

CPSC170B Fundamentals of Computer Science II

Spring 2019

MWF: 2:20-4:20pm in Trexler 166

Dr. Anil M. Shende

Trexler 365A

Email: shende at roanoke dot edu

Tel: 540-375-2341

Office Hours: MTTh 1:00 -- 2:00pm;

W 9:30 -- 10:30am and

by appointment.

Syllabus

Course Synopsis

This course is part two of a two part introduction to the discipline of computer science. The course focuses on object oriented design and implementation of programs. Programming topics include inheritance, polymorphism, abstraction, lists, maps, trees, searching, sorting, and recursion.

Required Texts

1. *How to Think Like a Computer Scientist: C++ Version, First Edition*, by Allen B. Downey

Prerequisites

CPSC120 or CPSC150. Familiarity with Unix/Linux.

Intended Learning Outcomes

At the end of this course successful students will be able to:

1. design, implement (in the C++ programming language), and test algorithms to solve problems. In particular, to implement the algorithms the student will be able to
 1. use inheritance, abstraction, polymorphism, exceptions, and recursion,
 - and 2. implement linear data structures and associated algorithms.
2. explain the fundamental concepts underlying memory usage, objects, classes, and methods,
3. analyze and compare the asymptotic performance of algorithms,
4. use the Linux command line interface for running C++ programs and navigating the Linux file structure, and
5. express integers in twos complement and vice versa and be able to perform and understand computer arithmetic.

Mechanics

The course will meet in class for six hours during the week. There will be one test in class during the semester. **The final exam is scheduled for Tuesday, April 30, 2:00pm - 5:00pm.**

Make-up tests will be available *by pre-arrangement only* in case of scheduling conflicts. After the test, makeups will be available only in case of documented medical emergencies.

Besides the exams, there will be pop quizzes in class, regular homework assignments, programming projects, and a co-curricular requirement.

This course expects you to spend *at least* 12 hours of work each week inside and outside of class.

Co-curricular Requirement: The Mathematics, Computer Science and Physics department offers a series of discussions that appeal to a broad range of interests related to these fields of study. These co-curricular sessions will engage the community to think about ongoing research, novel applications and other issues that face these disciplines. Each student is required to attend at least *three* of these sessions, and turn in a short paper describing the contents of the session, and his/her *critical reflections* about the topic and content. These papers are due in class within a week of the session. A paper submitted beyond a week from the event being discussed in the paper will not be accepted.

Grading

The final grade will be computed based on the grades in the quizzes, in-class test, the final exam, homeworks and programming projects according to the following weights.

Component	Weight
Co-curricular	4%
Home works	8%
Quizzes	8%
Test	15%
Projects	45%
Final Exam	20%

The final course grade will be calculated as follows:

< 60	60-62	63-65	66-69	70-72	73-75	76-79	80-82	83-85	86-89	90-92	> 92
F	D-	D	D+	C-	C	C+	B-	B	B+	A-	A

Class Attendance and Policies

Regular attendance in class and the lab sessions is highly recommended. Regardless of attendance, students are responsible for all material covered or assigned in class.

Cell phones should be kept in your backpacks or pockets (essentially, out of sight), and turned to the silent mode throughout the duration of the class and lab periods. Please do not remove your cell phones until you are outside the classroom/lab. Similarly, during office consultations or consultations in the lab (even when it is not during regular class or lab time), your cell phones should be out of sight and in the silent mode.

If you use an electronic device such as a tablet or a laptop for note-taking or to read the textbook, the content that is open on the screen should be strictly restricted to documents and pages of relevance to the class. For example, you should not have any social media websites open in your browser window, even if it is in a tab that is not currently in focus.

Academic Integrity

Students are expected to adhere to the Academic Integrity policies of Roanoke College. All work submitted for a grade is to be strictly the work of the student unless otherwise specified by the instructor. The policies as outlined in the Academic Integrity handbook will be enforced in the course.

Graded programs are subject to the Roanoke College Academic Integrity policies. Copying a program or a portion of a program (even a single line) or reading another person's program to obtain ideas for solving a problem is plagiarism. Other examples of integrity violation include writing code for someone else, using code written by someone else, telling someone else how to solve a problem or having someone tell you how to solve a problem (and using their method). These cases apply to any work that is handed in for a grade under

the instructor's assumption that the work is your own. Unless specified otherwise by the instructor, discussion among students should be limited to general a solution to the assigned problem.