

# ***HSTU Online Programming Campaign Syllabus***

Class 01 : Ad Hoc Techniques (Greedy, Prefix sums, contribution etc.)

Class 02 : Sorting, Searching & Usage of STL

Class 03 : Dynamic Programming

Class 04 : Graph

Class 05 : Number Theory

Class 06 : Combinatorics & Other math topics

Class 07 : Data Structures

Class 08 : Advance Dynamic Programming

Class 09 : Advance Graph

Class 10 : Constructive and Interactive Problems

## **1. Adhoc & Greedy:**

- a. Prefix sum technique
- b. Difference array
- c. Exchange Argument
- d. Contribution technique
- e. Problems with Bit manipulation
- f. Iterating over permutations.

## **2. Sorting, Searching & Usage of STL:**

- a. Binary Search
- b. Ternary Search
- c. STL (upper\_bound, unique, sorting etc.)
- d. Usage of Set, Map
- e. Usage of Stack and Queue
- f. Deque, Priority Queue
- g. Ordered Set

### **3. Number Theory:**

- a. Modular Arithmetic (Division/Expo/Inverse)
- b. Factorization
- c. Sieve
- d. Canonical form and Multiplicative Functions
- e. Euler Totient Function
- f. Exclusion with Sieve
- g. Powers of prime in factorial/LCM

### **4. Combinatorics & More Math:**

- a. Calculating Binomial Coefficients
- b. Stars and Bars Variations
- c. Basic Inclusion Exclusion
- d. Probabilities and Expected value
- e. Inclusion-Exclusion
- f. Basic Game Theory

### **5. Graph:**

- a. DFS ( DFS Tree, types of edges, traversals and start end time)
- b. BFS (Shortest Path, Level Graph)
- c. Dijkstra (Shortest Path Tree)
- d. Floyd Warshall
- e. Graph Modeling
- f. Bellman Ford

### **6. Dynamic Programming:**

- a. Basic 1D DP
- b. Multiple States (2D/LCS,LPS etc)
- c. Recursive DP (States, Transition, Memoization)
- d. Iterative DP (Evaluation Order)
- e. State Reduction
- f. Optimising Transitions
- g. Memory Optimisation

## **7. Data Structures:**

- a. Sliding Window
- b. Two Pointers
- c. Basic Segment Tree
- d. Sparse Table
- e. DSU
- f. Coordinate Compression
- g. Offline queries
- h. Other DS Techniques

## **8. Advance Graph:**

- a. Tree Properties
- b. LCA queries
- c. Flattening a Tree
- d. Minimum Spanning Tree
- e. Topsort + Condensation Graphs
- f. Euler Tours

## **9. Advance Dynamic Programming:**

- a. Coin Change/Knapsack Variations
- b. Interval DP
- c. Bitmask DP
- d. Optimising DP with Data Structures
- e. DP on DAG

## **10. Interactive, Constructive Problems & Geometry:**

- a. Constructive Problems (DFS Tree, Working Backwards)
- b. Interactive Problem (Search Space, Complex Queries)
- c. Basics of Geo (Points, Vectors, Area etc)