

b'

Microsoft Interview experience | Set 99 (On Campus for IDC and IT)

- Difficulty Level : [Medium](#)
- Last Updated : 20 Dec, 2018

Recently, Microsoft visited our campus for recruiting interns for their IDC and IT divisions.

Round 1

It was an online coding round hosted on HackerRank. It was 75 minutes long and had three coding questions.

- **Q1:** Given a Linked List of integers, [write a function to modify the linked list in-place such that all odd numbers appear before all the even numbers in the modified linked list](#). The relative order of even and odd numbers should remain the same.

For Example:

Input: 1->8->2->10->5->4->7->6->NULL **Output:** 1->5->7->8->2->10->4->6->NULL

- **Q2:** [Given a floating point number, calculate its square root without using in-built sqrt\(\) function.](#)
- **Q3:** Given a pattern string and a test string, implement RegEx substring matching. If the pattern is preceded by a ^, it will match with the starting position. Similarly, if it is preceded by a \$, it will match the ending position. If no such markers are present, it will check whether pattern is a substring of test.

For Example:

^coal\ncoaltar\nResult: True\n\$star\ncoaltar\nResult: True\nabcd\nefgh\nResult: False\n

Around 50 people qualified for the next round. From a total of 40 test cases across the three problems, successfully passing 20 test cases ensured qualification. All three problems had tricky, hidden edge cases and very few candidates could pass all the test cases in even one problem.

Round 2 Pen and Paper Round: Two coding questions were given whose solution was to be submitted on pen and paper. A total of 45 minutes was allocated for this round:

1. **Q1:** Rotate a square image 90 degrees clockwise
Similar Question: [GeeksforGeeks Link](#).
2. **Q2:** [Given an array of integers and a target sum, write a function to check whether there exists a pair of integers whose sum equals the target](#). Use extra space to reduce the time complexity as much as possible.

A total of 17 students qualified from this round.

Round 3 - Personal Interview 1: This was a technical round. There was a brief discussion on interests and project followed by two coding questions:

1. **Q1:** [Write a function to reverse the order of the words in a string](#)
My initial solution had a small bug, which I fixed later on.
2. **Q2:** [Given a linked list and an integer n, write a function to remove every nth node from the list](#).

For Example, 1->2->3->4->5->6->NULL n=3; **Result:** 1->2->4->5->NULL

Round 4 - Personal Interview 2: This was the second technical round. A brief discussion on coursework was followed by three coding questions.

1. **Q1:** [Given a binary tree and a target sum, return all the root-to-leaf paths which have the target sum](#). After solving this, the interviewer asked for some cases which could be used to test whether the code was working perfectly or not. Once he was satisfied, he modified the problem and now wanted to check whether any node-to-leaf path had the target sum. I briefly discussed my approach and it was good enough.
2. **Q2:** [Given a string of length n, generate all possible substrings of the string](#). I initially misunderstood the problem, thinking all the 2^n possible combinations of the string. Seeing the solution to this, the interviewer said that I am overcomplicating the solution and explained the problem once again. He seemed happy with the final solution I gave.
3. **Q3:** Imagine you have to build an API which would allow customers to book a hotel from all the available ones. How would you approach the problem and what data structures would you use for each functionality. It was an OOP problem and he was looking for a class design which could enable a customer to view hotel details and book one hotel.

Round 5- Personal Interview 3:

This was the HR round. There was a detailed conversation regarding the project followed by basic questions.

- He asked me what books I had read and what was the toughest assignment I had completed in college and what were the problems I faced during it.
- After this, he asked me what would be the optimal way to search an element in an array. It was a rather vague question and I answered that for smaller arrays, linear search would be efficient enough but binary search would be more efficient for larger arrays.
- Then I was asked to implement a library system which could allow anyone to issue or return a book and find the corresponding dues left. He kept adding certain functionalities as time went on which were to be implemented by modifying the classes as per the needs.

In the end, a total of 5 candidates were selected for MS-IDC and 3 for MS-IT.

All in all, it was a memorable experience for me. The entire procedure from start to end was carried out very professionally and the interviewers were friendly and supportive. They were more interested in looking at how a candidate could practically apply the concepts he/she knew instead of seeing how many diverse concepts he/she is aware of. My suggestions for other candidates would be to keep their cool during the process and to maintain their self-belief at all times. Keep your basics strong and learn to develop solutions from scratch instead of remembering several difficult algorithms. Providing correct approaches and thinking clearly matters more than actually giving the perfect solution.

If you like GeeksforGeeks and would like to contribute, you can also write an article and mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

[All Practice Problems for Microsoft](#) !

Related Practice Problems

[Reverse each word in a given string](#)

My Personal Notes

Add your personal notes here

Save