SAULT COLLEGE OF APPLIED ARTS & TECHNOLOGY
SAULT STE MARIE, ON

COURSE OUTLINE

Course Title: HIGH-LEVEL LANGUAGE PROGRAMMING

Code No.: CET331 Semester: 5

Program: ELECTRICAL/ELECTRONICS ENG. TECHNOLOGY

Author: Doug Faggetter

Date: Sept. 1999 Previous Outline Date: Sept. 1998

Approved: _____ _____
Dean Date

Total Credits: 5 Prerequisite(s): CET228
Length of Course: 16 Total Credit Hours: 80

I. COURSE DESCRIPTION:

This course will introduce students to high-level language programming by using "C++", to solve technical problems. Program development will use a PC based compiler.

II. LEARNING OUTCOMES AND ELEMENTS OF THE PERFORMANCE:

Upon successful completion of this course the student will demonstrate the ability to:

1) Write well structured C++ source code to solve a variety of technical problems.

Potential Elements of the Performance:
- Input and Output from a C++ program.
- Declare variables in a program.
- Apply the fundamentals of the C++ language with the use of *if statements, if-else statements, for loops, while loops, do-while loops, switches.*
- Write source code in a modular form with the use of function calls.
- Make use of pointers in C++ code.
- Make use arrays in C++ code.
- Make use of string manipulation in C++ code.
- Input and output to the floppy disk from the C++ program.

2) Apply all of the steps of program development to writing effective C++ programs.

Potential Elements of the Performance:
- Enter source code for program.
- Compile the program.
- Run the program.
- Debug the program.
- Link multiple object files.

3) Analyze the essential elements of Object-Oriented Programming in a C++ environment.

Potential Elements of the Performance:
- Define and recognize Classes and Objects
- Recognize and understand the use of Constructors and Destructors
- Recognize and understand the use of Private and Public members
- Recognize and understand the use of Inheritance, Virtual Functions and Pure Virtual Functions.

III. TOPICS:

1) C++ Fundamentals

2) Structured Programming

3) Relational operators (*if* statement, *if-else* statement, *switch* function)

4) Looping (*for* loop, *while* loop, *do-while* loop)

5) Pointers, Scope and Class

6) Strings

7) Numeric Arrays

8) Data Structures

9) Classes and Objects

10) Disk I/O

IV. REQUIRED RESOURCES/TEXTS/MATERIALS:

An Introduction to Programming Using C++
by Kenneth C. Mansfield Jr. and James L. Antonakos

V. EVALUATION PROCESS/GRADING SYSTEM

The grading weight will be:
Theory 75%
Lab 25%

The grading system will be as follows:

A+	90% - 100%	Outstanding Achievement
A	80% - 89%	Above Average Achievement
B	70% - 79%	Average Achievement
C	60% - 69%	Satisfactory Achievement
R	below 60%	Repeat

VI. SPECIAL NOTES:

- Special Needs
  If you are a student with special needs (eg. physical limitations, visual impairments, hearing impairments, learning disabilities), you are encouraged to discuss required accommodations with the instructor and/or contact the Special Needs Office, Room E1204, Ext. 493, 717, 491 so that support services can be arranged for you.

- Retention of Course Outlines
  It is the responsibility of the student to retain all course outlines for possible future use in acquiring advanced standing at other post-secondary institutions.

- Substitute Course Information is available at the Registrar’s Office.

VII. PRIOR LEARNING ASSESSMENT

Students who wish to apply for advanced credit in the course should consult the instructor.