CS 435 Spring 2024, Saturday 9-12 Noon, @KUPF 106, by Hans Raj Nahata
Teaching Assistant - Al Shahriar

Prerequisites: CS 241 and CS 288 with a grade of C or better.

Text Book: Intro to Algorithms, 4th ed, by Cormen, Lieserson, Rivest & Stein (aka CLRS)

Instructional Mode: I plan to run the course face-to-face in the classroom as much as possible unless something, such as a quarantine or unexpected weather, requires us to go online.
Webex Link: https://njit.webex.com/meet/hnahata

Office Hours: By appointment only.
Please start your office visit by sending me an e-mail request to hnahata@njit.edu . Put "CS 435" in your subject line to help me find your message. I will schedule a face-to-face meeting, a phone call, or a video meeting.

Main Lectures (Chapter #)

- Introductions!
- Sorting - Selection, Insertion, Merge sorts (2)
- Characterizing Running Times (3)
- Divide & Conquer (4)
- Heapsort (6)
- Medians & Order Statistics (9)
- Elementary Data Structures (10),
- Hash Tables (11)
- Binary Trees (12)
- Dynamic Programming (14)
- Greedy Algorithms (15)
- Elementary Graph Algorithms (20)
- Minimum Spanning Tree (21)
- Single Source Shortest Paths (22)
- All Pairs Shortest Path (23)
- Maximum Flow (24)
- Matching Bipartite (25)
- String Matching (32)
- Complexity (34)
- Review & Where do we go from here?

Mathematical Background

- Summations (A1, A2)
- Sets (B1, B2)
- Graphs (B3)

- Trees (B4)

Homework Assignments (Not graded)
4 Programming Assignments (4 * 5) = 20 %
4 Quizzes (best 3 * 10) = 30 %
1 Midterms (1 * 20) = 20 %
Final (30) = 30 %

Lecture #	Date	Notes
1	20-Jan	First Class
2	27-Jan	
3	3-Feb	Quiz 1
4	10-Feb	Due - Assignment 1
5	17-Feb	
6	24-Feb	Quiz 2
7	2-Mar	Due - Assignment 2
8	9-Mar	Mid-term
	16-Mar	Spring Recess
9	23-Mar	Due - Assignment 3
10	30-Mar	
11	6-Apr	Quiz 3
12	13-Apr	Due - Assignment 4
13	20-Apr	Quiz 4
14	27-Apr	Last Class
	4-May	Final Exam
	11-May	Grades Due

This course aims to teach the fundamentals of algorithms and data structures. This is not a basic course, though. It covers elementary data structures such as dynamic arrays, heaps, and hash tables, and algorithmic approaches to solve classical problems (sorting, graph searching, dynamic programming). Introduction to mathematical modelling of computational problems, as well as common algorithms, algorithmic paradigms and data structures used to solve these problems. Emphasizes the relationship between algorithms and programming, and introduces basic performance measures and analysis techniques of these problems. Many chapters in this

book include sections that delve into advanced topics more suitable for a graduate course, so we will either briefly touch on them or skip them altogether.

TA will grade programming assignments, quizzes and midterms and review the solutions during recitation hours. No make-up, late or early exams.

You lose 50% of points for missing the deadline for the programming assignments. You lose 100% of points if you submit by 4 or more days. [TA will explain this with an example]

No “special” projects or assignments will be devised or permitted to boost grades.

All exams are closed notes and books. However, a simple calculator and 1 page A4 sheet with your notes are allowed during all exams and quizzes.

My advice to do well in a class like this:

Read and study everything I assign as soon as possible. Read difficult material more than once! Attend all classes. Pay close attention.

Collaboration is good for learning the subtleties of the concepts. “Copying” from any source is not good. Towards this, you are strongly encouraged to do the homework assignments in groups - discuss the nuances, counter-examples, key points in proofs etc. I suggest you do programming assignments individually.

Take notes to help you think about concepts and remember the same. Get started on assignments early. Do a little work each day so that work never piles up. Finish assignments and reading on time. Be an active learner.

Visit : https://mitpress.mit.edu/9780262046305/introduction-to-algorithms/
For selected solutions, Prof Jokes!, Python Code, and more!

Refer to the NJIT Student Code Of Conduct for more dos and don’ts.