

Fundamentals of Computer Science

CSC 112

Course Syllabus

Fall 2019

Instructor: Dr. Grey Ballard

Email: ballard@wfu.edu

Office: Manchester 234

Office Hours: M 2-3:30 and W 3-4:30, or by drop-in or appointment

Teaching Assistant: John Farrell

CS Center Hours: MW 6-8pm

Class: 11-12:15 TR, Manchester 229

Lab: 3:30-4:45 R, Manchester 241

Text: Programming in C++ (zyBook)

Course Schedule: <http://users.wfu.edu/ballard/teaching/CSC112/>

1 Course Description

Problem solving and program construction using top-down design, data abstraction, and object-oriented programming. Memory addressing, dynamic memory allocation, and linear data structures are introduced.

2 Course Learning Objectives

In this course, students will review logical problem-solving, algorithm development, and programming basics, using C++ as their programming language. They will be able to implement increasingly complex programs using top-down design, functional decomposition, procedural programming, and object-oriented programming. They will extend their basic knowledge to an understanding of pointers, dynamic memory allocation, and data structures such as linked lists.

3 Course Topics

Problem Solving

- Review from CSC 111
 - The concept of an algorithm
 - Algorithmic problem-solving and problem-solving strategies
 - Structured decomposition and top-down design
 - Debugging strategies
 - Unit testing
- Searching and sorting algorithms

Fundamental Constructs

- Review from CSC 111
 - Basic syntax and semantics of a higher-level language, C++
 - Variables, types, expressions, and assignment statements
 - Simple input and output
 - Conditional and iterative control structures
 - Recursion
 - Functions and parameter passing
 - Call by value vs. call by reference
- Operator overloading

Data Representation, Data Types, and Data Structures

- Review from CSC 111
 - Binary and hexadecimal base systems
 - Primitive types such as character, integer, and floating point numbers
 - Arrays (including multi-dimensional ones) and vectors
 - Strings and string functions
 - References
- Type conversion and coercion
- Scope of identifier
- Namespaces
- Template classes
- Pointers
- C-strings
- Structs
- Linked lists
- Introduction to stacks, and queues
- Bit operations

Procedural and Object-Oriented Programming

- Procedural program layout
- Review from CSC 111
 - Object-oriented design
 - Classes, objects, and inheritance
 - Encapsulation, data protection, and information-hiding
 - Separation of interface and implementation
- Friend classes and functions
- Static data members
- Constructors and copy constructors
- Polymorphism

Other Topics

- Review from CSC 111
 - Types of errors: syntax vs logical vs run-time
 - Exception handling
 - Memory management (e.g., garbage collection, run-time storage management)
 - File input/output
- Command-line arguments
- Compiler directives
- Multiple file compilation
- Introduction to estimates of computational costs
- Run-time performance of a program

Tools and Skills

- Review from CSC 111
 - How to use an IDE effectively, including the debugger
 - How to use libraries and APIs, including their documentation
 - File structure fundamentals (directories and subdirectories)
- Working in the Linux environment, including basic Linux commands such as mkdir, cd, ls, cp, mv, grep, chmod, etc.
- Compiling, running, and debugging a program from the Linux command line
- Makefiles
- Introduction to network communication, file transfer, and shells in Linux (e.g., ftp and ssh)

4 Online Textbook

The required textbook for this class is available online. To access the text:

1. Sign up at `zybooks.com`
2. Enter zyBook code `WFUCSC112BallardFall12019`
3. Subscribe

A subscription is \$77. Subscriptions will last until Dec 28, 2019. You will be expected to register by the second class.

5 Assessment

There will be 9 projects, at least 4 quizzes, 2 midterms, and a final exam.

Projects will allow you to apply the concepts from lecture to solve problems with C++ programs. They will be completed both in and out of class, so you are expected to always bring your laptop to class.

Mastery of the material will be assessed with quizzes and tests. Quizzes will be in-class and last 10-15 minutes. The lowest quiz score will be dropped. The two midterms and final will be taken in class; the final exam will be cumulative. Make-up quizzes and tests will be administered only if excused in advance.

6 Collaboration Policy

I expect you to learn from each other throughout the semester, and I encourage you to work with a diverse set of your classmates. We will complete many ungraded, in-class exercises, including completing zyLabs built into the online textbook. These exercises are designed to be completed collaboratively, where code is developed and debugged in small groups.

