



**EFFECTIVE: SEPTEMBER 2004**  
**CURRICULUM GUIDELINES**

A. Division: **Instructional** Effective Date: September 2004

B. Department / Program Area: **Computing Science** Revision:  New Course:

If Revision, Section(s) Revised: **D, F, G, P**

Date of Previous Revision: **January 2003**

Date of Current Revision: **January 26, 2004**

C: **CMPT 1101** D: **Introduction to Programming Using Visual Basic.NET** E: **3**

Subject & Course No.	Descriptive Title	Semester Credits
----------------------	-------------------	------------------

<b>F:</b> Calendar Description:	<p><b>This course introduces students to a programming environment and language. Time is initially spent learning about hardware and software components of a computer system, files and directories/folders management in Windows and the software cycle. Most of the course is spent on object-based programming and fundamental building blocks for programming in Visual Basic.NET. Topics cover forms, controls, properties management, events and event handlers, data types including arrays, control structures, functions and procedures.</b></p>	
---------------------------------	--	--

<b>G:</b> Allocation of Contact Hours to Type of Instruction / Learning Settings  Primary Methods of Instructional Delivery and/or Learning Settings:  <b>Lecture / Laboratory</b>  Number of Contact Hours: (per week / semester for each descriptor)  <b>Lectures            2 hours / week</b> <b>Laboratory        2 hours / week</b>  Number of Weeks per Semester: <b>15</b>	<b>H:</b> Course Prerequisites: <b>MATH 1101 with a minimum grade of C or Math 12 with a minimum grade of C</b>  <b>I:</b> Course Corequisites: <b>None</b>  <b>J:</b> Course for which this Course is a Prerequisite: <b>CMPT 1110</b>  <b>K:</b> Maximum Class Size:  <b>Lectures            34</b> <b>Laboratory        34</b>	
---	---	--

**L:** PLEASE INDICATE:

	Non-Credit
	College Credit Non-Transfer
<b>X</b>	College Credit Transfer:

SEE BC TRANSFER GUIDE FOR TRANSFER DETAILS ([www.bccat.bc.ca](http://www.bccat.bc.ca))

**M:** Course Objectives / Learning Outcomes:

**Students should be able to**

- Create, locate, move, copy and delete a file or folder in Windows
- Create, navigate, and delete folder tree structure in Windows
- Understand the process of program design and development
- Design a GUI by using objects (forms and controls) and managing properties of controls
- Design a program to solve a well-defined problem
- Implement a program from a design using structured programming

**To implement a program a student should be able to**

- Understand what a form is, how to create it, and be familiar with the properties of a form
- Understand what menus are and how to add them to a form
- Understand what objects are, and be able to add controls to a form
- Understand the properties of a control and explore the events that can occur with a control
- Differentiate among the various data types
- Distinguish between variables and constants, and understand how to include them in programs
- Select appropriate scope for a variable, differentiate between variables that are global to a project and those visible only to a form
- Perform number and string manipulations including the use of built-in procedures
- Learn how to accept input through boxes
- Format values for output using formatting functions and output boxes
- Understand the purpose of procedures and be able to write reusable code in Sub procedures and Function procedures
- Understand the role of parameters and be able to differentiate between value and reference parameters
- Evaluate conditions using the relational operators and combine conditions using logical operators
- Understand and code selection logic using IF-THEN and SELECT CASE statements
- Understand and code looping routines DO-WHILE, DO-UNTIL, and FOR-NEXT
- Understand what arrays are and use one-dimensional arrays to hold data
- Pass arrays between procedures

**N:** Course Content:

The course content below applies to both the lecture and the lab. As they have equal contact time and are in phase.

1. Introduction to computers
  - 1.1. Hardware and software components of a computer system
  - 1.2. Levels of languages
2. Windows
  - 2.1. Files and folders management
3. Programming in general
  - 3.1. Program development cycle
  - 3.2. Programming tools – flowcharts, pseudocode
4. Programming in Visual Basic
  - 4.1. Forms, controls, properties, events
  - 4.2. Numbers, variables, constants
  - 4.3. Strings
  - 4.4. Input and output using text boxes
  - 4.5. Built-in functions – numeric functions, strings functions
5. Procedures
  - 5.1. Sub procedures
  - 5.2. Function procedures
  - 5.3. Scope of variables
  - 5.4. Value and reference parameters

<p><b>6. Decisions</b></p> <p>6.1. Relational and logical operators</p> <p>6.2. IF-THEN blocks</p> <p>6.3. SELECT CASE blocks</p> <p><b>7. Repetitions</b></p> <p>7.1. DO-WHILE and DO-UNTIL loops</p> <p>7.2. FOR-NEXT loops</p> <p><b>8. One-dimensional arrays</b></p> <p>8.1. Creating and accessing arrays</p> <p>8.2. Passing arrays between procedures</p>										
<p><b>O:</b> Methods of Instruction:</p> <p>There are three components to the course: lectures, labs, and self directed learning (i.e. programming assignments)</p> <p>The lecture is used to introduce new material, usually via a sequence of theoretical concepts and examples. The textbook is to be used as an additional source of study material, problems, and examples.</p> <p>The two-hour weekly lab parallels the lecture by considering the application of new material.</p> <p>Assignments are marked according to correctness of the algorithms, efficiency, and programming style.</p>										
<p><b>P:</b> Textbooks and Materials to be Purchased by Students:</p> <ul style="list-style-type: none"> <li>▪ Schneider, D.I., <u>An Introduction to programming Using Visual Basic.NET</u>, Fifth Edition, Prentice Hall, ISBN 0-130-30657-6</li> <li>▪ Portfolio for Programming Assignments</li> <li>▪ Two 3 ½ “ high density diskettes</li> </ul>										
<p><b>Q:</b> Means of Assessment:</p> <p>Evaluation will be carried out in accordance with Douglas College policy. The instructor will present a written course outline with specific evaluation criteria at the beginning of semester. Evaluation will be based on some of the following:</p> <table data-bbox="284 1234 893 1386"> <tr> <td>labs (12 to 14)</td> <td>15% - 30%</td> </tr> <tr> <td>assignments (3 to 4)</td> <td>15% - 25%</td> </tr> <tr> <td>tests (1 to 2) @ 15% - 30% each</td> <td>15% - 60%</td> </tr> <tr> <td>final examination</td> <td>25% - 40%</td> </tr> <tr> <td>class participation<sub>1</sub></td> <td>0% - 5%</td> </tr> </table> <p>Note #1: participation includes (but is not limited to) short pop-quizzes and/or attendance</p>	labs (12 to 14)	15% - 30%	assignments (3 to 4)	15% - 25%	tests (1 to 2) @ 15% - 30% each	15% - 60%	final examination	25% - 40%	class participation <sub>1</sub>	0% - 5%
labs (12 to 14)	15% - 30%									
assignments (3 to 4)	15% - 25%									
tests (1 to 2) @ 15% - 30% each	15% - 60%									
final examination	25% - 40%									
class participation <sub>1</sub>	0% - 5%									
<p><b>R:</b> Prior Learning Assessment and Recognition: specify whether course is open for PLAR</p> <p>Not at this time.</p>										

Course Designer(s):

Education Council / Curriculum Committee Representative:

Dean / Director:

Registrar: