SAULT COLLEGE OF APPLIED ARTS & TECHNOLOGY

SAULT STE. MARIE, ONTARIO

COURSE OUTLINE

Course Title: COMPUTER SYSTEMS III

Code No.: CET303-5 Semester: 5

Program: COMPUTER ENGINEERING TECHNOLOGY

Author: TYCHO BLACK

Date: JAN., 1993 Previous Outline Dated: JAN.,1992

APPROVED: _____ 93-01-05

Dean Date

COMPUTER SYSTEMS III
COURSE NAME

CET303-5
CODE NO.

TOTAL CREDIT HOURS: 80

PREREQUISITES: CET220

I. PHILOSOPHY/GOALS:

Computer Systems III follows two years of study in a variety of programming languages and operating systems and is intended to develop the student's skills as proficient users of VAX computers at the system's programming level. Two operating systems will be used; VMS and ULTRIX. The student will learn terminal and disk I/O techniques, screen management, process management and interprocess communications. The view of the VAX computer as the host of a single, dedicated system in an environment such as process control engineering will be developed. The concepts will be demonstrated by significant programming exercises in Fortran and C.

II. STUDENT PERFORMANCE OBJECTIVES:

Upon successful completion of this course the student will:

1. Understand the System's Level Programming environment of VMS and ULTRIX systems and the standard ways of calling system services and functions.

2. Be able to utilize VMS Screen Management Library routines to manage screens and implement windows.

3. Be able to write programs to manage, synchronize, and communicate between processes in multi-process VMS systems.

4. Be able to use available UNIX system calls to manage files and processes, and program terminals, in particular using the Curses library functions.

5. Be able to use UNIX system calls to do process management, interprocess communication and file I/O.

III. TOPICS TO BE COVERED:

1. General capabilities of VMS System Services, Run Time Library and RMS.

2. Utilization of the Screen Management Library facilities to create windows and manage I/O.

3. Managing process states and synchronizing processes using event flags and other techniques.

4. Interprocess communication techniques such as mailboxes, global memory and shared files.

5. VAX Input/Output System.

6. General C programming requirements and services in a UNIX environment.

7. UNIX terminal I/O techniques.

8. UNIX process management system calls and their utilization in C programming.

9. UNIX Interprocess communication techniques including pipes, named pipes, semaphores and signals.

COMPUTER SYSTEMS III
COURSE NAME

CET303-5
CODE NO.

IV. LEARNING ACTIVITIES

REQUIRED RESOURCES

Note: These Learning Activities will not necessarily be covered in the order shown below. Also, some minor objectives may have to be omitted if time does not allow.

BLOCK I - USING VMS SERVICES AND THE SCREEN MANAGEMENT LIBRARY

TEXT:

"PROGRAMMING VMS IN FORTRAN" (notes)

At the end of this block the student will understand the nature of the various services available on a VMS system, and the methods used to call them. In particular,

1. Discuss the general capabilities of the VAX Run-time Library, System Services and RMS, and the techniques for calling them.

2. Discuss the Screen Management features of the Run-Time Library, and the general windowing environment.

Write programs that utilize the screen management facilities.

BLOCK 2 - MANAGING AND SYNCHRONIZING PROCESSES IN VMS

In this block the student will study the VMS process, and how it is scheduled. Techniques of creating processes and controlling and synchronizing their execution will also be studied. Programming will be done in Fortran. In particular students will be able to:

1. Describe the types of VMS processes and be able to create processes from DCL or a program.

2. Discuss the various states a process may be in and use DCL and write Fortran programs that determine and modify process states, and that hibernate, wake, suspend and resume processes.

3. Use the VMS Monitor utility to monitor the states of processes.

4. Discuss the use of event flags as a process synchronization tool and utilize them in programs.

COMPUTER SYSTEMS III
COURSE NAME

CET303-5
CODE NO.

5. Describe the use of AST's and write programs that use them.

6. Describe the use of the VMS Lock Manager and write programs that use it.

7. Describe the use of exception and condition handlers and write programs utilizing them.

BLOCK 3 - INTERPROCESS COMMUNICATION and THE INPUT/OUTPUT SYSTEM IN VMS

In this block the student will study the methods used to pass information between processes. The student will use mailboxes, global memory areas and shared files as methods of communication and study the VAX input-output system. The student will be able to:

1. Discuss the use of mailboxes and write programs utilizing them.

Discuss the concept of global memory, the types of global sections and the services that can be used to call them and then write programs to utilize them.

3. Discuss file-sharing techniques and use them in programs.

4. Discuss the structure of the VAX I-O system and the process of performing input and output, in particular using QIO directives.

