SAULT COLLEGE OF APPLIED ARTS & TECHNOLOGY

SAULT STE. MARIE, ONTARIO

COURSE OUTLINE

Course Title: PROGRAMMING LANGUAGES

Code No.: CET 204

Program: COMPUTER ENGINEERING TECHNOLOGY

Semester: FOURTH

Date: WINTER 1994

Author: F. TURCO

Previous Outline Dated: SUMMER 1993

Approved: L P Crozutti
Dean

Date: 94-01-11

PROGRAMMING LANGUAGES **CET204**

COURSE OUTLINE

PREREQUISITES: **CET129**
or Permission granted by the Instructor
Pending completion of this course.

LENGTH of COURSE: 4 Hours per Week

TOTAL CREDIT HOURS: 75

I. PHILOSOPHY / GOALS:

The goals of this course are to continue improving the student's programming skills through the use of the COBOL programming language. We will do a comparison study of the similarities and differences in COBOL, FORTRAN and ASSEMBLER. The student will continue to follow and improve structured programming. The course will take a relatively non-mathematical approach but will include an in depth study of programming techniques such as modularization, file handling, error handling, string handling, table processing, and formatted output. The students will utilize a variety of VMS development tools such as EVE, the Debugger, and the LIBRARIAN.

II. STUDENT PERFORMANCE OBJECTIVES:

At the end of this course the student will:

1. Understand the various sections in a COBOL program such as:

   THE IDENTIFICATION DIVISION
   THE ENVIRONMENT DIVISION
   THE DATA DIVISION
   THE PROCEDURE DIVISION

2. Demonstrate the capability of Formatting Input and Output.

3. Be able to use a variety of command verbs to perform arithmetic such as:

   ADD
   SUBTRACT
   MULTIPLY
   DIVIDE
   COMPUTE

II. STUDENT PERFORMANCE OBJECTIVES (Continued):

4. Be able to do proper EDITING of reported output with the PICTURE Clause and the MOVE verb.

5. Be able to do Conditional Processing and Error Handling.

6. Understand what COBOL's REPORT WRITING features are used for.

7. Understand and Demonstrate proficiency in TABLE PROCESSING.

8. Be able to use the SORT Verb.

9. Be able to do STRING PROCESSING.

10. Understand the difference between the various FILE types such as:

    SEQUENTIAL
    RELATIVE
    INDEXED

III. TOPICS TO BE COVERED:

1. LANGUAGE FUNDAMENTALS
2. STRUCTURED PROGRAMMING TECHNIQUES
3. SUBROUTINES
4. FUNCTIONS
5. MODULARIZATION
6. LIBRARIES
7. DEBUGGING
8. ERROR HANDLING
9. FILE HANDLING
10. ARRAY HANDLING
11. STRING HANDLING

IV. LEARNING ACTIVITIES / REQUIRED RESOURCES

REQUIRED RESOURCES

TEXTBOOK:

1. "COMPREHENSIVE STRUCTURED COBOL" - GARY S. HOPKIN FOURTH EDITION

Additional reference material will either be given to the students or placed in the library for the student's use.

Instructor's Handouts, Guidance, and Material as it relates to the individual topics.

Individual and Group Assignments to be formally prepared as assigned.

BLOCK 1 LANGUAGE FUNDAMENTALS

LEARNING ACTIVITIES:

Languages in general have similarities in both structure and commands. There will be a comparison of how each language handles basic concepts such as defining data, program code, file handling, formatted output. Since the students have previous experience in FORTRAN, ASSEMBLER and C, the comparative study will evolve from that which they are accustomed to.

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

1. Illustrate the COBOL and FORTRAN approaches to:
   a) Program Identification
   b) Commenting
   c) Data Definition
   d) File Definition
   e) Assignment of Data
   f) Looping Control
   g) Decision Making

2. Write and develop programs in the COBOL environment.

RESOURCES

TEXTBOOK:

CHAPTERS:

1,2,3,4,5,19

INSTRUCTOR'S HANDOUTS, GUIDANCE, and MATERIAL

IV. LEARNING ACTIVITIES / REQUIRED RESOURCES (Continued)

BLOCK 2 STRUCTURED APPROACH TO PROBLEM SOLVING

LEARNING ACTIVITIES:	RESOURCES
Regardless of what language programs are written in, it is important that students prepare good habits when it comes to problem solving. A disciplined approach to problem solving can and will simplify the students effort. Several tools and techniques will be addressed.	INSTRUCTOR'S HANDOUTS, GUIDANCE, and MATERIAL

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

1. Describe and use the following structured programming tools and techniques:
   - a) Plan of ATTACK
   - b) Pseudocode
   - c) Inspections and Walkthroughs
2. Provide thorough and useful program documentation.
3. Produce Modularized and structured code using programming features such as:
   - a) Subroutines
   - b) Functions
   - c) Common Blocks of code
   - d) Passing of Parameters
   - e) Readable code by proper spacing and indentation
4. Use the Library Facilities as well as Debugging.

IV. LEARNING ACTIVITIES / REQUIRED RESOURCES (Continued)

BLOCK 3 FILE INPUT, OUTPUT AND ERROR HANDLING

LEARNING ACTIVITIES:

This block will introduce the student to the various types of files on the VAX, and the typical ways of using them. The students will also learn to write error handlers to respond to run time errors.

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

1. Format output using the variety of PICTURE clauses in COBOL.
2. Discuss the differences to the following types of file:
   a) Terminal Format
   b) Sequential
   c) Relative
   d) Indexed
3. Discuss the methods of performing typical file maintenance activities when using the various types of files such as:
   a) Add data records
   b) Modify records
   c) Delete records
4. Describe the methods of using error handlers to deal with run-time errors.
5. Write COBOL programs utilizing the various techniques described above.

RESOURCES

TEXTBOOK:

CHAPTERS:

7,8 in detail

9, 13, 14, 15, 16, 17 (will be discussed but it will not be necessary to read in any detail)

BLOCK 4 ARRAY AND STRING HANDLING

LEARNING ACTIVITIES:

The students have previously been exposed to one and two dimensional arrays as they are used in FORTRAN. They will now use them in the COBOL language. String manipulation is also a common programming required and will be addressed.

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

1. Discuss the concept of subscripted variables and the use of one and two dimensional arrays.

2. Discuss the common requirements of programs that process string data.

3. Write programs in both languages that utilize array processing and string manipulation.

RESOURCES

TEXTBOOK:

CHAPTERS:

10, 12, 18

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	70%
Assignments	30%

The tentative breakdown is as follows:

4	FORMAL THEORY TESTS	AT 15 % EACH
2	QUIZZES	AT 5 % EACH
6	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 reason thereof.

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. 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.