Course Syllabus

CSI-140 Introduction to Programming

Instructor: Dr. Vikas Thammanna Gowda Semester: Fall 2025
Contact: vthammannagowda@champlain.edu Office Location: West Hall 200
Office Hours: click here

1 Course Overview

First course in computer programming in a high-level language. Emphasizes analyzing problems, designing solutions, and implementing them in the form of a well-structured program, using the procedural aspects of C++. General topics include: the theory of computing, number systems, Boolean logic, algorithm design and implementation. Programming topics include: memory and variables, mathematical operations, basic input and output, decision making, repetitions, functions, and arrays.

2 Prerequisites and Audience

- **Prerequisite:** None. No prior programming experience required.
- **Target students:** Undergraduates from any discipline who want to understand and learn the basics of computing.

3 Learning Outcomes

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

- Define key programming concepts such as variable, constant, data type, function, loop, and array in C++.
- Explain the structure of a C++ program and the role of the compiler in translating code into executable instructions.
- Write and compile simple C++ programs that use variables, arithmetic operations, and input/output statements to solve basic problems.
- Implement decision-making in programs using `if`, `if-else`, and `switch` statements, and repetition using `for`, `while`, and `do-while` loops.
- Trace and debug given C++ code to identify syntax and logic errors, and determine the correctness of program output.
- Break down a problem into smaller subproblems and design modular solutions using functions, applying parameter passing by value and by reference.
- Compare alternative algorithms or approaches for solving a problem in terms of readability, efficiency, and maintainability.
- Design and implement a complete C++ program that integrates input/output, decision structures, loops, arrays, and functions to solve a moderately complex problem.

4 Course Components and Assessment

4.1 Course Structure

This course is organized into six instructional units. Each unit is designed to build progressively on the knowledge and skills acquired in previous units, combining conceptual understanding with practical application.

For each unit:

- **Assigned Readings & Practice Exercises:** Students are expected to complete assigned readings, and exercises prior to class to prepare for lectures and discussions.
- **Lectures and Discussions:** Class sessions will introduce and explain key concepts, provide demonstrations, and encourage questions and dialogue.
- **In-Class Activity:** Interactive activities will allow students to apply concepts in a guided environment and receive immediate feedback.
- **Homework:** Assignments will reinforce the topics covered in lectures and activities, providing additional practice outside of class.

In addition:

- **Tests:** Tests will be conducted at the end of every two units to assess comprehension and application of the material covered.

Note: A brief agend will be uploaded to Canvas before every class.

4.2 Grading Policy

Component	Weight
Practice Exercises	20%
Homeworks	30%
Tests	30%
Final Exam	20%

4.3 Grading Scale

Score Range (%)	Grade
≥ 93.00	A
90.00–92.99	A-
87.00–89.99	B+
83.00–86.99	B
80.00–82.99	B-
77.00–79.99	C+
73.00–76.99	C
70.00–72.99	C-
67.00–69.99	D+
63.00–66.99	D
60.00–62.99	D-
≤ 59.99	F

Note: Class participation will be subjectively evaluated and will be used in borderline cases to determine the final grade.

5 Tools and Technologies

Students are required to have complete access to a functioning laptop or PC with internet capabilities. It is strongly recommended that the computer is capable of having a C++ environment installed.

6 Weekly Schedule (Tentative)

Week	Topics
1–2	**Onboarding & Unit 1**: Course Overview, Intro to Computing, Algorithms, Program Translation, Software Development Life Cycle, Number systems (Binary, Decimal, Hexadecimal).
3–5	**Unit 2**: Variables, Rules for Identifiers, Basic Datatypes (int, double, char, boolean, string), Operators (arithmetic, comparision, logical, special), I/O, Errors
5–7	**Unit 3**: if, if-else, if-elseif ladder, switch, nested statements
7–9	**Unit 4**: for, while, do-while loops, nested loops
9–11	**Unit 5**: Inbuilt functions, User-defined functions, Call by Value, Call by Reference
11-13	**Unit 6**: Arrays, Structures and Applications
14	**Thanksgiving Break**
15	**Revision and Final exam study guide**

7 Major Assignments and Deliverables

7.1 Practice Exercises

Students will read the lecture notes and answer online exercises for each chapter before the lecture and discussion session. They are due **3 to 5 days from the day of assignment**.

