also keep a copy of the database under OS to allow for recovery from an erased database or from data errors in the existing database.

Another area of interest is improvements to the system. One improvement that deserves consideration is an interface between the Computing Center's Grader program and the interactive system. The Center's Grader program is used to grade examinations and is currently used for lecture exams because of the size of the classes. Output from this program can be passed to the interactive system by specifying the punch option in the Computing Center's program and having the punched output directed to the CMS Virtual Machine that the database resides in. The punched output would contain the student's identification number, the optional field on the mark sense card, the student's responses and his score. This data can then easily be interfaced with the database.

Another improvement would be the direct interface between the class enrollment records from Admissions and Records and the interactive system. Due to privacy requirements this cannot be accomplished directly but the Computing Center will provide data for students in the lecture and lab that consists of the student's name, ID number and line number. A program could be developed that would easily load the database using this information.

Other improvements could include statistical analyses on

particular fields in the database, such as assignments or exams, programs to verify information in the database and programs to produce various other reports.

BIBLIOGRAPHY

1. Data Definition Languages, Utilities and GCI Reference Guide, Release 3.1. Boston: Cullinane Corporation, 1975.
2. Data Manipulation Language Programmer's Reference Guide, Release 3.1. Boston: Cullinane Corporation, 1975.
3. Martin, James. Computer Data-Base Organization. Englewood Cliffs, New Jersey: Prentice-Hall, 1975.

APPENDIX A

CMS EXEC PROCEDURES

AND SPECIAL FILES

This Appendix gives recommended EXEC procedures for use with CMS. It is assumed that the EXEC procedures, program TEXT files and database all reside in File Manager space.

The PROFILE EXEC procedure is used to turn File Manager checkpointing off and to make several libraries global so that all external references can be resolved when the programs are loaded.

```
ECNTR0L OFF
FMSET C OFF
GLOBAL TXTLIB IDMS PLILIB
```


The REQ file when stored in the File Manager space indicates the files to be retrieved automatically by the File Manager when someone signs onto CMS. It contains a list of all files needed during the on-line session. It should contain, as a minimum, the following records:

VM421	IDMS	TXTLIB	(IDMS Library)
*	PROFILE	EXEC	(CMS Profile)
*	GRADEXEC	EXEC	(Executes programs)
*	WRAPUP	EXEC	(Logoff procedure)
*	GRADES	DATA	(Database)
*	ROSTER	TEXT	(ROSTER program)
*	GRADER	TEXT	(GRADER program)
*	GRDMCL	TEXT	(IDMS DMCL)
*	GRDSSUB	TEXT	(IDMS Subschema)

The GRADEXEC EXEC procedure is used to execute the application programs. This procedure accepts one or two parameters. The first must be the name of the program to be executed. The second parameter is optional and, if coded, specifies the Virtual Machine to which any print files, such as ROSTER grade reports, are to be spooled. If omitted, the print files will be spooled to OS where they may be printed using VMPRT.

This procedure first checks to see if a MODULE file for the specified program exists and, if so, it is executed. If the MODULE file does not exist, a check is made to see if a TEXT file exists for the program. If one does not exist, an error message is printed and the EXEC procedure is terminated. If a TEXT file does exist, a MODULE file is created and then executed.

```

&CONTROL OFF
&IF &INDEX NE 0 &GOTO -LAB1
&BEGTYPE
ENTER PROGRAM NAME AS PARAMETER
&END
&EXIT 4
-LAB1 STATE &1 MODULE A
&IF &RETCODE EQ 0 &GOTO -LAB2
STATE &1 TEXT A
&IF &RETCODE EQ 0 &GOTO -LAB3
&BEGTYPE
INVALID PROGRAM NAME
&END
&EXIT 4
-LAB3 &CONTINUE
LOAD &1 GRDMCL GRDSSUB (RESET IHECMS)
GENMOD &1
-LAB2 &CONTINUE
&A = OS
&IF &INDEX LT 2 &GOTO -LAB4

```

&A = &2
-LAB4 &CONTINUE
CP SPOOL PRINT TO &A
FILE SYSPRINT TERM (RECFM F BLOCK 133)
FILE SYSIN TERM (RECFM F BLOCK 80)
FILE PRT PRINT (RECFM F BLOCK 133)
FILE GRDDB DISK GRADES DATA (DSORG DA)
FILE GRDJRN DUMMY (RECFM F BLOCK 3156)
&1

