CPSC395A: Analysis of Algorithms
Syllabus

Instructor: Adewale Sekoni (Trexler 365B, sekoni@roanoke.edu)
Meeting Time and Location: MWF, 9:40 – 10:40 AM, Trexler 363
Office Hours: MWF 11:00 – 12:30 PM or by appointment, Trexler 365B
Prerequisites: CPSC 250 and MATH 131. Familiarity with C++ and Unix/Linux is assumed.
Text: *Introduction to Algorithms*, by Cormen, Leiserson, Rivest and Stein, McGraw Hill.

Intended Learning Outcomes

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

1. design algorithms for various problem types,
2. prove the correctness of algorithms,
3. analyze the efficiency of algorithms, and
4. analyze the computational complexity of problems.

Class Attendance: Regular attendance in class is highly recommended. Regardless of attendance, students are responsible for all material covered or assigned in class.

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

Homework: Homework deadlines are strict. You will receive no credit for late submissions. If you have a good reason, you can ask for an extension at least one day before the deadline. I expect your homework to be neat and legible. I will deduct points for rough work. Pictures will not be accepted. All assignments are due in class on the day of submission; they must be turned in physically and not digitally. Exceptions will be made on a case-by-case basis.

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 tests, the final exam, and home works according to the following weights.

Component	Weight
Co-curricular	2%
Home works (at least 10)	34%
Tests (2)	32% (16% each)
Final Exam	32%

Grade Scale:	93-100	A	83-86	B	73-76	C	63-66	D
	90-92	A-	80-82	B-	70-72	C-	60-62	D-
	87-89	B+	77-79	C+	67-69	D+	below 60	F

Outline

Introduction, Chapters: 3,4
Probabilistic analysis, Chapters: 5
Sorting, Chapters: 6,7,8
Dynamic programming, Chapters: 15
Greedy algorithms, Chapters: 16
Feb 25 Test 1
Amortized analysis, Chapters: 17
Single-source shortest paths, Chapters: 24
All pairs shortest paths, Chapters: 25
Maximum flow, Chapters: 26
Mar 22 Test 2
Matrix operations, Chapters: 28
Linear programming, Chapters: 29
NP-completeness, Chapters: 34
Apr 25 Final Exam

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 some 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 discussion of concepts and language details, not specific aspects of a solution to the assigned problem.