KAU CS 681 • Tentative (2024)

Selected Topics on Algorithmic Design

Dr. Muhammad Al-Hashimi

Credits 3

Prerequisite None • See *Undergrad Background*

Homepage www.hashimi.ws/cs681

Email cs681@hashimi.ws

Office Room 139 (B31) ✆ Ext. TBA

Graduate course on the design and complexity of computer-based solutions. It aims to help equip students with practical skills to pursue advanced work. We introduce students to the main tools and methods used to analyze algorithms. They explore algorithmic design techniques and other related topics. Case studies and short investigations guide their learning. The course relies on programming to help develop an experience-based understanding of core issues.

Selections Topics

- ✎ **Analysis Foundation** Formalizing efficiency, nonrecursive and recursive iteration, mathematical induction, sequences, sums, recurrences and generating functions, Master theorem, binary heaps and integer priority queues.
- ✎ **Intractability** NP-completeness concepts and the basic 21 NP-Complete problems, branch-and-bound/backtracking techniques.
- ✎ **Case Study Algorithms** Quickselect, mergesort, quicksort, matrix multiplication, Prim's minimum spanning tree, single-source-shortest-path (SSSP), the knapsack problem.

Resources Check announcement conversations in course group for schedules, and material.

Undergrad Background Anany Levitin, *Introduction to the Design and Analysis of Algorithms*, Pearson 3rd edition.

Assessment Short case-study investigations, presentations, and group discussions assess work quality and grasp of course material.

30% Project 1
`(CLO/Synth) Develop practical skills with standard methods and tools used to analyze algorithms`

30% Project 2
`(CLO/Anal) Examine various characteristics of interest to the design and performance of algorithms`

40% Final project/presentation
`(CLO/Appl) Demonstrate a satisfactory grasp of covered concepts, techniques, and main results`

References

1. Cormen, Leiserson, Rivest, and Stein, *Introduction to Algorithms*, The MIT Press 4th edition (2022).
2. Garey and Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman 1979 (1983 printing or later).
3. Graham, Knuth, and Patashnik, *Concrete Mathematics: A Foundation for Computer Science*, Addison-Wesley 1st or 2nd editions (a minor difference).
4. Bentley, Haken, and Saxe, *A General Method for Solving Divide-and- Conquer Recurrences*, SIGACT News (1980).
DOI 10.1145/1008861.1008865
5. Karp 1972, *Reducibility among Combinatorial Problems*, Complexity of Computer Computations (The IBM Research Symposia Series).
DOI 10.1007/978-1-4684-2001-2_9

Rev 1.1 (February 29, 2024)