The WRAPUP EXEC procedure is used to log off the CMS terminal. It requires one parameter, a 'Y' or a 'N', to indicate whether or not the database is to be copied into the File Manager space to replace the copy of the database there. A 'Y' indicates the database is to be saved while a 'N' indicates it is not to be saved. If the parameter is not coded or is entered incorrectly, the user is asked to enter the option again.

```
&CONTROL OFF
&IF &INDEX NE 0 &GOTO -LAB1
&BEGTYPE
SAVE DATABASE?
&END
&READ ARGS
-LAB1 &CONTINUE
&IF &1 EQ N &GOTO -LAB3
&IF &1 NE Y &GOTO -LAB2
FMS GRADES DATA (REPL)
&IF &RETCODE EQ 0 &GOTO -LAB3
&BEGTYPE
DATABASE NOT SAVED - FMSSAVE ERROR
&END
&EXIT 4
-LAB3 &CONTINUE
CP LOG
```

The following CMS statements are required to initialize the IDMS database. It is recommended that these statements not be kept in an EXEC procedure since if they are executed either accidentally or intentionally they will destroy all data currently in the database.

```
LOAD IDMSINIT
GENMOD IDMSINIT
FILE SYSIPT TERM (RECFM F BLOCK 80)
FILE SYSLST TERM (RECFM F BLOCK 132)
FILE GRDD8 DISK GRADES DATA (DSORG DA XTENT 25
BLOCK 3156)
IDMSINIT
```

When the prompt is typed by IDMSINIT enter the following:

```
PROCESS=TOTAL,DMCL=GRDMCL
```

APPENDIX B

IDMS SCHEMA

SCHEMA DESCRIPTION.

SCHEMA NAME GRDS200.

AUTHOR. DARRELL E MCINTOSH.

INSTALLATION. KANSAS STATE UNIVERSITY.

FILE DESCRIPTION.

FILE NAME IS GRDFILE

ASSIGN TO GRDDB.

FILE NAME IS JOURNAL

ASSIGN TO GRDJRN.

AREA DESCRIPTION.

AREA NAME IS GRDARA

RANGE IS 1001 THRU 1025

WITHIN FILE GRDFILE

FROM 1 THRU 25.

RECORD DESCRIPTION.

RECORD NAME IS MASTER-LINE.

RECORD ID IS 200.

LOCATION MODE IS CALC USING CLASS

DUPLICATES NOT ALLOWED.

WITHIN GRDARA AREA.

02 CLASS PIC X(10) .

02 PASS PIC X(8) .

02 POSSIBLES USAGE COMP-3 PIC 9(3) OCCURS 10.

02 BREAK-LECT USAGE COMP-3 PIC 99V99 OCCURS 5.

RECORD NAME IS LINE-NO.

RECORD ID IS 300.

LOCATION MODE IS CALC USING LINE

DUPLICATES NOT ALLOWED.
WITHIN GRDARA AREA.
02 LINE PIC X(4) .
02 PASSWORD PIC X(8) .
02 INSTRUCT PIC X(10) .
02 CLASS-TYPE PIC X(20) .
02 POSS USAGE COMP-3 PIC 9(3) OCCURS 9.
02 BREAK-LAB USAGE COMP-3 pic 99V99 OCCURS 5.
RECORD NAME IS STUDENT.
RECORD ID IS 400.
LOCATION MODE IS VIA STUD-SET.
WITHIN GRDARA AREA.
02 STUD PIC X(12) .
02 SSN USAGE COMP-3 PIC 9(9) .
02 CODE PIC X(8) .
02 GRADES USAGE COMP-3 PIC 9(3) OCCURS 19.
02 GRADE-LECT USAGE COMP-3 PIC 9.
02 GRADE-LAB USAGE COMP-3 PIC 9.
SET DESCRIPTION.
SET NAME LINE-SET.
ORDER SORTED.
MODE CHAIN.
OWNER MASTER-LINE NEXT POSITION 2.
MEMBER LINE-NO MANDATORY AUTOMATIC
NEXT POSITION 1
ASCENDING LINE DUPLICATES NOT.

SET NAME STUD-SET.

ORDER SORTED.

MODE CHAIN.

OWNER LINE-NO NEXT POSITION 2.

