

b'

Amazon Interview Experience (On Campus for SDE-1)

- Difficulty Level : [Medium](#)
- Last Updated : 26 Jul, 2019

Amazon visited IIIT Noida for on-campus process in 2019.

Round 1 (Online Coding Round- 90 minutes on HackerEarth):

This round consisted of 2 coding questions and 20 MCQs. The difficulty level of the coding questions was easy, and the MCQs tested the knowledge of CS Fundamentals (OS, DBMS, CN, OOPS etc).

The coding questions were:

1. [Given an integer N. You have infinite number of 3, 5 and 10 denomination coins. Print the no of ways you can form a sum of N by using the coin denominations. Soln: Standard DP coin-change problem.](#)
2. Given profits of a company for N days and Q queries. Each query contains two integers L and R. For each query, print the number of days on which the profit is greater than or equal to L and less than or equal to R. Soln: Sort the profits, and for each query, use binary search to find the upper-bound for R and lower-bound for L. Time Complexity: $O(n \log n + Q \log n)$

Some of the MCQs which I can remember:

1. Time complexity for $T(n) = 4T(n/2) + n^2$ (options : $O(n^2)$, $O(n \log n)$ etc)
2. Given: Prefix expression. Which of the following is the corresponding postfix expression.
3. Calculate the number of page faults that will occur, Given: OS uses FIFO for page replacement, no of pages per frame given. And system uses x pages in a specific order, and then some y pages in reverse order.
4. Purpose of Ping in networking.
5. Some questions on SQL commands (add column to a table etc)
6. In OOP, which of the following is used to achieve runtime polymorphism. Options: Friend function, operator overloading, function overloading etc)
7. If a machine sorts x entries in y seconds, how many entries will it sort in z seconds. Given: bubble sort is used for sorting.
8. Postorder traversal of tree given, which of the following is in-order.
9. Questions on deadlocks, semaphores etc
10. No of nodes in a complete binary tree with N leaves.
11. Although the questions were easy, the cutoff was very high. Out of ~400 students, 30 were able to clear this round.

Round 2 (Face to face Interview- 1 hour):

The interviewer was very friendly and started with "tell me about your projects". It sounded like a formality, I gave a very brief answer and then he started with the questions.

Q1: [Print all triplets in an array that sum to a given number k](#). The catch here was that ALL such triplets have to be printed. For e.g. if the given array is {1, 1, 1, 1, 2, 0} and $k = 3$, the output should be: {(1, 1, 1), (1, 1, 1), (1, 1, 1), (1, 1, 1), (1, 2, 0), (1, 2, 0), (1, 2, 0), (1, 2, 0), (1, 2, 0)}

I started thinking and was explaining my thought process to the interviewer as I was thinking. First, I talked about sorting the given array, then for each element $A[i]$ in the array, I found all pairs that sum

to $k - A[i]$. For finding the pairs, I fixed one pointer at $A[i+1]$ and one at $A[n-1]$ (if n is the size of the array). I had to modify the solution such that all possible pairs get printed, and the interviewer himself gave 2-3 hints for that.

Time Complexity: $O(n \log n + n^2) = O(n^2)$

Extra Space: $O(1)$

Then he asked me to code the solution, which I did, and after performing a dry run, submitted the solution. The interviewer told me to re-check my code. I analyzed it, a case was missing, which I added and then he was convinced.

\xc2\xa0

Q2: Given an array in which for all i , $A[i+1] = \text{either } A[i] + 1, \text{ or } A[i] - 1$. In such an array, find a number k .

Ans: It can be done in $O(n)$ using linear search. But it can be optimized by performing jumps. For example if $k = 10$ and the first element is 5, we can jump 4 elements because the 4th element can be at max 9. And so on

Then the interviewer modified the question. Now, $A[i+1] = \text{in the range } \{A[i] - t, A[i] + t\}$.

Ans: size of the jump will be modified.

Then he asked me to code the solution which I did.

\xc2\xa0

It took me approximately 30 minutes per question, and I reached at the solutions gradually with the help of hints from the interviewer. It might sound counter-intuitive, but the interviewer giving hints is a positive signal (as long as you are able to catch those hints and arrive at the solution he wants you to arrive at).

