Algorithms and Data Structures

(ENEE 351)

Spring 2017

Prof. Papamanthou

Course Goals: This course teaches fundamental concepts in computer engineering, including topics in discrete math, data structures and algorithms. The course will also include a hands-on programming component. This course will provide students with the tools to design modular, time and space-efficient algorithms for real-world problems.

Credits: 4

Course Prerequisites: ENEE 150 and ENEE 244

Topic Prerequisites: C programming

Main Textbook: Introduction to Algorithms (third edition), by Cormen, Leiserson, Rivest and Stein

Secondary Textbook: Algorithm Design and Applications, by Goodrich and Tamassia

Tentative Topics:

1. Fundamental Concepts: Mathematical induction; Recursion; Combinatorics (counting); Discrete probability; Recurrence relations; Concepts and tools for analyzing algorithmic performance such as: work-depth, asymptotic notation, worst case, randomized and probabilistic complexity.
2. Core Data Structures: Stacks; Queues; Graphs; Trees; B-trees; Binary-search trees; Hash tables; Dictionaries; Heaps.
3. Sorting Algorithms and their Analysis: Sorting: Insertion sort; Merge sort; Quicksort; Radix sort;
4. Graph algorithms: Depth-first search; Breadth-first search; Shortest path; Minimum spanning tree; Topological sort;Fast Fourier Transform (FFT).
5. Algorithmic approaches: brute-force algorithms; greedy algorithms; divide-and-conquer; dynamic programming.
6. Advanced Topics: Advanced (Tree) Data Structures, Max-flow/Min-cut, NP-Completeness, Parallel Computing

Tentative Topics for Programming Projects:

1. Sorting (Radix Sort)
2. Dynamic Programming
3. Hash tables
4. Graph algorithms
5. FFT

Grading Method:

5 Homeworks 30%
4 Programming projects 40%
1 Midterm 10%
1 Final 20%