DESIGN OF A DATA BASE MANAGEMENT SYSTEM FOR STUDENT PROGRAM DEVELOPMENT AND MAINTENANCE

Kenneth Mathew Golaszewski

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

DESIGN OF A DATA BASE

MANAGEMENT SYSTEM FOR STUDENT

PROGRAM DEVELOPMENT AND MAINTENANCE

by

Kenneth Mathew Golaszewski

June 1981

Thesis Advisor: N.F. Schneidewind

Approved for public release; distribution unlimited

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Design of a Data Base Management System for Student Program Development and Maintenance		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June 1981 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Kenneth Mathew Golaszewski Lieutenant, USN		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, CA 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, CA 93940		12. REPORT DATE June 1981 13. NUMBER OF PAGES 47
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, CA 93940		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) automated student academic program development and counciling system		

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

The purpose of this thesis is to develop a data base orientated system that aids the Academic Associates and Curricular Officers in program development and course selection for individual students within particular curricula. The data base system is written utilizing the Virtual Machine Facility/370 (VM/370) operating system and the IBM 3033 computer at the

Naval Postgraduate School. The automation of student curriculum matrices, course offerings, and tentative course scheduling is presented. A system user's manual for the IBM 3278 terminal is provided.

Approved for public release; distribution unlimited

Design of a Data Base Management System for Student Program Development and Maintenance

by

Kenneth Mathew Golaszewski
Lieutenant, United States Navy
B.S., S.U.N.Y. Maritime College, 1974

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
June 1981

ABSTRACT

The purpose of this thesis is to develop a data base orientated system that aids the Academic Associates and Curricular Officers in program development and course selection for individual students within particular curricula. The data base system is written utilizing the Virtual Machine Facility/370 (VM/370) operating system and the IBM 3033 computer at the Naval Postgraduate School. The automation of student curriculum matrices, course offerings, and tentative course scheduling is presented. A system user's manual for the IBM 3278 terminal is provided.

TABLE OF CONTENTS

I. INTRODUCTION

In an effort to improve student advicement and course scheduling procedures, an automated student advicement data management system has been developed. The primary objectives of this system are:

1. Development and maintenance of student academic programs within a given curriculum.

2. Tailoring of academic student programs to the individual student's degree requirements and interests.

3. Assisting the Academic Associate in his student counciling efforts as well as updating curricular programs.

4. Provide Curricular Officers with a readily available data base of student, course and schedule data, within their particular curricula.

The recently installed IBM 3033 AP System at the Naval Postgraduate School uses the Virtual Machine Facility/370 (VM/370) operating system. The components of VM/370 which are pertinent to this thesis are the control program (CP) and the conversational monitor system (CMS).

The conversational monitor system (CMS) is the controlling software for this academic student program development and maintenance system. CMS is a single-user operating system, that is capable of being executed in one or more virtual machines. It provides a wide variety of general-purpose, conversational timesharing functions that allows one to create, modify, and debug programs. Its use within this

system is as a manipulator of data files. By utilizing the XEDIT command the CMS editor is invoked and allows the user to create and modify data files.

The process of developing a student program, tailoring that program to the needs and desires of students, as well as complying with the Naval Postgraduate School's requirements is greatly improved by the use of the computer aided data management system.

II. THE STUDENT ADVISING PROBLEM

All incoming students are preregistered in the basic first quarter courses of their curriculum prior to arrival at the Naval Postgraduate School. This initial enrollment is made by the cognizant Curricular Officer in order to satisfy Special Assistant For Programs requirements. After the student arrives at the school and reports to his Curricular Officer, all subsequent program development is a manual process. This process typically has the student introduced to his core course matrix by the Curricular Officer, who appraises the student of his academic program requirements. The student develops an academic program with the Academic Associate. The student returns to his Curricular Officer with the program approved by the Academic Associate. The Curricular Officer fills out an Academic Program form (see Appendix A). This form is forwarded to the Special Assistant For Programs where it serves as input data to course scheduling and other computer programs.

It is important to note that student program development is a dynamic process. Students do not enter an academic pipeline and follow a rigid program without deviations. Instead, they begin with an academic program that consists of a set of core courses. Based upon the student's prior education, certain courses may be validated. As the student progress through his academic program and his goals become firmer, follow-on courses initially indicated for enrollment may be deleted, added or substituted. This tailoring and updating of a student's academic program may occur on a quarterly basis. The

ability to react and adapt to the fluctuating desires of the student and curriculum changes necessitates responsive administrative support.

The academic program development process can be made more efficient and more responsive if, instead of keeping a stack of hard copy Academic Record forms, these academic records are stored in a data base and maintained on the Naval Postgraduate School's IBM 3033 computer system. The other program development tools used by Curricular Officers and Academic Associates for development and maintenance of academic programs are the Naval Postgraduate School catalog, curricula matrices, Looseleaf Course Catalog, and the Tentative Course Schedule. These tools can also be converted to a data base and made available to the user.

The automation of this manual program development process provides an effective means to offset the ever-increasing clerical and administrative costs associated with student program development and counseling. Curricular Officers and Academic Associates will then have the capability of calling up and modifying any one of their student's academic programs from a remote computer terminal. Student counseling sessions can be greatly enhanced by directly accessing a course catalog file and a course offering file in order to obtain information regarding a course's particular prerequisites or availability.

III. SCOPE OF THE PROJECT

This thesis is specifically directed at presenting a data base management system that is dedicated to the development and maintenance of student academic programs. The procedures necessary for the development of the system's data base are provided in a User's Manual (see Chapter V). This manual is directed towards the user who will be working with the IBM 3278 computer terminal which is directly connected to the Naval Postgraduate School's IBM 3033 computer system.

The Registrar, Scheduler and Department Chairmen take the reports provided by the Special Assistant For Programs and develop course, professor and student schedules. The Special Assistant For Programs receives course enrollment requirements from Curricular Officers. It is these Curricular Officers and Academic Associates that are the primary users of this automated student program development and maintenance system.

It is acknowledged that with implementing any new system the capacity and utility of that system may not be recognized by all who come in contact with it. The paramount problem then becomes one of motivating acceptance and willingness to use the newly developed system.

IV. SYSTEM APPROACH

The administrative process of registering students for various courses, is as previously stated, directly aimed at fulfilling the needs of the Special Assistant For Programs. This process is extremely labor-intensive with regard to the efforts required of Curricular Officers and Academic Associates. The information sent to the Special Assistant For Programs is a detailed listing of courses requested by each student as indicated on the Academic Program form (see Appendix A). These forms contain vital identification information as well as a projected academic program that reflects those courses desired by the student as he progresses through his curriculum. This information is then utilized for student registration and course demand forecasting.

The existing program development system dictates that an Academic Program form be maintained for each student. The repetition experienced in completing this form for every student is significant, and readily lends itself to computerization. As Curricular Officers become aware of the impending arrival of perspective students, a new academic section begins to take form. It is proposed that as these new students become known, an appropriate student record be developed within the curriculum's student data base. These student records can then have a core course matrix appended (see Appendix B) thus forming, in a matter of minutes, a new student record with an appropriate tentative academic program. These newly created student records can in turn be grouped together to form a section that has not yet reported. This new

section information would then become immediately available for long range course load forecasting. The listing of a curriculum's student program data base for newly created sections could then be sent to the Special Assistant For Programs for preregistration.

