Douglas College

EFFECTIVE: SEPTEMBER 2011
CURRICULUM GUIDELINES

A.	Division:	Academic	Effective Date:		September 2011	
B.	Department / Program Area:	Faculty of Commerce and Business Administration / Computing Science and Information Systems	Revision	**X**	New Course	
			If Revision, Section(s) Revised:		A, B, D, H, K, P	
			Date of Previous Revision:		May 2005	
			Date of Current Revision:		June 2011	

C:	**D:**	**E:**
CMPT 1110	Introduction to Computing Science Using C++	3
Subject & Course No.	Descriptive Title	Semester Credits

F: Calendar Description:

This course provides the student with knowledge of program design and programming methodologies. Emphasis is placed on the analysis of problems, the design of algorithms, and the abstraction of control and data in computer implementations of the design. Initially structured programming top-down design and procedural programming is used followed by object-oriented design (OOD) and object oriented programming (OOP). C++ is used as the implementation language.

Note: CISY 1275 and CMPT 1110 will be treated as equivalent.

G: Allocation of Contact Hours to Type of Instruction / Learning Settings

Primary Methods of Instructional Delivery and/or Learning Settings:

Lectures and Seminars

Number of Contact Hours: (per week / semester for each descriptor)

Lecture 2 hours / week
Seminar 2 hours / week

Number of Weeks per Semester:

15

H: Course Prerequisites:

MATH 1110 with a minimum grade of C; or BC Pre-Calculus 12 with a minimum grade of B; or CSIS 1110

I: Course Corequisites:

None

J: Course for which this Course is a Prerequisite:

CMPT 1150 and CMPT 1210

K: Maximum Class Size:

35

L: PLEASE INDICATE:

	Non-Credit
	College Credit Non-Transfer
X	College Credit Transfer:

SEE BC TRANSFER GUIDE FOR TRANSFER DETAILS (www.bctransferguide.ca)

M: Course Objectives / Learning Outcomes:

At the end of the course, the student will be able to:
1. Explain and give examples of the various structured and O-O features of the C++ language covered in class;
2. Analyze a well defined problem and design a solution, as appropriate, using a top-down structured methodology or OOD methodology;
3. Write and debug introductory to intermediate C++ applications from a solution design;
4. Effectively describe and utilize C++ built-in functions and supplied class libraries;
5. Read, understand, and modify introductory to intermediate C++ code written by another programmer;
6. Create their own abstract data types and be able to explain/incorporate the concepts of extensibility, maintainability, and reusability.

N: Course Content:

All topics in the core area are covered, though not necessarily in the order stated. Topics in the optional area are covered at the discretion of the instructor.

Core Topics

1. Procedural programming and structured (top-design) design
1.1 Primitive data types, operators, and expressions
1.2 Control structures
1.3 Conditional
1.4 Repetition
1.5 User defined functions and procedures
1.6 Parameter passing by value and by reference
1.7 Introduction to pointers
1.8 System stack, scope, and lifetime of variables
1.9 Recursion
1.10 Function overloading

2. Data Structures
2.1 Files and I/O streams
2.2 Arrays and strings
2.3 Pointers to strings and dynamic allocation
2.4 Structures

3. Object Oriented Programming and Design
3.1 Abstraction, encapsulation, visibility, information hiding, instantiation
3.2 Constructors and destructors
3.3 Abstract data types
3.4 Inheritance
3.5 Dynamic allocation
  a) Shallow vs. deep copy
  b) Copy constructors

4. Optional Topics
4.1 Templates
  a) Function
  b) Class
4.2 Operator overloading
4.3 Virtual functions and polymorphism

O: Methods of Instruction:

The topics will be covered through in-class lectures, seminar sessions, laboratory assignments, reading, and research.

P: Textbooks and Materials to be Purchased by Students:

Eckel, Bruce, "Thinking in C++: Introduction to Standard C++, Volume One" (Current Edition) (Volume 1) ISBN 978-0139798092 (This book is available as a free download from the author at http://mindview.net/Books/TICPP/ThinkingInCPP2e.html)

or Textbooks suggested by the instructor

Q: Means of Assessment:

Assignments (minimum 2)	20 – 35%
Quizzes	0 – 20%
Participation	0 – 5%
Midterm examination	20 – 30%
Final examination	25 – 40%
TOTAL	100%

R: Prior Learning Assessment and Recognition: specify whether course is open for PLAR

Yes.

Course Designer(s)

Education Council / Curriculum Committee Representative

Dean / Director

Registrar