CS 135: Computer Science I
University of Nevada, Las Vegas
(Spring 2026)

Instructor

Name: Dr. Sohini Roy
Phone: 702-895-5409
Office: TBE-B364
Email: sohini.roy@unlv.edu
In-person Office hours: Tuesday: 2PM – 4PM
Online Office hours: Thursday: 2PM-4PM

Course Objectives

Problem-solving methods and algorithm development in a high-level programming language. Program design, coding, debugging, and documentation using techniques of good programming style. Program development in a powerful operating environment. Three hours lecture and one hour lab. 3 credits.

Pre-requisites

- C or better: MATH 127 (Pre-calculus II), MATH 128 (Pre-calculus and Trigonometry), or Calculus -or-
- SAT Math score of 630+ -or-
- ACT Math score of 28+

Recommended Text Books

- Fundamentals of C++ Programming by Richard Halterman December 2, 2018.
- C++ Programming: From Problem Analysis to Program Design by D.S. Malik

Student Learning Objectives

Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.

Specific Course Objectives and Expected Learning Outcomes

Upon completion of CS 135, students will be able to:

- Develop algorithmic solutions to basic problems and translate their algorithms into C++ programs that meet a provided set of specifications.
- Demonstrate and use good programming style and adequately document programs.
- Compile and execute their programs in the target operating environment and learn appropriate testing and debugging strategies.
- Use appropriate control structures (sequence, selection, and iteration) in their programs
- Develop modularized programs using functions and passing parameters.
- Understand and properly use strings and file streams.
- Understand and properly use one-dimensional arrays and records.

Labs

CS 135 includes a mandatory lab that is remote learning based and has two small assignments per week. The lab assignments provide additional programming practice and are intended to reinforce specific concepts from the lectures (due at the date and time specified on Canvas). Teaching assistants are available at set times during the week to provide assistance to students. See the lab syllabus for further details.

Academic Policies

Grading Criteria

The overall grade for the course will consist of four components:

- Lecture Assignments (15%)
- Labs (15%)
- Exam 1 (30%)
- Exam 2 (30%)
- Final Exam (10%)

Grades will be posted throughout the semester. Grading is as follows:

Grade	Percent	Grade	Percent
A	100% - 93%	C	76.99% - 73%
A-	92.99% - 90%	C-	72.99% - 70%
B+	89.99% - 87%	D+	69.99% - 67%
B	86.99% - 83%	D	66.99% - 63%
B-	82.99% - 80%	D-	62.99% - 60%
C+	79.99% - 77%	F	59.99% - 0%

Note: grades are not rounded.

Assignments

Throughout the semester, you will be provided with a series of carefully crafted assignments, including homework, worksheets, and other relevant materials (collectively referred to as "assignments"). These assignments have been designed to enhance your mastery of course content and to promote active engagement with the course material.
It is important to note that no makeup assignments will be offered if any are missed. As such, we strongly encourage you to prioritize timely and comprehensive completion of all assignments to ensure that you do not miss any crucial learning opportunities.

To facilitate fair and accurate evaluation of your progress, all assignment scores will be assigned as assignment points. Graded assignments will be returned to you expeditiously to allow ample time for self-reflection and learning. We kindly request that you carefully review your grade summaries for any potential posting errors, as it is ultimately your responsibility to ensure that all scores are recorded accurately. Any score discrepancies must be brought to the instructor's attention within one week of posting. Please note that after this time, all scores will be considered final.

Codegrade

In order to streamline and optimize the grading process for programming assignments in this course, all programs submitted will be conducted and evaluated via CodeGrade, a sophisticated and state-of-the-art tool for programmatic evaluation.

It is mandatory that all programs are submitted before the due date/time of the task to ensure prompt and efficient evaluation. Please be advised that any submissions that are tardy will be subjected to a penalty of a 2% deduction in grade for every hour beyond the deadline, up to a maximum of 48 hours. Submissions that exceed the 48-hour threshold will regrettably be assigned a score of zero. Furthermore, it is critical to note that no individual may submit code on behalf of any other student, including instructors, teaching assistants, or any other party. Please be aware that CodeGrade will assign the highest score achieved, even if a lower score is received upon subsequent submissions.

Exams

To ensure that you have ample opportunities to demonstrate your mastery of the course material, we have established a rigorous examination schedule consisting of three exams, including a final examination. Each examination will be designed to assess your understanding of the course material comprehensively, with a particular focus on the material covered since the last exam. We strongly encourage you to prioritize a thorough understanding of all course content to optimize your performance on each exam. Furthermore, exams will be considered closed resource, and students will not be permitted to access external materials during the examination period.

Please note that no makeup exams will be offered for any missed tests. As such, we strongly recommend that you prioritize your exam preparation and attendance to ensure that you do not miss any crucial evaluation opportunities.

Online Test Policy

We acknowledge that, in rare instances, it may be necessary to administer quizzes or exams online. In such cases, we will utilize a lockdown browser with robust audio and video monitoring to ensure the integrity and fairness of the examination process. To ensure the highest levels of examination security, we require all students to activate their audio and video functions throughout the exam period. Finally, we ask that all students have a valid Rebel Card to confirm their identity during the examination process, further ensuring that all academic evaluations are conducted in a fair and unbiased manner.

Artificial Intelligence (AI)

Students are only allowed to use cs50.ai for AI-related assistance in this course. Each student is expected to complete each assignment without substantive assistance from other automated tools. The student is expected to:

- Do their own work
- Not use any generative AI tools such as, but not limited to, chatbots, text generators, paraphrasers, summarizers, or solvers, except for cs50.ai.
- Consult with their professor before submitting work if there are any questions about what constitutes acceptable use of cs50.ai or other tools.