Subsequent to the student's arrival, the program development and iterative process begins. An in-depth review of the student's prior academic background and academic desires is conducted by both the Curricular Officer and the Academic Associate. It has been policy that any alterations to the student's preregistered academic program be made on the student's Academic Program form, which was forwarded to the Special Assistant For Programs, so that his records could be updated. With student programs maintained within an on-line data base, these interviews could be conducted with the student having his academic program displayed at a computer terminal. Any necessary changes to the student's academic program resulting from the interview could be immediately entered by the interviewer and then the revised student program could be made available to the Special Assistant For Programs.

Concurrent with the administrative effort of developing student academic programs is a coordinated counseling responsibility which also is time intensive. Both the Curricular Officer and the Academic Associate must maintain a familiarity with curriculum requirements, course prerequisites, and with projected course offerings in order to keep a high degree of effectiveness. This counseling function requires a thorough knowledge of the Naval Postgraduate School catalog and/or the Naval Postgraduate School Looseleaf Course Catalog, the course offering document (Tentative Course Schedule) as well as

curriculum matrices with their alternatives.

When a student seeks counseling in order to select meaningful electives in an effort to ensure compliance with requirements set forth for granting of a degree, Academic Associates and Curricular Officers must juggle between course catalog, course offering and student program requirements. Only after serious consideration of curriculum requirements, course prerequisites, and the impact a course change might have on the remaining academic quarters of a particular curriculum, should an advisor make his recommendation.

A. THE INFORMATION MANAGEMENT SYSTEM

The data base that makes up the student program development process consists of (1) a student program file (STUDENT PRGM), (2) a course catalog file (CATALOG DATA), and (3) a course offering file (COURSE SCHED). There are two additional smaller files that help the user in developing the other three files. These files are (1) a core course matrix file (CORE$nnn DATA), where nnn represents the appropriate curriculum number, and (2) a course schedule header file (COURSE SCHED2). See the user's manual for an in-depth description of these files.

The software that accommodates student/course record development, updating and retrieval is the Conversational Monitor System (CMS). CMS provides a wide variety of general purpose, conversational time sharing functions that support creating and modifying data files. By utilizing the XEDIT command the CMS editor is invoked and thus facilitates the building of the student program, course catalog, and course schedule files. The user performs these editing functions

while in the XEDIT environment. Access to the data base is achieved through the use of the hardwired IBM 3278 terminal or via dial-up communications. The operational status of the computer plays a significant role in this access. The computer is scheduled for twenty-four hours of operation per day.

B. SECURITY AND PRIVACY

Two important aspects of a data base system are security and disaster recovery. Data base access is controlled by the use of an authorized USERID and PASSWORD. The USERID is used to identify a user to the VM/370 operating system. The password is used as a protection device to ensure that only authorized users will be given access to a particular virtual machine. Both USERID and PASSWORD can be obtained from the Computer Center's Services Office, IN-147.

Physical file security is provided by the computer center's operators and the computer itself. A security procedure implemented 19 May 1981 has a back-up copy of all Virtual Memory (VM) user's files made every Monday. This procedure serves as a back-up/recovery vehicle in the event that the computer system crashes or loses user files. Another security option would be to have the data base written onto tape periodically. This tape would be stored off-line and made available for updating upon request.

V. USER'S MANUAL

The purpose of this USER's Manual is to assist the system user in overcoming the idiosyncracies associated with working on the IBM 3278 terminal. Various data handling features and examples are presented. The Conversational Monitor System (CMS) commands utilized are described in the context of their use. A more exhaustive description of each command and its general use can be obtained by invoking the HELP facility. This is accomplished by typing HELP XXXX on the COMMAND LINE of the terminal, where XXXX represents the CMS command in question. Upon pressing the ENTER-KEY a complete description of the command will be displayed on the screen.

A. ESTABLISHING COMPUTER COMMUNICATIONS

It is assumed that the user is familiar with the basic operation and configuration of the IBM 3278 terminal. A valid USERID (user identification) and PASSWORD are needed. The unique USERID is a symbol that identifies the user's virtual machine to the Virtual Machine Facility/370 (VM/370) operating system. This along with a correct PASSWORD permits access to the computer system. The use of a PASSWORD is an added security feature ensuring that only authorized users are given access to one's virtual machine and files. USERID and PASSWORD can be obtained from the Computer Center's Accounting Services Office, In-147.

1. Log-on Procedure:

Figures 1 and 2 represent the terminal and screen of the IBM

3278 terminal. The features and steps sighted below for the log-on procedure are indicated on the figures.

a. Turn on the terminal using the 1-O switch (big red toggle-switch), located to the left of the screen. The '1' means on.

b. When the NPS logogram appears press the RESET-KEY.

c. Press the ENTER-KEY.

d. The screen will clear and CP READ will appear at the lower right of the screen (STATUS AREA), type:

L nnnnP

Where nnnn represent the user's 4-digit user number.

e. Press the ENTER-KEY.

f. A prompt for your PASSWORD will appear. Type it in, noting that the characters will not appear on the screen.

g. Press the ENTER-KEY.

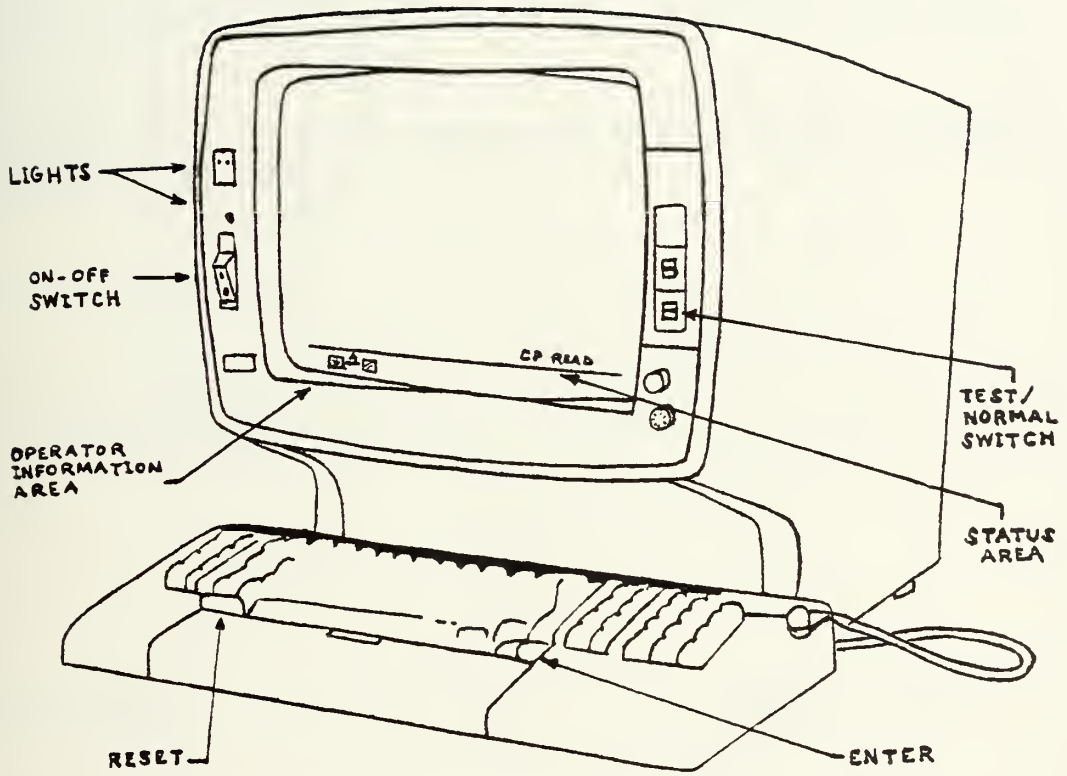
NOTES:

