Competitive Programing syllabus

By Aviram Magen

Lecture 1 - Basic c++ and competitive programming intro (3 hours)
We will learn basic c++ including STL and get familiar with ICPC and competitive programming.

Lecture 2 - Dynamic Programming: (3 hours)

1. Subset Sum
2. The Longest Increasing Subsequence
3. Longest Common Subsequence + Reduce the Space to One Dimension
4. Max Subarray Sum

Lecture 3 - Graphs 1: (3 hours)

1. Breadth First Search
2. Depth First Search
    a. Topological Sorting using DFS
    b. Strongly Connected Components using DFS (Tarjan's algorithm)
    c. Find bridges and cut points in graph using DFS

Lecture 4 - Dynamic Programming combined with Graphs (3 hours)

1. Given a directed acyclic graph, how many paths are there from u to v- using topological Sorting and DP.
2. Given a directed acyclic graph, find Longest paths are there from u to v- using topological Sorting and DP.
3. Find diameter of a tree - using DP and DFS
4. Maximum sum of nodes in tree such that no two are adjacent - DFS with DP

Lecture 5 - Graphs 2: (3 hours)

1. Shortest Paths
    a. Bellman-Ford
    b. Floyd-Warshall
    c. Dijkstra
2. The Minimum Spanning Tree

Lecture 6 - Graphs 3: (3 hours)

1. Bipartite Matching
    a. Check whether a given graph is Bipartite or not using BFS or DFS.
    b. equivalence between the maximum matching problem and the minimum vertex cover problem in bipartite graphs.(konig theorem)
2. Maximum Flows

3. Minimum Cost Maximum Flow

Lecture 7 - Numbers and Mathematics: (3 hours)

1. Fermat's Little Theorem
2. Euler's Theorem
3. GCD
4. Sieve of Eratosthenes - Generating the Prime Table
5. Repeated Squaring- To compute ab when b is big

Lecture 8 - Computational Geometry: (3 hours)

1. Points, Lines and Angles
2. Convex Hull
3. Line Sweep

Lecture 9 - Binary And Ternary Search (3 hours)

Lecture 10 - Segment Tree (3 hours)

Lecture 11 - Union Find and Sliding Window (3 hours)

Lecture 12 - String processing: (3 hours)

1. Kmp
2. Trie

Lecture 13- ICPC internal competition (6 hours)