Round 3(Face to face Interview- 45 minutes):

Question: Implement the algorithm of split-wise app (Algorithm, not system design). Given transactions in the format: $A \rightarrow B \ 45$, meaning A owes Rs.45 to B. Reduce the transactions to a minimum.

I started by drawing a graph of the transactions, and I figured that if we detect all cycles in the graph, then we can reduce one transaction per cycle. The interviewer asked me to code it, which I started doing. Then he stopped me in between and told me to come up with a different solution. He gave me a hint to think about how much net amount does each person owe/is owed. Using that hint, I was able to come up with a greedy solution: form two sets of people, those who owe some net amount and those who are owed. Then from those who owe, select one and give to one from those who are owed. But I was not sure about the order in which elements from the two sets had to be chosen. He hinted that I choose the ones with the largest values. I tried to prove it mathematically, and then proceeded with the solution. Then I used max-heaps to extract elements from both the sets.

Time Complexity: $O(n \log n)$

Extra Space: $O(n)$

Then he asked me to code the solution. In the solution, I had used heapify and percolate-up as functions for heap, he made me code them as well.

Round 4 (Face to face Interview- 1 hour):

Question: [Given a BST, find the kth largest element. Size of the BST not given.](#)

Ans: Using reverse-inorder traversal. Kept counter to keep track of no of elements visited.

Then the rest of the interview was based only on projects and my internship. I was also asked some subjective questions like, \xe2\x80\x9cwhat was the most challenging project you worked on and why?\xe2\x80\x9d, \xe2\x80\x9cwhich technology do you like the most\xe2\x80\x9d etc.

The interviewer asked in-depth questions about my projects, but since I had mentioned only simple projects which I had made myself, I answered all questions confidently.

\xc2\xa0

Round 5 (Face to face Interview- 1 hour):

There were two interviewers in this round. In this round they asked a lot of theoretical questions of CS Fundamentals. I am not good with theoretical subjects, and was only able to answer the questions partially.

Some of the questions were: Discuss semaphores and how do they work, what is critical section, what is race condition, what is the difference between a thread and a process, what is scheduling, discuss some scheduling algorithms, explain the working of round robin scheduling. What are atom properties of transactions, discuss normalization, 1NF, 2NF, etc, What is a DNS server, can two DNS servers communicate with each other, What happens when we type a web URL etc.

In one question, I was given a code segment in JAVA and asked whether critical section will be violated or not.

I don\xe2\x80\x99t know how much knowledge of Fundamentals is expected by Amazon, but I got through by answering only some of these questions. I was able to confidently answer some questions. For some questions, I gave partial answers, as much as I could recall. For some questions, like working of round robin, atom properties etc, I was not able to answer at all.

The round had not went well till this point, but as the interviewers were very friendly, I requested them to ask me some question on DS/Algo. Then they asked me to check if two BSTs are the same (i.e. have the exact same elements, the structures might differ).

Soln: Store in-order traversals of both trees in arrays and check if the two arrays are the same.

Time complexity: $O(m+n)$, Space Complexity: $O(m+n)$

The interviewer told me to optimize the space, and gave a hint to think about some other way to perform in-order traversal. Then I performed the inorder traversals using stack instead of recursion. I kept two stacks, and kept removing the same elements from both stacks, when they appeared in both. And this stack will have at max $O(\log n)$ elements (or $O(\text{height})$ elements), so space complexity is reduced. Then I was asked to code the solution which I did.

\xc2\xa0

Suggestions:

- Start with a brute-force solution and then keep optimizing it for as long as you can. You need to reach the optimized solution eventually, and it\xe2\x80\x99s perfectly fine if you start with a very simple solution and optimize it gradually.
- Try to prove every part of your logic mathematically, don\xe2\x80\x99t make wild guesses. If you form any hypothesis, don\xe2\x80\x99t proceed with it without proving it.

- Clarify the questions from the interviewer, ask him the output for some random test cases to be sure you have understood the question properly.
- Practice writing code on paper. It is very different from writing it on your computer, since you cannot edit much on paper.
- Always state the time and space complexity of your solution.
- Perform dry runs before submitting your code to the interviewer.

My Personal Notes\arrow_drop_up

Add your personal notes here

Save