MEMBER STUDENT OPTIONAL AUTOMATIC

NEXT POSITION 2

LINKED TO OWNER OWNER POSITION 3

ASCENDING STUD DUPLICATES NOT.

SET NAME CODE-SET.

ORDER SORTED.

MODE CHAIN.

OWNER LINE-NO NEXT POSITION 3.

MEMBER STUDENT OPTIONAL MANUAL

NEXT POSITION 1

ASCENDING CODE DUPLICATES NOT.

SET NAME SSN-SET.

ORDER SORTED.

MODE CHAIN.

OWNER MASTER-LINE NEXT POSITION 1.

MEMBER STUDENT MANDATORY AUTOMATIC

NEXT POSITION 4

ASCENDING SSN DUPLICATES LAST.

SET NAME NAME-SET.

ORDER SORTED.

MODE CHAIN.

OWNER MASTER-LINE NEXT POSITION 3.

MEMBER STUDENT MANDATORY AUTOMATIC
NEXT POSITION 5
ASCENDING STUD Duplicates LAST.

APPENDIX C

IDMS DEVICE-MEDIA CONTROL

DEVICE-MEDIA DESCRIPTION.

DEVICE-MEDIA NAME GRDMCL

OF SCHEMA NAME GRDS200.

AUTHOR. DARRELL E MCINTOSH.

INSTALLATION. KANSAS STATE UNIVERSITY.

BUFFER SECTION.

BUFFER NAME GRDS-BUFFER

PAGE CONTAINS 3156 CHARACTERS

BUFFER CONTAINS 4 PAGES.

AREA SECTION.

COPY GRDARA AREA.

APPENDIX D

IDMS SUBSCHEMA

SUBSCHEMA IDENTIFICATION DIVISION.

SUBSCHEMA NAME IS GRDSSUB OF SCHEMA NAME GRDS200.

DEVICE-MEDIA NAME GRDMCL.

AUTHOR. DARRELL E MCINTOSH.

INSTALLATION. KANSAS STATE UNIVERSITY.

SUBSCHEMA DATA DIVISION.

AREA SECTION.

COPY GRDARA AREA.

RECORD SECTION.

COPY MASTER-LINE RECORD.

COPY LINE-NO RECORD.

COPY STUDENT RECORD.

SET SECTION.

COPY LINE-SET SET.

COPY STUD-SET SET.

COPY CODE-SET SET.

COPY SSN-SET SET.

COPY NAME-SET SET.

APPENDIX E
SAMPLE ON-LINE SESSION

GRADEXEC GRADER
FILE 'GRADER MODULE A' NOT FOUND.

NEW DATA SET
ENTER MASTER PASSWORD
MASTER

ENTER SEMESTER AND YEAR
SPRING 77
ENTER POSSIBLE VALUES FOR 10 LECTURE FIELDS
15 25 0 21 14 25 100 100 200 0
GRADER
A 3100
INVALID PASSWORD
GRADER
A 3100 LINE1
ENTER POSSIBLE VALUES FOR 9 LAB FIELDS
30 40 40 35 45 40 80 80 10
ENTER INSTRUCTOR NAME
MCINTOSH
ENTER CLASS TYPE
PL/I BUSINESS
ENTER STUDENT DATA
ADAMS 111111112 CODE1
JONES 111111113 CODEC
SMITH 111111114
MARTIN 111111115

GRADER
A 3101 LINE2
ENTER POSSIBLE VALUES FOR 9 LAB FIELDS
30 40 40 35 45 40 80 80 15
ENTER INSTRUCTOR NAME
SMITH
ENTER CLASS TYPE
COBOL
ENTER STUDENT DATA
ROBERT2 222222223
TYLER 222222224
ADAMS 222222225

GRADER
U 3101
UPDATE
C ROBERT2 NAME ROBERTS

I ROBERTS NAME
ROBERTS

C * AS1
ADAMS
12
ROBERTS
0
TYLER
14

C * CZ1
ADAMS
31
ROBERTS
28
TYLER
40

C * QZ1 +3
TYLER GRADE > MAXIMUM
I TYLER QZ1
43

E
GRADER
U 3100
UPDATE
C * PG1
ADAMS
26
JCNE
23
MARTIN
30
SMITH
27

C * EXT 10

I JONES EXT
10

E
GRADER
T
END OF ON_LINE SESSION
R;

GRADEXEC ROSTER
FILE 'ROSTER MCDULE A' NOT FOUND.

