CSE5610 APM5610 Advanced Algorithms and Data Structures

From the OU catalog for APM5610 Algorithms and Complexity:

A general introduction to algorithm design and analysis, including study of the following algorithmic techniques: divide-and-conquer, greedy methods, backtracking, preconditioning and precomputation, probabilistic and approximation algorithms. Topics covered also include: the fast Fourier transform, lower bound theory, reduction and NP-completeness. Required background: a course in discrete mathematics and knowledge of data structures. Formerly APM 567. Cannot receive credit for both APM 5610 and CSE 5610.

From the OU catalog for CSI5610 Advanced Data Structures and Algorithms:

The course systematically studies advanced data structures and the design and analysis of algorithms. The main focuses are the techniques for designing algorithms using appropriate data structures, proving correctness, and analyzing the computational complexity of algorithms. Topics covered include: hash tables; data structures for combinatorial optimization; search trees; recurrence relations; divide and conquer; dynamic programming; greedy methods; advanced graph algorithms; linear programming. This course will be supplemented by algorithms selected from recent technical literature. Formerly CSE 561.

> Computer science is no more about computers than astronomy is about telescopes.
> – Edsger Dijkstra

Summary

Instructor:	Dr. Serge Kruk
Email:	kruk@oakland.edu
Office:	MSC 548
Office hours:	By appointment
Textbook:	Introduction to Algorithms by Cormen, Leiserson, Rivest and Stein (Any edition.)
Times:	MW 1730h-1917h

Course Objectives

We start where the undergraduate version of the course left off and pursue more advanced data structures and algorithms.

Material covered

- Some classical data structures. Hash tables. Red-black trees. Biray search trees. Tries.
- Applications: Image data structures. Computational Biology data structures.
- Advanced analysis of algorithms.
- Advanced heaps. Binomial, Fibonacci.
- String matching.
- Data mining algorithms.
- Sorting networks.
- Fast Fourier transforms. Discrete cosine transforms.
- Algorithms for crypto.

Grading

- Presentations: 40%
- Test 0: 20% Jan 31
- Test 1: 20% Feb 28
- Final: 20% April 25 at 1900h

Presentations

I expect graduate students to be able to read and understand material on their own, as well as presenting material to their peers. The presentations will test those skills. Each student will do two presentations. I will list, when the time comes, a series of subjects from which students will choose on a first-come first-serve basis. That is, no two students will present the same material.

You will be responsible to read and understand the material. Of course, visiting me in my office for clarifications is permitted. Then you will implement whatever algorithm or data structure you choose to present. You will test it exhaustively and you will report on the behavior. You are expected to figure out the experimental runtime and correlate with the therotical runtime. More details to follow.

The material of the presentations is deemed part of the course, hence students should expect questions on this material to be on the tests and final exams. It is therefore in the interest of each student to ask questions and clarifications during the presentations.

Corrections

After a test, I will often allow students to turn-in a corrected version of the test. The reason for this, contrary to students' belief, is that a test is a learning tool, not only an evaluation tool. A good correction may earn you back some of the points missed during the test, and will allow you to learn material that you failed to learn while preparing for the test. The rules for a test correction:

- You must do the corrections alone. No help of any kind. No discussion with peers. Find the answers in the book or in your notes.
- You must turn in a paper that states the questions, then answers them, in the order presented on the test.
- You must answer **all** questions, not only those you missed on the test. (While grading a test, I allow some slips because I understand the difficulty of answering under time constraints. Do not expect such munificence on a test correction.)
- Your answer must be written in English, be complete and be correct.
- Your corrections must be typed or written with impeccable penmanship. If your handwriting is bad, type your solutions.

I will also offer "corrections" of presentations, in the following sense. If you give a presentation that I feel is inadequate, I will list the inadequacies and allow you to present it again, in my office to regain some of the points lost.

Late Policy

Late homework assignments are **not** accepted. Missed tests or quizzes may **not** be made up. If an emergency forces you to miss a test, permission to be excused should be sought from me in advance, if possible. If granted, the final exam score will replace the score of the missed midterm. Final exams cannot be missed, cannot be taken before nor taken after the date scheduled by the registrar's office.

Academic Integrity:

Cheating is a serious academic crime. Oakland University policy requires that all suspected instances of cheating be reported to the Academic Conduct Committee for adjudication. Anyone found responsible of academic misconduct in this course may receive a course grade of 0.0, in addition to any penalty assigned by the Academic Conduct Committee. Students should read the Academic Conduct Regulations of Oakland University.

Working with others on homework is not cheating unless indicated by the instructor; handing in an assignment that has essentially been copied from someone else is cheating. Looking at someone else's work during an exam is cheating. Receiving help from someone else or consulting unauthorized material during an exam is cheating. Providing such assistance for someone else also constitutes cheating.