

CS 3320-001 Object Oriented Programming

Sprint 2016, Sul Ross State University

Instructor: Dr. Kennard Laviers

Office Location: ACR 107 **Office Phone:** 432-837- 8500

Email: kennard.laviers@sulross.edu

Office Hours:

MW 8:00am-10:00am

TTH 8:00am-9:30am

F 1:00pm - 2:00pm

Class: *TTH* 9:30 pm - 9:45 am BAB 302;

Textbook: None

Online Tools: Google Doc's Draw is a free app offered by Google.

Program Learning Objective

1. Understand the fundamental concepts of computer science including algorithms and data structures.
2. Understand modern computer systems, databases and networking.
3. Display an understanding and ability to implement current programming methodologies.
4. Become proficient with systems design based on object-oriented programming.
5. Work as a team in workgroup environments.

Course Objectives

Week 1

Subject: Course Overview/ Programming Refresher I Basics

Goal: Allow students to understare the full scope of this course and what the expectations are for successful completion of the class.

Learning Objective:

1. Students will be able to declare and use variables
2. Students will understand strings, integers, and floats
3. Students will know how to write information to the Console

Week 2

Subject: Programming Refresher I Objects/ Using Static Classes, Methods, Variables

Goal: To ensure all students are able to make a program ask a user for information and will have a basic knowledge of what flowcharts are.

Learning Objective:

1. Students will know how to implement code to ask a user for information
2. Students will be able to cast strings from a users input into integers and floats when appropriate
3. Students will be able to explain why it is helpful to use a flow chart to plan out complex algorithms

Week 3

Subject: Generics

Goal: Students will understand how to make methods that use other objects that are not defined until the method is called.

Learning Objective:

1. Students will be able to write methods using the generics operator
2. Students will be able to return objects that are the same type as the generic operator.

Week 4

Subject: Abstraction/ Interfaces -- Behavior Project

Goal: This module will focus on appropriate use of abstraction and when an interface is more appropriate to use.

Learning Objective: Students will understand and be able to effectively implement:

1. Abstract Classes
2. Be able to override methods in an abstract class
3. Effectively implement interfaces

Week 5

Subject: Partial Classes/Methods/Extension Methods

Goal: This module will introduce to partial classes and methods and how to implement extension methods in C#

Learning Objective:

1. Be able to effectively implement partial classes
2. Implement partial methods
3. Extend class functionality using extension methods and be able to explain why and when it is a good idea to do so.

Week 6

Subject: Reflection -- Console Project

Goal: This section will teach students the idea of reflection and how we can use reflection to determine what type of class a generic object is and also, what methods are contained in an object. We will also learn how to invoke those methods at runtime via a command console.

Learning Objective:

1. Student will learn how to determine what class an object is an instance of.
2. Invoke methods by name.
3. Cast generic objects to their specific classes.

Week 7

Subject: File System operations

Goal: We will explore ways of accessing files, folders, changing the current folder and a list of other basic file operation tasks.

Learning Objective:

1. Know how to show the current folder
2. Be able to implement code to list all files in the current directory
3. Be able to implement code to change directories
4. Be able to move files and folders from one location to another
5. Be able to delete and rename files and directories

Week 8

Subject: Property Types and indexers

Goal: This module will provide an understanding of how to use property types in C# to simplify and make code more streamlined. Additionally students will learn the possible pitfalls of using property types in situations where this could significantly slow down the application at run time. Students will also learn how to declare and use indexers to provide a greater degree of complexity to Classes.

Learning Objective:

1. Students will know how to setup a property type and use the get call inside the property type.
2. Students will know how use the set call inside the property type.
3. Students will know how to setup and use indexers in their Classes when needed.

Week 9

Subject: Midterm

Notes: For Weeks 1 thru 8

Week 10

Subject: Chaining objects <Pointers>

Goal: We will introduce students to the idea of references and how objects can be chained together by simple pointers.

Learning Objective:

1. Be able to create a simple double linked list using pointers
2. Be able to create and search a binary tree. <using recursion>

Week 11

Subject: Overloading the operators

Goal: Students will continue to add more complexity to their classes by learning how to overload the operators to allow much more sophisticated operations by allowing classes to be able to get added and other such operations together.

Learning Objective:

1. How to overload the operator to allow adding/subtracting/multiplying/etc. two classes together.
2. Will be able to go beyond just the arithmetic to perform interesting object interactions with operators.

Week 12

Subject: Comparators

Goal: Week we learn to use comparators as separate objects so we can use the built in sort for everything.

Learning Objective:

1. Know how to implement the comparer.
2. Know how to implement iComparable.

Week 13

Subject: Sorting

Goal: This section will introduce the containers built in sort mechanism and will show students how to sort based on any member variable within an object that makes up a List<object>;

Learning Objective:

1. Be able to populate a list of some kind of object List<object>
2. Be able to implement a comparator to us to sort a list
3. Be able to sort the list based on any field in the objects in the list

Week 14

Subject: Spinning new Threads

Goal: This code will take students to the next level with a start at working with parallel threads and get them used to writing in a distributed and asynchronous environment.

Learning Objective:

1. Understand when it is appropriate to use multiple threads.
2. Be able to implement methods that are designed to run in parallel and execute them on new threads.

Week 15

Subject: Final

Notes: For weeks 8 thru 14

Attendance

Any student who accumulates 10 **unexcused** absences (MWF Classes) or 7 **unexcused** absences (MW classes) will be automatically dropped from this course.

Need for Assistance

Qualified students with disabilities needing academic or other accommodations to ensure full participation in the programs, services and activities at Sul Ross State University should contact the Disabilities Services Coordinator, in Counseling and Prevention Services, Ferguson Hall 112, Box C-117, Alpine, Texas 79832. Please notify me before the third day of classes.

Course Policies

Quizzes and assignments must be submitted on time. I have set up rules in Blackboard so that assignments cannot be submitted after the due date.

Academic Dishonesty: Honesty in completing assignments is essential to the mission of the university and to the development of the personal integrity of the student. Cheating, plagiarism, or other kinds of academic dishonesty will not be tolerated and will result in appropriate sanctions that may include failing an assignment, failing the class, or being suspended or expelled. Suspected cases in this course may be reported to Student Life.

Posting of Grades

As soon as assignments, exams, and quizzes are graded, the grades will be posted in Blackboard.

Grading

Letter grades will be determined using a standard percentage point evaluation as outlined below. Please note that this is a tentative schedule and can change. Any changes that happen will be updated in Blackboard. Due Dates for assignments will also be posted in Blackboard.

Grades are derived as:

50% Homeworks/Projects

25% Attendance and participation

12.5% Midterm

12.5% Final

Your final grade will be determined by calculating points based on the following weights:

- A 90 - 100 %
- B 80 - 89 %
- C 70 - 79 %
- D 60 - 69 %
- F < 60%