Documenting Assistance from cs50.ai

Students are required to clearly document any assistance received from cs50.ai in their code. To maintain academic integrity and transparency, please follow these guidelines:

Commenting Assistance

- Include a comment in your code that specifies where cs50.ai was used and how it helped.
- Be specific about the nature of the assistance (e.g., "cs50.ai helped debug this loop" or "cs50.ai suggested this sorting algorithm").

Examples of Proper Documentation

```
// Assistance from cs50.ai:
// Helped debug the logic error in the if-else condition.
if (x > y) {
// End assistance from cs50.ai
   // ...
}
```

Submission Requirements

- Ensure all instances of cs50.ai assistance are documented before submitting your work.
- Assignments lacking proper documentation of cs50.ai usage may be subject to the academic integrity policy.

Class Rules

1. All students are expected to do their own work on exercises, assignments and examinations (see next page).
2. All assignments must be submitted via the class web page on Canvas. No email or hardcopy submissions.
3. All programs must be submitted for CodeGrade (Ubuntu Linux) automated grading.
4. See the "Coding Style Documentation and Grading" document on the class WebCampus/Canvas website for information on the commenting style specified for the class.
5. Check the class WebCampus/Canvas page for notes/assignments/quizzes/announcements on a regular basis.
6. Personal electronic devices are not permitted in class except for written note taking. No audio/video recording.
7. If you are distracting others in class you may be asked to refrain from distracting and/or be asked to leave class.

Department of Computer Science Academic Integrity Policy

Each student enrolled in a course offered by the Department of Computer Science is expected to do their own work when preparing written or programming assignments, as well as examinations. Students must adhere to the academic integrity policy provided by the instructor and the university. It is also each student's responsibility to notify the instructor if they become aware of any activities that would violate the academic integrity policy of the class.

Permitted and Prohibited Assistance

Students may use the permitted websites for assistance in this class. All other sites should not be used for assistance in this class. Instructors may provide additional links in assignment instructions or on the class WebCampus/Canvas website. All other sites should not be used, and their use may be treated as academic misconduct.

Each student enrolled in a course offered by the Department of Computer Science is expected to do their own work when preparing written or programming assignments, as well as examinations. Students must adhere to the academic integrity policy provided by the instructor and the university. It is also each student's responsibility to notify the instructor if they become aware of any activities that would violate the academic integrity policy of the class.

Permitted websites:

- Any website provided to you on the class canvas site or posted by the instructor in the official Discord server. That is it. Do not use any websites for this class other than the ones provided.
- C++ Official Documentation (cplusplus.com)
- The official UNLV WebCampus/Canvas site for your class.
- The official UNLV Discord site for class.
    - Discord sites that are not the official site should not be used. Discussion boards and/or social media sites should not be used unless they are specifically approved by the instructor. All other websites may not be used for assistance in this class.

Assistance that is **allowed** in the preparation of coursework:

- Information/code provided in the textbook.
- Information/code provided in the class notes (on the Canvas).
- Assistance provided by the course instructor.
- Assistance provided on the official Discord site by the instructor, other instructors of different sections, and teaching assistants for the class.
- Websites on the permitted websites list above.

Examples of prohibited practices include, but are not limited to:

- Copying answers/code from a fellow student, friend, or relative.
- Providing answers/code to a fellow student.
- Collaborating (sharing) answers/code.
- Using the Internet to develop a strategy for solving a problem (finding an algorithm).

- Using the Internet to solicit a solution to an assignment.
- Using the Internet to find an example of code that solves a portion of an assignment.
- Asking and/or paying someone to complete your assignments.
- Creating/using a non-instructor provided social media site (i.e. Chat server such as Discord).

Consequences of violating the academic integrity policy

1. An Alleged Academic Misconduct Report will be completed and a copy sent to the Office of Student Conduct.
2. You will receive an F in the class with no additional class participation allowed.

CS 135 Tentative Schedule

Week	Topic	Text Chapter	Homework
1	Intro/Syllabus, Setting up a Programming Environment (VSC/Terminal/WSL), Basic Linux Commands, Program Creation Process	1-2	
2	Basics of a C++ Program, Input/Output, Variables/Constants, Primitive Data Types, Assignment, Binary/Unary Arithmetic, Mixed Expressions, Casting	2-4	HW0 Due
3	Libraries (-and-)	3-5	HW1 Due
4	Logical/Relational Operators, Selection (n-way/switch), Error Handling, Short Circuit Evaluation	5, 7	HW2 Due
5	Loops (while/for/do,while), break/continue, Recovering Failed Input	6, 7	HW3 Due
6	Review for Exam 1 & Exam 1		
7	Nested Loops, File I/O, Strings as Arrays	11	HW4 Due
8	1D Arrays, Array Linear Search, Array Bounds, Range-Based for Loops, Parallel Arrays	11, 12	HW5 Due
9	Comma Separated Values, Bubble Sort, 2D Arrays	11, 12	
10	Functions (value/void), Scope (local/global), Parameters (formal/actual and value/reference), Arrays as Function Parameters	10	HW6 Due
11	Default Parameters, Function Overloading, Structs Intro	8, 9, 13	
12	Structs, Structs and Functions, Structs and Files, Arrays of Structs, Structs within Structs	14, 15	HW7 Due
13	Review for Exam 2, Exam 2		
14	Review for Final Exam		

Week	Topic	Text Chapter	Homework
15	Study Week - No classes		HW8 Due
16	Final Exam - Check final exam schedule on MyUNLV		

University Policies
Please see: https://www.unlv.edu/policies/students