CSE 340 / MATH 340: Fall 2019
Design and Analysis of Algorithms
Section 010: MW 10:45-12:00, Sinclair Auditorium
Section 011: TR 10:45-12:00, Drown Hall Rm 210

Description:

Algorithms are methods for solving information processing problems. Every algorithm must be automatable (for example, a computer can run it), and provably correct (that is, it must find the right answer for every instance of the problem it is given). We also often want it to run as fast as possible, and use as little memory as possible, even on huge instances. Fast algorithms for hard, practically important problems are among the key discoveries of computer science research. This course presents algorithms for searching, sorting, manipulating graphs and trees, scheduling tasks, finding shortest paths, matching patterns in strings, etc---and gives proofs of their correctness and analysis of their runtime and memory demands. General strategies for designing algorithms---e.g. recursion, divide-and-conquer, greediness, dynamic programming---are stressed. Limits on algorithm efficiency are explored through elementary NP-completeness theory. Prerequisites: (MATH 021 or MATH 031 or MATH 076) and CSE 140 and CSE 017

Administrative Information, Section 010:

Lecture: Monday/Wednesday, 10:45am-12:00pm
Instructor: Arielle Carr
Web Page: www.cse.lehigh.edu/~acarr (website currently under construction)
Office: PA 200B
Office hours: F 11:00am-12:30pm
Contact information:
Routine course queries: Use Piazza and make message 'private to instructors', if necessary
Email: arg318@lehigh.edu (for personal issues, otherwise please use Piazza)

Administrative Information, Section 011:

Lecture: Tuesday/Thursday, , 10:45am-12:00pm
Instructor: Hank Korth
Web Page: http://www.cse.lehigh.edu/~korth/
Office: PA 414
Office hours: TR 9:30 - 10:30 (T in PA 414, R in Rauch Lobby or Common Grounds)

Contact information:
Routine course queries: Use Piazza and make message 'private to instructors' if necessary
Email: hfk2@lehigh.edu (for personal issues, otherwise please use Piazza)

TA/Graders:

The TA and graders work for both sections. Their office hours are (or soon will be) posted on Piazza under the Staff tab.

- Samuel Chebruch, grader
- Yidong Gong, TA
- Paul Grocholske, grader
- Siraphob (Kop) Limprapaipong, grader
- Alex Koby, grader

Text:

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, third edition, 2009. The MIT Press, ISBN 978-0-262-03384-8 (hardcover), 978-0-262-53305-8 (paperback)

Class Work:

We shall be using two online sites: Lehigh's CourseSite and the externally-supported Piazza. Yes, having two sites is a bit annoying, but Piazza has a much more usable question/answer feature and great app support. CourseSite will be used **only** for submitting work (because Piazza does not support that). All other forms of communication, including the posting of homework assignments and supplemental materials, will take place on Piazza. It is your responsibility to check Piazza regularly.

The University sets up your CourseSite access automatically. For Piazza, we must arrange for an invitation to be sent to you. Once you log into Piazza, you will find instructions on its use.

Readings from the textbook are listed on the syllabus. Lectures will be related to the assigned readings, but in many cases the lecture will not be a direct repetition of the book contents. We shall assume that you have at least skimmed the required readings before lecture (except for the first class of the semester) and then, after lecture, that you will re-read the required readings carefully to fill in the details that we must necessarily omit in a lecture. Assignments will be posted on CourseSite. There will be two 4 o'clock exams (scheduled as part of the university's 4 o'clock quiz process), and a final exam. Exams will be based on the material covered in lecture and the assigned readings.

Attendance:

Attendance in class is expected. If you need to miss a class, it is your responsibility to get notes and announcements from a classmate. The two sections of the course will follow the same weekly schedule. The graders work for both sections and we shall share single Piazza and CourseSite sites for both sections.

