ST/ALGO:DESGN, ANALYSIS & IMPL (CS_419_X001_F2019)

Jump to today

This course is a Masters level introduction to algorithm design, complexity analysis and implementation.

References:

- [CLRS] Introduction to Algorithms, 3rd or 2nd edi. (http://mitpress.mit.edu/algorithms/) (default reference).
- (http://www.aw-bc.com/info/kleinberg/) [KT] Kleinberg and Tardos, Algorithm Design (DP chapter online, all slides online (http://www.cs.princeton.edu/~wayne/kleinberg-tardos/))
- [DPV] Dasgupta, Papadimitriou, and Vazirani (DPV). Algorithms (http://cseweb.ucsd.edu/~dasgupta/book/) (full text online via berkeley)
- [E] Jeff Erickson. Algorithms, Etc. (http://jeffe.cs.illinois.edu/teaching/algorithms/) (full text online)
- How to Think Like a Computer Scientist: Learning with Python (http://www.openbookproject.net/thinkcs/python/english2e/) (full text online)
- Introduction to Mathematical arguments (https://math.berkeley.edu/~hutching/teach/proofs.pdf) by Michael Hutchings
- Algorithms for approximate string matching (https://www.cs.helsinki.fi/u/ukkonen/InfCont85.PDF) by Esko Ukkonen
- Python Tutorial by Liang Huang
- Introductory materials

Prerequisites:

The CS 519 version of the course is designed for Masters students in Computer Science. Seniors can take CS 419 with almost the same content. I only assume the knowledge of Data Structure and programming experience in a high level language. All programming will be in Python. If you do not know Python already, you will be expected to learn on your own.

Assessment:

- Midterm: 20%
- Weekly Homeworks and Programming: 30% (**no late submission is accepted).**
- Quizzes: 20%
- Final: 30%

Office Hours:

Instructor Office Hours: T,F 12-1 PM, KEC 3069.

Class room: Bexell 102

GTA: Yilin Yang's Office hours: , KEC Atrium. MW, 3-4 PM.

Course contents

- Big-O notation, worst case and average case complexity analysis
- Divide and conquer, Binary search, Master equation
- Sorting algorithms including selection sort, quick sort, merge sort, heap sort, etc.
- Graph algorithms (BFS/DFS, Prim, Kruskal, Dijkstra's algorithm)
- Dynamic programming (knapsack problems, edit distance, traveling sales person, etc.)
- Network Flow problems and algorithms
- NP-hard and NP-complete problems (satisfiability, traveling sales person, etc.)

Measurable Student Learning Outcomes

After this class the students will be able to:

- Describe classic algorithms for standard problems such as sorting and searching
- Analyze the complexity of algorithms
- Develop algorithms for new problems based on standard algorithmic schemas such as dynamic programming and divide-and-conquer
- Implement and test algorithms in a high level programming language
- Give simple proofs of NP-completeness

Students with disabilities:

Students with documented disabilities who may need accommodations, who have any emergency medical information the instructor should know, or who need special arrangements in the event of evacuation, should register with **the Office of Services for Students with Disabilities (http://ds.oregonstate.edu/)** and discuss their needs with the instructor in the first week of classes.

Collaborations and Student Conduct:

Each student is responsible for their own homework.You should not use any web sources for answering the homework questions unless explicitly instructed to do so. You might talk to other students orally about solution approaches, but any communication that requires written content is strictly prohibited and will automatically result in a zero grade for the homework and academic dishonesty report. I expect professional and respectful conduct towards the instructor and other students during and after the class. Please read the **university's student conduct code (http://studentlife.oregonstate.edu/sites/studentlife.oregonstate.edu/files/student_conduct_code_1.pdf)** for more details.

Course summary:

Date	Details

Date	Details	
Thu, 3 Oct 2019	Homework #1 (https://oregonstate.instructure.com/courses/1736993/assignments/7705191)	due by 23:59
Thu, 10 Oct 2019	Homework #2 (https://oregonstate.instructure.com/courses/1736993/assignments/7710693)	due by 23:59
	Quiz1 (https://oregonstate.instructure.com/courses/1736993/assignments/7725575)	due by 23:59
Thu, 17 Oct 2019	Homework #3 (https://oregonstate.instructure.com/courses/1736993/assignments/7715436)	due by 23:59
Thu, 24 Oct 2019	Homework #4 (https://oregonstate.instructure.com/courses/1736993/assignments/7718599)	due by 23:59
	Quiz 2 (https://oregonstate.instructure.com/courses/1736993/assignments/7725576)	due by 23:59
Thu, 31 Oct 2019	Midterm (https://oregonstate.instructure.com/courses/1736993/assignments/7711467)	due by 8:00
Tue, 5 Nov 2019	Homework #5 (https://oregonstate.instructure.com/courses/1736993/assignments/7720230)	due by 23:59
Tue, 12 Nov 2019	homework #6 (https://oregonstate.instructure.com/courses/1736993/assignments/7725948)	due by 23:59
Tue, 19 Nov 2019	Quiz 3 (https://oregonstate.instructure.com/courses/1736993/assignments/7725624)	due by 8:00
	Homework #7 (https://oregonstate.instructure.com/courses/1736993/assignments/7716898)	due by 23:59
Thu, 5 Dec 2019	Quiz 4 (https://oregonstate.instructure.com/courses/1736993/assignments/7725625)	due by 8:00
Mon, 9 Dec 2019	Final exam (https://oregonstate.instructure.com/courses/1736993/assignments/7726951)	due by 23:59