

## Amazon Interview Experience (For SDE Intern)

- Difficulty Level :[Hard](#)
- Last Updated :17 Nov, 2019

Amazon visited our campus recently(November 2019) for hiring SDE Interns. It was a three round hiring process.

### Round 1 (Online Coding Round)

It was an online coding round on mettl platform. We were given 90 minutes to solve 2 coding questions and 28 mcqs.

The mcqs were solely based on data structures and algorithms as well as finding error/output in given c/c++ snippet there was no negative marking.

The coding question for me were,

1. It was a simple question based on string manipulation, you just have to take care about corner cases, otherwise it was a cakewalk.
2. The second question was to [find the next greater number using the same digits as given in the input string](#). Constraints on input string length was  $1 \leq \text{length} \leq 100000$  so an  $O(N \log N)$  solution would be good to go.

Example

Input: 1532

Output:2135

I'm not sure exactly but, students who solved both the coding questions completely and also did well in mcqs were selected for next round. The results of this round were declared after a week. And the next two rounds were after a week of results.

### Round 2(F2F interview):

This round was a f2f personal interview and was of nearly 1 hour for each candidate.

First he started with his introduction about his background and his current work in amazon as SDE, Then I was told to introduce myself. After that he began with the interview.

First he went through my resume as I mentioned about my summer internship he asked about my work there and about the application I worked on. He was friendly and seemed to be keen about everything I said. Then he took a glance on my project and found one of my project interesting so he asked me to show the logic and calculations on paper, I explained it for around 10 minutes, he was convinced with the answer and then we moved to the data structure and algorithm questions.

1. Keeping the story aside the question was basically to find [two nodes which sums up to given value in a balanced binary search tree](#).

\xc2\xa0 \xc2\xa0 \xc2\xa0 \xc2\xa0 \xc2\xa0 \xc2\xa0 \xc2\xa0 \xc2\xa0 \xc2\xa0 \xc2\xa0 I suggested an approach to take inorder traversal and get the sorted array. And then to apply [this](#) two pointer O(N) time approach as the array already being sorted. He asked me if I can improve space complexity instead of O(N) to something better? I told him a method to maintain two stacks one for inorder traversal and the other for reverse inorder traversal and doing the same two pointer approach as one stack would give array elements in ascending order and the other would give it in descending order. I was not sure about the iterative implementation of this traversals so he said it was fine and told me to code the first approach. He reviewed the code thoroughly and after some explanation on code from my side he was satisfied and moved on next question.

2. He asked me to design a data structure which supports insert, search and delete operation in best possible way. I suggested a doubly linked list with hashmap then he said if I can improve on delete time complexity. Then I suggested Self balancing BST (AVL Tree) which will support all of these in O(LogN) time. He was convinced and told me to show how will I balance the tree? I told him about the rotations to perform after every insert and delete operations. I didn't have to code this.

The results of this round were declared after around an hour of completion of the entire round for all the candidates.

### Round 3(F2F interview):

This round was also based on data structure and algorithms and it lasted for more than an hour.

1. The first question was that I was given an array of non negative integers where each integer represents the maximum number of jumps that I can take from that position and I have to find the number of ways to reach the nth step. where n is the length of the array. I suggested him a DP solution, I took an array count[n] which stores number of ways to reach that particular step and let's take the input array as steps[n].
  - o And the relation was,  
count[0] = 1; (As there is one way to reach there and we are already there by that way!)  
for each i from 1 to n  
count[min(i+1, n-1)] to count[min(i+steps[i], n-1)] would be incremented by count[i] as the ways to reach here were count[i] and we can go up to steps[i] from here.

He was convinced with the solution and told me to write a working code without any mistake.

2. The second question was about [joining the parts of a rope by taking two parts at a time to join them and finally merging them to a single rope in minimum cost](#), where the cost is defined as sum of the two ropes we chose to join. I told him greedy approach using priority queue, to take two min length pieces from the priority queue and adding the merged piece again to the priority queue until there is a single rope. Then he asked me how would I implement priority queue? I told him that I'll use min heap. Then he asked me about insertion and deletion in min heap and after that we moved to the next question I didn't have to code this one.
3. The third question was that I was given an n-ary tree each node has an integer value and I had to find a subset of node which has maximum sum but I can't take both parent and child in the set (i.e. I have to exclude at least any one them). This problem resembled to [this](#)

[problem](#) which I had solved earlier. So I suggested the same by taking two extra properties in Node class (i.e. maximum sum including this node and excluding this node) and then taking level order traversal of the tree and changing the value of the including and excluding variables of Nth level while being in (N-1)th.(i.e. while inserting them in the queue, here we have to do it by considering all the childs of the current node. And the root node being initialized with the value of excluding as zero and including as the value of that node) And finally taking the sum of max(including, excluding) for all the leaf nodes. He then told me to write working code with all the corner cases covered. And he also told me to write the Node class.

After this 3 question he asked me about my internship and projects and as when the terms kept coming he asked me various questions on them,

- What is react?
- What is component based development?
- What is the difference between SQL and NO-SQL database?
- What is AWS? etc.

At last he asked if I had any questions for him, and then the round was over.

After around two hours of this round the results were declared, I and other 6 students were selected.

I would suggest to be confident and think loudly while being there and also express your thoughts by taking examples on paper and ask questions wherever there is a doubt. Best of luck for your interviews!

My Personal Notes\ *narrow\_drop\_up*

Add your personal notes here

Save