New York University Tandon School of Engineering

EL-GY 9343: Special Topic Course in Telecom Networks

Data Structure and Algorithm

Fall 2016

Instructor: Yong Liu

Course Prerequisites

1) Basic knowledge of fundamental data structures.
2) Basic programming language skills, such as C/C++, Java, Python

If you are not sure you have the proper preparation, you must talk to me before taking this course. Additionally, you should not take this course if you have taken a similar course, such as CS6033 with a 'B' or better grade.

Course Description

- Review of basic data structures and mathematical tools.
- Data structures: priority queues, binary search trees, balanced search trees, and B-trees. Algorithm design and analysis techniques illustrated in searching and sorting: heapsort, quicksort, sorting in linear time, medians and order statistics.
- Design and analysis techniques: dynamic programming, greedy algorithms.
- Graph algorithms: elementary graph algorithms (breadth-first search, depth-first search, topological sort, connected components, strongly connected components), minimum spanning trees, shortest paths.
- Brief introduction of complexity and NP-completeness.

Textbook

Cormen, Leiserson, Rivest, and Stein,
Introduction to Algorithms, 3rd Edition, MIT Press, 2009;
ISBN-13: 9780262033848;
The paperback international version has ISBN-13 9780262533058. It is known as CLRS.
We have free access to CLRS on books24x7 (on the library web site http:// library.poly.edu, go to Find Databases, then letter B, then books24x7).

Course Work and Grading

Your final grade will be determined roughly as follows:

Homework	10%
Midterm	40%
Final	50%

Tentative Schedule

- Week1: Introduction to algorithm: correctness and performance. Best-, worst-, and average-case performance. Asymptotic notation: big-O, big-Ω, and big-Θ; little-o, and little-ω.
- Week 2: Recurrence and solving methods: iteration, substitution and master
- Week 3: Divide and conquer algorithms, introduction to sorting: insertion sort, bubble sort
- Week 4: Sorting: MergeSort, Heap and HeapSort,
- Week 5: Sorting: quick sort, randomized algorithms, lower bound for comparison sorting, counting sort and radix sort
- Week 6: Order statistics and selection problem, midterm review
- Week 7: Midterm
- Week 8: Hashing and Universal Hashing, Binary search trees
- Week 9: Graph basics, Breath-First Search, Depth-First Search
- Week 10: Directed-acyclic graph and topological ordering, strongly connected components,
- Week 11: Intro to dynamic programming, greedy algorithm
- Week 12: Greedy algorithm, Huffman coding, Minimum Spanning Tree
- Week 13: Single-source shortest paths, all-pairs shortest paths
- Week 14: NP-Completeness and Final Review
- Week 15: Final