REFERENCE ONLY
Syllabus is subject to change Students currently enrolled in this course should reference NYU Classes syllabus only**

CS6033 Design & Analysis of Algorithms I
Fall 2020
CSE Dept., Tandon School of Engineering, NYU

Prof. Yi-Jen Chiang (chiang@nyu.edu)
Office: 370 Jay St. 1103 (11F)
Tel.: 646-997-3395
Office Hours: Wed. 5–6pm

Course Syllabus

Catalog Description: Review of basic data structures and mathematical tools. Data structures: priority queues, binary search trees, balanced search trees. B-trees. Algorithm design and analysis techniques illustrated in searching and sorting: heapsort, quicksort, sorting in linear time, medians and order statistics. Design and analysis techniques: dynamic programming, greedy algorithms. Graph algorithms: elementary graph algorithms (breadth-first search, depth-first search, topological sort, connected components, strongly connected components), minimum spanning trees, shortest paths. String algorithms. Geometric algorithms. Linear programming. Brief introduction to NP-completeness.

Course Objectives:
• Learn the use of algorithmic design strategies: divide-and-conquer, dynamic programming, greedy algorithms, etc.
• Learn and be able to apply fundamental algorithms and data structures for solving other problems: searching, sorting, priority queues, hash tables, balanced search trees, etc.
• Learn to apply algorithmic design ideas to solve new problems.
• Learn to use rigorous mathematics for algorithm design and analysis: observe, state, and prove structural properties that are used for designing efficient algorithms; prove the correctness and analyze the efficiency of algorithms (running time, space complexity, etc.), in the best case/worst case, or expected case, etc.

Prerequisites: You MUST have these to take the course:
(1) CS 5403: Data Structures and Algorithms, or equivalent knowledge of fundamental data structures, and CS 6003: Foundations of Computer Science, or equivalent knowledge of discrete mathematics for computer science.
(2) Familiarity with programming beyond an "Introduction to Programming" class.

We will briefly review the material from prerequisites, but this does *NOT* replace taking them.

This course is *not* intended for students who have already taken and passed with a grade of B or above (or equivalent) a comparable algorithms course, graduate or undergraduate; such a course need not cover precisely the same topics.

Textbook: Cormen, Leiserson, Rivest, and Stein, *Introduction to Algorithms*, 3rd Edition, MIT Press, 2009; ISBN-13: 9780262033848; international version ISBN-13: 9780262533058. The textbook is known as *CLRS*.

Homeworks and Exams: There will be 7 regular (approximately weekly) homework assignments (it's a lot of work!), one Midterm Exam, and one Final Exam (**remote (synchronous), closed-book and limited notes**). No formal programming assignments will be given. If you miss an exam without a good reason (documentation is required!), you get zero for it.

Grading Distribution: 7 Homeworks: 30%; Higher Score of the Midterm and Final Exams: 40%; Lower Score of the Midterm and Final Exams: 30%.

Homework Policy: Homeworks are typically due on Thursdays by noon, right before the recitation session to present the solutions. No late homeworks are accepted. Feel free to discuss the homeworks and write the answers, in groups. **Each group can have at most 4 people.** Put the names of ALL the people participating in working on a homework, on the solution handed in (by uploading to NYU Classes). Your name should appear only on **one** homework at a time. (Do not hand in/upload multiple copies of the same homework solution.) Homeworks handed separately with similar answers will be considered **academic dishonesty**, see below.
Note: Homeworks are a **key component** to **mastering the course materials and building up your problem-solving ability**, and **prepare you for the Midterm and Final Exams.** It is **to your benefit** that you **think hard** and **try to solve the homework questions before discussing with others**. Do not let your team members do all the homework work for you —– this will ruin your chances of passing the exams.

Important Tips on How to Proceed When Solving a Homework/Exam Creative Problem:
Do not just think abstractly. Try to **construct and look at a good number of small examples** to get a better understanding and better feeling about the problem, which can often lead to key insights to a solution.
(**Note:** Even for researchers, this is usually how they approach a research problem in Algorithms.)

Recitation Sessions: The TA will give recitation sessions to present homework solutions. These sessions are in the TA's weekly office hours (same time) in the same week when the corresponding homework is due. We will also announce in NYU Classes (with an email sent out) before each recitation session.

Course Policy on Academic Dishonesty on Homeworks: Any homework submitted is expected to be the work of the students taking the course whose names are on the homework. If there is any evidence that the work is not your own (for example: copying from other students in the course, from students not taking the course, from students at another institution, from the Internet, paying someone to do the work, etc.), this will be treated as academic dishonesty. It does not matter who copied from whom. First instance would result in a **zero** for the **entire homework** in question. Second instance will result in a grade of **F** for the course. All instances will be reported to the department and to the Dean of Student Affairs.

