

## Microsoft SWE Interview Experience

- Difficulty Level : [Easy](#)
- Last Updated : 10 Dec, 2020

**Round 1 (Shortlisting):** Had three coding questions. The level of difficulty was easy-moderate. The first question was on maps, the second on the Coin change problem but with prime numbers instead of coin values, and the third was repeated from last year.

**Interview Round(December 2020):** I had a single interview. The interviewer was from Redmond, having 22 years of experience. He asked me one question :

- Given an unsorted array, find the binary searchable elements in it. Binary search works for sorted arrays because the middle element is always greater than all elements on the left of it and smaller than all elements on the right of it (for arrays having distinct elements). Therefore, finding the leftmax and rightmin arrays and elements that are on positions where the leftmax and rightmin coincide with the array elements would give binary searchable elements.

**The Experience and thinking process:** I was lucky to have an interviewer who was really calm when explaining the question. He gave me a codility link, which is the platform on which Microsoft usually asks questions. I asked him some clarifications on the range of numbers, the number of elements in the array, and whether they can be negative or not. I proceeded to give him a trivial solution, telling him the time and space complexity for it. He then told me he wanted a  $O(n)$  solution. This is when I opened a notepad window and shared it on teams (the interview was virtual). While thinking about a solution, I made sure I was saying everything that I was thinking upon. That way he would stop me if my train of thought steered in the wrong direction. I started with how binary search works and as this was the first time I came across that question, I tried to be as basic in my start as possible. I tried telling him that elements can be searched when the partition (the middle element) is in the right position. I was typing all this on my notepad as I spoke about it. Then I took an example and showed him how it works, making separate rows for the given array, the leftmax and the rightmin, and a bool array representing the elements that are searchable. At this point, he was really impressed with my thought process and the structure with which I built my solution. I told him that now I will code it on the platform. They also want clean codes, so I started with taking the input, defining arrays with good variable names, not just `arr` or `left` or `right`. Naming helped me ignore the comments as everything was clear from the code. I made the leftmax first, then the rightmin array, and finally counted the number of elements. The interviewer told me he loved my coding style. I told him all that could be done in a single loop but making three helped the code be clear to anyone seeing it for the first time. After this, he told me that running the code wasn't needed because he knew it looks correct. I ran it anyway and it was accepted in the first go.

Further, he asked me if I had any questions, and I asked him if he was happy at Microsoft and what factor enabled him to work for 22 years at a single place. He gave a very long answer to it. Try to be interested when the interviewer says something. After this, he told me he was done for the interview. 7 anxious hours later, I directly had my offer. Overall, a very positive experience.

**TLDR?** It is essential to tell the interviewer about the thinking process and your approach to the problem as clearly as possible. Solving isn't enough.

**Preparation:** I hadn't done a single coding problem before August 2020. I had done a basic data structure course in July 2018, and that was all. I started coding, first on **GeeksforGeeks**, then on **leetcode** and **interviewbit**. The first month was a mess. I couldn't do a single

question in the first week. I started with the easy questions, then to medium, and then to hard level problems. Even after the first month, I could only do some easy questions. I continued coding, looking at solutions for most questions in the beginning. I started off with top \xe2\x80\x9cmust do\xe2\x80\x9d questions on GFG, completed all of them, did some more, and then switched to leetcode. I was able to solve some easy questions, some medium ones, but the hard ones were still challenging, even after 2 months. In the third month, I gave it my all, coding from the morning till night. I tried to think about the way others approached similar problems, in the discuss section of leetcode and in the comments of GFG. Slowly, I could see myself getting better at solving. I preferred C++. My friends from CSE helped me a lot. I used to ask every small thing I couldn\xe2\x80\x99t understand, read about data structures again, and looked at solutions over and over again until I was able to do problems on my own. Finally, in the end, from August to November I had solved a total of around 800 questions, many of them over twice.

My Personal Notes\narrow\_drop\_up

Add your personal notes here

Save

,