ENTER PASSWORD
MASTER
LIST BY LINE NUMBER?
Y
PRINT GRADES?
N
LIST BY CODE?
Y
ENTER LINE_NUMBER, *, CR END
*
LINE NUMBER 3100 BEING PRINTED
LINE NUMBER 3101 BEING PRINTED
NO STUDENTS WITH CODES IN THIS LINE
ENTER LINE_NUMBER, *, CR END
END
END OF ROSTER
PRT FILE 9838 TO OS
R;
GRADEXEC ROSTER

ENTER PASSWORD
LINE1
PRINT GRADES?
Y
LIST BY CODE?
Y
ENTER LINE_NUMBER, *, OR END
*
LINE NUMBER 3100 BEING PRINTED
ENTER LINE_NUMBER, *, OR END
3101
ACCESS NOT AUTHORIZED
ENTER LINE_NUMBER, *, OR END
END
END OF ROSTER
PRT FILE 9839 TO OS
R;

APPENDIX F
SAMPLE GRADE REPORTS

SPRING 77
 CS 20X GRADES - LINE # 3100
 PL/I BUSINESS
 05/16/77

NAME	LAB										LECTURE										SPC	TOT	PERCENT	GRD
	PG1	PG2	PG3	PG4	PG5	CZ1	QZ2	QZ3	EXT	TOT	PERCENT	GRD	AS1	AS2	AS3	AS4	AS5	AS6	EX1	EX2				
MCINTOSH	30	40	40	35	45	40	80	80	10	36	90.00	15	25	0	21	14	25	100	100	200	0	0	0.00	
ADAMS	26	M	M	M	M	M	M	M	10	33	82.50	M	M	M	M	M	M	M	M	M	M	0	0	0.00
JONES	23	M	M	M	M	M	M	M	10	40	100.00	M	M	M	M	M	M	M	M	M	M	0	0	0.00
MARTIN	30	M	M	M	M	M	M	M	10	37	92.50	M	M	M	M	M	M	M	M	M	M	0	0	0.00
SMITH	27	M	M	M	M	M	M	M	10	37	92.50	M	M	M	M	M	M	M	M	M	M	0	0	0.00

SPRING 77	CS 20X GRADES - LINE # 3100																				05/16/77				
	PL/I BUSINESS																								
	LAB										LECTURE														
NAME	PG1	PG2	PG3	PG4	PG5	QZ1	QZ2	QZ3	EXT	TOT	PERCNT	GRD	AS1	AS2	AS3	AS4	AS5	AS6	EX1	EX2	EX3	SPC	TOT	PERCNT	GRD
MCINTCSH	30	40	40	35	45	40	80	80	10			15	25	0	21	14	25	100	100	200	0				
CODE0	23	M	M	M	M	M	M	M	10	33	82.50	M	M	M	M	M	M	M	M	M	M	0		0.00	
CODE1	26	M	M	M	M	M	M	M	10	36	90.00	M	M	M	M	M	M	M	M	M	M	0		0.00	

SPRING 77
 CS 20X GRADES - LINE # 3101
 CCROL
 05/16/77

NAME	LAB										LECTURE										TOT	PERCENT	GRD
	PG1	PG2	PG3	PG4	PG5	C21	Q22	Q23	EXT	TOT	PERCENT	GRD	AS1	AS2	AS3	AS4	AS5	AS6	EX1	EX2			
SMITH	30	40	40	35	45	40	00	80	15				15	25	0	21	14	25	100	100	200	0	
ADAMS	M	M	M	M	M	34	M	M	M	34	85.00		12	M	M	M	M	M	M	M	M	M	12
RCBERTS	M	M	M	M	M	31	M	M	M	31	77.50		0	M	M	M	M	M	M	M	M	M	0
TYLER	M	M	M	M	M	43	M	M	M	43	107.50		14	M	M	M	M	M	M	M	M	M	14
																							93.33

