

CS 1309-001 Computer Science I

Fall 2019, Sul Ross State University

Instructor: Ms. G

Office Location: MAB 109B

Office Phone: 931-237-3324

Email: tglenn2@sulross.edu

Office Hours:

M & W 8 – 9am & 1-2:30 pm

You may call from 10 am to 10 pm Mon-Sun if you need assistance with your homework or have issues that need to be discussed.

Call 931-237-3324 Be brave be bold call if you need help.

Class: *MW* 9:00 am - 9:50 am BAB 302;

Lab001: **BAB 302** *W* 3:00 pm - 4:50 pm.

Textbook: [C# Programming for Beginners: An Introduction and Step-by-Step Guide to Programming in C#](#)

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 provides the beginning programmer with a guide to developing structured program logic. It assumes no programming experience and will focus on the C# programming language. Students will be introduced to programming concepts such as structure, decision-making, looping, arrays, and files, with emphasis on good style and logical thinking. Students will be introduced to basic object oriented programming techniques. Students will learn how to code basic algorithms and will be able to implement primitive variables, arrays, if conditions, for loops, while loops, input and output to the screen and to files, and will know how to implement a basic Class and demonstrate how to use it.

Schedule:

Week 1

Subject: Course Overview

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

Notes:

Week 2

Subject: Keyboard Input and Flow Charts

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

Notes:

Week 3

Subject: Boolean Logic and Branching

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: Branching Continued

Goal: This module is designed to ensure students understand how to implement more complex branching if conditions in order to make more complex computer programs.

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

1. If statements

2. If/then/else statements
3. Compound if statements

Notes:

Week 5

Subject: Branching and 1-Dimensional Arrays

Goal: This module will introduce to nested if statements and 1 dimensional arrays which is the first and basic data storage structure students will be exposed to.

Learning Objective:

1. Be able to effectively implement a nested if statement.
2. Know how to implement and use a one-dimensional array.

Notes:

Week 6

Subject: 2-Dimensional and Parallel Arrays

Goal: This module will introduce students to more complex implementation and usages of arrays.

Learning Objective:

1. Students will be able to implement and use a 2-Dimensional array.
2. Students will be able to implement and use a parallel array.

Notes:

Week 7

MIDTERM

Week 8

Subject: Code Repetition (Loops)

Goal: This module will introduce the students to looping within a program to perform repeated functions. This module will also introduce the students to the while loop to loop when there is not a known number of times that the loop much be repeated.

Learning Objective:

1. Understand and be able to implement a while loop
2. Understand and be able to implement a do/while loop
3. Students will know how to implement a for loop
4. Students will know how to implement a foreach loop
5. Students will be able to use a for loop to iterate through an array
6. Students will be able to use a foreach loop to iterate through an array

Notes:

Week 9

Subject: File Input/Output

Goal: This module will introduce the students to persistent storage. They will learn how to save information to the hard drive and be able to read information from the hard drive.

Learning Objective:

1. Be able to write code to write out to a file
2. Write code to append data to a file
3. Write code to check to see if a file exists
4. Write code to read from a file and save to an array

Week 10

Subject: Bubble Sort

Goal: This module will introduce the students to the typical first sorting algorithm known as bubble sort. Students will also get an introduction to Classes and Object Oriented Programming.

Learning Objective:

1. Why sorting is important and be able to implement a bubble sort without using the internet or other reference material.

Week 11

Subject: Classes/Constructors and Methods

Goal: This module will introduce students to effectively using Constructors and provide a brief introduction to using Methods to modularize code.

Learning Objective:

1. Know how to implement a Class with multiple constructors.
2. Know how to write a basic method without parameters or a return value.

Week 12

Subject: Methods Continued

Goal: This module will introduce students to methods with return values and parameters.

Learning Objective:

1. Be able to write a method that returns a value
2. Be able to write a method that passes in parameters
3. Be able to write overloaded methods. That is methods with the same name but takes different parameters.

Week 13

Subject: Protection Levels

Goal: This module will introduce students to how parts of code are hidden from other parts of a program.

Learning Objective:

1. Be able to write a methods with specific protection levels appropriately

Week 14

COURSE REVIEW

Week 15

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.

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%

Homeworks: 50%

Participation/Attendance: 25%

Midterm/Final: 25%