ILLINOIS INSTITUTE OF TECHNOLOGY

Spring 2023

Syllabus: CS430-02/03/04
Introduction to Algorithms

Instructor: Lan Yao Email: lyao8@iit.edu

Office Hours (online): 17:00-19:00 CST Tuesday Office Location: SB206e

Class Schedule

3:15pm-4:30pm TR , Jan. 10, 2023 - May 6, 2023

- Study resources are posted prior to each lecture according to the schedule.
- Students are required to participate in each lecture either in-person or on live zoom. Failing in participation will result in a deduction from the attendance credit.
- The instructor will hold office hours at the following time in-person and online. It is an opportunity for Q&A. Please wisely use it.

Office hours: 5:00 p.m. to 7:00 p.m. (CST)
Every Tuesday from Jan. 10 to May 6.
Please use the link to join in:
https://iit-edu.zoom.us/j/86473477036?pwd=WnJ5K2VjRzUrWWRwKytJM3NGUklJZz09

Meeting ID: 864 7347 7036
Passcode: 265303

Email Guideline: upon sending an email to your instructor, you may **Include course ID in the subject field followed by the subject (question/late submission/discussion).**
e.g. "CS430-02: discussion"

Textbooks:

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein
"Introduction to Algorithms", 3rd edition, MIT Press
(You may use earlier edition with your own risk)

Current Catalog Description

Introduction to the design, behavior, and analysis of computer algorithms. Searching, sorting, statistics, optimization and combinatorial algorithms are emphasized. Worst case and average bounds on time and space usage.

Prerequisites: ((CS 330 or MATH230) and CS 331) or CS401 or CS403.

Grading:

5 Homework assignments	25%	Project	20%
		Midterm Exam	20%
Attendance	5%	Final	30%

- No late assignments accepted.
- Prospective extra credits are provided for adequate oral presentation of the

project.

- No make-up exams.
- All exams are taken in-classroom.
- Failure to attend class will result in lowering your final grade by one letter grade.
- Any submission with original statements, code, proof or other solutions from the Internet is considered as cheating. The instructor reserves the right to fail any student on his cheating behavior.

Grading Scale: 90 ≤ A ≤ 100%
80 ≤ B < 90%
70 ≤ C < 80%
60 ≤ D < 70%
E < 60%

* If final exam is lower than 60%, "A" grade may not be qualified.

CS 430 Course Outcomes

1. Use big O, omega, and theta notation to give asymptotic upper, lower, and tight bounds on time and space complexity of algorithms.
2. Determine the time complexity of simple algorithms, deduce the recurrence relations that describe the time complexity of recursively defined algorithms, and solve simple recurrence relations.
3. Design algorithms using the brute-force, greedy, dynamic programming, divide-and-conquer, branch and bound strategies.
4. Design algorithms using at least one other algorithmic strategy from the list of topics for this unit.
5. Use and implement the fundamental abstract data types -- specifically including hash tables, binary search trees, and graphs -- necessary to solve algorithmic problems efficiently.
6. Solve problems using techniques learned in the design of sequential search, binary search, O(N log N) sorting algorithms, and fundamental graph algorithms, including depth-first and breadth-first search, single-source and all-pairs shortest paths, and at least one minimum spanning tree algorithm.
7. Demonstrate the following abilities: to evaluate algorithms, to select from a range of possible options, to provide justification for that selection, and to implement the algorithm in simple programming contexts.
8. Understand the notion of complexity, especially in the context of NP-Hard problems.

CS 430 Program Outcomes

1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
3. Communicate effectively in a variety of professional contexts.
4. For undergraduates, be prepared to enter a top-ranked graduate program in Computer Science

Timeline

Weeks	Chapters	Tuesday	Thursday	Assignment
1	1,2,3	Introduction	Topic 1 Introduction to Algorithm Design: Complexity Analysis; Insertion sort, merge sort, theta notation, complexity definition.	
2	1,2,3 4.3-4.5 6,7	Topic 1 Introduction to Algorithm Design: Complexity Analysis; Insertion sort, merge sort, theta notation, complexity definition.	Topic 2 Recurrence Relations and Divide & Conquer Sorting Methods - Quicksort, Heaps and Heapsort	HW #1 (topic 1)
3	4.3-4.5 6,7	Topic 2 Recurrence Relations and Divide & Conquer Sorting Methods - Quicksort, Heaps and Heapsort	Topic 2 Recurrence Relations and Divide & Conquer Sorting Methods - Quicksort, Heaps and Heapsort	
4	4.3-4.5 6,7	Topic 2 Recurrence Relations and Divide & Conquer Sorting Methods - Quicksort, Heaps and Heapsort	Topic 2 .Recurrence Relations and Divide & Conquer Sorting Methods - Quicksort, Heaps and Heapsort	HW #2 (topic 2)
5	8.1-8.3	Topic 3 Lower Bound on Sorting	Topic 3 Lower Bound on Sorting	
6	9	Topic 4 Medians and Order Statistics	Topic 4 Medians and Order Statistics	HW #3 (Topics 3 & 4)
7	11, 12.1-12.3,13	Topic 5 Binary Search Tree and Balanced Binary Search Trees (Red-Black trees, AVL trees)	Topic 5 Binary Search Tree and Balanced Binary Search Trees (Red-Black trees, AVL trees)	
8	11, 12.1-12.3,13	Topic 5 Binary Search Tree and Balanced Binary Search Trees (Red-Black trees, AVL trees)	Topic 5 Binary Search Tree and Balanced Binary Search Trees (Red-Black trees, AVL trees)	
9	Midterm week	Midterm Exam #1	Midterm Exam #2	Midterm (Topics 1~5)
10	No Classes Spring Break			
11	14.1-14.2 15.2-15.5	Topic 6 Introduction to Dynamic Programming	Topic 6 Introduction to Dynamic Programming	Project Review
12	16.1-16.3	Topic 7 Introduction to Greedy Methods	Topic 7 Introduction to Greedy Methods	HW#4 (topic 6)

13	17.1-17.2 19	Topic 8 Amortized Analysis	Topic 8 Amortized Analysis	
14	appendix B.4, 22,23	Topic 9 Graphs, Depth First Search, Breadth First Search	Topic 9 Graphs, Depth First Search, Breadth First Search	HW#5 (topics 8& 9)
15	23,24,25	Topic 10 Shortest Paths	Topic 10 Shortest Paths	
16	23,24,25 34	Topic 10 Shortest Paths	NP and NP Completeness	
17	Final Week			

Academic Honesty: Any indication of copying lab (including previous semester work) or project work or any behavior during exams that could be considered copying or cheating will result in an immediate zero on the assignment/exam for all parties involved. In addition, a failing grade (E) in the course will be assigned, and the student's advisor/department and the Dean will be notified. Cheating on assignments is defined to be copying from someone else or providing someone else copies of your answers. **DO NOT show your assignments and programming work to anyone else!** You may answer questions on labs or project homework asked by other students.

Any behavior on the homework, projects, or exams that could be considered copying or cheating will result in an immediate zero on the assignment for all parties involved, failure in the class, and reported to academichonesty@iit.edu .

Please read the IIT Code of Academic Honesty policy carefully.

https://web.iit.edu/student- affairs/handbook/fine-print/code-academic-honesty

Reasonable accommodations will be made for students with documented disabilities. In order to receive accommodations, students must obtain a letter of accommodation from the Center for Disability Resources. The Center for Disability Resources (CDR) is located at 3424 S. State Street - 1C3-2 , 312 567.5744 or disabilities@iit.edu