SPRING 77		CS 20X GRADES - LINE # 3100 PL/I BUSINESS														05/16/77													
NAME	PG1	PG2	PG3	PG4	PG5	LAB					TOT	PERCENT	GRD	LECTURE										TOT	PERCENT	GRD			
						Q21	Q22	Q23	EXT	RC				AS1	AS2	AS3	AS4	AS5	AS6	EX1	EX2	EX3	SPC						
MCINTOSH	30	40	40	35	45	40	80	80	10	RC	10	36	93.00	P	15	25	0	21	14	25	100	100	200	0					
ADAMS	26	M	M	M	M	M	M	M	10	M	M	10	36	93.00	P	M	M	M	M	M	M	M	M	M	0	0	0.00	F	
JONES	23	M	M	M	M	M	M	M	10	M	M	10	33	82.50	B	M	M	M	M	M	M	M	M	M	0	0	0.00	F	
MARTIN	30	M	M	M	M	M	M	M	10	M	M	10	40	100.00	A	M	M	M	M	M	M	M	M	M	0	0	0.00	F	
SMITH	27	M	M	M	M	M	M	M	10	M	M	10	37	52.50	A	M	M	M	M	M	M	M	M	M	0	0	0.00	F	

CS 20X GRADES - LINE # 3101
COROL

05/16/77

[illegible]

SPRING 77

CS 20X GRADES

05/16/77

NAME	LAB										LECTURE										LINE				
	PG1	PG2	PG3	PG4	PG5	QZ1	QZ2	QZ3	EXT	TOT	PERCENT	GRD	AS1	AS2	AS3	AS4	AS5	AS6	EX1	EX2		EX3	SPC	TOT	PERCENT
ADAMS	26	M	M	M	M	M	M	M	10	36	90.00	M	M	M	M	M	M	M	M	M	M	M	0	0.00	3100
ADAMS	M	M	M	M	M	34	M	M	M	34	85.00	12	M	M	M	M	M	M	M	M	M	M	12	80.00	3101
JONES	23	M	M	M	M	M	M	M	10	33	82.50	M	M	M	M	M	M	M	M	M	M	M	0	0.00	3100
MARTIN	30	M	M	M	M	M	M	M	10	40	100.00	M	M	M	M	M	M	M	M	M	M	M	0	0.00	3100
ROBERTS	M	M	M	M	M	31	M	M	M	31	77.50	0	M	M	M	M	M	M	M	M	M	M	0	0.00	3101
SMITH	27	M	M	M	M	M	M	M	10	37	92.50	M	M	M	M	M	M	M	M	M	M	M	0	0.00	3100
TYLER	M	M	M	M	M	43	M	M	M	43	107.50	14	M	M	M	M	M	M	M	M	M	M	14	93.33	3101

APPENDIX G

USERS' GUIDE

GRADER Program

The GRADER program allows the user to add line numbers and students to the database, change values in the database, move students from one line number to another, and to remove students or entire line numbers from the database.

The program operates in either a MASTER or USER mode depending on the password entered at the beginning of the on-line session. A person in the USER mode has access only to those line numbers with a password matching the password entered by the user and has the use of a limited number of commands. A person in the MASTER mode has access to all line numbers in the database and also has the use of several more powerful commands in addition to those available to a USER. All commands in this guide indicate if they are available in USER or MASTER mode.

A number of commands can be used only after the UPDATE command has been entered. The UPDATE command simply notifies the GRADER program that the commands following it apply only to the line number specified in the UPDATE command. To end operations on the current line number use the END command. Another line number may then be UPDATED or non-UPDATE commands may be entered.

All commands are format-free with blanks used as delimiters.

If the database has not been initialized when the GRADER program begins execution it is assumed the user is loading the database and the user's password becomes the master password. The user will also be prompted to enter the semester and year (20 characters) and possible scores for the 10 lecture fields (1 to 3 digits).

ADD

GRADER COMMANDS

Purpose:

The ADD command allows addition of new line numbers and students to the database.

Format:

- (1) A <line number>
- (2) A <line number> <password>
- (3) A <name> <ssn> <code>

Usage:

Format (1) allows a person in the USER mode to add a new line number to the database. The user's password becomes the password for the new line number.

Format (2) allows a person in the MASTER mode to add a new line number to the database. The password specified becomes the password for the new line number.

Format (3) allows a person in UPDATE to add a new student to the line number.

For formats (1) and (2), after the line number has been added the message 'ENTER STUDENT DATA' is typed. At this point data for students may be entered one at a time in the form

<name> <ssn> <code>

After all student data has been entered type a blank to terminate insertion of students. The same rules that apply to format (3) apply here.

