June

Using this document

The document specifies topics with section references to the course text book - “Algorithm Design Manual” by Steven Skiena. The practice problem set will contain problems based on the topics specified. For each of the topics, there are ample problems to try on CodeChef, CodeForces and other websites. For effectiveness, use the problem-tag feature to practice particular topics. Going through all the resources **is not necessary but recommended.**

From The Algorithm Design Manual by Steven Skienna

5 Graph Traversal
5.1 Flavors of Graphs
5.2 Data Structures for Graphs
5.5 Traversing a Graph
5.6 Breadth-First Search
5.7 Applications of Breadth-First Search
5.8 Depth-First Search
5.9 Applications of Depth-First Search
5.10 Depth-First Search on Directed Graphs

6 Weighted Graph Algorithms
6.1 Minimum Spanning Trees
6.3 Shortest Paths

The above includes the following topics :
Applications of BFS - connected components, two-coloring/bipartiteness testing
Applications of DFS - cycle finding, articulation vertices, strongly connected components, topological sorting
Weighted graph algorithms - Prim, Kruskal, Dijkstra, Floyd-Warshall

You can also look for videos and tutorials on these topics, if you prefer learning it from other/multiple resources.

Segment Tree

You need to study only the basics. Segment trees offer the following functionalities-

1. Range querying (eg. what is the minimum in the array between index 12 to 43)
2. Point updates (eg. multiply the array element at 15th index by 7)
3. Range updates (eg. multiply the array elements between 15th and 72nd index by 2)

Range queries work in O(logn) time. Point updates also work in O(logn). Range update can be done in 2 ways, doing point update for every element in the range. This is O(nlogn). The other option is called lazy propagation.

Lazy propagation and Range updates aren't part of this month's practice set.

1.Algosaurus -
Start here- http://algosaur.us/segment-tree/
Stop here- http://algosaur.us/segment-tree/#lazy (from here on it's lazy propagation)

2.Codeforces -
http://codeforces.com/blog/entry/15890 (stop at the Lazy propagation heading)

3.HackerEarth -
https://www.hackerearth.com/practice/data-structures/advanced-data-structures/segment-trees/tutorial/