Instructions:

- You will have around 8 exercises throughout the semester.
- **Two of the lowest scores will be dropped.**
- These are online quizzes on Canvas.
- Expect 5–10 multiple choice questions
- **You have unlimited attempts and the highest score counts.**

Late submission policy: NO late submissions unless prior arrangements made, it defeats the entire purpose.

Practice Exercise Schedule: The exact dates will be announced via Canvas.

7.2 Homeworks

Assignments will reinforce the topics covered in lectures and activities, providing additional practice outside of class. They are due **1 week from the day of assignment**.

Instructions:

- **Handwritten:**
  - Use the handout provided by the instructor and complete it by hand. (There will be a few additional copies in the file folder outside my office door — West Hall 200.)
  - Print a blank template from Canvas and write on it by hand.
  - Use the soft copy from Canvas and write directly on a tablet (e.g., using a stylus).
- **Use of word processor:**
  - The homework should be answered in chronological order.
  - Each question must be added in **bold** before answering.
  - Submission must be a single PDF file.
  - Use of AI to generate report/answers will be considered as academic dishonesty.
- **Why PDF?** PDF files are universally compatible, meaning they can be opened and viewed on virtually any device with a PDF reader. This makes them ideal for sharing documents with a wide range of recipients, regardless of their software or hardware.
- **Individual Work:** This is an individual homework assignment. While you are encouraged to discuss the problem and possible approaches with your classmates, all work must be completed independently.
- **Plagiarism Policy:** Any form of plagiarism — including copying code, solutions, or text from another student — will be considered academic dishonesty and will be reported according to college policy.

Submission Guidelines:

- Drop off your completed assignment in the file folder outside my office door (West Hall 200).
- Submit it during class.
- Scan your work into a PDF and upload it to Canvas.
- **If you wish to upload as images to Canvas, combine all pages into a single, high-resolution PDF file that is clear and easy to read.**
- Do not upload a zip folder that contains multiple files.

Late submission policy: 50% penalty for late submissions within 1 week; no credit after 1 week unless prior arrangements made.

Failure to follow the instructions and submission guidelines may result in a reduction of up to 100% of the points.

Tentative Homework Schedule:

HW #	Topic	Assigned	Due Date
1	Unit 1	09/05	09/11
2	Unit 2	09/23	09/29
3	Unit 3	10/07	10/13
4	Unit 4	10/21	10/27
5	Unit 5	11/07	11/13
6	Unit 6	11/18	12/01

7.3 Tests

Tests will be conducted at the end of every two units to assess comprehension and application of the material covered.

Instructions:

- Each test is worth **30 points.**
- Tests will be held at the beginning of class and will last **45 minutes**.
- **Tests are closed-book and closed-notes, but you will be allowed one A4 size cheat sheet.**
- **Format:** All questions will be multiple choice questions in the form of fill in the blanks, match the following, trace/debug the code, True/False.
- **Timing:** Tests will start promptly at the scheduled time. Latecomers will not receive extra time. **All tests will be collected only at the end of the allotted time.**
- **Early Completion:** If you finish early, you must remain seated and respect others by not causing any disturbance.
- **Prohibited Items:** The use of smartphones, smart calculators, smartwatches, or any other electronic devices is strictly prohibited during the entire test period. Any use will be subjected to academic integrity.
- **Missed Tests:** If a student misses a test, it cannot be made up under normal circumstances. Exceptions will be considered on a case-by-case basis ONLY WHEN NOTIFIED WELL IN ADVANCE and, if granted, the make-up test will be in a different format than the regular test.

To excel in tests you must be on top of lecture notes, practice exercises, in-class coding, activities, and homeworks.