Grading:

Your grade will be computed as follows:

- **Homework (10)** 20%. Homework is to be submitted electronically on coursesite. Hard copy will not be accepted. Students using handwritten solutions are asked to scan their work and verify that the scanned version is clearly readable prior to submitting it. An 11th optional homework will be offered based on the project. The lowest grade of these 11 homeworks will be dropped.
- **Programming Project** 5% (This project and its intermediate and final deadlines will be discussed immediately prior to the first quiz)
- **Common Hour Exams (formerly known as 4 o'clock Quizzes)** (Wed Sep 25 and Wed Oct 30, both 5:50 - 7:05 pm) 20% each. (Rooms to be announced by the university after the 10th day of classes). Both sections will take the same quiz.
- **Final Exam** 35% (Time and place to be announced by registrar later)
- TRAVEL: Please do not make winter break travel reservations until the exam schedule is announced unless you will stay at Lehigh for the entire exam period. No makeup exams will be offered to allow early departure for breaks.

Assignments:

Homework assignments will be posted and submitted on CourseSite. Typical due times will be at **10:45am on Monday** so that both sections are free to discuss the homework that day. But check Piazza for any changes. CourseSite will enforce deadlines and, thus, prevent late work from being submitted; thus, late work will not be accepted. Please verify that your work has actually been uploaded. Failure to upload successfully is not a valid excuse for late work.

Late Policy:

Students are expected to hand in all homework assignments on the day and time they are due. No credit will be given to late work unless an excuse is granted in advance. Students are advised to back up their files to the university-supported H drive, a USB drive, cloud service, and/or an external hard disk on a regular basis. Failure of one's personal machine is not an acceptable excuse for late work. There are numerous university and departmental labs available to you as an alternative if your personal machine should fail.

Computer and Cell-Phone Policy:

Cell phones must be kept silent and should not be used except for emergencies. Although computers (including tablet computers) are useful note-taking tools, they often distract from the lecture. For this reason, we ask that computers not be used during lecture. Exceptions to the rule will be made to accommodate disabilities.

Collaboration Policy:

All homework assignments are to be an individual effort. You are encouraged to discuss assignments with one another, your friends, and with the instructor and graders for the course. Indeed, this may be the most effective method of learning. You may share concepts, approaches and strategies for producing a solution. However, all work submitted in your name must be your own. You may not copy solutions in whole or in part from another student or from a Web site or other archive. Violations will be considered as cases of

academic dishonesty. If any aspect of this policy is not clear to you, don't make assumptions; consult with the instructor.

Weather, Delayed Opening, and Illness:

- In the event of a late opening of the University due to weather, we shall start class at university opening time or normal class time, whichever comes later. Students driving to class should of course make prudent choices about travel in adverse conditions.
- If you are ill, your classmates will appreciate you not sharing your flu or virus. You do not need a formal excuse for a missed lecture unless there is a scheduled quiz or exam on that date. You are responsible for getting notes from a classmate for any classes missed.

ABET-format Syllabus

CSE 340 Design and Analysis of Algorithms

Catalog description: Algorithms for searching, sorting, manipulating graphs and trees, finding shortest paths and minimum spanning trees, scheduling tasks, etc.: proofs of their correctness and analysis of their asymptotic runtime and memory demands. Designing algorithms: recursion, divide-and-conquer, greediness, dynamic programming. Limits on algorithm efficiency using elementary NP-completeness theory. Credit will not be given for both CSE 340 (MATH 340) and CSE 441(MATH 441). Prerequisites as noted below.

Prerequisites: (MATH 021 or MATH 031 or MATH 076) and CSE 140 and CSE 017

Credit hours: 3

Class/laboratory schedule: This class has two 75-minute meetings a week. No laboratories.

Instructor: Arielle Carr and Hank Korth

Required course for all majors in Computer Science and Business and Computer Science

Prerequisites: **:** (MATH 022 or MATH 032) and (CSE 261 or MATH 261)

Textbooks/required materials: Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, third edition, 2009. The MIT Press, ISBN 978-0-262-03384-8 (hardcover), 978-0-262-53305-8 (paperback)

Course objectives:
Upon completing this course, students will: (*denotes primary objectives)

- Design new algorithms, prove them correct, and analyze their asymptotic and absolute runtime and memory demands.
- Locate in the literature provably correct and – to the extent possible – efficient algorithms to solve a wide range of computational problems.
- Understand how to judge whether or not a problem is likely to possess an efficient algorithm
- Have a grasp of basic engineering issues arising in the implementation, adaptation, and application of algorithms

Relationship between course outcomes and program outcomes:

CSE 340 substantially supports the following program outcomes:

- An ability to apply knowledge of computing and mathematics appropriate to the discipline.
- An ability to analyze a problem and identify and define the computing requirements appropriate to its solution.
- An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the trade-offs involved in design choices.

CSE 340 provides modest support to the following program outcomes:

- An ability to design, implement, and evaluate a computer-based system.
- Recognition of the need for, and an ability to engage in, continuing professional development.
- An ability to use current techniques, skills, tools necessary for computing practices.
- An ability to apply design and development principles in the construction of software systems of varying complexity.

Prerequisites by Topic

- (MATH 021 or MATH 031 or MATH 076) and CSE 140 and CSE 017
- Sets: basic operations, Cartesian products, relations, functions, graphs, trees
- Summations: formulas & properties, bounds
- Counting: strings, permutations, combinations, Binomial coefficients
- Matrices: basic operations, inverse, rank, determinant
- Derivative, integral, differentiation, integration

- Proof techniques: direct, by contradiction, by contrapositive, by induction

Major Topics Covered in the Course

- Problems & Algorithms :definitions, properties
- Algorithm Performance Analysis:
- Asymptotic Growth of Functions, O(.), etc.
- Iterative vs Recursive algorithms;; Divide and Conquer, Insertion Sort & MergeSort
- Recurrences & their solution: by Substitution;; the Master Theroem
- Sorting:Heapsort, Priority Queues;; Quicksort;;
- Lower Bounds, Sorting in Linear Time
- Data Structures: Stacks, queues, lists, graphs, trees, heaps
- Dictionaries;; Hashing;; Disjoint-Sets Union/Find
- Graph Algorithms: Bread-first & Depth-first Search;; Topological Sort
- Shortest Paths: Bellman-Ford, Dijkstra
- Greedy Algorithms: Huffman Trees, Minimum Spanning Trees (Kruskal)
- Dynamic Programming: Matrix-chain multiplication;; All-Pairs Shortest Paths;; Floyd-Warshall
- P,NP,NP-completeness:
    - The Complexity Classes P & NP
    - Polynomial-time Reductions among Problems
    - NP-complete Problems, P=NP?
    - Approximation Algorithms for NP-complete problems
- Miscellaneous (as time permits):
    - String Matching: Knuth-Morris-Pratt, Rabin-Karp
    - Computational Geometry: Convex Hull, Voronoi diagrams
    - Polynomials & the Fast Fourier Transform

Assessment Plan for the Course

There are weekly written homework assignments, to be handed in electronically (students who choose to use hardcopy are asked to scan their papers).

Exams: There are two (2) 75-minute exams and a final 3-hour exam: all are closed-book, written exams.Grading: 75% Exams (20% 1st 75-min exam, 20% 2nd exam, 35% final exam); 20% Homework; 5% Programming Project

University-required syllabus statements:

- If you have a disability for which you are or may be requesting accommodations, please contact your professor and the Office of Academic Services, Room 212, University Center or call (610-758-4152) as early as possible in the semester. University policy states that you must notify your professor seven (7) days prior to the exam.

- Lehigh University endorses The Principles of Our Equitable Community (http://www4.lehigh.edu/diversity/principles). We expect each member of this class to acknowledge and practice these Principles. Respect for each other and for differing viewpoints is a vital component of the learning environment inside and outside the classroom.

Topics	Date*	Reading	Homework and notes
Course overview, GCD example, Correctness proofs illustrated by insertion sort, Review of mathematical induction	M 8/26 T 8/27	Ch 1, 2.1 - 2.2	H1 distributed (induction and correctness proofs, divide and conquer)
Divide-and-conquer, illustrated by merge-sort	W 8/28 R 8/29	2.3	
Growth of functions	M 9/2 T 9/3	Ch 3	H1 due at 10:45am M H2 distributed (growth of functions and recurrences)
Divide and Conquer, Recurrences	W9/4 R 9/5	4.1 – 4.2	
Solving recurrences, Master Theorem	M 9/9 T 9/10	4.3 - 4.5	H2 due at 10:45am M H3 distributed (recurrences, heaps and applications)
Heapsort and priority queues	W 9/11 R 9/12	Part II intro, Ch 6	
Quicksort	M 9/16 T 9/17	Ch 7	H3 due at 10:45am M H4 distributed (quicksort and linear sorts)
Sorting in linear time	W 9/18 R 9/19	8.1-8.4	
Red-black trees	M 9/23 T 9/24	Ch 13.1 - 13.3	H4 due at 10:45 M no HW due to quiz; practice RBT question will be posted for study)
QUIZ 1 on W 9/25: both sections have the same quiz from 5:50 - 7:05 pm in Neville 1	W 9/25		Hand out sorting project (due date TBA) W class will be an optional group office hour hosted by both instructors R class period will not meet
Dynamic Programming	M 9/30 T 10/1	Ch 15 intro, 15.1	H5 distributed (dynamic and greedy)
Greedy algorithms	W 10/2 R 10/3	16.1 – 16.3 (exclude correctness in 16.3)	
Disjoint sets	M 10/7 T 10/8	21.1 - 21.3	H5 due at 10:45am M H6 distributed (union/find, {D,B}FS)
Breadth-first search, Depth-first search, Topological sort	W 10/9 R 10/10	22.1 - 22.2, 22.3 - 22.4	
Pacing Break			no class M 10/14 nor T 10/15
Minimum spanning trees, Kruskal's algorithms	W 10/16 R 10/17	Ch 23 excluding Prim's algorithm	H6 due at 10:45am W H7 distributed (MST, shortest paths, due 10/28)
Shortest paths, Bellman-Ford and Dijkstra's algorithms	M 10/21 T 10/22	Ch 24 intro, 24.1 - 24.3	
All-pairs shortest paths, Warshall's algorithm	W 10/23 R 10/24	25.1 - 25.2	
P and NP	M 10/28 T 10/29	Ch 34	H7 due at 10:45am M NOTE:. Quiz 2 covers only definitions of P, NP, and

			NP-complete but not NP completeness reductions
QUIZ 2 on W 10/30: both sections have the same quiz from 5:50 - 7:05 pm in Neville 1	W 10/30		W class will be an optional group office hour hosted by both instructors R class period will not meet
Proving NP-completeness	M 11/4 T 11/5	Ch 34	H8 distributed (serious NPC stuff)
More NP-complete problems	W 11/6 R 11/7	Ch 34	
Example NP-completeness proofs	M 11/11 T 11/12	Ch 34	H8 due at 10:45am M H9 distributed (more NPC and comp geom)
Computational Geometry	W 11/13 R 11/14	Ch 33	
Multithreaded algorithms	M 11/18 T 11/19	27.1, 27.2	H9 due at 10:45am M H10 distributed (threads and Bentley question)
Algorithm Engineering, John Bentley, guest lecturer	W 11/20 R 11/21		dates tentative, pending confirmation with Jon, who has already agreed to come two consecutive days
Approximation algorithms	M 11/25 T 11/26	35.1-34.3	H10 due at 10:45am M H11 distributed (Approx. algs, NC)
No Class	W 11/27 R 11/28		Thanksgiving Break
Parallel polylogarithmic-time algorithms and Nick's Class (NC)	M 12/2 T 12/3	not in book	
Summary: final review, and course evaluation	W 12/4 R 12/5		H11 due at 10:45 W
final	Time, Room TBA		No makeups for travel-related issues, wait for exam date before booking tickets(or book for after the exam period)

* Note that Monday/Wednesday dates pertain to Section 010, and Tuesday/Thursday dates pertain to Section 011.