Syllabus: CS5800-03 Algorithms, Spring 2020 – Silicon Valley

Course Description

Presents the mathematical techniques used for the design and analysis of computer algorithms. Focuses on algorithmic design paradigms and techniques for analyzing the correctness, time, and space complexity of algorithms. Topics include asymptotic notation, recurrences, divide-and-conquer, searching and sorting, advanced data structures, hashing, greedy algorithms, dynamic programming, graph algorithms, and NP-completeness.

Prerequisite: Restricted to students in the College of Computer and Information Science and in the Network Science Program.

Course Logistics

- Term: Spring 2020, Northeastern University, Silicon Valley Campus
- Catalog number/title/section: CS5800 Algorithms, Section 3
- Course number: 38139
- Number of credits: 4
- Class period: 1/6/2020 - 4/20/2020
- Class meeting time: 6-9pm, almost every Monday except official holidays.
- Office Hours: by appointment through email, TA's to announce their office hours separately
- Place: TBD, SV Campus, 6024 Silver Creek Valley Rd, San Jose, CA 95138
- Final exam: TBA between 4/21 & 4/25 at the Silicon Valley Campus

Instructional Staff

- Instructors:
    - Anurag Bhardwaj, Ph.D., a.bhardwaj@northeastern.edu
- Teaching assistant:
    - Vishal Annamaneni, annamaneni.v@husky.neu.edu
    - Zijun Wan, wan.zi@husky.neu.edu
    - Gongzhan Xie, xie.go@husky.neu.edu

Evaluation

- Midterm exam: 25%
- Final exam: 25%
- Individual homework assignments: 40% (may include a few coding assignments)
- In-class/online participation: 10%
- This distribution may change slightly during the term (±5%).

Course Materials

- Blackboard material, class notes, online discussion, collaborative work, etc.
- Suggested book: Introduction to Algorithms, Third Edition, by Cormen, Leiserson, Rivest and Stein, from MIT Press (https://mitpress.mit.edu/books/introduction-algorithms)

Note on Online Content and Instructional Methods

This course is designated as an on-ground & traditionally-instructed course. However, we may use lecture materials developed for online offering of this course, and we may use some of our in-person class time for tutored problem solving and collaboration activities. This may be considered a flipped-classroom instructional method. Students who have questions or concerns on this instructional method are expected to contact the first instructor immediately.

Topics and Schedule

Date & time	Topic	Remarks
Jan 6, 6-9PM	**Module 1:** What is an algorithm, analysis of algorithms, asymptotic notation, correctness, lower bounds	
Jan 13, 6-9PM	**Module 2:** Review of well-known algorithms and their analysis (toy problems, basic searching and sorting algorithms) **Module 3:** Divide-and-conquer algorithm design techniques and their analyses using advanced recurrences	**HW-1 Out**
Jan 20, 6-9PM	NO CLASS	**MLK Birthday**
Jan 27, 6-9PM	**Module 4:** All about sorting and their analysis: Quicksort, Mergesort & Heapsort **Module 5a:** Basic data structures (stacks, queues, priority queues, binary search trees)	**HW-1 Due** **HW-2 Out**
Feb 3, 6-9PM	**Module 5b:** Balanced Binary Search Tree **Module 6:** Hashing and expected constant time search	
Feb 10, 6-9PM	**Module 7:** Optimization problems with their Greedy algorithms and analyses	**HW-2 Due** **HW-3 Out** **Last class before Midterm**
Feb 17, 6-9PM	NO CLASS (MIDTERM THIS WEEK)	**HW-3 Due** **President's Day** **Midterm Date TBA**
Feb 24, 6-9PM	**Module 8:** Optimization problems with their Dynamic Programming solutions and analyses	**HW-4 Out**
Mar 2, 6-9PM	NO CLASS	**Spring Break**
Mar 9, 6-9PM	**Module 9:** Graphs, elementary graph traversal algorithms and applications	**HW-4 Due** **HW-5 Out**

Mar 16, 6-9PM	**Module 10:** Algorithms to find a minimum spanning tree and a shortest path in a graph	
Mar 23, 6-9PM	**Module 11:** Introduction to complexity theory, classes of problems (P & NP), reductions	**HW-5 Due** **HW-6 Out**
Mar 30, 6-9PM	**Module 12:** NP-completeness and examples with proof	
Apr 6, 6-9PM	**Module 13:** Searching text: inverted indexes	**HW-6 Due** **HW-7 Out**
Apr 13, 6-9PM	**Module 14:** Searching large databases: B+-trees, external hashing	**Last Class**
Apr 20	NO CLASS (FINALS THIS WEEK)	**HW-7 Due** **Patriot's Day / Finals Date TBA**

Note: The list/schedule of topics may change slightly.

Miscellaneous Notes

- This course assumes that students have taken prerequisite courses as follows:
    - Discrete mathematics
    - Data structures
    - Introduction to computer programming (using some programming language, such as Java, C++, C, C#, Python, ...)
- If you haven't taken any of the above prerequisites, please contact the first instructor immediately.
- All assignments should be conducted individually, unless noted otherwise.
- Searching the Internet or other resources for solutions to assignment problems is highly discouraged.
    - However, we understand that there's not an easy way for use to enforce that.
    - To cope with this issue, whenever we suspect potential plagiarism, we will call upon any suspected students and ask to solve the problems in person, to make our own decision whether the suspected submission is truly original or plagiarized.

- It's OK to talk with your classmates or anyone regarding the meaning of any assignment problem, its clarification, and solution ideas.
    - However, it is NOT OK to see someone else's written solution (or work in progress), or to show your solution (or work in progress) to someone else. Keep your written work to yourself only and do not try to see someone else's work.
- This course doesn't assume proficiency in a specific programming language, but still assumes some proficiency in at least one programming language.
    - That is, we don't require a specific programming language as a prerequisite, but students are expected to have used some programming language to create computer programs at an undergraduate intro programming level.
    - We may use a certain specific programming language in lecture to study actual coding/implementation.
    - If that specific programming language is not what you used before, don't worry about it, and the level of the actual programming language usage in our course will be restricted to an elementary level.
- We may still have some coding assignment, and you will be allowed to use a programming language of your choice for such an assignment.
- The suggested book is optional as many books cover most of the content of the class.