Notes:

- (1) This command is available in both USER and MASTER modes. When in USER mode format (1) or (3) must be used, while in MASTER mode format (2) or (3) must be used.
- (2) Line number is the four-digit number of the class to be added.
- (3) Name is the student's name to be added and is limited to 12 characters.
- (4) SSN is the student's ID number and must be nine numeric characters long.

- (5) Code is an optional 1 to 8-character code used for posting students grades and may consist of any printable characters except #, ", & or @.
- (6) If the code field is omitted grades for that student will not be listed by code.
- (7) The code may be entered at some other time by using the CHANGE command. The code may be deleted by using the CHANGE command and using a blank code.
- (8) When adding a new line number the user will be prompted to enter the instructor's name (12 characters), class type (20 characters), and possible scores for the 9 lab grades (1 to 3 digits each).
- (9) The command

A 3131 HELP

adds line number 3131 to the database with a password of 'HELP'. At this point the user would be prompted to enter data for students in the new line number. Student data should be entered as

JONES 123456789 MYSCORE

- (10) The sequence

U 3132

A SMITH 987654321

would add a student named Smith to the line number 3132 without an optional code.

Purpose:

The BREAKPOINTS command allows the user to list the breakpoints for determining letter grades or to change the breakpoints.

Format:

B <LIST>

Usage:

When the BREAKPOINTS command is used while in UPDATE, breakpoints for that line number will be listed or modified. If in MASTER mode and not in UPDATE breakpoints for lecture grades will be listed or modified.

If LIST is specified the current breakpoints are typed. If LIST is not specified the message 'ENTER NEW BREAKPOINTS' is typed. The user must then enter four breakpoints separated by blanks for grades of A, B, C, and D.

Notes:

- (1) This command may be used in either USER or MASTER mode.
- (2) Breakpoints are initially set for lecture when the database is created and for each lab section when a line number is added. The breakpoints are initially set at the following:

A	91
B	81
C	71
D	61

- (3) If changing the breakpoints the user will be prompted to enter all 4 new breakpoints.
- (4) The sequence

```
B
ENTER NEW BREAKPOINTS      (program response)
90 80 70 60
```

would establish new breakpoints as indicated.

CHANGE

GRADER COMMANDS

Purpose:

The CHANGE command allows modification of a student's data.

Format:

C <name|*> <field> <value>

Usage:

The CHANGE command may be used in UPDATE in either USER or MASTER mode to modify a single student's data or the data of all students in the line number.

The CHANGE command may be used in MASTER mode while not in UPDATE to modify data for all students in the database.

Notes:

- (1) If name is entered it must be the up-to-12-characters name of the student whose data is to be modified. If a name is entered, only one student is affected during the CHANGE operation.
- (2) If * is entered while in UPDATE all students in the line number will be modified for the field indicated. If a value is given in this case all students will be updated by the value. If no value is given the user is prompted with each student's name and a value may then be entered.
- (3) If * is entered while in MASTER mode but not in UPDATE a value must be entered and the value will be applied to all students in the database.
- (4) Field may be any of the names of the grade fields appearing on the ROSTER grade report or any of the following:

FIELD	VALUE
NAME	12 characters
SSN	9 digits
CODE	8 characters
LECT	ADD, WDP, WDF
LAB	ADD, WDP, WDF

- (5) If a lab or lecture grade (LAB or LECT) is not given, grades on the ROSTER grade report are determined from break points. WDP or WDF will appear on the grade report if entered in the database. If WDP or WDF has been entered for a student and it is desired to compute his grade using breakpoints again, change the appropriate lecture or lab grade to ADD.
- (6) Values for grades may be 1 to 3 digits, either signed or unsigned, or the letter M. Unsigned values are entered directly into the database. Signed values are added to the current value for that field already in the database. If no value currently exists in the database for a field and a signed value is entered, the field remains unchanged. If M is entered, the grade is changed to Missing.
- (7) The C * form is especially useful when adding or subtracting points for all students. For example, the sequence

```
U 3110
C * QZ1 +5
```

adds five points to the scores of all persons in line number 3110 who have scores for lab quiz 1. The grades for any students with missing scores for lab quiz 1 will remain unchanged.

- (8) The C * may also be used to enter values in the same field for all students in a single line number. The sequence

```
U 3220
C * QZ2
```

enables the user to enter grades for lab quiz 2 for all students in the class. A student's name is typed by the program. The user then enters his grade or, if there is no grade for the student, a M.