1. The terminal requires a few seconds to completely energize after being turned on. It is properly functioning when the three small red lights on the left of the screen are illuminated.

2. If a screenfull of characters appear rather than the NPS logogram, change the setting of the screen test switch from 'test' to 'normal'. This switch is located to the right of the screen and is blue in color.

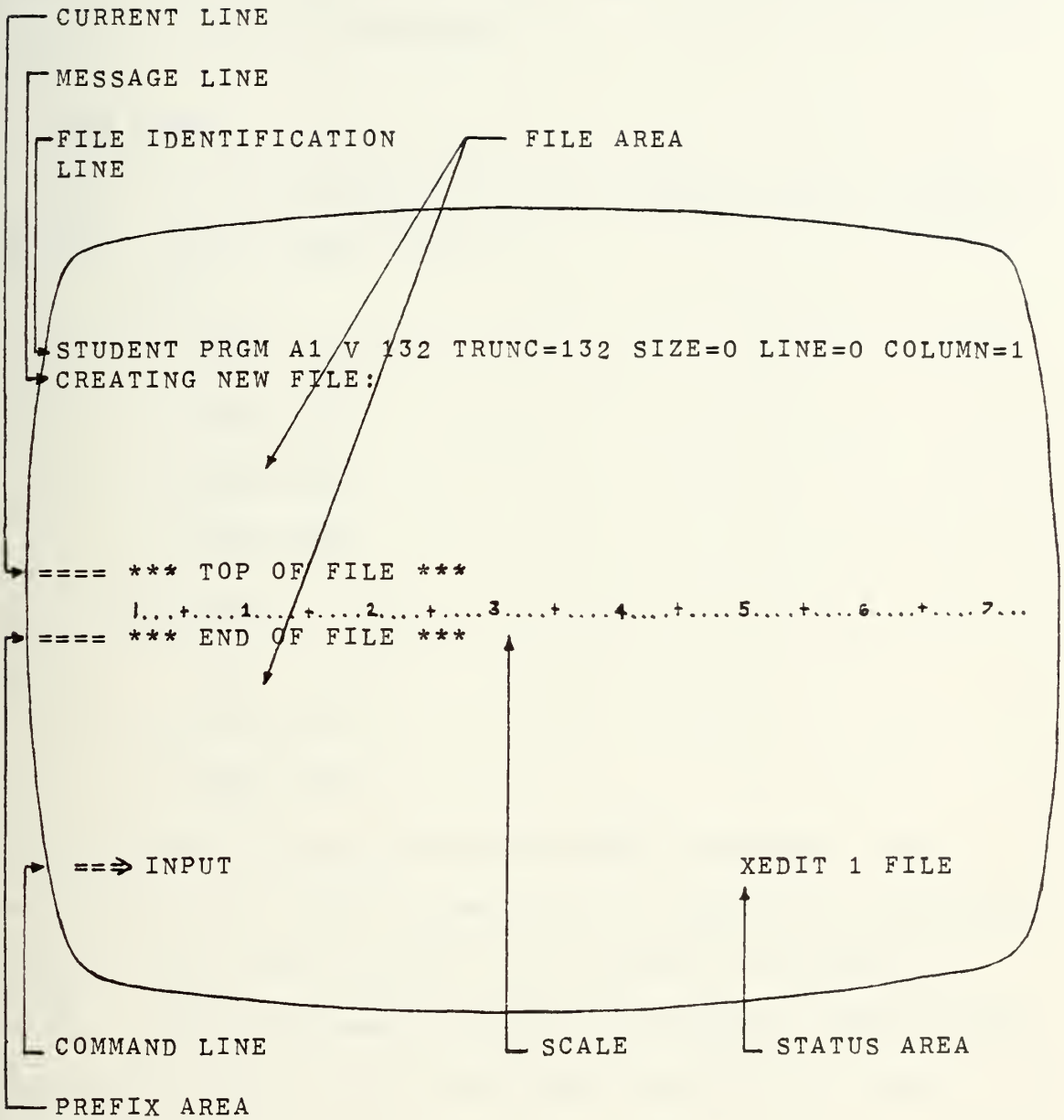
3. At this point the user should see within the Operater Information Area (see Figure 1) the following symbol:

A
4 --- ZZ



IBM 3278 TERMINAL

Figure 1



IBM 3278 SCREEN LAYOUT

Figure 2

This signifies that the user is logged-on and communicating with the computer. If any other symbol appears press the RESET-KEY and continue. The user is in the CMS environment.

B. FILE ACCESS

There are three primary files and two secondary files that make up the basic system. These files are:

1. Primary

[FN]	[FT]
a. STUDENT	PRGM
b. CATALOG	DATA
c. COURSE	SCHED

2. Secondary

[FN]	[FT]
a. CORE$367	DATA
b. COURSE	SCHED2

All files are accessed through the XEDIT environment. This environment can be called by issuing the CMS command XEDIT immediately followed by the identification of the data file to be created or modified. This CMS command is typed on the terminal's COMMAND LINE, located at the bottom left of the screen, identified by a large arrow (===>).

EXAMPLE:

```
===>  XEDIT STUDENT PRGM
```

After pressing the ENTER-KEY, notice that in the case of creating a new file that the file identification line appears on the first line

of the screen along with the message 'CREATING NEW FILE' on the second line (the message line). The cursor will be positioned at the command line.

C. CREATING THE STUDENT PROGRAM FILE

The student program (STUDENT PRGM) file is the mainstay of this system. It possesses an academic program for each student within a particular curriculum. Students are grouped into sections and student academic programs are subdivided into academic quarters. Each academic quarter lists those courses a student has taken, is presently taking or will take depending on how far along the student is within his academic program. Each student section begins with a roster. Section leaders and assistant section leaders are identified by one or two asterisks respectively. The procedure for creating this file is:

1. Create a new file by typing (XEDIT STUDENT PRGM) on the command line of the terminal.

2. Press the ENTER-KEY.

Note that a file identification line appears on the first line of the screen with the message 'CREATING NEW FILE' on the second line. The cursor is positioned on the command line (see Figure 2).

