COMP 2580 INTRODUCTORY C++ PROGRAMMING FOR ENGINEERS(3 credit hours)

Elmira College

SPRING 2025

Required Text:

Gary J. Bronson (2013). ***C++ for Engineers and Scientists*** (4th ed.). Cengage Learning.
Supplemental readings might be included to illustrate or expand on textbook readings.

Pre-requisites: MATH 1535 Calculus with Analytic Geometry I.

Course Description

This course is designed to equip students with the essential programming skills needed to tackle engineering problems through a procedural-oriented approach in the C++ programming language. C++ is known for its efficiency, flexibility, and wide-ranging applications, making it a vital tool in various engineering disciplines. And the course emphasis on numerical algorithms, which are crucial for solving complex engineering challenges.

Course Objectives and Goals

- ➢ Develop a strong foundation in algorithmic thinking and problem-solving, particularly in the context of numerical engineering problems.
- ➢ Master the art of procedural programming in C++, including topics such as data types, functions, and libraries relevant to engineering.
- ➢ Explore a range of numerical algorithms used in engineering.
- ➢ Apply numerical methods and C++ programming to practical engineering challenges.
- ➢ Learn software development practices and principles that help engineers create efficient, maintainable, and robust applications.

Evaluation of Performance

Your grade will be based upon your performance on exams, assignments, and participation.

2 Assignments	10%
4 Labs	20%
2 Projects	20%
Midterm Exam	25%
Final Exam	25%
Total	100%

Grades will be assigned as follows:

A	93% and above	B-	80 - 82%	D+	67 - 69%
A-	90 - 92%	C+	77 - 79%	D	63 - 66%

B+	87 - 89%	C	73 - 76%	D-	60 - 62%
B	83 - 86%	C-	70 - 72%	F	59% or below

Withdrawal Policy: Please see Elmira College Bulletin for information on this policy.

Academic Honesty: Please read the section on Academic Honesty in the **Code of Conduct**. Briefly, academic dishonesty includes: cheating, fabrication, facilitating academic dishonesty, and plagiarism. Ask if you have any questions on whether something constitutes as academic dishonesty. All work must be original and new. Past assignments from current or other courses will not be accepted. Academic dishonesty will not be tolerated. It will result in zero on the assignment, and a report will be filed with the school. Continued practice will result in failure of the class. Institutional penalties may also apply with repeated acts of academic honesty.

Student Responsibility:

- It is your responsibility to keep track of assignments and due dates.
- You should ask questions concerning assignments and lectures, if you need any clarifications.
- If you are struggling in class, have concerns, and/or unsure about expectations, please stop by during office hours or make an appointment for another time.

Tentative Schedule of Topics

Topic	Materials	Tasks & Evaluations
Fundamentals of C++ Programming	Chapter 1	
Preliminary Four: Algorithms	Chapter 2	Assignment#1
Introduction to C++	Chapter 3	
Programming Style	Chapter 4	Lab#1
Data Types	Chapter 5	
Arithmetic Operations	Chapter 6	Project#1
Formatting Numbers for Program Output	Chapter 7	
Using Mathematical Library Functions	Chapter 8	Lab#2
Selection Structures	Chapter 9	
Repetition Statements	Chapter 10	Midterm Exam
Modularity Using Functions	Chapter 11	
Arrays	Chapter 12	
I/O Streams and Data Files	Chapter 13	Assignment#2
Introduction to Classes	Chapter 14	
Adding Functionality to Your Classes	Chapter 15	
Structures	Chapter 16	Lab#3
Structures (Cont.)	Chapter 17	
Structures as Function Arguments	Chapter 18	Project#2
Dynamic Data Structure Allocation	Chapter 19	
Introduction to Root Finding	Chapter 20	Lab#4
Introduction to Numerical Integration	Chapter 21	
The Trapezoidal Rule	Chapter 22	Final Exam