

Course Information

A: Division: **INSTRUCTIONAL** Date: **MARCH 1997**
 B: Faculty: **COMMERCE AND BUSINESS ADMINISTRATION** New Course:
 Program: **COMPUTER INFORMATION SYSTEMS** Revision of Course Information form: **MAY 1996**
 C: **CISY 470** D: **C++ AND OBJECT ORIENTED PROGRAMMING** E: **3**
~~SEE CISY 570~~

Subject & Course No. Descriptive Title Semester Credit

F: Calendar Description: This course provides the student with knowledge of the enhancements from C to C++. Object oriented terminology, concepts and language constructs will be covered. Students will learn how to develop object oriented programs in C++ and implement their own abstract data types as well as how to access supplied class libraries. *Students who have taken CISY 570 will not get credit for CISY 470.*

Summary of Revisions:
 1997-03 Sections: C,H,M,N,O,P,R

G: Type of instruction: Hrs per week

Lecture:	2	Hrs.
Laboratory:		Hrs.
Seminar:	2	Hrs.
Clinical Experience:		Hrs.
Field Experience:		Hrs.
Practicum:		Hrs.
Shop:		Hrs.
Studio:		Hrs.
Student Directed Learning:		Hrs.
Other (Specify)		
Total:	4	Hrs.
Semester Total (4 x 15 wks):	60	Hrs.

H: Course Prerequisites:
CISY 270 or CISY 370

I: Course Corequisites:
nil

J: Course for which this Course is a Prerequisite:
nil

K: Maximum Class Size:
35

L: College Credit Transfer
 College Credit Non-Transfer
 Non-Credit

M: Transfer Credit: Requested:
 Granted:

Specify Course Equivalents or Unassigned Credit as appropriate:

BCOU
 SFU
 UBC
 UNBC
 UVIC
 Other:

E. Meyer
 Course Designer(s): E. Meyer

J. Sator
 Dean: J. Sator

[Signature]
 Vice-President, Instruction: J. McKendry

[Signature]
 Registrar: P. Angus

N: TEXTBOOKS AND MATERIALS TO BE PURCHASED BY STUDENTS

Deitel, H.M. and P.J. Deitel. C++ How to Program, Latest Ed. Prentice Hall

O: COURSE OBJECTIVES

The student will be able to:

1. read, write and modify introductory to intermediate level object oriented C++ programs;
2. describe and utilize the code re-usability mechanisms of classes, inheritance and templates;
3. develop classes and class hierarchies associated with their own conceived abstract data types;
4. access existing class libraries;
5. describe and use the power of overloaded functions and operators as well as the concept/implementation of virtual functions and polymorphism.

P: COURSE CONTENT

1. Keyboard and Screen I/O - manipulators, flags, operators, member functions.
2. Classes/Objects - abstract data types, declarations, constructors, destructors, access protection.
3. File I/O Streams.
4. Function Name Overloading.
5. Operator Overloading.
6. Inline Functions.
7. References and Reference Arguments.
8. Default Arguments.
9. Comments, symbolic constants.
10. Pointers.

11. Base and Derived Classes.
12. Single and Multi-Level Inheritance.
13. Dynamic Memory Allocation.
14. Rules of Scope and Scope Resolution.
15. Friends - functions and classes.
16. Polymorphism, Virtual Functions.
17. String Streams.
18. Exception Handling.
19. Building a Class Library.

Q: METHOD OF INSTRUCTION

Lecture and seminar.

R: COURSE EVALUATION

Assignments (4-6)	20%-30%
Tests (2-6)	10%-20%
Midterm Examination	25%-30%
Final Examination	25%-30%
Participation	<u>0%- 5%</u>
	<u>100%</u>