

# CS 1320-001 Beginning 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](mailto:kennard.laviers@sulross.edu)

## Office Hours:

MW 8:00am-10:00am

TTH 8:00am-9:30am

F 1:00pm - 2:00pm

**Class:** MW 10:00 pm - 10:50 am BAB 317;

**Lab01:** W 2:00 pm - 3:45 pm. ACR 104

**Lab02:** W 12:30 pm - 1:45 pm. BAB 303

**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

This course introduces concepts and techniques of object-oriented programming including objects, classes, methods, inheritance, polymorphism and memory management of objects.

## Schedule:

### Week 1

**Subject:** Course Overview

**Goal:** Allow students to understand 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

**Notes:**

Week 2

**Subject:** Programming Refresher I

**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

**Notes:**

Week 3

**Subject:** Programming Refresher II

**Goal:** Students will understand how to use Boolean algebra to simplify logic problems and will know how to implement logic branching in C#.

**Learning Objective:**

1. Students will be able to write a simple if statement
2. Students will be able to simplify Boolean algebra statements using Boolean algebra

**Notes:**

Week 4

**Subject:** Classes

**Goal:** This module teaches students how to define and instantiate standard classes in C#.

**Learning Objective:** Students will be able to implement and use a class using c#.

**Notes:**

Week 5

**Subject:** Constructors

**Goal:** This module will introduce using more sophisticated constructors in c# to gain an understanding of how to use constructors to simplify code.

**Learning Objective:**

1. Be able to implement a default constructor
2. Implement a constructor with parameters

**Notes:**

#### Week 6

**Subject:** Member Variables/Enumerated Type

**Goal:** This module will introduce students to declaring class-level variables and local method level variables.

**Learning Objective:**

1. Students will be able to implement class member variables.
2. Students will be able to implement local variables in methods.
3. Students will be able to declare and use enumerated types

**Notes:**

#### Week 7

**Subject:** Methods

**Goal:** This module will introduce the students to breaking their programs into modules that can be defined in methods. Students will learn how to pass in parameters and return values from the methods.

**Learning Objective:**

1. Students will know how to implement a method.
2. Students will learn how to pass in parameters to a method.
3. Students will learn how to use information returned from a method.

**Notes:**

#### Week 8

**Subject:** Midterm

#### Week 9

**Subject:** Polymorphism

**Goal:** This module will teach students about how classes can be constructed in multiple ways to do multiple tasks and a method can be defined multiple times and the method will act as it should based on how it is called.

**Learning Objective:**

1. Be able to write methods with multiple definitions.
2. Students will be able to write constructors using multiple definitions.

**Notes:**

Week 10

**Subject:** Information Hiding

**Goal:** This module will teach students when and why it is appropriate to hide information in a class and when it is appropriate to show member information to outside classes.

**Learning Objective:**

1. Know when private should be used for a variable or method.
2. Know how to make methods and variables private.
3. Know when protected should be used in methods and variables
4. Know how to make methods and variables protected.
5. Know when public should be used in methods and variables
6. Know how to make methods and variables public.

**Notes:**

Week 11

**Subject:** Abstraction

**Goal:** This module will introduce students to effectively using generalization of classes (referred to as abstraction)

**Learning Objective:**

1. Know how to implement an abstract class
2. Know how to inherit from an abstract class
3. Know how to decide when it is important to make an abstract class and know how to appropriately place methods and variables at the correct level of abstraction.

**Notes:**

Week 12

**Subject:** Overloading Methods

**Goal:** This module will introduce students to specializing methods inherited from super-classes by overloading them.

**Learning Objective:**

1. Be able to write a method that overloads a super-class method.
2. Effectively use method overload in code

**Notes:**

Week 13

**Subject:** Static Classes, Methods, Variables

**Goal:** This module will introduce the students to better coding practices by showing how to declare static methods, classes and variables when it is practical to do so. The emphasis of this module is to provide an understanding of how instance objects and values defer from static classes and values.

**Learning Objective:**

1. Know how to make methods and variables static
  2. Know when it is appropriate to use static classes, methods and variables.
- Be able to implement and call static classes, methods and variables.

**Notes:**

Week 14

**Subject:** Project Demos

Week 15

**Subject:** Final

**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%