

Cvent Interview Experience

Anushka Gurjar

M.Sc. CS (2019 - 2021)

Round 1: Aptitude Test

Number of Questions: 30

Time: 1 Hour

The questions were based on C/C++/Java, DBMS, OOPS, SQL Queries, and basic Aptitude. The time given was more than enough. Most of the questions were easy, just basic knowledge is enough. There was no negative marking in this round.

28 students were shortlisted after this round.

Round 2: Coding Test

Number of Questions: 1

Time: 1 Hour

Platform: Codility

Problem statement was similar to [Minimum Window Substring](#).

Problem Statement:

You want to spend your next vacation in a foreign country. In the summer you are free for N consecutive days. You have consulted a Travel Agency and learned that they are offering a trip to some interesting location in the country every day.

For simplicity, each location is identified by a number from 0 to $N - 1$. Trips are described in a non-empty array A : for each K ($0 \leq K < N$), $A[K]$ is the identifier of a location which is the destination of a trip offered on day K . Travel Agency does not have to offer trips to all locations, and can offer more than one trip to some locations.

You want to go on a trip every day during your vacation. Moreover, you want to visit all locations offered by the Travel Agency. You may visit the same location more than once, but you want to minimize duplicate visits. The goal is to find the shortest vacation (a range of consecutive days) that will allow you to visit all the locations offered by the Travel Agency.

Example 1: Given $A = [7, 3, 7, 3, 1, 3, 4, 1]$, the function should return 5. One of the shortest vacations that visits all the places starting on day 2 and ending on day 6.

Example 2: Given $A = [2, 1, 1, 3, 2, 1, 1, 3]$, the function should return 3. One of the shortest vacations that visits all the places starts on day 3 and lasts for 3 days.

Note that -

1. Your code will be judged on hidden test cases only, which means passing the base cases doesn't mean your code is correct.
2. After figuring out the approach, do think about all possible edge cases.
3. You can run custom test cases there, do run some test cases before final submission.

11 students were shortlisted after this round.

Round 3: Technical Round 1

Duration: 1 Hour

Platform: Zoom, Google doc

The interviewer was friendly. He introduced himself and asked me to do so.

Then he asked me for a brief summary of my projects. I explained all of my projects very precisely. After that he asked me some random questions related to my projects only.

He asked me a coding question: [Lowest Common Ancestor of a Binary Tree](#). I started with brute force and then gave him the optimal approach. Time and space complexities were discussed for each solution. He asked me to code the optimal one.

Then he asked some Database questions -

- Difference between DELETE and TRUNCATE statement.
- What are joins, why do we use it? Give an example of left join.
- What is the use of group by, give an example of it.
- 3-4 Queries were asked based on Joins, Group by and SQL functions.

Example: Given a department table (dept_id, dept_name) and an Employee table (emp_id, emp_name, dept_id), Print the department id and count of employees in that department. (in DESC order of count of employees)

Then he asked me a Puzzle: You are given 8 identical looking balls. One of them is heavier than the rest of the 7 (all the others weigh exactly the same). Find the heavier ball in minimum number of comparisons/iterations.

7 students Qualified this round.

Round 4: Technical Round 2

Duration: 35-40 mins

Platform: Zoom, Google doc

No introduction. He already checked out my resume, he asked me about the challenges I faced as a PC. Then he asked me the following coding problems -

1. Design a stack that supports push, pop, top, and retrieving the minimum element in constant time. [Min stack](#)
2. Given an m x n matrix. If an element is 0, set its entire row and column to 0. Do it in-place. [Set matrix Zeros](#)
3. Given a string, check if it contains any duplicate character or not in O(1) space. [Unique Characters or not](#)
4. Given an unsorted integer array nums, find the smallest missing positive integer. [First Missing Positive](#)

For each question I explained the approach followed by time and space complexity.

Then he asked me one OS question -

“You are given a paragraph, which contains a number of words, you are given ‘m’ threads. What you need to do is, each thread should print one word and give the control to the next thread, this way each thread will keep on printing one word, in case the last thread comes, it should invoke the first thread. Printing will repeat until all the words are printed in paragraphs. Finally all threads should exit gracefully. What kind of synchronization will I use?”

I approached ‘m’ semaphore solution. Then there was a discussion digging around threads, lock, mutex, semaphore.

At last he asked me if I had any questions. I asked some and the round ended.

3 students cleared this round.

CCAT: Criteria Cognitive Aptitude Test

It was divided into two sections:

1. Aptitude Test (22 Minutes, 50 MCQs)
 - It covered basic verbal, math and logic, and spatial reasoning.
 - No negative marking. Try to attempt as many questions as possible.
 2. Personality Test (no time limit, 140 MCQs)
 - Questions will repeat, so do not lie while answering.
-

Result: 3 students were offered Intern + FTE.

Tips -

1. Revise everything you have written on your resume, Specially projects and skills.
2. Do not rush into the optimal solution (unless directly asked) if you have already solved the question, always give a quick brute force approach followed by optimizations.
3. While coding, use meaningful variable names and write modular code.
4. Never say that you can't do it. Even if there is a problem that you have not solved before, keep attacking the problem from different angles, the interviewer will give you hints. But if you would say that you can't solve the problem it is a big red flag.

All The Best !! 😊