Tentative Test Schedule:

Test #	Topic	Date
1	Unit 1 & Unit 2	09/30
2	Unit 1 & Unit 2	10/28
3	Unit 1 & Unit 2	12/02

7.4 Final Exam

It will be held during the officially scheduled finals week. It will be **cumulative**, covering material from all units of the course, including concepts, problem-solving techniques, and programming skills discussed throughout the semester.

The format and style of the questions will be similar to the tests, ensuring familiarity with the types of problems and the level of detail expected. Students should review all notes, tests, homework assignments, in-class coding, and activities as part of their preparation.

Details for the final exam including revision and study guilde will provided during Week 15.

8 Collaboration and Academic Integrity

In addition to skills and knowledge, Champlain College aims to teach students appropriate Ethical and Professional Standards of Conduct. The Academic Honesty Policy exists to inform students and Faculty of their obligations in upholding the highest standards of professional and ethical integrity. All student work is subject to the Academic Honesty Policy. Professional and Academic practice provides guidance about how to properly cite, reference, and attribute the intellectual property of others. Any attempt to deceive a faculty member or to help another student to do so will be considered a violation of this standard. The student's work must match the instructor's intended purpose for an assignment. While the instructor will establish the intent of an assignment, each student must clarify outstanding questions of that intent for a given assignment. The full text of the Academic Honesty Policy is in the Student Handbook.

I have zero tolerance for academic dishonesty. This means that a first offense results in an F for the course and the student shall be reported.

Responsible AI Collaboration

Use LLMs as assistants, not oracles: always verify computed statistics manually or with independent code. While I have no direct control over your use of AI tools or (LLMs), you are solely responsible for the originality, accuracy, and understanding of your submitted work. Please note that your understanding will be assessed through in-class tests. A significant mismatch between your homework performance and test performance may indicate overreliance on AI without genuine comprehension, which could result in additional questioning, a zero on the assignment, or further academic review.
If you choose to use AI tools, treat them as supplementary aids—you must still fully understand and be able to explain your solution without assistance.

LLMs can be valuable tools for supporting your learning in this course when used appropriately. They can help you:

- **Understand Concepts:** Ask follow-up questions about lecture topics, request simplified explanations, or explore alternative examples of programming concepts, number systems, and algorithms.
- **Practice Additional Exercises:** Request extra practice problems or coding challenges, including variations on homework problems, to strengthen your understanding.
- **Debugging Assistance:** Paste short snippets of your code to ask for clarification about errors or unexpected behavior (ensure you understand the explanation rather than just copying fixes).
- **Explore Real-World Applications:** Use LLMs to find examples of how concepts from the course are applied in industry, such as binary representation in networking or C++ in game development.
- **Preparation for Tests:** Use LLMs to test yourself by generating practice questions or asking for explanations of key terms and syntax.

9 Books / References

- Any Introduction to Programming in C++ textbook .
- Problem Solving with C++ (10th Edition), by Walter Savitch. Addison-Wesley/Pearson publishers.

10 Instructor Contact & Communication Protocol

- **Office Hours:** via prior appointments only. Click here to schedule a meeting.
- Preferred communication: institutional email; include course name and brief subject in subject line.
- Expect replies within 48 hours on weekdays.
- For quick debugging: include minimal reproducible example, expected vs actual behavior, and any LLM prompt + response if AI-assisted.

11 Strict No-Recording Policy

Prohibited: Recording of any kind—audio, video, screen capture, photographs, or live streaming—is not allowed during class meetings, labs, or discussions. This includes the use of smart pens and AI note-taking/transcription apps (e.g., live captions, Otter, Notability audio).
Purpose: To protect student/instructor privacy, foster open discussion, and safeguard course materials and intellectual property. Because audio can be AI-manipulated (e.g., voice morphing/deepfakes) and misused against the instructor or students, no personal exceptions will be granted.

This syllabus may be refined slightly during the course of the semester to accommodate class pace and student interests. Significant changes will be communicated in writing.