SNDT acm -W STUDENT CHAPTER acm MPSTME acm MHSSCOE

ACM MPSTME

PRESENTS

Your Guidebook to the World of Competitive Coding

Dear Participant,

We welcome you as you begin your exhilarating journey into the vast domain of competitive programming. We are happy to have you along for the ride of Semicode 2025, which we ensure will be a life-changing experience for you! However, we know that it can be daunting to enter a competition when you're just starting off - so don't worry, we've got you covered. This guidebook is designed to help you map your path and do things the ACM way, where you can take new wings in the technological landscape.

We have highlighted most of the important concepts that you'll find in data structures and algorithms in this guidebook. However, we want you to remember that studying this resource is not the only thing that will help you win Semicode. The key lies in consistency and practice. We have full faith in you, all the very best!

Phase 1: Basics of Programming & Complexity Analysis

Topics Covered:

- **Core Concepts:** Input/Output, Data Types, Operators, Loops, Conditionals
- **Complexity Analysis:** Time Complexity & Space Complexity (Big O Notation)

Start by nailing the basics. Understanding how to handle inputs, outputs, and the different types of data is your first step. Also, get comfortable with time and space complexities—trust me, you'll thank yourself later.

Phase 2: Basic Data Structures

Topics Covered:

- **Arrays & Strings**
- **Sorting:** Bubble, Selection, Insertion, Merge, Quick
- **Searching:** Linear & Binary
- **Two Pointers Technique**

This is all about building a strong foundation. Arrays and strings are everywhere in coding problems. Master these, and you'll be solving a lot of questions with ease!

Phase 3: Intermediate Data Structures

Topics Covered:

- **Stacks & Queues:** Implementation & Applications
- **Linked Lists:** Singly, Doubly, Circular
- **Hashing:** Hash Maps, Collision Handling
- **Recursion & Backtracking**

By now, you'll start dealing with more complex problems. Linked lists and hashing are super common in interviews, so give them some extra love.

Phase 4: Advanced Data Structures

Topics Covered:

- **Trees:** Binary Trees, BSTs, Tree Traversals
- **Graphs:** BFS, DFS, Dijkstra, Floyd-Warshall
- **Segment Trees & Fenwick Trees**
- **Disjoint Set Union (DSU)**

This is where things get real! Trees and graphs can be a bit intimidating, but once you crack them, you'll feel like a god.

Phase 5: Dynamic Programming & Advanced Algorithms

Topics Covered:

- **Dynamic Programming:** Knapsack, LIS, LCS, Matrix Chain
- **Greedy Algorithms**
- **Bit Manipulation**
- **String Algorithms:** KMP, Rabin-Karp, Z-Algorithm

Dynamic programming is like the boss level of DSA. Start simple and work your way up. The key is to practice—there's no shortcut here!

Phase 6: Competitive Programming & Contest Practice

Tasks to Focus On:

- Participate in **CodeChef Long Challenges & Starters**
- Try **CodeChef Div3, Div2 & Div1 contests**
- Solve **AtCoder Beginner Contests (ABC)**
- Take the **CodeChef DSA Certification Exam**

The best way to get better at CP is by actually competing. Don’t worry about the ratings initially—just focus on learning.

📚 Recommended Resources

1. NeetCode.io 🛠️

- A goldmine for interview prep with categorized problems like **"Blind 75"**, **"NeetCode 150"**, and more.
- Patterns-based learning (Sliding Window, Graphs, Dynamic Programming).
- Video explanations + solutions in Python, Java, C++, and JavaScript.

🔗 Visit NeetCode.io

---

2. CP-Algorithms

- In-depth explanations of algorithms and data structures.
- Regularly updated with new articles and improvements.

🔗 Explore CP-Algorithms

---

3. Practice Platforms

- **Codeforces:** Regular contests, problem archives, and a rating system.
  🔗 Codeforces
- **CodeChef:** Long challenges, Cook-Offs, and a vast problem set.
  🔗 CodeChef
- **AtCoder:** Known for its clean problem statements and a mix of difficulty levels.
  🔗 AtCoder
- **TLE Eliminators:** Helps find Codeforces problems by difficulty.
  🔗 TLE Eliminators

---

4. YouTube Channels for Learning

- **Apna College:** Data Structures and Algorithms explained simply.
  🔗 Watch here
- **Take U Forward:** Deep dives into DSA and interview prep.
  🔗 Check it out
- **CodeHelp:** Friendly tutorials for coding beginners.
  🔗 Visit CodeHelp