BLOCK 4 - PROGRAMMING C IN AN ULTRIX ENVIRONMENT

In this block the student will study the basic concepts needed to program UNIX from C and will review basic file I/O. In particular, students will be able to:

"ADVANCED UNIX PROGRAMMING" by Marc. Rochkind

1. Discuss in general terms the following terms: UNIX files; programs and processes; signals; process-ids and groups; permissions; process attributes and interprocess communications.

CHAP. 1

COMPUTER SYSTEMS III
COURSE NAME

CET303-5
CODE NO.

2. Describe the techniques of using system calls in UNIX and write programs demonstrating that capability.

3. Discuss and use the following system calls to do basic file I/O: **creat, unlink, open, write, read, close, lseek.** CHAP. 2

4. Discuss and use the following system calls to do advanced file I/O: **link, access, mknod, chmod, chown, utime, stat, fstat, fcntl.** CHAP. 3

5. Discuss the programming of time in UNIX.

BLOCK 5 - UNIX TERMINAL I/O

In this block the student will study the concepts and techniques needed to program the terminal in UNIX. The student will be able to:

Discuss the general concepts of doing terminal I/O in Unix and the use of the **ioctl** system call. CHAP. 4

2. Describe the facilities of the Curses Library for screen management and terminal I/O and write programs demonstrating its use.

BLOCK 6 - UNIX PROCESS MANAGEMENT

In this block the student will study the techniques for creating and managing processes in UNIX. At the end of this block the student will:

1. Discuss the UNIX environment from a process point of view and write programs that manage processes with the use of the following system calls: **exec, fork, exit, and wait.** CHAP. 5

2. Discuss the use of various system calls for finding and setting process identification values: **getgid, getegid, getpid, getpgrp, getppid, setuid, setgid, and setpgrp.**

3. Use the **nice** system call to modify a process's priority.

BLOCK 7 - INTERPROCESS COMMUNICATION AND SIGNALS IN UNIX

This block involves the study of UNIX interprocess communication techniques including pipes, named pipes, semaphores, messages, signals, shared memory, and record locking. The student will be able to:

1. Discuss the concepts of pipes and FIFO's (or named pipes) in UNIX and write programs utilizing them. — CHAP. 6 and CHAP. 7

2. Discuss and use semaphores for interprocess communication.

3. Describe shared memory and record locking concepts in UNIX.

Discuss the various types of signals and write programs demonstrating their use. — CHAP. 8

5. Discuss the use of **kill, pause and alarm** system calls and write programs utilizing them.

V. METHOD OF EVALUATION:

3 THEORY TESTS	66%
ASSIGNMENTS	34%

(The percentages shown above may vary slightly where circumstances warrant.)

GRADING SCHEME

A+	90 -	100%
A	80 -	89%
B	70 -	79%
C	55 -	69%
I	Incomplete	
R	Repeat	

COMPUTER SYSTEMS III — COURSE NAME
CET303-5 — CODE NO.

UPGRADING OF INCOMPLETES

When a student's course work is incomplete or final grade is below 55%, there is the possibility of upgrading to a pass when a student meets the following criteria:

1. The students attendance has been satisfactory.
2. An overall average of at least 40% has been achieved.
3. The student has not had a failing grade in all of the theory tests taken.
4. The student has made reasonable efforts to participate in class and complete assignments.

ASSIGNMENTS:

All assigned work must be completed satisfactorily to pass this course. A penalty will be applied to assignments handed in late. The exact nature of this penalty will be clearly stated before assignments are due.

It is acceptable that students consult with each other in relation to their assigned problems. However, it is unacceptable to copy programs written by someone else and submit them as your own work. Where plagiarism or copying is found and it is impossible to determine whose original work it is, a mark of zero will be assigned to all assignments involved.

VI. REQUIRED STUDENT RESOURCES:

TEXT BOOKS:

1. "PROGRAMMING VMS IN FORTRAN" (Printed Notes)

2. "ADVANCED UNIX PROGRAMMING"
   by Marc. Rochkind (Prentice-Hall)

VII. ADDITIONAL RESOURCE MATERIALS AVAILABLE:

DEC VMS and ULTRIX Manuals (to be itemized during the course)

VIII. SPECIAL NOTES:

Students with special needs (eg. physical limitations, visual or hearing impairments, or learning disabilities) are encouraged to discuss any required accommodations confidentially with the instructor. Your instructor reserves the right to modify the course as deemed necessary to meet the needs of students or take advantage of new or different learning opportunities.