I also expect you to demonstrate your mastery of the material on quizzes, tests, and projects. All tests and quizzes are to be done independently, and they will be completed in class. Projects may be discussed with other students, however the work submitted must be your own work and reflect your understanding of the material. This means

- you may solicit help and show your code to the instructor, the section TA, and TAs in the CS Center;
- you may **not** share code with other students (this includes copying-and-pasting, seeing others' code, and letting others see your code);
- you may discuss ideas, syntax, debugging strategies etc. with other students and outside tutors, as long as you're not looking at the same code;
- you may **not** copy code from Internet-based resources (aside from the textbook).

Violation of the collaboration policy is not acceptable and will be dealt with through the Honor System. I recommend that you retain drafts of your work and programs until the end of the semester in case a question arises as to authorship. (Committing frequently in GitHub provides a clear trace of your progress.)

7 Participation

Regular attendance of class is expected, as is verbal participation during lectures. There will also be frequent in-class exercises, done individually and often in small groups. Reading and interacting with the online textbook is also required: the sections of the text that correspond to each lecture will be listed on the course schedule (other sections will be marked Optional). Each week, you will be expected to complete at least 90% of the assigned reading and complete the associated “Participation Activities” and “Challenge Activities.”

8 Computer Science Experience

The purpose of the Computer Science Experience (CSE) is to give you an opportunity to learn about other areas of Computer Science and prepare for a career in CS or related field. You will complete four tasks during the semester to fulfill the CSE requirement, using a combination of three options: 1) writing biographies/reports of important figures of CS (current or past), 2) attending CS seminars, and 3) interacting with the Office of Personal and Career Development (OPCD).

The subject matter of biographies/ reports must be at least 250 words long. You may complete **at most 2** biographies.

To receive credit for a seminar, you must attend the seminar and provide a brief summary (due one week after the seminar date). Acceptable seminars include CS Colloquia, Brown Bag Meetings, and Colloquia in other departments (with prior approval).

You may receive credit for **at most 2** interactions with OPCD, which must include an informational session held by Brian Mendenhall (along with a brief summary due one week after the session). The second interaction with OPCD can be (a) a resume consultation, (b) a completed Individual Development Plan (IDP), or (c) completed LinkedIn and OPCD Handshake profiles.

All summaries and reports must be written in plain text (not Word) and emailed to the instructor.

9 Grading

Course grades are determined using the following weightings:

- 30% projects
- 5% CSE

- 10% participation
- 10% quizzes
- 25% midterms
- 20% final

Letter grades are assigned based on the following categorization:

A	93 or above	C	73–76.99
A⁻	90–92.99	C⁻	70–72.99
B⁺	87–89.99	D⁺	67–69.99
B	83–86.99	D	63–66.99
B⁻	80–82.99	D⁻	60–62.99
C⁺	77–79.99	F	below 60

10 Contacting Me

In general, email is the best way to reach me, and I'm happy to take questions over email. The easiest way to find me in person is to stop by my office during office hours, though please feel free to drop by any time. If you want to be sure to find me then you can also email ahead to schedule a time; it helps to propose a few times that work for you so that I can choose one that works for me too. Please contact me as soon as possible if you know you will miss class due to a university-sponsored activity, such as athletics.

11 Learning Assistance Center

If you have a disability that may require an accommodation for taking this course, then please contact the Learning Assistance Center (336-758-5929) within the first two weeks of the semester and bring it to my attention as appropriate.

12 Supporting Fellow Students in Distress

As members of the Wake Forest community, we have a personal responsibility to ensure that this classroom and the campus as a whole remains a healthy and safe environment for learning. Occasionally, you may come across a fellow classmate whose personal behavior concerns or worries you, either for the classmate's wellbeing or yours. If this should occur, you are encouraged to send your concern to the Wake Forest CARE Team at <http://careteam.wfu.edu/how-to-make-a-report/>. By utilizing your insights and observations, we can work together to help individuals get connected to appropriate resources and keep our community safe.

13 Emergency Preparedness Policy

In the unlikely event of a major disruption of normal university activities (such as might result from a health emergency or other disaster), a course continuation contingency plan will be enacted in order to allow completion of the course. During this time, students should continue with the reading and other assignments listed on the syllabus and monitor email, Sakai, and the WFU website for information. If students have questions or are in doubt about how to proceed, they should contact the instructor by email if available, otherwise they should contact by phone.