3. Enter the input mode.

This is accomplished by typing (Input or I) on the command line followed by pressing the ENTER-KEY. Note that the display splits with the lower half of the screen becoming the input zone. The cursor will be positioned on the first line of this zone. Any further cursor positioning can be accomplished through the use of four keys located to

the right of the main key-board. These keys cause the cursor to move forward, forward rapidly, backward, backward rapidly, up, and down. They can be identified by the arrow that appears on the face of the key. A scale appears on the screen beneath the current line. It is similar to the margin scale on a typewriter and marks columns 0 through 79 (see Figure 2). The procedure for skipping lines when in the input-mode is to (1) have the cursor positioned on a new line, (2) depress the space bar once, and (3) depress the Enter-Key. Repeat these three steps for each line to be skipped.

4. Enter the following data and perform the steps indicated:

a. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
5	CURRICULUM	COMPUTER SYSTEMS

b. Press the ENTER-KEY and skip two lines.

c. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
1	SECTION: XXXX	SECTION: PL11

Where XXXX is the section descriptor (i.e. PL03, CS01, WT93, etc.)

30	(label)	ENTRY QUARTER
45	(Quarter started)	WINTER 1981 (811)

Where 811 represents the first quarter of academic year 1981

d. Press the ENTER-KEY.

e. Enter the following data in the columns indicated.

(COLUMN)	(DATA)	(EXAMPLE)
30	(label)	* - INDICATES SECTION LEADER
30	(label)	** - INDICATES ASSISTANT SECTION LEADER

f. Press the ENTER-KEY and skip two lines.

g. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
5	XXXX ROSTER:	PL11 ROSTER:

Where XXXX is the section descriptor.

h. Press the ENTER-KEY and skip one line.

i. List the students that constitute section XXXX. The format being (last name, first initial. middle initial.), with two names per line beginning in columns 10 and 40 respectively.

j. At the end of the roster skip two lines.

k. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
1	(Label)	STUDENT PROGRAMS:

l. Press the Enter-Key and skip one line.

m. Create a student academic record for each student. This is accomplished by entering the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
1	STUDENT NAME	CRAMER, STEVEN R.
30	RANK	CAPT, LCDR, LT, LTCOL
35	SERVICE or COUNTRY	USN, USCG, RSK
40	SOCIAL SECURITY NUMBER	123-45-6789
55	DESIGNATOR	1110, 1360
62	SMC NO.	1234
68	CURRIC. NO.	#367, #368

(1) After entering the student curriculum number press the ENTER-KEY twice. This exits the input-mode and positions the cursor at the command line. The student record data is positioned at

the screen's current line.

(2) To insert the core course matrix beneath the newly created student record, the XEDIT subcommand (GET) is utilized.

EXAMPLE:

===> GET CORE$367 DATA

Where CORE$367 is the file name representing the core course matrix for the Computer Systems Management curriculum and DATA is the file type.

Upon pressing the ENTER-KEY, the core course matrix for the desired curriculum will be inserted beneath the student record (current line). This core course matrix resides within a separate file (see CREATING A CORE COURSE MATRIX). The message (EOF) meaning end of file, will be displayed to signify that the entire core course matrix file (CORE$367 DATA), has been inserted as requested.

n. Exit the input mode.

This is accomplished by pressing the ENTER-KEY twice. Note that the screen is no longer split and the cursor is positioned at the screen's command line.

5. With the file completed all that remains is to write it onto disk. This is accomplished by typing the subcommand (FILE) on the command line followed by pressing the ENTER-KEY. The user is returned to the CMS environment, the file is written onto disk and the edit session is terminated.

NOTES:

1. At the beginning of the editing session the user might

desire to issue the subcommand (AUTOSAVE nn), where nn is a number. This command causes the CMS editor to save the user's input after every nn change or addition.

D. CREATING THE COURSE CATALOG FILE

The course catalog file (CATALOG DATA) is similar in form to the Naval Postgraduate School's Looseleaf Course Catalog. An indicator is provided to reflect the last time the file was revised due to catalog updates. The file consists of a course listing grouped according to academic department. The primary access key for locating a particular course is the course number. The file is accessed through the XEDIT environment and a particular course can be located by utilizing the subcommands (Find or FINDUP). A complete description of these subcommands and their general use is provided in the section (LOCATING A SPECIFIC STUDENT/COURSE RECORD). Each course description consists of a course number, course title, hours, course definition, and prerequisites if applicable. The procedure for creating this file is:

1. Create a new file by typing (XEDIT CATALOG DATA) on the command line of the terminal.

2. Press the ENTER-Key.

Note that a file identification line appears on the first line of the screen with the message 'CREATING NEW FILE' on the second line. The cursor is positioned on the second line (see Figure 2).

3. Enter the input mode.

This is accomplished by typing (Input or I) on the command line followed by pressing the ENTER-KEY. Note that the display splits with the lower half of the screen becoming the input zone. The cursor will be positioned on the first line of this zone.

4. Enter the following data and perform the steps indicated:

a. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
45	(label)	REVISED:
55	DATE	5 DECEMBER 1980

b. Press the ENTER-KEY and skip two lines.

c. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
5	ACADEMIC DEPT.	COMPUTER SCIENCE DEPARTMENT:

d. Press the ENTER-KEY.

e. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
20	(label)	COURSE CODE:
35	XX	CS

Where XX is the two letter course code descriptor.

40	COURSE CODE DEFINITION	COMPUTER SCIENCE

f. Press the ENTER-KEY and skip one line.

g. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
1	COURSE NUMBER	CS 3020 (see note 1)
10	COURSE TITLE w/ HOURS	SOFTWARE DESIGN (3-2)

h. Press the ENTER-KEY.

i. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
10	COURSE DEFINITION	(see note 2)

j. Press the ENTER-KEY.

k. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
10	(label)	PREREQUISITES:
24	PREREQUISITES	(see note 3)

l. Press the ENTER-KEY and skip one line.

m. Repeat steps g through l for all courses offered by a particular academic department. Skip two lines between the end of a course description and the beginning of another academic department's course listing.

n. Exit the input mode.

This is accomplished by pressing the ENTER-KEY twice. Note that the screen is no longer split and the cursor is positioned at the screen's command line.

5. With the file completed all that remains is to write it onto disk. This is accomplished by typing the subcommand (FILE) on the command line followed by pressing the ENTER-KEY. The user is returned to the CMS environment, the file is written onto disk and the edit session is terminated.

NOTES:

1. The space between the two letter course code and the four digit number of the course number is required.

2. Course definitions will require more than one line,

therefore maintain a left margin at column 10 throughout the definition.

3. Not all courses possess prerequisites. where they are required and more than one line is needed to list them maintain a left margin in column 24.

4. At the beginning of the editing session the user might desire to issue the subcommand (AUTOSAVE nn), where nn is a number. This command causes the CMS editor to save the user a input after every nn change of addition.

