Placement Interview Experience

Name: Bishnu Tiwari

Department: Computer Science and Technology

Batch: 2022

Company Name: Deutsche Bank

Job Role: Graduate Analyst

Domain of Your Role*(Software Development/Consulting/Core Electronics, etc)***: Software**

Cut-off:

Interview Process*(How many rounds were there and what were the rounds?)*:

Once you qualify the coding round and get selected for interviews, there are the following 4 rounds:

1. Technical Round 1 (45 mins)
2. Technical Round 2 (45 mins)
3. Professional Fitment Round (30 mins)
4. HR Round (15 mins)

The rounds 3 and 4 don't necessarily follow this order, and will be informed during the interview process which round you have to go for first. The technical rounds deal with your knowledge in the basics of coding and also the work you have done so far, and your knowledge about them.

Questions asked in the each stage of the process:

Technical Round 1:(45 mins)

1. Tell me briefly about yourself
2. Which is your favourite subject? (answered DSA)
3. What are the basic data structures that you know?
4. If I need to make frequent insertions, what data structure would you suggest?
5. If I need to perform a lot of read operations, which data structure would you suggest? (answered array)
6. Do you know what hashmaps are? Why wouldn't you use a hashmap in the previous question?
7. Can you tell me the difference between heaps and priority queues? Give me an example where priority queues can be useful.

8. What are the different sorting algorithms that you know?
9. Which according to you is the best sorting algorithm and why?
10. Do you know the string parenthesization problem? Share your screen, write the code for it in an IDE and run it. (input given by interviewer)
11. Do you know about greedy algorithms? Solve the coin change problem using greedy method. Share screen and code.(interviewer gave the denominations)
12. Why do you say that a greedy approach will fail for this problem? And how do you suggest this problem can be overcome?
13. Do you know DBMS? Explain normalization and 1NF, 2NF, 3NF.
14. Do you know OOPs? What is the difference between encapsulation and abstraction?
15. Explain with an example where you would use encapsulation.
16. Explain with an example where you would use abstraction.
17. Questions to the interviewer.

Technical Round 2:(45 mins)

1. What do you understand by API, what are they used for? (resume mentioned a project in fullstack, and using APIs)
2. What is the code that is returned from the backend for success?
3. What does error 404 mean? When do you get error code 500?
4. What are the different (http) methods available to communicate with the backend?
5. Which (http)method does one use in case profile data has to be deleted from the backend?
6. Explain briefly what API you had implemented for your project.
7. Describe the end-to-end process of how the API you developed worked. From getting data from the user to updating the database to returning proper messages to the frontend.
8. While communicating with the backend, you send a json object, how would you handle the data, or what will you convert the data into, to ensure no data type mismatch error arises? (answered a json array)
9. Do you know hash maps? Explain hashmaps. Why wouldn't you use a hashmap to store the values of the json object? (the right answer would be hashmap, as it would save the iterating time through the json array)
10. What are the main differences between Java and C++? (both were mentioned on my CV)
11. What happens if you don't take care of garbage collection while handling objects? What error will you get?
12. What are the different types of system exceptions that you know?
13. Where do you suggest exceptions should be handled, the frontend or the backend? Why?
14. In case your application gets an exception, how would you ensure that the app remains functional, and maintain the beauty/structure of the UI rather than the app crashing?

15. The user wouldn't understand what a null pointer or out of memory exception means, how would you get the user to take the necessary steps to take care of the exception?
16. Share screen and write queries to :
    a. Given an employee table and a list of employee ids, print the name and salary of the employees whose id is in the list
    b. Select the name, id and salary of all employees with salary > 30000
    c. Display the id, name and salary of all employees in ascending and then in descending order.
17. Do you know linux commands? (from CV)
    a. What is the command to change directories?
    b. What is the command to list the directory contents?
    c. What does ls -rt do?
    d. How do you start writing in vi editor? What do you do to save and exit, exit without saving?
    e. Do you know the grep command?
18. Do you know git?
    a. What are the git commands that you are aware of?
    b. Explain the whole process, with commands in sequence, you will follow to add a new file in a repository and push it to the remote branch
    c. Let's say you create a new file, which doesn't yet exist on a branch. What is the term associated with such files/changes that are yet to be added? The changes are stored in kind of a buffer, and in any popular ide you will see this term for the changes. What is that term? (staged/staging was the answer)
19. What are the sorting algorithms you are aware of? Share screen and code bubble sort. (used C++)
20. While coding, I had used a reference to the array, so the interviewer then asked me about how this also highlights a difference between C++ and Java, and if I could identify it. (answer was pointers)
21. Code binary search. In case of duplicate elements, return the index of the first occurrence of the target value.
22. Explain all the sorting algorithms that you know, and which one do you think is the best.
23. Questions to the interviewer.

