Programming II – Spring 2018

Course Description

This course will provide the students with an understanding of structured, procedural, and event-driven programming. The students will develop techniques for problem solving through the application of programming methods and will gain experience in the nuts-and-bolts of program design as they complete lab-work and assignments. Students will learn to use the C# language and programming environment.

Course Requirements

- All program source code will be documented
- All program source code will contain comments listing the author, assignment number/title, and date
- You are required to keep a notebook for this course. Your notebook shall contain the following:
    - All class handouts
    - All lab exercises and source code
    - All quizzes
    - All tests
    - All homework assignments
- Your notebook will have a section for each category of item as described above. All material placed in your notebook will be dated and in order
- At the end of the course, I will review your notebook with you and you will receive a notebook grade for that semester.
- All program source code will be kept in a notebook in order

Desired Learning Outcomes

1. Learn how to design, plan, code and document a C# program
2. Create and debug C# programs using the Visual Studio Integrated Development Environment (IDE)
3. Create and debug C# programs using the SharpDevelop Integrated Development Environment (IDE)
4. Understand C# syntax, controls, and functions;
5. Understand the techniques required to use variables and their scope
6. Use C# constructs such as loops, selections and decisions
7. Manipulate basic built-in functions
8. Use logical and relational operators
9. Implement decision and repetition structures
10. Define and use arrays
11. Gain an introduction to concepts of class, objects, events, and properties
12. Develop graphical interfaces
13. Create and read data from a sequential file using C#
14. Learn to develop dynamic web pages using ASP.NET
15. Learn to develop database applications using ADO.NET

Effective: July 4, 2017

Class Methodology

The class will be comprised of a combination of lecture, discussion, exercise, reading, and projects. Students are expected to come to class each day fully prepared to participate in the day's activities. In this course, you will be apply fundamentals that you learn by developing solutions to a variety of programming challenges.

Dual Enrollment Credit

Credit for CIS118 – Introduction to .NET Programming is offered for this course through Great Bay Community College.

This course will provide students with an understanding of structured, procedural, and event-driven programming. Students will develop techniques for problem solving through the application of programming methods and will gain experience in the nuts-and-bolts of program design as they complete lab work and assignments. Students will learn to use the C#.NET language and programming environment.

Topic List

1. C# Language fundamentals
    a. .NET Framework
    b. Managed vs. Unmanaged Code
    c. Blocks of code
    d. C# statements
    e. Identifiers
    f. C# class library
2. Data Types and Operators
    a. C# value types
    b. Variables
    c. Literals
    d. Arithmetic operators
    e. Relational operators
    f. Logical Operators
    g. Operator precedence
    h. Type Casting
3. Program Control
    a. Input and Output
    b. if statement
    c. switch statement
    d. for loop
    e. while loop
    f. do while loop
    g. break and continue
    h. nested loops

4. Introduction
    a. Computer Systems: Hardware and Software
    b. Programs and Programming Languages
    c. Controls and Programming
    d. The Programming Process
    e. Visual Studio
    f. SharpDevelop

5. Creating Applications with Visual Basic
    a. Problem Solving
    b. Responding to events
    c. Clickable Images
    d. Debugging your application

6. Variables and Calculations
    a. Getting Text Input
    b. Variables and Data Types
    c. Performing Calculations
    d. Mixing Data Types
    e. Formatting Numbers and Dates
    f. Exception Handling
    g. Group Boxes and Loading Events
    h. Logic Errors

7. Making Decisions and Working with Strings
    a. The Decision Structure
    b. The If – Then statement
    c. The If – Then – Else statement
    d. The If – Then – ElseIf statement
    e. Nested If statements
    f. Logical Operators
    g. Comparing, Testing, and Working with Strings
    h. The Message Box
    i. The Select Case statement
    j. Radio Buttons and Check Boxes
    k. Class level variables