E. CREATING THE COURSE SCHEDULE FILE

The course schedule file (COURSE SCHED) is similar in form to the Naval Postgraduate School's Tentative Course Schedule. An indicator is provided to reflect the last time the file was revised due to schedule updates. The file consists of a course listing grouped according to academic department. The primary access key for locating a particular course is the course number. The file is accessed through the XEDIT environment and a particular course can be located by utilizing the subcommands (Find and FINDUP). A complete description of these commands and their use is provided in the section (LOCATING A SPECIFIC STUDENT/COURSE RECORD). Each course record consists of a course number, course title, hours, indicator as to what quarter the course is offered and course offering periodicity. The procedure for creating this file is:

1. Create a new file by typing (XEDIT COURSE SCHED) on the command line of the terminal.

2. Press the ENTER-KEY.

Note that a file identification line appears on the first line of the screen with the message 'CREATING NEW FILE' on the second line. The cursor is positioned on the command line (see Figure 2).

3. Enter the Input mode.

This is accomplished by typing (Input or I) on the command line followed by pressing the ENTER-KEY. Note that the display splits with the lower half of the screen becoming the input zone. The cursor will be positioned on the first line of this zone.

4. Enter the following data and perform the steps indicated:

a. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
45	(label)	REVISED:
55	DATE	15 SEPTEMBER 1980

b. Press the ENTER-KEY and skip two lines.

c. Enter the following data in the column indicated:

(COLUMN)	(DATA)	(EXAMPLE)
1	ACADEMIC DEPARTMENT	COMPUTER SCIENCE

d. Press the ENTER-KEY.

e. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
20	DEPT. CODE: nn	DEPT. CODE: 53

Where nn represents the academic department code.

f. Press the ENTER-KEY.

g. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
20	(label)	COURSE CODE:
34	XX	CS

Where XX is the two letter course code descriptor.

38	COURSE CODE DEFINITION	COMPUTER SCIENCE

h. Press the ENTER-KEY.

i. Insert the course schedule header utilizing the XEDIT sub-command (GET).

EXAMPLE:

===> GET COURSE SCHED2

Where COURSE is the file name and SCHED2 is the file type. This header is used to identify the beginning of a new course type (i.e. COMPUTER SCIENCE, MATHEMATICS, INFORMATION SYSTEMS, MANAGEMENT, etc.) (see CREATING THE COURSE SCHEDULE HEADER).

j. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
1	COURSE NUMBER	CS 0810
10	COURSE TITLE	THESIS RESEARCH
45	HOURS	0-0
50	INDICATOR - 'X'	(see note 1)
55	INDICATOR - 'X'	(see note 1)
60	INDICATOR - 'X'	(see note 1)
65	INDICATOR - 'X'	(see note 1)
67	PERIODICITY	ALL QTR (see note 2)

k. Press the ENTER-KEY.

l. Exit the input mode.

This is accomplished by pressing the ENTER-KEY twice. Note that the screen is no longer split and the cursor is positioned at the screen's command line.

5. With the file completed all that remains is to write it onto disk. This is accomplished by typing the subcommand (FILE) on the command line followed by pressing the ENTER-KEY. The user is returned to the CMS environment, the file is written onto disk and the edit session is terminated.

6. Repeat steps j through k for all courses that are offered by a particular academic department.

NOTES:

1. The indicator 'X' is used to signify that a particular course will be offered during the academic quarter flagged.

2. The periodicity of a course offering is indicated as follows:

(CODE)	(MEANING)
ALL QTR	EACH QUARTER
FALL/SP	FALL and SPRING
WNTR/SU	WINTER and SUMMER
FAL/SP/SU	FALL, SPRING and SUMMER
FAL/WN/SU	FALL, WINTER and SUMMER
WN/SP/SU	WINTER, SPRING and SUMMER (or any combination of the above)
AS REQ	AS REQUESTED or AS REQUIRED
*	An asterisk represents courses that will be given upon request, as reading courses only and only if instructors are available.

3. At the beginning of the editing session the user might desire to issue the subcommand (AUTOSAVE nn), where nn is a number. The command causes the CMS editor to save the user's input after every nn change or addition.

F. CREATING A CORE COURSE MATRIX FILE

The core course matrix file is used in conjunction with creating new student records. It reflects the core courses within academic quarters that are required for students of a particular curriculum. By calling this file with the XEDIT subcommand GET, an entire curriculum's core course matrix can be attached to a newly created student record.

File names can reflect the curriculum they represent (i.e. CORE$367, CORE$368, CORE$530, etc.), where the numbers 367, 368, and 530 represent unique curriculum descriptors. The procedure for creating this file is:

1. Create a new file by typing (XEDIT CORE$nnn DATA) on the command line of the terminal, where nnn represents the applicable curriculum number.

2. Press the ENTER-KEY.

Note that a file identification line appears on the first line of the screen with the message Creating New File on the second line. The cursor is positioned on the command line (see Figure 2).

3. Enter the input-mode.

This is accomplished by typing (Input or I) on the command line followed by pressing the ENTER-KEY. Note that the display splits with the lower half of the screen becoming the input zone. The cursor will be positioned on the first line of this zone.

4. Enter the following data and perform the steps indicated:

a. Skip one line.

b. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
10	(label)	ADVISOR:
19	ACADEMIC ADVISOR's NAME	
35	(label)	EMPHASIS AREA: (optional)
50	EMPHASIS AREA	(if applicable)

c. Press the ENTER-KEY.

d. The following data will be repeated for each academic quarter.

(COLUMN)	(DATA)	(EXAMPLE)
5	QTR nnn:	QTR 811:

Where nnn represents a particular academic quarter, which can be filled in or left blank.

e. Press the ENTER-KEY.

f. Repeat as necessary the following entries which reflect the core courses and emphasis/elective courses that constitute an academic quarter within a particular curriculum.

(COLUMN)	(DATA)	(EXAMPLE)
10	COURSE NUMBER	CS 2800
18	ACADEMIC CREDIT	(3-2)
30	COURSE TITLE	COMPUTER SCIENCE FUND.

Note that where emphasis courses are to be taken enter 'EMPHASIS COURSE' in lieu of the course number. This entry can be updated at a later date.

g. Press the ENTER-KEY.

h. Skip one line.

i. Exit the input-mode.

This is accomplished by pressing the ENTER-KEY twice. Note that the screen is no longer split and the cursor is positioned at the screen's command line.

4. With the file completed all that remains is to write it onto disk. This is accomplished by typing the subcommand (FILE) on the command line followed by pressing the ENTER-KEY. The user is returned to the CMS environment, the file is written onto disk and the edit session is terminated.

NOTES:

1. At the beginning of the editing session the user might desire to issue the subcommand (AUTOSAVE nn), where nn is a number. This command causes the CMS editor to save the user's input after every nn change or addition.

