# CompE361 Windows Programming

**Course Prerequisite**

* CompE260 Data Structures and Object Oriented Programming
* CompE271 Computer Organization

Student must be comfortable with Data Structures (Arrays, Lists, Stacks, Queues) and basic OOP principals, and should be able to use Visual Studio to develop basic applications.

**Textbook**

* Sharp, John et al, Microsoft Visual C# Step by Step 9th Edition, Prentice Hall, 2018

**Grading**

* Programming assignments 40%; midterm 25%; final exam 25%; attendance 10%
* All work must be submitted on the date due, anything received afterwards will receive a grade of 0.
* All Assignments must be turned in to receive a passing grade in the class, if you do not submit an assignment on time you will still need to submit a working copy to pass the class.
* All written assignments must have your full name and id written legibly in the upper right corner of each page.
* All programming assignment must be submitted fully commented for example see page included with syllabus.
* All assignments uploaded to blackboard must be zipped, no other archival format will be accepted. All zip files should be named as follows, your last name and redid. So my files would be amack123451234.zip. No deviations from this format will be accepted.

**Plagiarism and Cheating**

* Plagiarism will not be tolerated, if you are not sure if you can use something please ask, you must document all use of outside code in your assignments, you must include the source, and how you accessed the material.
* Assignments are to be done individually unless otherwise specified
* Reusing work from other classes is also plagiarism, again if you are not sure ask.
* You will be asked to sign a student contract agreeing to not cheat
* There is a zero-tolerance policy for cheating, you will receive a 0 grade for the class and will be referred to the academic dishonesty board of the university, if you are a graduate student your thesis advisor will also be notified of your actions

**Extra Credit**

* I reserve the right to offer extra credit anytime for any reason, to receive extra credit when offered you must be present in class

**Attendance Policy**

* I firmly believe that class attendance is the key to success. I reserve the right to give pop quizzes and extra credit during class and if you are not present you will not get credit and cannot make up these points later.

**Instructor**

* Scott Amack, E409, (509)270-3570 samack@gmail.com
* Office hours: By Appointment Only

**Teaching Assistant**

* TBD

**Course Documents and Programs**

* Documents referred to in class and demonstration programs done in class will be available on Blackboard

**Course Software**

* Visual Studio Community Edition 2017

**Course Objective**

The objective of this course is to provide a deeper understanding of Object Oriented Programming (OOP) using C# and .NET Framework. Graphical User Interface (GUI) and event-driven programming. Visual Studio Integrated Development Environment (IDE) and Graphics Programming.

**Topic Outline**

***Basic Concepts and Perspectives***

* Computing Industry History
* Command-Line interfaces
* Graphical User interfaces
* Event-drive operation(s)
* Object-Oriented Programming and Class Libraries
* Static Linking
* Dynamic Linking
* Integrated Development Environments
* Visual Programming

***.NET Runtime Environment***

* Intermediate Language
* Just-in-time Compiling
* Common Language Runtime (CLR)
* Platform Independence
* Language Interoperability
* Assemblies
* Garbage Collection

***C# language and OOP***

* Primitive data types
* Classes and structs
* Reference types and value types
* properties
* common root class, Object;
* ToString and Parse methods
* Rectangular and jagged arrays
* Foreach statement
* Indexers
* Strings
* Inheritance
* Polymorphism
* Virtual functions
* Interfaces
* Abstract classes
* Enumerations
* Reflection
* Delegates
* Exceptions
* Collection classes
* Generics
* multithreading

***GUI Programming***

* console applications
* windows applications
* Forms
* Packaging source code
* DLL Components
* GUI Controls
* Mouse Events
* Keyboard Events
* Common Dialogs
* User-Defined Dialog Boxes
* User-Defined Controls
* Debugging Tools

***Graphics Programming***

* Basic structs and classes
* Drawing tools
* Graphics class and drawing methods
* Fonts
* Paint Event
* Coordinate systems and transformations
* Timers and animation
* Bitmaps and images

## Tentative Lecture Schedule for Fall 2017

|  |  |  |
| --- | --- | --- |
| 8/28 |  | Syllabus and Class Intro – Start Chapter 1 Intro to C# |
| 8/30 |  | Chapter 2. Variables, Operators and Expressions |
| 9/4 |  | Chapter 3. Writing methods and applying scope |
| 9/6 |  | Chapter 4. Using decision statements |
| 9/11 |  | Chapter 5. Using compound assignment and iteration statements |
| 9/13 |  | Chapter 6. Managing errors and exceptions |
| 9/18 |  | Chapter 7. Creating and managing classes and objects |
| 9/20 |  | Chapter 8. Understanding values and references |
| 9/25 |  | Chapter 9. Creating value types with enumerations and structures |
| 9/27 |  | Chapter 10. Using arrays |
| 10/2 |  | Chapter 11. Understanding parameter arrays |
| 10/4 |  | Chapter 12. Working with inheritance |
| 10/9 |  | Chapter 13. Creating interfaces and defining abstract classes |
| 10/11 |  | Chapter 14. Using garbage collection and resource management |
| 10/16 |  | **MIDTERM** |
| 10/18 |  | Chapter 15. Implementing properties to access fields |
| 10/23 |  | Chapter 16. Handling binary data and using indexers |
| 10/25 |  | Chapter 17. Introducing generics |
| 10/30 |  | Chapter 18. Using collections |
| 11/1 |  | Chapter 19. Enumerating collections |
| 11/6 |  | Chapter 20. Decoupling application login and handling events |
| 11/8 |  | Chapter 21. Querying in memory data by using query expressions |
| 11/13 |  | Chapter 22. Operator Overloading |
| 11/15 |  | Chapter 23. Improving throughput by using tasks |
| 11/20 |  | Chapter 24. Improving response time by performing asynchronous operations |
| 11/27 |  | Chapter 25. Implementing the user interface for a UWP app |
| 11/29 |  | Chapter 26. Displaying and searching for data in a UWP app |
| 12/4 |  | Chapter 27. Accessing a remote database in a UWP app |
| 12/6 |  | Bonus: TBD |
| 12/11 |  | Bonus: TBD |
| 12/12 |  | Bonus: TBD |
| 12/18 |  | **Comprehensive Final Exam 7:00 PM** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

All submitted programs must be fully commented, see below for a simple example. You will lose 25% of the points available for the assignment if you do not follow these guidelines. Note that all methods and classes are commented and you should include your name redid and date at the top of every class file.