Professional Fitment Round:(30 mins)

1. How were the previous interviews?
2. What is one thing in life that you are proud of?
3. Describe one such project from the past, end-to-end, where you think you couldn't perform well, or which you will be able to do differently now and improve upon.
4. Describe one project where you worked in a group and your learnings from it.
5. Is there any project you can demo or show the code or results and explain it?
6. Questions to the interviewer.

HR Round:(15 mins)

1. Give me a brief introduction of yourself, your background, who you are, your interests and hobbies.
2. Given your interests and passion, why do you want to join DB? How do you think DB will help you in your targets?
3. Family background
4. Which would you prefer, working on a project alone or in a group? Why?
5. If one of your group members is unable to perform well, and your team has to meet the deadline, how would you handle the situation?
6. If given a chance to replace the above group member, would you replace the person or choose him/her again?
7. Questions to the interviewer.

Reason behind selection results*(Which skills helped you the most to get the job?)*:

Although it's difficult to evaluate by oneself what might have worked in the interviews, I think more than any particular skill, my composure, confidence in answers and the way I presented myself was the key to getting this opportunity. It's important to have good knowledge and be clear about the skills and work listed on one's resume. The interviewer isn't looking for one particular thing in every candidate, but the objective is to evaluate the candidate based on what he/she has worked on. Although, for a software role, it is necessary to know a decent level of coding, and be good at not only thinking of a logic to solve a given problem, but also be good at coding what one is able to think. Another important thing to keep in mind is that the interviewer is not looking for perfect answers, but your ability to understand and explain concepts. So it's always better to be honest and clear about if you know the answer or not, and give the interviewer as many chances as one can to ask questions, as more the number of questions, more the chances one has to showcase one's knowledge. Also, trying to actively discuss the thoughts at every stage of answering a question or solving a problem helps. In an online scenario, being able to handle situations of internet problems, clearly communicating them to the interviewer and not letting the interruptions hamper your flow are also important in making a good impression. Lastly, working on the basics, being calm and confident, and answering clearly and honestly are keys to doing good in interviews. An added bonus in the HR rounds is if one is able to prepare beforehand how to justify applying to the company, and having thought about the popular HR questions beforehand.

Preparation Strategy*(How did you prepare for the non-interview rounds? How did your preparation evolve for the interview rounds?)*:

For the coding round, it's most helpful if one has been solving coding questions from popular resources like Leetcode, GFG, etc., and looking up past problems from the company. It's easier for a person who has been active in competitive coding. There is a set of popular concepts, and

questions are generally picked from that pool. The more one has practiced, the better the chances.

Once you clear the coding round, look up the past interview questions from the company online, from sites like gfg. Most companies have this set of concepts which they prefer to ask. Going into the interview, your theoretical knowledge should also be good, so revise college notes of popular subjects like DSA, DBMS, OS, OOPs, and in some cases Computer Networks too. Taking a very customised approach to interview preparation based on the past interview questions from the company helps, as most of the questions are then from the pool of questions you arrive at. It is also important to know the popular questions using Strings, Arrays, Dynamic Programming, Greedy approach, Divide and Conquer, and important data structures such as Linked Lists, Trees, Hash Maps, Arrays, etc.

Resources to prepare from -

Necessary Resources*(Must Do)*:

Top Interview Questions cards on Leetcode (at least the easy card)
Striver's CP Sheet
Past interview questions from GFG

Advanced Resources*(Can Do)*:

Competitive Coding competitions on Codeforces/Codechef
Leetcode Daily challenges
Top Interview Questions card on Leetcode(medium and hard)

Anything else that you'd like to share with the students: