

U701-221: Introduction to Programming

Course Format: Online

Course Author/s: Suresh Chalasani, *Ph.D.*

Course credits: 3

Prerequisites: College Algebra I (or) Equivalent

Course Description: This course provides a conceptual and practice-oriented approach to define a business problem, design and test solution logic, implement and code the logic through sound structured programming techniques. It uses the C# .NET programming language to develop programs that are robust and easy to maintain using the Windows GUI framework. This course covers programming constructs such as data types, control structures, exception handling, methods and even handlers, arrays and collections, and file I/O. This course has a significant component of object-oriented programming and covers classes, inheritance, interfaces, and polymorphism.

Required Course Materials

- IMurach's C#, 7th Edition, Murach Press, 2021. ISBN: 978-1-943872-53-4
- Microsoft Visual Studio .NET. Software will be provided to within the Virtual Lab. You can also download the software from <https://visualstudio.microsoft.com/vs/features/net-development/>

Hardware Requirements

- You will need a webcam, speakers, and a microphone. You will need a laptop or computer to run the Microsoft Visual Studio platform and C# .NET programs developed using this platform.

Provided Software

- This course utilizes the use of software programs within the IL program-provided virtual lab. Information regarding accessing the virtual lab can be found in the course information module.

Course Learning Objectives

- CO1: Utilize the .NET platform for program design to develop Windows GUI applications
- CO2: Develop programs using different data types including strings and dates
- CO3: Apply the basic programming constructs such as control structures.
- CO4: Be able to develop methods, event handlers and exception handlers
- CO5: Implement arrays and collections in programs
- CO6: Develop object-oriented programs using the principles of inheritance and polymorphism
- CO7: Develop programs using I/O structures

Course Overview

Module #	MODULE TOPIC(s)	EVALUATED ACTIVITIES
----------	-----------------	----------------------

1	Getting Started with Visual Studio .NET (Chapter 1) Design a Windows Forms Application (Chapter 2)	Programming Assignment Coding Quiz 1 Coding Quiz 2
2	Code and test a Windows Forms application (Chapter 3)	Programming Assignment Coding Quiz
3	Work with numeric and string data (Chapter 4)	Programming Assignment Coding Quiz
4	Program control structures (Chapter 5)	Programming Assignment Coding Quiz
5	Program methods and event handlers (Chapter 6)	Programming Assignment Coding Quiz
6	Develop code for exceptions and data validation data (Chapter 7)	Programming Assignment Coding Quiz
		Midterm Exam #1
7	Program arrays and collections (Chapter 8)	Programming Assignment Coding Quiz
8	Program arrays and collections (Chapter 8) --- Continued	Programming Assignment Coding Quiz
9	Working with dates and string (Chapter 9)	Programming Assignment Coding Quiz
10	OO Programming: Create and use classes (Chapter 12)	Programming Assignment Coding Quiz
11	OO Programming: Indexers, delegates, events, and operators (Chapter 13)	Programming Assignment Coding Quiz
		Midterm Exam #2
12	OO Programming: Inheritance (Chapter 14)	Programming Assignment
13	OO Programming: Interfaces and Generics (Chapter 15)	Programming Assignment Coding Quiz
14	Working with file I/O (Chapter 17)	Programming Assignment Coding Quiz
		Final Exam

Evaluation Methods

Your final grade will be based on your performance on the following:

- 1) Quizzes – 25% - 275 points
- 2) Midterm Exams (Two midterm exams) – 24% - 200 points
- 3) Final Exam (One final exam) – 16% - 100 points
- 4) Programming Assignments (14 assignments) – 35% - 1400 points

Quizzes

Quizzes are used as basic knowledge checks. These will you give the ability to test your knowledge of module

concepts before you attempt the programming assignments. You will have an unlimited amount of attempts on each quiz, but you must score at least 80% before you can move on to the programming assignments.

Programming Assignments

Programming assignments are to be turned in by uploading your zipped project along with all source code to Canvas. Follow the specifications included in the programming assignment description. Each programming assignment covers only one module and the number of points can range from 30 to 40 points, depending on the complexity of the module and the assignment.

Exams

For this course, there are two midterm exams (each with 150 points or 15% of the grade). Each midterm exam covers concepts in a few modules. The final exam is comprehensive and will cover concepts from all modules. Final exam is worth 20% of the grade (worth 200 points). Because this is a programming class and student ability varies, the exams will be open book and open notes. Students will be given a window of two days to complete the exam, once they start the exam. The exams will not be proctored.

Exam Method: Online Without Proctoring

Because this is a programming class and student ability varies, the exams will be open book and open notes. Students will be given a window of two days to complete the exam, once they start the exam. The exams will not be proctored. This course requires all students to complete both midterm exams and the final exam. Students need to submit all their exam work as .zipped files (see the instructions for the exam). Students receive two attempts on each exam. If you elect to take a second attempt, the average score of both exams will be recorded.

Grading Scale

The following grading scale is used to evaluate all course requirements and determine your final grade:

A = 93–100	B = 83–87.9	C = 70–77.9	D = 60–69.9
AB = 88–92.9	BC = 78–82.9		F = Below 60

Pass/Fail Option

Students who enroll in an Independent Learning (IL) course under the pass/fail option will receive a final grade of S in place of a final grade equivalent to an A, AB, B, BC, or C and a final grade of U in place of a final grade equivalent to a D or F.