

0806C303

Programming in C++

Instructor: TBA

Email: TBA

Time: May 12, 2025 - June 13, 2025

Office Hours: 2 hours (according to the teaching schedule)

Contact Hours: 60 (50 minutes each)

Credits: 4

Course Description

Covers core programming skills including procedural programming in C++, algorithms, modularity, and abstraction. Out-of-class assignments give the student a chance to tie the concepts together and learn basic programming style, documentation, and development skills necessary for working in a team environment.

Required Textbook(s)

Programming in C++, by Nell Dale and Chip Weems, 5th Edition, 2010.

Prerequisites

0806C220 Introduction to Programming.

Course Goals

- Develop algorithms to solve "computer-solvable" problems. Test algorithms.
- Translate algorithms to C++ programs.
- Debug, run and test C++ "procedural" programs.
- Understand the software development process by using requirements to design, implement and test C++ programs.

Course Schedule

Please note that the schedule is meant to give an overview of the major concepts in this course. The actual weeks and days on which the topics will be covered are subject to change at the discretion of the course instructor.

Week 1	<ul style="list-style-type: none">• Introduction to computer programming; Programming life cycle• c++ program elements with "Hello World" example• Input, output• Assignment and interactive input: Assignment operators• Data types, variables, declarations• Arithmetic operations, expressions and precedence• Complete assignments
Week 2	<ul style="list-style-type: none">• Relational Expressions: The if-else statement; switch case statement and nested if statements• The iterative loops: for loop and while loop• while loop and do-while loop• Complete assignments

<p>Week 3</p>	<ul style="list-style-type: none"> ● Mathematical library functions and symbolic constants ● Modularity using functions: Function and parameter declarations ● Modularity using functions: function returning a single/multiple values Variable scope ● Arrays: One-dimensional arrays, initialization and arrays as arguments ● Complete assignments and test
<p>Week 4</p>	<ul style="list-style-type: none"> ● Array based list operations ● Arrays & Pointers: Introduction to pointers and array name as pointers ● Arrays & Pointers: pointer arithmetic and pass addresses ● Two-dimensional arrays ● Introduction to classes: Object-based programming, classes, constructors and class scope ● Complete assignments
<p>Week 5</p>	<ul style="list-style-type: none"> ● Adding functionality to classes: Creating class operators, methods and this pointer ● File I/O ● more on string class ● Inheritance ● Polymorphism ● Complete assignments and Test

Course Requirements

About Computing Facilities

You will need a computer for this course. The course will contain lab sessions as well as practice coding assignments during lectures.

About Assignment Submission

Every homework/lab assignment must be submitted on the day it is due. Late work will be penalized one grade notch (e.g., B- to C+) for each day it is late. The weekend counts as one day. Only late work can be submitted electronically to the instructor's official email address.

How to Dispute a Grade for an Assignment or Exam?

If you wish to dispute the grade given to a project/ homework assignment, you must do so by coming to the office hours and disputing the grade in person within a day after the assignment is graded and returned.

If you wish to dispute the grade assigned to a paper or a question on an exam, you must do so IN WRITING within a day after the exam or paper has been returned. You must include a specific rationale for why your answer is correct, or why the paper deserves a higher grade. "I think I deserve a better grade" does NOT constitute a rationale.

Grading Policy

Final letter grades are determined from your final cumulative score that is computed using the following breakdown:

Gradable Contents

Programming Assignments (including labs and projects)	50%
Midterm Exam	25%
Final Exam	25%
Total	100%

Letter grades are assigned at the end of the semester. We do not curve individual assignments or exams. At the end of the semester after all of the scores are recorded, the thresholds are set given the difficulty of the course work during the semester. Thresholds are raised if the course work was easier or lowered if it was harder. By adjusting and setting thresholds at the end of the semester we can account for varying difficulty among semesters to ensure grading consistency.

Grading Scale

The instructor will use the grading system as applied by JNU:

Definition	Letter Grade	Score
Excellent	A	90~100
Good	B	80~89
Satisfactory	C	70~79
Poor	D	60~69
Failed	E	Below 60

Academic Integrity

As members of the Jinan University academic community, students are expected to be honest in all their academic coursework and activities. Academic dishonesty, includes (but is not limited to) cheating on assignments or examinations; plagiarizing, i.e., misrepresenting as one's own work any work done by another; submitting the same paper, or a substantially similar paper, to meet the requirements of more than one course without the approval and consent of the instructors concerned; or sabotaging other students' work within these general definitions. Instructors, however, determine what constitutes academic misconduct in the courses they teach. Students found guilty of academic misconduct in any portion of the academic work face penalties that range from the lowering of their course grade to awarding a grade of E for the entire course.