

## Placement Interview Experience

**Name:** Raghavendra Singh Chauhan

**Department:** Electronics and Telecommunication Engineering

**Company Name:** Deutsche Bank

**Batch:** 2021

**Job Role:** Software Engineering

**Cut-off:** Online Test Based Shortlisting On HackerEarth and Min-70% upto 6th Sem. For online test link.

### Interview Process

*(How many rounds were there and what were the rounds?): 4 rounds*

Technical Rounds - 2; Professional Fit - 1; and HR Round.

### Questions asked in the each stage of the process:

Online Test: 20+50+50 - 3 problems.

1. One was a Math based problem.
2. One was on DP(LC Medium)
3. Another one as far as I can recall probably related to Combinatorics or DP

Technical Rounds -

1. Broadly Binary Search Tree. Its implementation and code walkthrough, Graph Search Algorithm DFS based question.
2. Application based questions through presentation of use case scenario for sorting algorithms.
3. Time Complexity
4. A basic String Based Problem related to frequency

Some Resume specific questions

1. Computer Networks,DBMS and Operating Systems.
2. Quite Fundamental questions related to TCP IP layers,
3. SQL queries of medium nature. Theoretical questions on DBMS.
4. I skipped a OS based question as I had not covered it much.
5. Java fundamentals and internal implementation of HashMap.
6. Code Snippet based output calculation question.

Professional Fit Round was kind of a behavioral round as you may find on LeetCode presenting you with such scenarios that you might face at work and expecting your reaction to it and your way of dealing with these situations, such as team conflicts, Disagreement, etc.

HR round is a short round focusing on the non technical aspects of your personality.

## Reason behind selection results

*(Which skills helped you the most to get the job?):*

- Being acquainted thoroughly with one programming language is essential. I had Computer Application in 9-10th and then Computer Science in 11-12th so my acquaintance with Java language and basics of programming has been quite natural.
- That doesn't mean that one cannot learn during graduation. I mean, in my case, it was quite comfortable to carry on. But in normal circumstances we have Basic Programming in 1st year and that's enough as I have seen my friends excelling who didn't have any prior experience with coding. You learn C you can easily switch towards C++ or Java and it helps.
- Data Structures and Algorithms are the fundamentals for clearing most Interviews. Consistently solving problems on online platforms since Summer last year has helped me a lot. Skipping for long durations disturbs the flow of solving problems.

## Preparation Strategy

*(How did you prepare for the non-interview rounds?)*

*How did your preparation evolve for the interview rounds?):*

- Data Structures and Algorithms are basically the most important tool for most interviews.
- For that I practiced through LeetCode(best), GeekForGeeks, Codeforces, InterviewBit, Hackerrank and HackerEarth. Primarily from LeetCode and Codeforces after clearing basics from other platforms. I didn't participate in contests but kept myself time bound.
- Apart from that for core subjects I had some overlap with my electronics curriculum and my optional for extras I referred to GFG and YouTube.
- LeetCode Discussions are also a great source for Interview experiences and learning on any topic.

## Resources to prepare from -

### Necessary Resources(Must Do):

- DSA and practice on LeetCode/GFG and then Competitive Programming(if you like it) and InterviewBits.
- (Design and Analysis Of Algorithms- CMI NPTEL; MIT OCW Algorithms; Coursera Algorithms courses are also nice; Tushar Roy; GeekForGeeks; I have referred to many sources and links because one has to read multiple sources to enhance understanding for some particular topics. I will add the links for discussion threads if I come across the good ones.

Introduction to Algorithms CLRS, CTCI Gayle McDowell, Narasimha Karumanchi

Above stated is a long list which one may refer for any particular topic but not as a whole)

- A good hold over one programming language such as Java/C++. C++ helps in CP for those looking towards higher rankings.(Head First Java or Herbert Schildt)

### Advanced Resources(Can Do):

- Competitive Programming is essential in these times I feel and what I have missed, because a good rating on Codeforces or Codechef opens up a lot of opportunities and enhances the problem solving ability. If someone likes it one should spend time on CP as a hobby.(Codeforces or Codechef)

- Coursera Algorithms course by Univ Of San Diego(some specific parts are quite good if someone had already covered basics)( I came across it recently)
- cp-algorithms is a good site for DSA
- Grokking The System Design Interview (Though not required in case of DB but will help as some companies may conduct a round on it.)

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

The resources I have shared or the views I have put is to generalize the approach towards Interview preparation.

Solving daily is quite helpful and one must focus on revision of concepts, as sometimes it may be necessary to code quickly (for conventional problems) so that you can have more time for good level questions in Interview or online assessment.

I have not mentioned other programming languages such as JavaScript and Python which are quite popular but sometimes some library constraints might be an issue. So for that please ask those who use them primarily to clarify.

Thanks for reading.

If you have any doubts or queries or clarifications please feel free to connect.