

Microsoft Interview experience | Set 91 (2 Yrs Experienced for IDC)

- Difficulty Level : [Hard](#)
- Last Updated : 19 Oct, 2021

This is the main forum where I did all my preparation. Thanks a lot GeeksForGeeks! Here's giving back.

I had got a call from the Bing team in IDC. I have 2 years of job experience post a BTech in CSE. I had 5 interviews in all, as described below.

1) The first one was a fly/ no fly round. At first, there was some discussion around my previous work experience, with the interviewer asking me some technical and design questions here and there. Then I was asked to write code to find the smallest subarray to sort that would sort an entire array. The answer is available at <https://www.geeksforgeeks.org/minimum-length-unsorted-subarray-sorting-which-makes-the-complete-array-sorted/>. I had to write optimized code for this.

The second was to do an ER diagram for a furniture scenario. The important part here being that Material of a furniture can be a separate class, rather than a property in furniture items like bed, chair, table. So Material can have associated properties like IsFlammable, IsWaterResistant, etc.

2) I got just one question in this interview which I was asked to solve on the board. The question was to add a digit to a number that is represented by a linked list, where each node is a digit of the number. The constraint here was that the linked list couldn't be modified, except the digits to be modified in answer and the number could be infinitely long. So you had to avoid the problem of memory overflow. So no recursion, no stack, no copy of linked list, no change to original linked list, as suggest the solutions on geeksforgeeks. To keep track of carry to be added to previous digit iteratively, you basically have to keep a track of the digit before the last consecutive 9 that precedes the last digit. For example in 9719938996, this would be the 4th digit from right, that is 8. So now in a single pass, you can figure out such a position of 9(if any) and also the last digit. Then the required digits can be changed easily.

3) In this round, the interviewer opened a console app on his PC and asked me to code there. I believe he was actually taking note of the time I took to write the code and whether it compiled in the first time or not. I paid more attention to accuracy of code than to the time I took. The questions I was asked are:

Check if a number N is a power of another number X. Then we spoke of the case where X was 2. For this case, we discussed how this could be done by checking the bits.

Check if two binary trees are identical. I also had to initialize and run it with a test case. There was a discussion on line wise optimization after completing the code.

Rotate a matrix by a given distance. A similar answer where distance to rotate is 1 can be found at <https://www.geeksforgeeks.org/rotate-matrix-elements/>

Then there was a design discussion on designing a library's online portal. After I gave him a basic architecture, he asked how I would implement the search feature. I suggested doing that with the book's name, author, ISBN, etc. He asked me how to do it based on a book's content. I told him I'd store tags for each books that could be used while searching. He liked the idea, but prodded me about how I'd do a proper text search. I told him about using a hash table and blobs. Perhaps he was looking for tries, but since I was not much familiar with them, I avoided that. He seemed satisfied and ended the interview.

4) He asked me to introduce myself, talk about my projects and of the greatest technical challenge that I had faced. Then asked me to write code on paper for the LinkedIn feature where we are

shown what is the connection depth of two people. The solution was a version of shortest path algorithm for a graph. He asked me to optimise my answer for $O(1)$ extra space (besides the space of the graph structure).

The next scenario included a series of questions. I was to give the informal schema of Customer, Item, Order for an online retailer like Amazon. Apart from the foreign key relationships, he was interested in how I'd store the orders because each purchase order can have multiple items purchased. The answer was to have a relational table like ItemizedOrder which stored mappings of the order and item Ids. Then he asked me to write a query to get all customers who had not bought a certain item by a given date. Then he said that for any item like a book, the company will have to get their books from many book sellers who will have different schemas and ways of storing the books' information. He asked me to design a system that will allow handling such variation. I gave him a portal or a web service as an end point where the different vendors could upload their data. Then my web service would have a rule engine which would map different schemas, assign default values wherever possible for missing values like Subtitle, ignore missing values of unimportant fields like NoOfPages, else return error messages for items that had insufficient mandatory data, like Price. This rule engine would then store the processed input in our DB in our schema.

5) This was with a very senior guy, but was more of a formality. The interviewer told me I was selected in the first 10 minutes and then was just trying to analyze my work and preferences for a suitable team and set my expectations. He answered some questions I asked and ended the interview in half an hour.

In general, I was asked 2 common questions every time- Describe your previous work experience and why do you want to join Bing. I did not get the approach to most questions instantly, but I spoke of my thoughts aloud. They all guided me when I got stuck and prompted me towards optimizations possible. Twice, I was also given examples of scenarios which my code did not cover. Since I knew I was interviewing for Bing, I asked the interviewers relevant questions after each round. Each interview was scheduled for 1 hour, though the third one got stretched to 1.5 hours and the last one was quite short. Except for the first interview, the outcome of the others is collective. So even if you don't do very well in one, if you can manage the others decently, you will be hired. All the best!

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

[All Practice Problems for Microsoft !](#)

My Personal Notes *narrow_drop_up*

Add your personal notes here

Save