G. CREATING THE COURSE SCHEDULE HEADER FILE

The course schedule header file (COURSE SCHED2) is used in conjunction with the (CREATING THE COURSE SCHEDULE FILE). By calling this file with the XEDIT subcommand (GET), a header containing the labels for course number, course title, hours, the four academic quarters, and the periodicity of course offering will be inserted into the course schedule file beneath the academic department label. This header is used to identify and separate the course types offered by an academic department (i.e. department: Administrative Sciences, course types: Administrative Sciences, Communication Management, Information Systems and Management). The procedure for creating this file is:

1. Create a new file by typing (XEDIT COURSE SCHED2) on the command line of the terminal.

2. Press the ENTER-KEY.

Note that a file identification line appears on the first line of the screen with the message 'CREATING NEW FILE' on the second line. The cursor is positioned on the command line (see Figure 2).

3. Enter the input mode.

This is accomplished by typing (Input or I) on the command line followed by pressing the ENTER-KEY. Note that the display splits with the lower half of the screen becoming the input zone. The cursor will be positioned on the first line of this zone.

4. Enter the following data and perform the steps indicated:

a. Skip one line.

b. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
1	(label)	COURSE
10	(label)	COURSE TITLE
45	(label)	HRS
54	(label)	FALL
54	(label)	WNTR
59	(label)	SPRG
64	(label)	SUM
67	(label)	PERIOD

c. Press the ENTER-KEY.

d. Enter the following data in the columns indicated:

(COLUMN)	(DATA)	(EXAMPLE)
1	(label)	NUMBER
44	(label)	(STD)
50	(label)	80
55	(label)	81
60	(label)	81
64	(label)	81
67	(label)	ICITY

e. Press the ENTER-KEY.

f. Enter a string of astericks beginning in column 1 and ending in column 73.

g. Press the ENTER-KEY and skip one line.

h. Exit the input mode.

This is accomplished by pressing the ENTER-KEY twice. Note that the screen is no longer split and the cursor is positioned at the screen's command line.

5. With the file completed all that remains is to write it onto disk. This is accomplished by typing the subcommand (FILE) on the command line followed by pressing the ENTER-KEY. The user is returned to the CMS environment, the file is written onto disk and the edit session is terminated.

NOTES:

1. At the beginning of the editing session the user might desire to issue the subcommand (AUTOSAVE nn), where nn is a number. The command causes the CMS editor to save the user's input after every nn change or addition.

H. LOCATING A SPECIFIC STUDENT/COURSE RECORD

In order to locate a particular student record one must first access the STUDENT PRGM file, which contains the alphabetical listing of those students within a particular curriculum grouped according to section. The procedure for locating a specific student record is:

1. Access the STUDENT PRGM file (see FILE ACCESS, pg. 17.

2. Use the XEDIT subcommands (Find or FINDUP) whichever is appropriate.

a. The Find subcommand searches for the line that starts with the specified text that follows the command word, searching in a forward direction.

b. The FINDUP subcommand searches for the line that starts with the specified text that follows the command word, searching in a backward direction.

EXAMPLE:

Find CLAVELLI, DANIELLE
or
FINDUP CS 3020

As a result of the way in which the Find and FINDUP subcommands execute their record search algorithm, all key records (i.e. students within the STUDENT PRGM file and course designators within the CATALOG DATA and COURSE SCHED files) begin in column one.

3. If the need arises to search for a particular course or academic quarter within the student program file, the subcommand LOCATE should be utilized. Use the (LOCATE) subcommand to scan a file for a specified target, which (if found) becomes the new line. The search starts with the first line following the current line. The target of a LOCATE subcommand can be a 'string expression' defining a group of characters to be located. The characters in the string must be delimited by a character that does not appear in the string itself. However, when a string target is entered by itself (without the optional subcommand name LOCATE), the delimiter must be a diagonal (/). In the

following example a diagonal (/) is used.

EXAMPLE:

/IS 2000/

If the above command is issued while examining a student's academic program the line containing the course IS 2000 will become the new current line.

NOTES:

1. If after using the Find subcommand end-of-file (EOF) appears on the screen's current line without locating the record of interest along with the message (NOT FOUND) being displayed on the terminal's message line one of the following has occurred:

a. The record of interest resides within the file preceding the location at which the search was initiated. In this case use the XEDIT subcommand FINDUP.

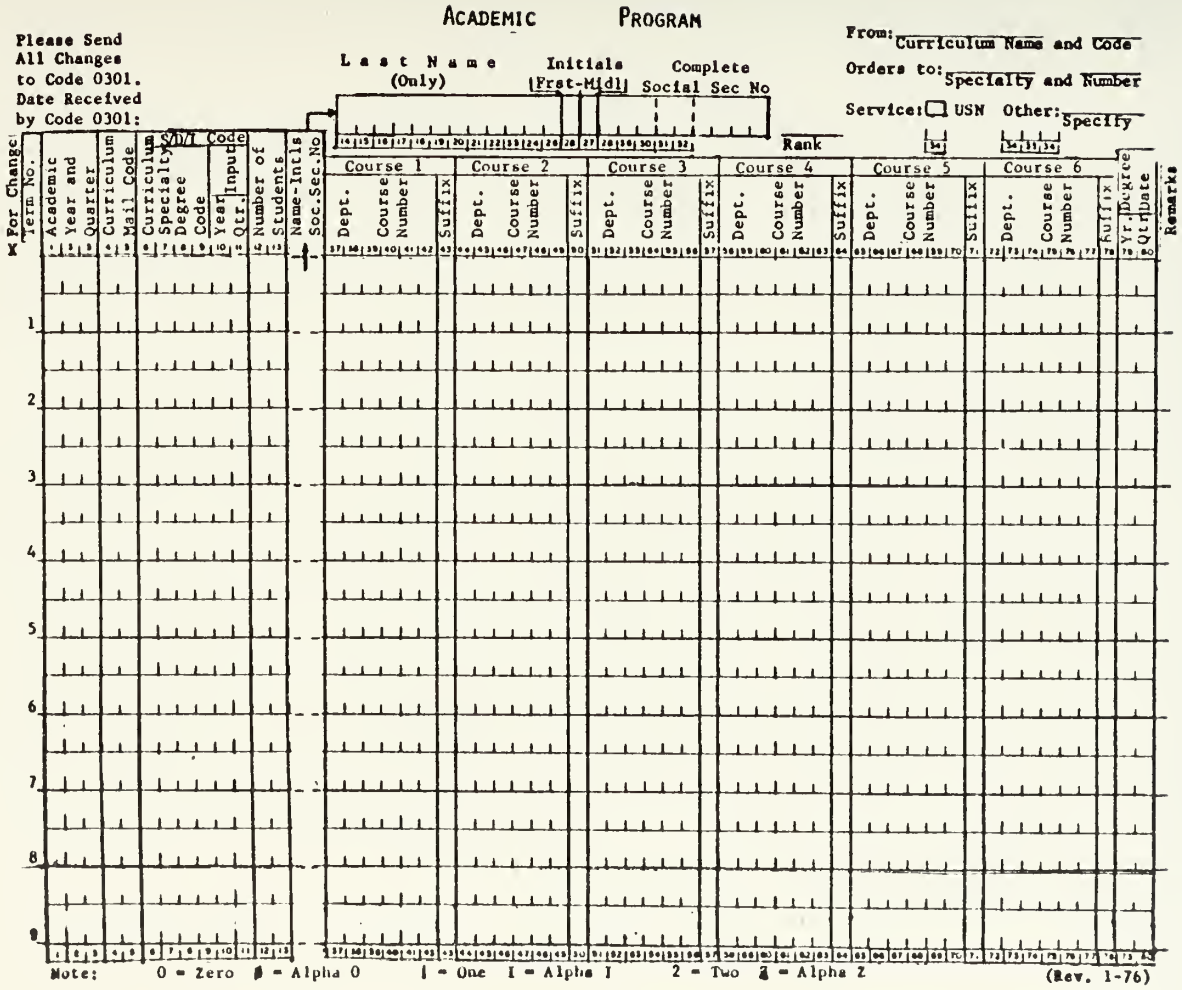
b. The name of the student was misspelled. In this case use either the XEDIT subcommand (Find or FINDUP) depending on the direction of search required.

c. No such record exists within the file, therefore the student record needs to be created (see CREATING A NEW STUDENT RECORD).

3. Once a particular student record has been located the user can scroll through the academic program by using the predefined PF-KEYS 7, 8, 10, and 11, where PF7 and PF8 scroll one line backward and forward respectively and PF10 and PF11 scroll one-half screen backward and forward respectively.

I. MODIFYING A RECORD

The simplest way to modify a data field within a record is to move the cursor to the desired location utilizing the cursor positioning keys found to the right of the main key-board. These four keys cause the cursor to move forward, forward rapidly, backward, backward rapidly, up, or down. Once the cursor is positioned, the user needs only to type over the existing data in order to change the entry. After the new data has been typed depress the ENTER-KEY. This causes the cursor to be positioned on the command line. Now, type the XEDIT subcommand SAVE, which writes the change onto disk upon depressing the ENTER-KEY. The cursor will again be positioned at the beginning of the command line and the user remains within the file he was editing.



ACADEMIC PROGRAM

Please Send All Changes to Code 0301. Date Received by Code 0301:

Last Name (Only) | Initials (Frst-Mid) | Complete Social Sec No

From: Curriculum Name and Code

Orders to: Specialty and Number

Service: ☐ USN Other: Specify

Rank

* For Change	Term No.	Academic Year and Quarter	Curriculum Mail Code	Curriculum Specialty Degree Code	CNT Code Year	CNT Code Qtr.	Input Number of Students	Name-Initls Soc.Sec.No	Course 1 Dept.	Course 1 Course Number	Course 1 Suffix	Course 2 Dept.	Course 2 Course Number	Course 2 Suffix	Course 3 Dept.	Course 3 Course Number	Course 3 Suffix	Course 4 Dept.	Course 4 Course Number	Course 4 Suffix	Course 5 Dept.	Course 5 Course Number	Course 5 Suffix	Course 6 Dept.	Course 6 Course Number	Course 6 Suffix	Yr. Degree Qtr. Date	Remarks
	1																											
	2																											
	3																											
	4																											
	5																											
	6																											
	7																											
	8																											
	9																											

Note: 0 = Zero Ø = Alpha O 1 = One I = Alpha I 2 = Two Z = Alpha Z

(Rev. 1-76)

APPENDIX B

The following are several representative curriculum matrices which are currently used as definitive aids to individual student program development.

NAVAL POSTGRADUATE SCHOOL
COMPUTER SCIENCE PROGRAM (#368)

Quarter					
1.	Finite Mathematics MA 2025 (4-1)	Digital Machines EE 2810 (3-3)	Probability & Statistics OS 3104 (4-0)	Introduction to Computer Science CS 2810 (3-0) CS 2813 (0-2)	
2.	Discrete Mathematics MA 3026 (5-0)	Intro to Computer Organizaton CS 3200 (3-2)	Data Structures CS 3300 (3-1)	Fundamental Concepts in Programming Languages CS 3111 (4-0)	FORTRAN Lab CS 2811 (0-2)
3.	Automata, Formal Languages & Computability CS 3601 (4-0)	Comparative Computer Architectures CS 3400 (4-0)	Operating Systems CS 3112 (3-2)	Introduction to Compilers CS 3113 (3-2)	
4.	Numerical Analysis MA 3232 (3-2)	Option Elective(s)	At Least Two of These Four Courses Computer Systems CS 4112 (4-1) Advanced Language Topics CS 4113 (3-2)		COBOL CS 2103 (1-2)
5	Artificial Intelligence CS 3310 (4-0)	Operations Research OS 3001 (4-0)	Software Engineering CS 4500 (3-2) Data Base Systems CS 4300 (4-0)		Thesis CS 0810
6.	Systems Analysis & Design IS 4200 (4-0)	Option Elective	Elective	Thesis CS 0810	
7.	Option Elective	Elective	Thesis CS 0810	Thesis CS 0810	

COMPUTER SYSTEMS (367)

First Quarter (Fall. Spring)	CS 2810 (3-2) Fund. Comp. Sci CS 2812 (0-2) Comp. Sci. Lab	IS 2000 (3-0) Computer Management	MN 3105 (4-0) Organization & Management	MN 3155 (4-0) Financial & Managerial Accounting	IS 0001 Seminar
Second Quarter (Winter, Summer)	CS 3010 (4-0) Computing Devices & Systems	PS 3011 (5-0) Probability & Statistics for Management	MN 3307 (4-0) ADP Acquisition	MN 3143 (4-0) Managerial Economics	IS 0001 Seminar
Third Quarter (Spring. Fall)	CS 3020 (3-2) Software Design	OS 3210 (5-0) Operations Research	Emphasis Course	IS 3170 (4-0) Econ. Eval of Computer & Telcomm. Sys.	IS 0001 Seminar
Fourth Quarter (Summer, Winter)	CS 3030 (4-0) Operating Sys. Structures	IS 4200 (4-0) Systems Anal. & Design	Emphasis Course	Project/Thesis & Exper. Tour	IS 0001 Seminar
Fifth Quarter (Fall, Spring)	Emphasis Course	IS 4182 (4-0) Data Processing Management	MN 3310 (4-0) Man. Pers. Plan or MN 4154 (4-0) Financial Mgt. in the Navy	Project/Thesis	IS 0001 Seminar

ANTISUBMARINE WARFARE

(#525)

