CONESTOGA

COURSE OUTLINE

Course Programming: Fundamentals (2013-2014)

Code / Version PROG1780 (100)

Total Hours 90

Credits 6

PreRequisite(s)

CoRequisite(s)

Course Description

The student will use an object-oriented language to teach programming concepts and to develop programs to solve business problems. Assignments in programming will be given to explain the concepts of variables, decisions, scope, functions, repetition, and arrays. Particular attention will be paid to problem solving techniques using logical solutions with either pseudo code and/or flowcharting to design the logic of the program. The student will also be required to test, and debug, their projects and alternative solutions.

PLAR Eligible: Yes

Course Outcomes

Successful completion of this course will enable the student to:

1. Explain the concepts behind GUI development.
2. Create computer programs from a written problem description.
3. Declare variables and use operators to manipulate variables.
4. Discuss flow of control and use sequence, selection and iteration to solve logical problems.
5. Create user interfaces using various GUI elements.
6. Use arrays.
7. Use program debugging techniques to locate and correct programming errors.

Unit Outcomes

Successful completion of the following units will enable the student to:

1.0 Introduction to Programming and Visual C#

1.1 Describe the process of visual program design and development.

1.2 Explain the concepts of classes, objects, properties, methods, and events.

1.3 List and describe the three steps for writing a C# program.

1.4 Identify the elements in the Visual Studio environment.

1.5 Define design time, run time, and debug time.

1.6 Identify syntax errors, run-time errors, and logic errors.

2.0 User Interface Design

2.1 Use text boxes, masked text boxes, group boxes, check boxes, radio buttons, and picture boxes effectively.

2.2 Select multiple controls and move them, align them, and set common properties.

2.3 Make programs easy for the user to understand and operate by defining access keys, setting an Accept and a Cancel button, controlling the tab sequence, resetting the focus during program execution, and causing ToolTips to appear.

2.4 Disable and enable controls at design time and run time.

2.5 Change text colour during program execution.

2.6 Concatenate (join) strings of text.

3.0 Variables, Constants and Calculations

3.1 Differentiate among the various data types.

3.2 Apply naming conventions incorporating standards and indicating scope and data type.

3.3 Declare variables and constants and select the appropriate scope.

3.4 Convert text input to numeric values.

3.5 Perform calculations using variables and constants.

3.6 Use try/catch blocks for error handling.

3.7 Display message boxes with error messages using event-driven and compound messages.

3.8 Accumulate sums and generate counts, applying correct formats.

4.0 Decisions and Conditions

4.1 Use if statements to control the flow of logic.

4.2 Use nested if statements.

4.3 Evaluate Boolean expressions using the relational operators or comparison operators.

4.4 Combine expressions using logical operators && (and), | | (or), and ! (not).

4.5 Perform validation on numeric fields using if statements.

4.6 Use a switch structure for multiple decisions.

4.7 Debug projects using breakpoints, stepping program execution, and displaying intermediate results.

5.0 Creating and Calling Methods

5.1 Passing an argument to a method.

5.2 Use an output parameter.

5.3 Writing a value-returning method.

6.0 Lists and Loops

6.1 Create and use list boxes and combo boxes.

6.2 Determine which item in a list is selected.

6.3 Use the Items.Count property to determine the number of items in a list.

6.4 Use do, while, and for loops to execute a series of statements.

7.0 Arrays

7.1 Establish an array and refer to individual elements in the array with subscripts.

7.2 Use the foreach loop to traverse the elements of an array.

7.3 Create a structure for multiple fields of related data.

7.4 Accumulate totals using arrays.

7.5 Write a table lookup for matching an array element.

7.6 Store and look up data in multidimensional arrays.

Required Student Resources

Tony Gaddis. Starting out with Visual C# 2012. Pearson.

Optional Student Resources

Evaluation

The minimum passing grade for this course is 55 (D).

In order to successfully complete this course, the student is required to meet the following evaluation criteria:

Tests/Group Project	60.00
Assignments	40.00
	100.00 %

A passing grade in both the test and assignment portion independently is required in order to attain standing in this course. If the student fails one or both portions, then the lowest failing mark is submitted.

Other

Conestoga College is committed to providing academic accommodations for students with documented disabilities. Please contact the Accessibility Services Office.

Prepared By Liz Stacey

School Information Technology

Date 2013-08-07