SAULT COLLEGE OF APPLIED ARTS & TECHNOLOGY

SAULT STE. MARIE, ONTARIO

COURSE OUTLINE

Course Title: STRUCTURED PROGRAM DEVELOPMENT

Course No.: CET129

Program: COMPUTER ENGINEERING TECHNOLOGY

Semester: SECOND(2)

Date: JANUARY 1994

Author: FRANK TURCO

Previous
Outline Dated: JANUARY 1993

APPROVED: _____   _____
          Dean   Date

STRUCTURED PROGRAM DEVELOPMENT   CET129

C O U R S E   O U T L I N E

PREREQUISITES: CET106

LENGTH OF COURSE: 4 HOURS PER WEEK

TOTAL CREDIT HOURS: 64

I. PHILOSOPHY/GOALS

This course is intended to provide the student with a firm
base of VAX/VMS concepts and structured programming concepts.
The student is also taught the fundamentals of VAX usage
through the study of DCL programming techniques and the use of
VAX utilities. The student applies structured programming
concepts to more complex problems and will improve his/her
skills in the Fortran programming language.

II.   PERFORMANCE OBJECTIVES

Upon successful completion of this course the student will:

1. Use the EVE editor.
2. Understand and apply a variety of commonly used DCL
    commands such as:
        Assign, GOTO, IF, Inquire, Purge, SET commands,
        SHOW commands,
3. Create DCL command procedures.
4. Understand and apply file and directory protection
    options.
5. Follow proper Structure and problem solving Techniques.
6. Create Fortran programs that demonstrate capabilities in:
        Using Arrays, Using Functions, Using Subroutines
        Using Format Statements, Reading and Writing Files.
7. Solve programming problems using the VAX symbolic
    debugging program.

STRUCTURED PROGRAM DEVELOPMENT   CET129

III.   TOPICS TO BE COVERED

This particular course is an extension of CET106 which introduces the student to VAX/VMS and programming. It follows where that particular course ends and covers in detail the following topics:

1.   VAX/VMS Commands.
2.   EVE editor.
3.   Command Procedures.
4.   Structured Programming and Problem solving techniques.
5.   Arrays.
6.   Functions and Subroutines.
7.   Libraries.
8.   File Input and Output.
9.   Debugging Concepts and using the Symbolic Debugger.

IV.   LEARNING ACTIVITIES / REQUIRED RESOURCES

The learning activities have been blocked in related topic areas. The sequence in which they will be delivered may not necessarily correspond to these blocks. However, we will attempt to cover all of activities by the end of the course.

BLOCK 1   USING THE VMS EVE EDITOR

LEARNING ACTIVITIES

RESOURCES

Throughout this course we will use the EVE Editor to make the program coding and debugging easier.

Instructor's Handouts, Guidance, and Material covered in the class and Labs.

At the end of this block the student will be able to use the EVE editor to do the following features:

1.   General editing.
2.   Cut and Paste.
3.   Replace and Substitute.
4.   Find.
5.   Buffers.
6.   Include Files.
7.   Create Windows and move from Window to Window.

VMS USER"S GUIDE Chapter 4

STRUCTURED PROGRAM DEVELOPMENT   CET129

BLOCK 2   DEBUGGING TOOLS AND TECHNIQUES

LEARNING ACTIVITIES   RESOURCES

Throughout this course and in future courses the student will find debugging techniques an essential tool to programming and problem solving.

At the end of this block the student will be able to:

1. Understand that it is imperative to know what the program is supposed to do before you can start to effectively solve the problem through debugging.
2. Define what the outcome is supposed to be and identify what the input is.
3. Establish test data and test outcomes.
4. Produce desk copies of the required documentation that will help in solving the problems.
5. Isolate what the program does right and what it does wrong.
6. Focus energies on solving one problem at a time.
7. Proficiently desk check the code and walk through the code by manually memory mapping the iterative outcomes.
8. Proficiently work with the VAX SYMBOLIC DEBUGGER.
9. Understand what the common VAX DEBUGGER commands are and what they do.

Instructor's Handouts, Guidance, and Material covered in the class and Labs.

ONLINE VAX HELP FACILITY RELATED TO DEBUGGING

STRUCTURED PROGRAM DEVELOPMENT   CET129

BLOCK 3   MODULAR FORTRAN PROGRAMMING

LEARNING ACTIVITIES   RESOURCES

Throughout this course the student will
continue to strengthen their programming
skills and problem solving.

At the end of this block the student will
be able to:

1. Understand what FUNCTIONS are and
    how to use them.
2. Understand what SUBROUTINES are and
    how to use them.
3. Understand what Libraries are and
    how to use them.
4. Appreciate how subroutines and
    Functions are essential to proper
    structured programming.
5. Write programs using functions and
    subroutines.

Instructor's
Handouts,
Guidance, and
Material
covered in the
class and
Labs.

VAX FORTRAN
Text Chapters
9 to 10

BLOCK 4   ARRAY, STRING and INPUT/OUTPUT PROCESSING

LEARNING ACTIVITIES   RESOURCES

Array and string processing become
important tools when dealing with more
advanced problems.

At the end of this block the student
will be able to:

1. Understand what ARRAYS are and when
    to use them.
2. Understand the difference between
    one dimensional and multi-
    dimensional arrays.
3. Write programs utilizing one and
    two dimensional arrays.
4. Understand what STRING processing
    is.
5. Write programs utilizing String
    Manipulation.
6. Write programs that will process
    sequential files.