NYU School of Engineering Policy on Academic Dishonesty
See **Student Code of Conduct:**

https://engineering.nyu.edu/sites/default/files/2018-06/code-conduct2-2-16.pdf

NYU School of Engineering Policies and Procedures on Excused Absences
Complete policy is found here: https://engineering.nyu.edu/campus-and-community/student-life/office-student-affairs/policies
with associated form:
https://engineering.nyu.edu/sites/default/files/2018-09/Excused%20Absence%20Form%20DR.pdf

Deanna Rayment, deanna.rayment@nyu.edu, is the *Coordinator of Student Advocacy, Compliance and Student Affairs* and handles excused absences. She is located in 5 MTC, LC240C and can assist you should it become necessary.

Important Dates:

No class	Mon., Sept. 7 (Labor Day)
First class	**Wed., Sept. 9**
Midterm Exam	**Mon., Oct. 19**
Midterm Grade Deadline	Tue., Nov. 3
Last class	Mon., Dec. 7
No class	Mon., Dec. 14 (Reading Day)
Final Exam	**Mon., Dec. 21**

If you have two exams at the same time, **report the conflict to your professors as soon as possible. Do not make any travel plans until the exam schedule is finalized.**

Also, please pay attention to notable dates such as Add/Drop, Withdrawal, etc. For confirmation of dates or further information, please contact Susana M. Garcia-Henriquez at sgarcia@nyu.edu.

Instruction Mode and Contingency Plan:
The instruction mode of this course is **Blended**. All lectures will be given by the professor in-person in the classroom, with a Zoom meeting turned on to enable remote students to participate synchronously (where they can interact in class). All lectures will be video-recorded, and all course materials (including the lecture videos) will be posted on-line on NYU Classes, so remote students can also participate asynchronously. The office hours of the professor and of the TA will be conducted via Zoom meetings. The TA recitation sessions (via Zoom meetings) will also be video-recorded and posted on-line.

In the event of a quarantine or stay-at-home order for which the course must be switched to fully remote, it will be announced on NYU Classes (with an email sent out), and the lectures will be given by the professor remotely via Zoom meetings. Students will then participate remotely either synchronously or asynchronously.

Tentative Course Outline: The precise order and content, especially the later parts of the course, may change. Please check for updates during the semester.

- Introduction: What's an algorithm? Why do we want to study algorithms? Termination. Correctness. Performance. How to measure performance of an algorithm? Models of computation, abstract machines. RAM. Best-, worst-, and average-case performance. Review of asymptotic notation: big-O, big-Ω, and big-Θ; little-o, and little-ω. [Ch 1, 2, 3]

- Review of basic data structures. Abstract data types (ADTs). Common ADTs: Arrays. Stacks and queues. Linked lists. Priority queues. Heaps. Heapsort. [Ch 10, 6]

- Dictionaries and ordered dictionaries. Hashing. Balanced search trees (tentatively, AVL-trees, 2-3-4 trees, and more generally (a, b)-trees; possibly red-black trees). B-trees. [Ch 11, 12, (13), 18, Handouts and Lecture Notes]

- Divide-and-conquer algorithms. Review of recurrences and how to solve them. Master's theorem. Binary search. Mergesort. Fast integer multiplication. Fast matrix multiplication. Median and order statistics. Deterministic QuickSelect. QuickSort, randomized QuickSort. Randomized QuickSelect. Comparison-based sorting lower bound. Integer sorting in linear time (counting sort), radix sort. Closest-pair problem. [Ch 4, 7, 8, 9, Section 33.4]

- Graph algorithms: elementary graph algorithms (breadth-first search, depth-first search, topological sort, strongly connected components), minimum spanning trees, shortest paths. Some graph algorithms will be presented later in the course as illustrations for different algorithm design paradigms. [Ch 22, (23, 24, 25)]

- Dynamic programming: Rod cutting. Matrix-chain multiplication. Longest common subsequence. Optimal binary search trees. Shortest path problems in graphs. Transitive closure. [Ch 15, 25]

- Greedy algorithms: Activity selection. Huffman coding. Minimum spanning trees. [Ch 16, 23, (24)]

- Randomized algorithms and geometric algorithms (not covered as separate topics).

- Undecidability and fundamentals of NP-completeness (both very briefly; one lecture).

The Moses Center Statement for Student Accessibility
If you are student with a disability who is requesting accommodations, please contact New York University's Moses Center for Student Accessibility (CSA) at 212-998-4980 or mosescsa@nyu.edu. You must be registered with CSA to receive accommodations. Information about the Moses Center can be found at https://www.nyu.edu/students/communities-and-groups/student-accessibility.html. The Moses Center is located at 726 Broadway on the 2nd and 3rd floors. **Please do this at the start of the semester.**