8. Lists, Loops, Validation, and More
    a. Input Boxes
    b. List Boxes
    c. Do – While loop
    d. Do – Until loop
    e. For – Next loop
    f. Nested loops
    g. Input validation
    h. Tool Tips

9. Procedures and Functions a. Procedures b. Passing Arguments to Procedures c. Functions d. Debugging
10. Multiple Forms, Standard Modules, and Menus a. Multiple Forms b. Standard Modules c. Menus
11. Arrays and Timers a. Arrays b. Array Processing c. Using Procedures and Functions with Arrays d. Multidimensional arrays e. Random numbers f. The Timer Control
12. Files, Printing, and Structures a. Using Files b. Dialog Controls c. PrintDocument control d. Structures
13. Working with Databases a. Database Management Systems b. Database Concepts c. DataGridView control d. Data-Bound controls e. Structured Query Language (SQL)
14. Developing Web Applications a. Programming for the Web b. Creating ASP.NET applications c. Web Server Controls d. Designing Web Forms e. Using Databases
15. Object Oriented Programming a. Classes b. Objects c. Creating a class

Tentative Schedule

Week	Topics
1	Creating C# Console applications
2	Creating C# applications using decision making constructs
3	Creating C# applications using repetition constructs
4	Creating Windows Form Applications with Visual C#
5	Variables and Calculations
6	Making Decisions and Working with Strings
7	Making Decisions and Working with Strings
8	Loops
9	Loops
10	Procedures and Functions
11	Procedures and Functions
12	Multiple Forms, Standard Modules
13	Arrays and Timers
14	Files, Printing and Structures
15	Working with Databases
16	Developing Web Applications
17	Console Applications and the Graphics class
18	Object Oriented Programming
19	Object Oriented Programming
20	Introduction to Java

Textbook
Various E-Books

Course Grading

Participation	3 Daily Projects
Notebook	4 Daily Projects
Homework	1 Daily Project
Quizzes	1 Daily Project
Projects/Labs	1 Daily Project

Effective: July 4, 2017

Grading Standards

It is expected that you do your best on all project/lab activities. I generally assign one of four possible grades to your labs.

√+ all project requirements completed in an exemplary manner (100)
√ most project requirements completed in an acceptable manner (85)
√- minimal project requirements completed in a substandard manner (70)
0 project not completed or completed in an unacceptable manner (0)

Assignment Due Dates

All assignments are due on the day they are assigned unless otherwise stated. Late assignments will be penalized one grade for each two days they are late. **No assignments will be accepted after 5 school days late**.

Classroom Rules

No food at computers. If you need a snack, eat it at the desk.
Start each session at the desk (not at computers)
Drinks are allowed but they must be in a container that is capable of being capped
No Internet use except for specific class assignments
No installation of programs; this includes the installation of files from USB drives.
No games!!!
Keep a class notebook of notes, handouts, assignments, tests, quizzes, etc.
Bring a writing instrument each day to class.
If you must leave the classroom, you must sign out.
If you wish to go to the school store, you must have a store pass.
Only 3 people out of the classroom at any time.
If you make a mess, clean it up.
See me for any exceptions to the above rules

Plagiarism and Collaboration

When working on various projects in this class, you are encouraged to collaborate at some level. Plagiarism is offering someone else's work as your own, whether one sentence, whole paragraphs, or blocks of code. Whether from an internet source, book, periodical, the writing of other students, or source code shared over the network. It is also dishonest to submit your own paper (or program) as original work in more than one course. There is a fine line sometimes between plagiarism and collaboration. Plagiarism is unacceptable here or at any time in your future career. Plagiarism will not be tolerated. In this class, plagiarized work will receive a grade of 0. **It is unacceptable for one person to write a program and share it over the network with other students placing your name on code that you did not write.** If you have any question as to whether or not you are plagiarizing someone's work, ask!!! Better ask for clarification than receive a 0 on your test.

Things you need to do:

- Install Visual Studio 2010 and SharpDevelop on your home computer