

Introduction to C++

Lecturer	Dr. Wendy Fisher
Institution	Colorado School of Mines
Email	wfisher@mines.edu

Course Description:

This course introduces fundamental computer programming concepts using a high-level language and a modern development environment. Programming skills include sequential, selection, and repetition control structures, functions, input and output, primitive data types, basic data structures including arrays and pointers, objects, and classes. Software engineering skills include problem solving, program design, and debugging practices.

Course Learning Outcomes:

The goal of this course is to open your mind to computational thinking, to educate you to leverage programs as tools in your field of study, and to empower you with a fundamental knowledge of programming. Concepts and skills will be developed through a series of small projects throughout the semester and a final project.

By the end of this course, students will be able to:

- Identify and construct proper object-oriented C++ syntax. Explain the components that comprise C++ syntax and how the components operate together.
- Design and write pseudocode to accomplish a given task or solve a defined problem using common programming design structures including conditionals, loops, functions, arrays, and classes.
- Translate pseudocode into valid and correct C++.
- Analyze & trace existing code and calculate the output given an initial input while explaining what the code does.
- Identify and correct errors in C++ syntax, program structure, and algorithm.
- Discuss at a high level how C++ code becomes an executable program and how data is stored in computer memory.
- Discuss ethical practices in code development and code usage.

Materials/Text:

Textbook (electronic zyBook):

Introduction to Programming in C++

Purchasing link and code to register will be provided prior to the first day of class.

Expectations:

You are expected to attend all classes and come prepared to actively participate in the activity and

discussion for the day. To do well in this course, you must keep up with the out of class material and engage in the in class activities. We promise to prepare you and to provide you with the tools needed to succeed.

Evaluation:

The final course grade will be computed from the following course percentage breakdown :

5% zyBook Completion	10% Quizzes
25% Assignments and Labs	15% Midterm Exam
10% Final Project	15% Final Exam

Final grades will be determined using a straight scale; letter grades will be posted in Canvas:

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

* You must pass both the final exam and the final project with at least a 60% grade on each in order to pass this course.

For a discrepancy in any grade in which you think you deserve more points than you received, you must raise the issue within one week of the day the item was returned. No claims, justifiable or not, will be considered after this deadline.

Grading will be done on an absolute, but adjustable scale. This means that there is no curve. Anyone earning 90% or more of the total number of points available will receive a grade of A; 80% or higher a B, etc. This scale may go down, but it will not go up.

Late policy:

Late work is strongly discouraged. All work will be accepted within 4 working days after the deadline (weekend and holiday days do not count in the 4 days). Students will lose 10 percentage points per day late. After 4 days, the work will not be accepted at all.

Coursework Return Policy:

Homework and exams will be graded and returned as quickly as possible, generally within a week.

Academic Integrity:

All students are advised to be familiar with university policy on Academic Integrity. In addition, The following Collaboration Policy exists for this course. This policy is a minimum standard; your instructor may decide to augment this policy.

1. If the project is an individual effort project, you are not allowed to give code you have developed to

another student or use code provided by another student. If the project is a group project, you are only allowed to share code with your group members.

2. You are encouraged to discuss homework and project assignments with other students in the class, as long as the following rules are followed:
 - a. You view another student's code only for the purpose of offering/receiving debugging assistance. Students can only give advice on what problems to look for; they cannot debug your code for you. All changes to your code must be made by you.
 - b. Your discussion is subject to the empty hands policy, which means you leave the discussion without any record [electronic, mechanical or otherwise] of the discussion.
3. Any material from any outside source such as books, projects, and in particular, from the Web, should be properly referenced and should only be used if specifically allowed for the assignment.
4. To prevent unintended sharing, any code stored in a hosted repository (e.g., on github) must be private. For group projects, your team members may, of course, be collaborators.
5. If you are aware of students violating this policy, you are encouraged to inform the professor of the course. Violating this policy will be treated as an academic misconduct for all students involved.

Violations of this policy result in one of a range of punitive measures, from a zero score for an assignment, up to and including a course letter grade drop for all students involved.

Academic misconduct associated with an exam grade will likely result in course failure.

Student Absences:

All students are advised to be familiar with university policy regarding the make-up of work missed due to excused absences.

Learning Environment:

Fundamentally, I expect and require respect in this course for yourself, your classmates, and your instructor.

- Respect for yourself includes taking care of yourself physically and mentally and advocating for an environment that facilitates learning for you.

-Respect for your classmates includes recognizing and appreciating the diversity of backgrounds and experiences of your classmates and making it your interest to foster a learning environment for everyone; all are welcome.

-Respect for your instructors (as well as your classmates) includes not participating in disruptive or distracting behavior: talking, playing games, or web surfing during lecture, for instance, make it difficult for others to focus on the reason we are all here.

-Respect must be mutual to be effective; we (your instructors) will be held to the same standards of respect.

Please let your instructor know if you become aware of an issue with the classroom (or out-of-classroom) environment with regards to these policies.