- (9) Particular attention should be paid to how percentages are computed on the ROSTER grade report. If a student has a missing score in a particular field that field is not used in computing his percentage, therefore, if a student failed to do an assignment or missed a test a value of zero should be entered so it will be reflected

in his percentage.

- (10) As scores are entered or modified in the database the new score is compared against the maximum points possible for that field. If the new score is greater than the maximum possible the score will be entered into the database and a warning message will be printed.

Purpose:

The CHANGE-MASTER command allows the user to change various fields in the Master record.

Format:

P <field> <value>

Usage:

This command may be used to change fields in the Master record.

Notes:

- (1) This command may be used only in Master mode while not in Update.
- (2) The field may be any of the lecture grade names given on the ROSTER grade report or one of the following with the corresponding values:

CLASS	20 characters (semester and year)
PASSWORD	1 to 8 characters (new password)

DROP

GRADER COMMANDS

Purpose:

The DROP command allows the user to remove a student from the database.

Format:

D <line> <name>

Usage:

This command completely removes the student from the database.

Notes:

- (1) This command may be used only in the Master mode.
- (2) Line is the line number of the class the student is in.
- (3) Name is the name of the student to be dropped from the database.

END

GRADER COMMANDS

Purpose:

The END command ends updating of records for a line number and allows other line numbers to be updated or non-UPDATE commands to be entered.

Format:

E

Usage:

The END command terminates updating of records for a line number.

INQUIRE

GRADER COMMANDS

Purpose:

The INQUIRE command allows the user to determine the value of any field in a student's record.

Format:

- (1) I <name> <field>
- (2) I <line number> <name> <field>

Usage:

Format (1) is used in UPDATE to inquire into a student's record.

Format (2) is used when not in UPDATE to query a student's record.

Notes:

- (1) This command may be used in either USER or MASTER mode.
- (2) If in USER mode, the user's password must match the password for the line number when format (2) is used.
- (3) Line number is the four-digit number of the class the student is in.
- (4) Name is the student's name whose record is to be queried.
- (5) Field is the name of the field to be queried. These values are the same as for the CHANGE command.
- (6) The sequence

```
U 3131
I JONES QZ2
```

would result in the score for lab quiz 2 being printed.

- (7) If not in Update, the command

```
I 3131 JONES QZ2
```

would give the same results as above.

Purpose:

The LINENUMBER-CHANGE command allows the user to modify fields in the Line number record except the grade breakpoints.

Format:

- (1) L <field> <value>
- (2) L <line number> <field> <value>

Usage:

Format (1) is used while in Update to change the possible number of points for lab grade fields.

Format (2) is used when in Master mode and not in Update to change any field for a line number.

Notes:

- (1) Field for format (1) may be the name of any of the lab grade fields as specified on the ROSTER grade report or PASS.
- (2) Value for format (1) may be a 1 to 3 digit number specifying the possible points for the field or the new 1 to 8 character password.
- (3) Field for format (2) may be any of the lab grade fields or any of the following with the appropriate value:

INSTR	12 characters
LINE	4 digits
TYPE	20 characters

MOVE

GRADER COMMANDS

Purpose:

The MOVE command may be used to move a student from one line number to another.

Format:

M <name> <from-line> <to-line>

Usage:

This command is used to move a student from one line number to another within the database.

Notes:

- (1) This command may be used only in the Master mode.
- (2) If the student's name is the same as a name in the to-line number the student is not moved and an error message is printed.
- (3) If the student to be moved has a code entered, the student is entered into the set of students in the to-line number with codes. If this action would cause a duplicate code, the student is not entered into the code set and a message is printed.
- (4) The command

M JONES 3131 3132

would move Jones from line number 3131 to line number 3132.

REMOVE

GRADER COMMANDS

Purpose:

The REMOVE command may be used to remove a line number from the database.

Format:

R <line>

Usage:

This command allows a line number and all students in it to be removed from the database.

Notes:

- (1) This command may be used only in the Master mode.
- (2) If any students currently exist in the line number a message is printed and the user is given the option of continuing with the removal or to allow the line number to remain in the database. A response of 'y' will cause the line number and all students in it to be removed from the database. A response of 'n' will leave the students and line number in the database.

TERMINATE

GRADER COMMANDS

Purpose:

The TERMINATE command is used to terminate the on-line session of the GRADER program.

