SAULT COLLEGE OF APPLIED ARTS & TECHNOLOGY
SAULT STE. MARIE, ONTARIO

COURSE OUTLINE

COURSE TITLE: COMPUTER PROGRAMMING 2

CODE NO.: CSD101 SEMESTER: WINTER 97

PROGRAM: COMPUTER ENGINEERING TECHNOLOGY/COMPUTER PROGRAMMER

AUTHOR: DENNIS OCHOSKI/FRANK TURCO

DATE: JANUARY 1997 PREVIOUSLY DATED: JANUARY 1996

APPROVED: Joseph C Fruchter
DEAN

970106
DATE

COMPUTER PROGRAMMING 2 CSD101

COURSE NAME COURSE CODE

TOTAL CREDITS: 4

PREREQUISITE(S): CSD100

I. **COURSE DESCRIPTION**: This course is intended to extend the foundation of computer programming skills needed in the computer studies area. It is the second course in the C programming language, and further develops the student's problem-solving, computer programming, and software utilization skills.

II. TOPICS TO BE COVERED:

1. Advanced data-manipulation operators.
2. Library and user-defined functions.
3. One- and two-dimensional arrays.
4. Pointers and strings.
5. Data structures and files.

III. LEARNING OUTCOMES AND ELEMENTS OF THE PERFORMANCE:

A. Learning Outcomes:

	Approx. % of Course Grade
1. Discuss and apply the concepts of special C operators used to manipulate data.	10%
2. Discuss and use additional C library functions, and write programs incorporating user-written, separately-compiled functions.	25%
3. Develop algorithms and write C programs to solve problems involving one- and two-dimensional arrays.	20%
4. Develop algorithms and write C programs to solve problems involving the use of pointers and string manipulation.	25%
5. Develop algorithms and write C programs to solve problems involving the use of data structures and file manipulation.	20%
	100%

B. Learning Outcomes and Elements of the Performance:

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

1. Discuss and apply the concepts of special C operators used to manipulate data. (units 9 and 10)

Elements of the performance:

- define or describe the meaning of the following terms:

  TRUE
  FALSE
  bit manipulation
  bitwise AND
  bitwise OR
  bitwise XOR
  boolean operators
  bit shifting
  bitwise complement
  overloaded

- explain automatic promotion and apply the typecast operator to define data types
- use the *sizeof* operator to determine how much memory is needed to hold a value
- apply conditional operators to relational tests
- discuss the concept of truth tables
- apply bitwise and compound bitwise operators
- write, test, and debug programs using advanced operators

2. Discuss and use additional C library functions, and write programs incorporating user-written, separately-compiled functions. (units 15 and 16)

Elements of the performance:

- define or describe the meaning of the following terms:

  scope
  local vs global variables
  class
  auto vs static variables
  pass by value
  pass by reference
  random number generation
  arguments/parameters

- discuss and apply additional standard library functions supplied with Turbo C++ such as the math, string, and ctype libraries, and how to determine the libraries that are available and which library a particular function is located

- discuss and apply the concepts of passing arguments to functions both by value and by reference

3. Develop algorithms and write C programs to solve problems involving one- and two-dimensional arrays.
(unit 17)

Elements of the performance:

- define or describe the meaning of the following terms:

one-dimensional array
two-dimensional array
subscript
index value
null character

- discuss the purpose and concepts relating to one- and two-dimensional arrays
- declare and initialize both numeric and character arrays
- pass arrays between C functions
- write, test, and debug programs containing arrays

4. Develop algorithms and write C programs to solve problems involving the use of pointers and string manipulation.
(unit 18)

Elements of the performance:

- define or describe the meaning of the following terms:

offset
pointer constant
address operator

- discuss and apply the concept of pointers and pointer arithmetic
- apply the concept of pointers to arrays

Elements of the performance(cont'd):

- discuss and apply the concept of strings and pointers in C
- discuss and apply the use of the following string functions: strcpy, strcat, strchr, strcmp, and strlen
- write, test, and debug programs using pointers and strings

5. Develop algorithms and write C programs to solve problems involving the use of data structures and file manipulation.
   (units 19, 21 , and 22)

Elements of the performance:

- define or describe the meaning of the following terms:

  structure
  member
  record
  union
  open
  close
  append
  internal pointer

- discuss the concept of structures in C
- apply the use of arrays of structures
- discuss and apply methods of passing and returning structures to and from functions
- create a disk file
- write data to, and, read data from a disk file
- perform disk I/O with records
- discuss and apply the use of the following functions: stdin, stdout, and stderr
- differentiate between sequential and random access files
- write, test, and debug programs containing structures and files

IV. EVALUATION METHODS:

The mark for this course will be arrived at as follows:

Quizzes:	
outcome #1 & #2	25%
outcome #3	15%
outcome #4	20%
outcome #5	15%
Assignments:	
outcomes #1 & #2	10%
outcomes #3	5%
outcomes #4	5%
outcomes #5	5%
Total	100%

The grading scheme used will be as follows:

A+	90 - 100%	Outstanding achievement
A	80 - 89%	Excellent achievement
B	70 - 79%	Average achievement
C	55 - 69%	Satisfactory achievement
R	Repeat	
X	Incomplete	A temporary grade limited to special circumstances that have prevented the student from completing objectives by the end of the semester. An X grade reverts to an R grade if not upgraded within a specified time.

V. SPECIAL NOTES

1. In order to pass this course the student must obtain an overall **quiz** average of 55% or better, as well as, an overall **assignment** average of 55%.

2. Assignments must be submitted by the due date according to the specifications of the instructor. Late assignments will normally be given a mark of zero. Late assignments will only be marked at the discretion of the instructor in cases where there were extenuating circumstances.

3. The instructor reserves the right to modify the assessment process to meet any changing needs of the class. Consultation with the class will be done prior to any changes.

4. The method of upgrading an incomplete grade is at the discretion of the instructor, and may consist of such things as make-up work, rewriting tests, and comprehensive examinations.

5. Students with special needs (eg. physical limitations, visual impairments, hearing impairments, learning disabilities) are encouraged to discuss required accommodations confidentially with the instructor.

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

VI. PRIOR LEARNING ASSESSMENT:

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

VII. REQUIRED STUDENT RESOURCES

Text: Programming C in 12 Easy Lessons
by Greg Perry

Diskettes: minimum of 3, 3 1/2"