UNIVERSITY COLLEGE of the FRASER VALLEY

PAC FORM #2

COURSE IMPLEMENTATION DATE:[Winter 1996
COURSE TO BE REVIEWED DATE:[Winter 2000]

(Four years after implementation date)

OFFICIAL COURSE OUTLINE INFORMATION

Students are advised to keep course outlines in personal files for future use.

Shaded headings are subject to change at the discretion of the department and material will vary - see course syllabus available from instructor

FACULTY/DEPARTMENT: **COMPUTER INFORMATION SYSTEMS**

COMP 155		**4**
COURSE NAME/NUMBER	FORMER COURSE NUMBER	UCFV CREDITS

OBJECT-ORIENTED PROGRAMMING

COURSE DESCRIPTIVE TITLE

CALENDAR DESCRIPTION:

This course continues the systematic study of structured programming in C++ started in COMP 150 with an introduction to object-oriented programming (OOP). Advanced features and applications are introduced including file processing and database engine access. An introduction to visual tools is also given. Credit with only be given for one of COMP 155 and COMP 160.

PREREQUISITES: COMP 150 with C+ or bette and Math 12 with C+ or better

COREQUISITES: None

SYNONYMOUS COURSE(S)

(a) Replaces: N/A *(Course #)*

(b) Cannot take N/A *(Course #)* for further credit

SERVICE COURSE TO:

(Department / Program)

(Department / Program)

TOTAL HOURS PER TERM: 56

STRUCTURE OF HOURS:

Lectures:	42	hrs
Seminar:		hrs
Laboratory:	14	hrs
Field Experience:		hrs
Student Directed Learning:		hrs
Other (Specify):		hrs

TRAINING DAY-BASED INSTRUCTION

LENGTH OF COURSE:

HOURS PER DAY:

MAXIMUM ENROLMENT: 35

EXPECTED FREQUENCY OF COURSE OFFERING:

WILL TRANSFER CREDIT BE REQUESTED? YES X NO

TRANSFER CREDIT EXISTS IN BCCAT TRANSFER GUIDE: YES NO X

AUTHORIZATION SIGNATURES:

Course designer(s): Chris Watkiss

Chairperson: *(Curriculum Committee)*

Department Head: Paul Franklin

Dean: K. Wayne Welsh

PAC Approval in Principle Date:

PAC Final Approval Date: October 27, 1999

COMP 155

COURSE NAME / NUMBER

LEARNING OBJECTIVES / GOALS / OUTCOMES/ LEARNING OUTCOMES:

To develop and maintain large programs
To develop transaction-based programs
To write advanced C + + programs in a Windows environment
To use several software tools (e.g., Visual C + +, Visual Basic, Access or Paradox database engine)

METHODS:

The course will be delivered in lecture-lab format, with numerous demonstrations and hands-on activities. The lab portion gives students and the instructor the ability to view and interact with current projects.

PRIOR LEARNING ASSESSMENT RECOGNITION (PLAR):

Credit can be awarded for this course through PLAR YES X NO

METHODS OF OBTAINING PLAR:

Course challenge with permission of the department.

TEXTBOOKS, REFERENCES, MATERIALS:

SUPPLIES / MATERIALS:

STUDENT EVALUATION:

Term tests (2)	20%
Assignments (5)	40%
Final exam	50%

COURSE CONTENT:

Review of C + +
Introduction to classes in C + +
Object-oriented design
Operator overloading
Inheritance: derived classes
Polymorphism, virtual functions
Pointers and dynamic memory allocation
Templates in C + +
Container classes: linked lists