Instructor's
Handouts,
Guidance, and
Material
covered in the
class and
Labs.

VAX FORTRAN
Text Chapters
8, 11, 12, and
14

STRUCTURED PROGRAM DEVELOPMENT   CET129

BLOCK 5   STRUCTURED APPROACH TO PROBLEM SOLVING

LEARNING ACTIVITIES   RESOURCES

Throughout this course we will attempt to
focus our energy on ensuring that we take
the time to solve the problems in our
mind and on paper prior to sitting down
and coding. Experience has shown that a
student who takes the time to really
understand the problem and follows
structured programming techniques saves
time in the long run and becomes a better
programmer.

At the end of this block the students
will be able to:

1. Properly define the problem in
    his/her own words.   Instructor's
2. Describe what has to be done to   Handouts,
    solve the problem in English.   Guidance, and
3. Understand what program design is   Material
    and why it is important.   covered in the
4. Describe and use the following   class   and
    structured programming tools and   Labs.
    techniques:
    a) Pseudo Code
    b) Modular Code
    c) Peer Walk Through   VMS   USER"S
5. Develop larger and more complex   GUIDE   Chapter
    programs.   6

STRUCTURED PROGRAM DEVELOPMENT

BLOCK 6   COMMAND PROCEDURES

LEARNING ACTIVITIES

RESOURCES

Throughout this course the student will continue to strengthen their skills in using the VAX.

At the end of this block the student will be able to:

1. Discuss the concept of a command procedure, and relate it to the use of a LOGIN.COM file.
2. Understand the use of the logical names SYS$INPUT, SYS$OUTPUT, SYS$ERROR, and SYS$COMMAND.
3. Describe the form and use of the DCL commands such as the following:
    a) INQUIRE   b) IF   c) EXIT
    d) GOTO   e) WRITE
4. Describe the ways of defining symbols in DCL, and the method of displaying symbols and logical names.
5. Define keys using the DEFINE/KEY command.
6. Assign logical names with the ASSIGN or DEFINE commands.
7. Understand and use the SET PROTECTION command to protect files and directories.
8. Write a login command procedure that creates a personal working environment, including a menu driven procedure that allows them to select the working environment.

Instructor's Handouts, Guidance, and Material covered in the class and Labs.

VAX VMS USERS GUIDE Chapters

6 to 7

Appendix A.

STRUCTURED PROGRAM DEVELOPMENT   CET129

VI.   SPECIAL NOTES

1.   Students with special needs are encouraged to discuss required accommodations confidentially with the instructor.

2.   Your instructor reserves the right to modify the course as he/she deems necessary to meet the needs of students.

VII.   ASSESSMENT

Theory Tests, Practical Tests and Quizzes   65 %
Assignments   35 %

The tentative breakdown is as follows:

3	FORMAL THEORY TESTS	at 15 % each.
2	PRACTICAL TESTS	at 5 % each.
2	QUIZZES	at 5 % each.
7	ASSIGNMENTS	at 5 % each.

Some minor modifications to the above percentages may be necessary. The instructor reserves the right to adjust the mark up or down 5 % based on attendance, participation and whether there is an improving trend.

*   All Assignments must be completed satisfactorily to complete this course. Late hand in penalties will be 5 % per day. Assignments will not be accepted past one week late unless there are extenuating and legitimate circumstances.

*   The instructor reserves the right to adjust the number of tests, practical tests and quizzes based on unforseen circumstances. The students will be given sufficient notice to any changes and the reasons thereof.

STRUCTURED PROGRAM DEVELOPMENT   **CET129**

VII.   ASSESSMENT (CONTINUED)

GRADING SCHEME

1.   TESTS
Written tests will be conducted as deemed necessary;
generally at the end of each block of work. They will be
announced about one week in advance. Practical on-line tests
will be conducted in which time to complete the assigned
problems will be a factor in the evaluation. Quizzes may be
conducted without advance warning.

2.   ASSIGNMENTS
Assignments not completed by the assigned due-date will be
penalized by 5% per day late.   All assignments must be
completed satisfactorily to complete the course.

3.   GRADING SCHEME

A+	90	-	100%	Outstanding achievement
A	80	-	89%	Excellent achievement
B	70	-	79%	Average Achievement
C	55	-	69%	Satisfactory Achievement
R	Repeat			
X	A temporary grade that is limited to			

instances   where   special
circumstances have prevented the
student from completing objectives
by the end of the semester. An X
grade must be authorized by the
Chairperson. It reverts to an R if
not upgraded in an agreed-upon time,
less than 120 days.

4.   UPGRADING OF INCOMPLETE
When a student's course work is incomplete or final grade is
below 55%, there is the possibility of upgrading to a pass
when the student's performance warrants it. Attendance and
assignment completion will have a bearing on whether upgrading
will be allowed. A failing grade on all tests will remove the
option of any upgrading and an R grade will result.   The
highest grade on re-written tests or assignments will be 56%.
Where a student's overall performance has been consistently
unsatisfactory, an R grade may be assigned without the option
of make-up work.

The method of upgrading is at the discretion of the teacher
and may consist of one or more of the following options:
assigned make-up work, re-doing assignments, re-writing of
tests, or writing a comprehensive supplemental examination.