PrepBytes

How to Master

Data Structure & Algorithms

Aveek Goyal TCE-PrepBytes

Mentored more than 3000 students.

Agenda

- Importance of Competitive Coding
- Topics to learn
- How to learn and practice different topics?
- Different coding contests
- What to do when you are stuck at a question?
- How to improve logic building?
- Time management
- How many questions we should solve each day?
- How to concentrate for longer duration?
- QnA

Interview Experiences

- TCS
  - TCS came to my college to recruit for the digital profile. We had an aptitude test on a day and a coding test on another day. Aptitude had easy MCQs and the **coding test had 2 coding questions** on the TCS Ion platform. The first question was easy based on the array and you need to complete the first question in order to move to the second question which was tough, and the question was on a string. Both aptitude and coding round marks decide your chances of being called for an interview.

Interview Experiences

- TCS
  - Round -1
    - Aptitude
    - English
    - Reasoning
    - Coding(1 Easy +1 Medium)

Interview Experiences

- TCS
  - **Round 1:** NQT Exam (Online Mode)Aptitude
    - NQT (cognitive skills): Numerical Ability, Verbal Ability, and Reasoning Ability.
    - Subject NQT (for IT companies)- Programming Logic (10 questions), Coding (2 Questions-1 Easy and 1 Medium)

Interview Experiences

- Wipro
  - **There are 5 sections in Wipro Online Test Paper**
    - Online Test
      - Quants
      - Verbal
      - Logical
    - Essay Writing
    - Coding
    - Technical Interview
    - HR Interview

Interview Experiences

- Wipro Elite National Talent Hint Interview
  - **Section 1:** Aptitude Test and Logical Ability, Quantitative Ability, English (verbal) Ability Duration: 48 mins
  - **Section 2:** Written Communication Test: Essay writing – Duration: 20 mins
  - **Section 3:** Online Programming Test, Two programs for coding Duration: 60 mins

Interview Experiences

- Wipro Elite National Talent Hint Interview
  - The test consists of 5 rounds which we have to do in **128 minutes**.
  - Verbal Ability (**22 ques** in **18 min**), Logical Reasoning (**14 ques** in **14 min**), Aptitude (**16 ques** in **16 min**), Coding (**2 ques** in **60 min**) and Essay writing (**1 essay** in **20 min**).
  - The level of online test was medium. The coding ques are of easy to medium level. Some students got very easy ques like count the characters in a string, convert lowercase characters to uppercase characters and vice-versa.
  - I got 1st ques to return index of the elements whose sum is equal to given sum in an array and 2nd ques was to return the difference between maximum and minimum number's count in an array. I was able to solve one and half coding ques.

Interview Experiences

- Amazon
  - **Round 1(Online Coding Round):**
    - **Five components (2.5 hours):**
    - **Coding**: 2 Medium level story-based problems. One on **Binomial Coefficients** (use DP to avoid TLE) and another on **Heaps**. I passed 10/10 on Q1 and 9/10 on Q2. Practice the past year's archives extensively to get an overall idea.
    - **Debugging**: 7 questions. Not hard as long as you have a good understanding of any of the OO languages (preferably C++)
    - **Workstyle Assessment (HR)** : 50 questions. You have to move the slider to your inclination for the particular question. Keep in mind the 14 Amazon Leadership Principles while attempting this section.
    - **Reasoning Ability (Aptitude)**: 24 questions. Try to solve as much as you can, as many of the questions are follow-ups of the previous question.
    - **Feedback Form:** I solved all the questions in the given time frame (and had 20 minutes to spare). Each section has an individual timer, so always check how long a problem is taking to be solved. You have to solve every question to get a chance to clear this round, as the competition was very high. Only **40/1000** candidates were selected from this round.

Interview Experiences

- Amazon
    - **Round 2(Online Round 1): Amazon Chime – 45 mins**
    - The interviewer was very strict on the timing. He stated that he would give two problems, and I was supposed to provide him with the optimized solution in the first go (unlike the usual method of starting from naive and reaching optimized)
    - https://www.geeksforgeeks.org/in-place-convert-a-given-binary-tree-to-doubly-linked-list/
    - Given an array of strings with each string denoting a number, find the second largest number without doing type conversion (as the string value can be extremely high, and converting will not be efficient). I gave him an answer using the strcmp() method. He gave me an edge case, and I fixed it by checking the string size also in the condition. There were only 5 minutes left, so I typed out the program as fast as I could while explaining the logic on the side. He asked the TC and SC, and I answered it right as well.

Interview Experiences

- Amazon
  - **Round 3(Online Round 2): Amazon Chime – 45 mins**
    - This time, the interview started with introducing ourselves to each other. The interviewer asked me about my previous projects and internships. As I had done Open-Source in the past, she asked me about my work's impact. I then talked about Javascript, Typescript, and React. She then asked in-depth about a project I put on my resume and about my current internship at a startup (responsibilities, progress in my work, etc.). After this discussion, I was given my first problem.
    - Given a complex string with characters going up and down, what must be done to print it. For this, I decided to use a char matrix and figured out the standard pattern of the up and down traversals. I coded it up quickly as I talked through the logic and wrote the complexities within 10 minutes.
    - https://www.geeksforgeeks.org/a-program-to-check-if-a-binary-tree-is-bst-or-not/

Importance of Coding

- The higher the package more coding questions will be asked in 70-80% of the companies.
- For almost all the companies in the first round there will be a coding question for SDE role.

