

Amazon Interview Experience | SDE (1.5 Year Experienced)

- Difficulty Level : [Hard](#)
- Last Updated : 26 May, 2020

Round 1 Online Coding Round on Amcat Platform:

1) Top N competitors similar to this

<https://leetcode.com/discuss/interview-question/460127/>

2) A very similar question to this, the same concept of BFS will apply. Given a 2D grid, each cell is either a zombie 1 or a human 0. Zombies can turn adjacent (up/down/left/right) human beings into zombies every hour. Find out how many hours does it take to infect all humans?

Example: Input: $\begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix}$

Output: 2

Both top N competitors and the Zombie problems were quite interesting and challenging. Although the zombie problem was twisted in terms of description, the scope of the problem was exactly the same. Instead of Zombies, it was updating the adjacent servers in the least possible number of days. After this, in each round they were expecting the most optimized and productive code as I was 1.5 years experienced.

Round 2 Asked for my Work Experience and Projects in beginning and then jumped to questions. This was taken by 2 SDEs PEN-PAPER-ROUND:

1) Convert BST to Doubly Linked List without deforming tree and without using extra space except used for creating List. So this shouldn't be done in-place. Time and Space Complexity of my solutions -: $O(n)$ & $O(1)$ respectively.

2) You are given a subarray which has only 0s and 1s, Maximise the subarray containing 1s and in this you can only flip one 0, tell the index of that 0. Similar to this <https://www.geeksforgeeks.org/find-zeroes-to-be-flipped-so-that-number-of-consecutive-1s-is-maximized/>

Time and Space Complexity of my solutions -: $O(n)$ & $O(1)$ respectively.

Round 3 (It was a Managerial Round) -Discussion on one of my Projects and then jumped straight to questions, it was more of WHITE-BOARD-ROUND:

1) <https://www.geeksforgeeks.org/minimum-swap-required-convert-binary-tree-binary-search-tree/> Firstly asked about my approach and then code.

2) Implement LRU Cache.

Detailed discussion on Idea and then code <https://leetcode.com/problems/lru-cache/> I gave an approach similar to what needed in this. Worst time complexity for get() is $O(1)$ and for put() can go to $O(n)$.

3) Find the next greater element of every element in an array. I gave stack-based approach with time and space complexity of $O(n)$ each traversing array from left to right, he wanted to work with the same approach but optimize it further, so I thought and told him that we can traverse array from right to left, it reduces the number of comparisons <https://www.geeksforgeeks.org/next-greater-element/>

Then we had a long discussion on Leadership Principles.

Round 4 This was taken by a SDE, he was very helpful and cool. Firstly he told me about himself and then asked about me and then he jumped to questions. He asked me the difference between Trees and Graphs and then he gave me this question.

1) We have been given a list of pair, which denotes edges of a tree, there can be an edge which leads to the creation of cycle, I have to find a pair, which causes cycle. If there are multiple edges

that form a cycle, then I have to choose last from the list. First I told him, DFS Approach and then I told him Disjoint Set, he was satisfied with the second approach and then I implemented it and wrote a solution for this problem. Time Complexity $O(n^2)$ where n is the length of the list given in input.

2) Find minimum length subarray whose elements sum up to $\geq k$ similar to

<https://www.geeksforgeeks.org/minimum-length-subarray-sum-greater-given-value/>

First I gave him $O(n^2)$ approach and then $O(n)$ windowing technique and then I wrote code for the second technique.

Round 5(It was taken by a different manager)-As this round started so late, I got tired till then. He was really very cool. He asked about me and how is my experience till now. then he gave me a question and he wanted robust productive code considering each and every test case carefully. This code should never crash or give wrong output.

1) Replace every element in array with next greater element present to its right in an array(greater by position not value)so I told him that it is very similar to question asked to me in round 2, still, he wanted to listen my approach and wanted me to code it.I gave him approach and I coded it , then he found some loopholes in it which can be potential points for crashing and some things which he didn't like in my code , then I coded it again as he wanted. He was happy with my this attempt.First code was working fine, the only error in it is no check for null or empty array and some duplicate code.

Time Complexity $O(n)$, Space Complexity $O(n)$

Then again discussion on projects and leadership aspects. The process ended after this.

Verdict: Selected

Thanks, GeeksforGeeks. It helped me a lot in preparation.

My Personal Notes

Add your personal notes here

Save