April 1981

1.	MA 1112 (2-1) Calculus Review		OC 2120 4-0 Oceanography Survey	ST 1810 4-1 Calculators & Programming	
	MA 2129 2-0 ODE, Laplace	MA 2181 2-1 Vectors			
2.	MA 3139 4-0 PDE, Fourier Series Trans.	PS 2411 4-1 Applied Probability	EE 2721 4-1 Electronic Systems	PH 2119 3-1 Oscillations, Waves	
3.	OS 3390 4-2 Programming & Simulation	OS 3661 4-0 Decision Anal. & Data Anal.	PH 2471 3-0 Sonar Equation	EE 3714 4-1 Signals & Noise	
4.	EE 4716 4-1 Signal Processing	OS 3651 4-0 Search, Detection and Localization	PH 3472 4-2 Underwater Acoustics	OC 4265 4-0 Env. Factors in Underwater Acoustics	
5.	PH 3366 (4-0) Electromagnetic Wave Propagation		OC 3625 2-1 Env.Predictions	MR 2413 3-0 Meteor for ASW	
	/ / / / / / / / / / / E X P E R I E N C E T O U R S / / / / / / /				
6.	EE 4451 4-2 Sonar Systems Engineering	PH 3281 4-0 Non-Acoustic Sensor Systems	PH 4473 4-0 Adv. Topics in Underwater Acoustics	OS 3665 3-1 Human Factors in ASW	
7.	OS 4653 4-0 Operational Test & Eavl	OS 3652 4-1 Combat Models and Weapons Effectiveness	ELECTIVE	THESIS	
8.	NS 3070 4-0 Naval Warfare and Threat Environment	ELECTIVE	THESIS	THESIS	

Approved by ASWAC
May 21, 1980

590 Curriculum
Typical Program
Guidance, Navigation & Control

May 1981

QTR				
1	EE2101 (3-2) Circuit Analysis I	PH2241 (4-0) Modern Physics for Engineers	CS2810/11 (3-2) Intro to Computer Science (FORTRAN)	MA2047 (4-0) Linear Algebra and Vector Analysis
2	EE2102 (3-2) Circuit Analysis II	EE2211 (4-2) Electronics Engineering I	EE2810 (3-3) Digital Machines	MA2125 (3-0) Differential Equations
3	EE2400 (4-2) Linear Systems Analysis	EE2212 (4-3) Electronics Engineering II	EE2812 (3-2) Logic Design	MA2151 (4-0) Intro to Complex Vari- ables and Numer- ical Analysis
4	EE2500 (4-2) Communications Theory	EE2621 (4-0) Intro to Fields and Waves	CS2813 (0-2) PASCAL EE2401 (2-0) Discrete Systems	OS2102 (4-1) Intro to Applied Probability
5	EE2411 (4-2) Control Systems	EE2622 (3-1) Electromagnetic Engineering	EE2215 (2-4) Applied Electronics	EE3500 (4-1) Analysis of Random Signals
6	EE4413 (3-2) Linear Optimal Estimation and Control	EE3600 (3-2) Electromagnetic Radiation, Scat- tering, & Propa- gation	EE3800 (3-2) Microprocessor- Based Systems Design	EE3400 (4-0) Intro to Digital Signal Processing
7	EE3473/72 (3-2) Navigation, Missile and Avionics Systems	Elective	Core Elective	EE0810 Thesis Research
8	EE4411 (4-0) Digital Control	Elective	EE0810 Thesis Research	EE0810 Thesis Research
9	EE4460 (4-0) Principles of Systems Engineering	Elective	Elective	EE0810 Thesis Research

Attachment 2

FILE: CCRE1367 DATA A NAVAL POSTGRADUATE SCHOOL

ADVISOR: EMPHASIS AREA:
SEQ. OPTION NO.:

QTR :		
	CS 2810 (3-0)	COMPUTER SCIENCE FUND.
	CS 2812 (0-2)	PRGM. LAB. FOR INFORMATION SCIENCES
	IS 2000 (3-0)	INTRO. TO COMPUTER MANAGEMENT
	MN 3105 (4-0)	ORGANIZATIONAL SYSTEM II
	MN 3155 (4-0)	FINANCIAL & MANAGERIAL ACCOUNTING
QTR :		
	CS 3010 (4-0)	COMPUTING DEVICES & SYSTEMS
	PS 3011 (5-0)	PROBABILITY & STATISTICS FOR MGMT.
	MN 3307 (4-0)	A.D.P. ACQUISITION
	MN 3143 (4-0)	MANAGERIAL ECONOMICS
QTR :		
	CS 3020 (3-2)	SOFTWARE DESIGN
	IS 3170 (4-0)	ECON. EVAL. OF COMPUTER AND TELECOMM. SYS.
	OS 3210 (5-0)	OPERATIONS RESEARCH FOR CSM
	EMPHASIS COURSE	
QTR :		
	CS 3030 (4-0)	OPERATING SYS. STRUCTURES
	IS 4200 (4-0)	SYS. ANAL. & DESIGN
	IS 0810 (0-0)	THESIS RESEARCH FOR CSM
	EMPHASIS COURSE	
QTR :		
	IS 4182 (4-0)	DATA PROC. MGMT.
	IS 0810 (0-0)	THESIS RESEARCH FOR CSM
	MN 3310 (4-0)	MANPOWER PERS. PLAN. & ANAL.
	OR	
	MN 4154 (4-0)	FINANCIAL MGMT. IN THE NAVY
	EMPHASIS COURSE	

BIBLIOGRAPHY

Chapman, Jack Albert, Logical Data Base Design for Relational Data Base Systems, MS Thesis, Naval Postgraduate School, 1978.

Daniels, Thomas Howard, The Feasibility of Employing A Data Base Management System in Student Program Development, MS Thesis, Naval Postgraduate School, 1980.

International Business Machines Corporation, IBM Virtual Machine Facility/370: CMS User's Guide, 3rd ed., 1979.

International Business Machines Corporation, IBM Virtual Machine/ System Product: System Product Editor User's Guide, 1st ed., 1980.

International Business Machines Corporation, IBM Virtual Machine/ System Product: System Product Editor Command and Macro Reference, 1st ed., 1980.

International Business Machines Corporation, IBM Virtual Machine/ System Product: Terminal User's Guide, 1st ed., 1980.

Kroenke, David, Database Processing, Science Research Associates, 1977.

Vincent, Charles David, The Design and Implementation of the Registrar's Information System at the Naval Postgraduate School, MS Thesis, Naval Postgraduate School, 1971.

INITIAL DISTRIBUTION LIST

1.	Dean of Academic Planning and Dean of Information and Policy Sciences, Code 013 David Alan Schrady Naval Postgraduate School Monterey, California 93940	1
2.	Dean of Academic Administration, Code 014 Abraham Sheingold Naval Postgraduate School Monterey, California 93940	1
3.	Department Chairman, Code 54 Department of Administrative Sciences Naval Postgraduate School Monterey, California 93940	1
4.	Curricular Officer, Code 37 Computer Technology Programs Naval Postgraduate School Monterey, California 93940	1
5.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
6.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
7.	Professor Norman F. Schneidewind, Code 54Ss Department of Computer Science Naval Postgraduate School Monterey, California 93940	2
8.	Associate Professor Norman Robert Lyons, Code 54Lb Naval Postgraduate School Monterey, California 93940	1
9.	Special Assistant For Programs, Code 0301 Naval Postgraduate School Monterey, California 93940	1
10.	Kenneth Mathew Golaszewski, Lieutenant, USN 84 Twin Hills Drive Groton, Connecticut 06340	1

Thesis
G5345 Golaszewski
c.1 Design of a data base management system for student program development and maintenance.
193792