Importance of Competitive Coding

- Improve Problem Solving Skills.
- Interview coding round will appear just another contest.
- For F2F interview coding rounds, you will be able to give multiple answers and explain in detail algorithms.
- Speed and accuracy of solving various problem increases.
- Usage of different data structures and algorithms becomes clear.
- A good rating of competitive coding platforms give a boost to your resume.

Topics to Learn

- Basic Topics
  - Data types, Operators
  - If-else
  - Loops
  - Functions
  - Arrays
  - Strings
  - Recursion
  - Searching
  - Sorting

- OOPs (C++, Java, Python)
- Structure (C, C++)
- Pointers (C, C++)
- Math
  - Prime Number
  - GCD, LCM
  - Modular Arithmetic
  - Probability
  - Permutation and Combination
- Bit Manipulation

Topics to Learn

- Data Structures
    - Linked List
    - Stack
    - Queue
    - Tree
    - Heap
    - Graphs
- STL (C++)
- Collection Framework (Java, Python)

Topics to Learn

- Algorithms
  - Divide and Conquer
  - Greedy Algorithms
  - Backtracking
  - Dynamic Programming
- Advance topics
  - Segment Trees
  - Fenwick Trees
  - Game theory
  - Computational Geometry
  - FFT

Learn and Practice

- Basic Topics
  - Data types, Operators
  - If-else - 5
  - Loops - 10
  - Functions
  - Arrays - 15
  - Strings - 15
  - Recursion - 15
  - Searching - 15
  - Sorting - 15

- OOPs (C++, Java, Python)
- Structure (C, C++)
- Pointers (C, C++)
- Math
  - Prime Number - 5
  - GCD, LCM - 5
  - Modular Arithmetic - 10
  - Probability - 5
  - Permutation and Combination - 5
- Bit Manipulation -10

Learn and Practice

- Data Structures

Linked List	-10
Stack	-10
Queue	-8
Tree	-15
Heap	-10
Graphs	-15

- STL (C++)
- Collection Framework (Java, Python)

Learn and Practice

- Algorithms
  - Divide and Conquer -15
  - Greedy Algorithms -15
  - Backtracking -15
  - Dynamic Programming -25

Which language to choose?
[C, C++, Java or Python]

Coding Contests

- Long Coding Contests (more than a weeklong)
  - Give you time to think, research and learn new algorithms to solve the questions.
  - Can start giving after Basic Topics are learned and practiced.
- Short contests (2 to 3 hours)
  - Practice to think faster in a pressure environment.
  - Most coding rounds during interviews will be like short contests.
  - They should be given once you have enough practice of long contests and are done with DSA.
  - You can give them after learning basic topics also, first 2 questions mostly will be from those topics.

What if you are not able to solve some questions in a contest?

- After the contest see the editorials for the questions which you are not able to solve during the contest and implement the solution yourself.
- If not able to implement see the solution of other coders.
- For the question you were able to solve see the editorial/solutions to know if there was some other logic to solve the same problem.

How to improve logic building?

- This whole process/roadmap that we discussed will improve your logic building.
- The more you learn and practice the better logic building will be.
- Make a group of friends who are equally interested in coding and discuss various approaches and algorithms regularly.

General Timetable that most can follow

- 6am to 8am – Uninterrupted Study. Watch videos, try to create logic and solve questions on your own. Note down the doubts somewhere.
- 8am to 6:30pm – Your time: college, friends, food, ask doubts which you encountered in the morning.
- 6:30pm to 7:30pm – Work on the doubts, solve the problems. (Uninterrupted)
- 9:00pm to 10pm - Work on the doubts, if not doubts try to solve some questions or watch videos. (Uninterrupted)
- Sleep between 10:30pm to 11:pm.

- 6am to 8am , or 7am to 9am – Uninterrupted Study. Watch videos, try to create logic and solve questions on your own. Note down the doubts somewhere.
- Then its your choice, do ask doubts whenever you get time in between.
- If you can give 2 hours in noon, then 2pm to 4pm is a good time.
- Then again, your choice. For the remaining day.
- Morning two hours should be uninterrupted.
- In rest of the day either divide into several hours or again give 2 hours uninterrupted.

How many questions we should solve each day?

- Do not have the number of questions to be solved in a day as a target.
- It is not in your hands; it depends on the question. If you are solving some easy questions, you can solve 3 to 4 in a day, on the other hand if it is a hard question, then maybe even 1 question will not be completed.
- Goal should be the uninterrupted time you can invest in the whole day. I personally believe each student can easily invest 4 hours each day.
- The trick we have already discussed, the 2 hours of morning is the best time to study without any kind of disturbance. And then remaining 2 hours you can divide in the remaining day.

How to concentrate for longer duration?

- Take a quick break.
    - Listen to 1 soothing song.
    - Or watch 5 mins Friends clip/ Standup Comedy/ Poetry
    - Close your eyes, count from 10 to 0 in reverse and starting studying again.
- In 2 hours slot your break time should be only of 15 mins at maximum. And study time of 1 hour 45 mins.
- If your mind diverts to thoughts other than the problem you are solving or video you are watching, starting counting in reverse to gain the concentration back.
- Watch some motivation videos in between, it also helps.
    - https://www.youtube.com/c/TomBilyeu
        - Impact Theory
    - https://www.youtube.com/c/BeInspiredChannel

QNA