Competitive Programming Fundae

1. Recommendation: Do CP on Linux/Mac. For windows, it is difficult to use bits/stdc++.h header, Google or ask batchmates to resolve this issue.
2. Recommended IDE: VS Code.
3. The code should be clean and well indented.
4. Languages generally used: C++, Python, Java (Good online community support).
5. Take care of the time complexities of functions used in your code.
6. C++: Must know ALL PDS concepts.
7. Learn STL (Standard Template Library): Stacks, Queues, Vector, Set, Map and their functions (Geeksforgeeks).
8. Data Structures: Array, Strings, Trees.
9. https://en.cppreference.com/w/: Details about all functions and data structures.
10. No Need to learn everything in one go. Learn on the go while solving problems.
11. Algorithms to learn:
    - Greedy
        - Priority Queue
        - Set
    - Dynamic Programming
        - Vector
        - Map
    - Divide & Conquer
        - Binary Search
        - Recursive Functions
    - Sorting Algorithms
        - Insertion Sort
        - Bubble Sort
        - Quick Sort
        - Merge Sort

    Inbuilt function sort() for C++.
    Avoid lambda function for the user-defined comparator.
    The comparator function should return False for equal inputs.
    (return a>b; OR return a<b;)
12. Time complexity: O-notation, Ω-notation
13. Use Codechefs as little as possible, but for contests only.
14. Atcoder beginner contests (every Saturday, 100 minutes). Target at least A & B for beginners.
15. Practice Codeforces(CF):
    - Contest
        - Future contest
        - Running contests
        - Past contest
    - Problem set

    Explore the website.

16. In CF problem set, start solving from 800 rated problems. For CDC, we should be comfortable with 1600 rated problems and solve 2000 rated problems with effort and time.
17. For someone doing CP for passion, 1800 rated problems should be comfortably solved and should solve 2400 rated problems with enough time and effort.
18. Rating increases with contests and rankings.

19. Contests impose time constraints - good preparation for CDC.
20. Don't get demotivated even if no problems were solved in a contest. Practice and have patience.
21. Efficient code writing and implementation is necessary
22. Youtube Channels you can follow:
    - Errichto: https://www.youtube.com/c/Errichto
    - Neal Wu: https://www.youtube.com/c/NealWuProgramming
23. At least 6 easy-level questions as a beginner every day (1-2 hours daily).
24. Even after solving, look into the solution for a possible better solution and implementation.
25. Observe the time taken for a question.
26. Username of Satvik Bansal: satvikb (Codeforces, Codechef)
27. VS Code Extension: Competitive Programming Helper
28. Chrome extension: Competitive Companion (for efficient input/output and running the test cases)
29. Improve typing speed (try without looking at the keyboard).

RESOURCES:

1. ****Use your Institute account to access these.***
CS21203-ALGORITHMS-I (Autumn 2021): -
Prof. Animesh Mukherjee and Prof. Pawan Goyal
https://iitkgpacin-my.sharepoint.com/:f:/g/personal/garggopal2001_kgpian_iitkgp_ac_in/EmienSdzpwhEkoPYmA4Hz4MBBDDPZD_SEFLDZagglh5k5A?e=GZzAhQ

CS29203-ALGORITHMS LABORATORY (Autumn 2021): -
Prof. Animesh Mukherjee and Prof. Pawan Goyal
https://iitkgpacin-my.sharepoint.com/:f:/g/personal/garggopal2001_kgpian_iitkgp_ac_in/Ej8saB8JhcxMlfx-zZZRvnwBrC_MfMML-CxLGPdCp0bfEA?e=QGO7vO

CS10003-Programming and Data Structures (Spring 2021): -
Prof. Abhijit Das
https://iitkgpacin-my.sharepoint.com/:f:/g/personal/garggopal2001_kgpian_iitkgp_ac_in/EuO08BPg8llAieHbGd1yoPQBczl2TKUZSXnlealASc2HVA?e=ZU4tqO

CS19003-Programming and Data Structures Laboratory (Spring 2021): -
Prof. Pallab Dasgupta
https://iitkgpacin-my.sharepoint.com/:f:/g/personal/garggopal2001_kgpian_iitkgp_ac_in/EgNJTnYuddZAuKRvR_hu-ZYBkWt0kIwNEdPqSmD5MmmhTw?e=hAxb0V

2. Competitive Programmer's Handbook: https://cses.fi/book/book.pdf
3. Important Websites:
    - Codeforces: https://codeforces.com
    - Codechef: https://www.codechef.com
    - Atcoder: https://atcoder.jp/home
    - Geeksforgeeks: https://www.geeksforgeeks.org

- ***All the good tutorials found for Competitive Programming - Codeforces https://codeforces.com/blog/entry/57282 (This is mainly for topic-specific preparation).***
- https://cp-algorithms.com/ (This is mainly for topic-specific preparation).

4. For more resources and tips: https://wiki.metakgp.org/w/Competitive_Programming