Format:

T

Usage:

The TERMINATE command terminates the on-line session of the GRADER program.

Notes:

- (1) The TERMINATE command may be used in either USER or MASTER mode.
- (2) This command may be entered only when not in UPDATE.

UPDATE

GRADER COMMANDS

Purpose:

The UPDATE command allows updating of data in a specific line number.

Format:

U <line number>

Usage:

The UPDATE command readies the GRADER program to update data for a specific line number. A check is made to insure a person in USER mode is authorized access to the line number by determining whether the password he signed on with matches the password for the line number.

Notes:

- (1) Line number is the four-digit number of a class loaded in the database.
- (2) This command is available in both USER and MASTER modes.
- (3) The program remains in UPDATE until the END (e) command is entered. Commands used prior to END must be valid for UPDATE.

ROSTER Program

The ROSTER program is used to print grade reports for students in the database. Several options allow the user to select the output desired.

The ROSTER program operates in either a Master or User mode depending on the password entered when execution of the program begins. A person in the Master mode may access all data in the database while a person in the User mode may access only those line numbers with passwords matching the one entered when the program began executing.

The user is prompted with several questions concerning which options are desired. These queries should be answered with a 'y' or 'n'.

If in the Master mode the user will be asked if a listing is desired by line number. A 'y' response will cause all reports produced to list students by line number. If a 'n' is entered an alphabetical listing of all students in the database will be generated. Before the report is made the user will be asked if a letter grade is to be computed and printed for each student.

If in Master mode and the decision was made to list by line number or if in User mode, several options are available. The user is asked if letter grades are to be computed and printed and if a report is to be generated listing grades in alphabetical order by code. A request is then made for the user to enter a line number, an asterisk, or 'end'. If a line number is specified a grade report for that line number by student name and code, if requested, are generated. If an asterisk is given grade reports by name and code, if requested, are generated for each line number with a password matching the user's password if in User mode or for all line numbers if in Master mode. 'end' terminates execution of the ROSTER program.

Total points and percentages on the grade report are computed as follows: If a student has a grade for a field, those points and the corresponding possible points are accumulated. Any missing grades are ignored along with the possible points for that grade. Percentages are then computed by dividing the total points earned by the total points possible.

Letter grades are assigned as follows. If a student has a grade of WDP or WDF that grade will be placed on the grade report. Otherwise, the appropriate breakpoints are used with the percentage of points to determine the letter grade.

CMS Operations

The GRADER program requires a Virtual Machine consisting of 340K, therefore, when logging onto CMS the following should be entered:

LOGON CMS 340K

If only the ROSTER program is to be run the 340K may be omitted.

To execute an application program enter the following while under CMS:

GRADEXEC program-name spool-to

where program-name is GRADER or ROSTER. Spool-to is the name of the Virtual Machine to which the grade report from the ROSTER program is to be spooled. If omitted, the print file will be spooled to OS for later printing using the VMPRT program.

To sign off CMS enter the following:

WRAPUP option

where option is 'Y' or 'N' to indicate if the updated database is to be saved in the File Manager space. If a 'Y' is coded the database will be saved; if a 'N' is coded the database will not be saved. In either case the user will also be logged off of CMS.

AN ON-LINE SYSTEM FOR
MAINTAINING GRADES FOR
BASIC COMPUTER SCIENCE CLASSES

by

DARRELL E. MCINTOSH

B.I.E., Auburn University, 1971

AN ABSTRACT OF A MASTER'S REPORT
submitted in partial fulfillment of the
requirements for the degree
MASTER OF SCIENCE
Department of Computer Science

KANSAS STATE UNIVERSITY

Manhattan, Kansas

1977

The purpose of this paper is to discuss the design and implementation of an on-line system to maintain grades for the CS 200 lecture classes and corresponding computer language laboratories.

The system runs on the IBM System/370 under the Conversational Monitoring System (CMS). The database containing the students' grades is managed by Cullinane Corporation's Integrated Database Management System (IDMS).

The on-line system consists of two application programs. The first program is used to load, update and query the database. The second program produces grade reports either alphabetically for all students in the database or alphabetically by student name for each line number. If listed by line number a report may also be created by an optional eight-character code for students desiring to have their grades posted.

The system utilizes a password system that allows teaching assistants access only to records for students in their laboratories while the lecture instructor may access all records in the database.