

Internship Interview Experience

(Mohit Kumar)

About Me:

I am a B.Tech undergraduate 2nd year student at Chandigarh University pursuing Computer Science Engineering. I am a passionate algorithmic coder and problem solving is my life.

Former Software Engineer Intern at Microsoft, Google, Goldman Sachs, Coding Ninjas, Tech Mahindra, Nutanix.

I have 24 Patents and 6 Startups, Featured By Many Newspaper's

Awarded by President of India for Young Entrepreneur 2019

Awarded by India Excellence Book Award for Entrepreneur of The Year 2020

I am creating this document to share my journey with all the ambitious students out there, giving a sneak peek into the corporate world. I hope this helps and guides you in your interview process.

Good luck to all!

“Everything comes to him who hustles while he waits.” -Edison

Goldman Sachs

● Coding Round

- MCQs (aptitude+advanced probability+os+dbms+coding) + 2 coding questions + 1 situation-based question on Hacker rank
- Probability questions were challenging. Rest was pretty straight forward. Coding questions included 1 easy question(arrays) and 1 difficult question (graphs).
- Roles offered were Quant and CS. Quant role required good performance in the MCQs and CS role required good performance in the coding questions. If you perform well in both, they'll shortlist you for both.

● Interview Round 1 (CS role)

- Resume + OOPs concepts + 1 question related to sorting an array
- Interviewer was from a Java background and was more focused on OOPs concepts and methodologies. I gave him answers for the Python language which he accepted.

● Interview Round 2 (CS role)

- 1 question based on infinite input sequence of integers (in most standard cases heaps/queues are the solutions)
- 1 standard question on binary search tree

“Genius is one percent inspiration and ninety-nine percent perspiration” -Edison

Uber

● Coding Round

- 3 coding questions (1 was array-based bit manipulation, 2 were dp based)

● Interview Round 1

- Purely CP based (on Code pair platform)
- 1 question on a 2-D array traversal featuring obstacles (was quite easy, kind of basic BFS)
- 1 question on Minimum spanning tree

● Interview Round 2

- CP+Design (on code pair platform)
- 1 question to design amazon book service (insert, update, delete in DB) - Heap was all they wanted - I struggled to write the heapify function. Interviewer helped a bit.
- 1 question on arrays. (In particular, lower bound)

● Interview Round 3

- Resume (projects in detail) + CP
- Interviewer asked only one project but in depth. It is therefore very important that you must know (in detail) every single line mentioned in your CV.
- 1 question on string - Given two strings s and t. Find the length of the smallest substring in t which contains the same frequency of letters as in s. I gave him the $O(n)$ solution, which he queried a bit (maybe because the code I wrote was a bit crappy; minus point for me).

"It's fine to celebrate success but it is more important to heed the lessons of failure." -Bill Gates

- 1 question to find k maximum elements in an infinite input sequence of numbers (heap was all he wanted but I didn't use at all)
- Total 6 people were invited for this round, out of which they took 3 due to position constraints.

Google

● Resume shortlisting

- Google did not have a coding round. They shortlisted approximately 15 people based on their resumes out of ~100 people.
- Main things they looked at in the resume (according to me) are - CGPA, competitive coding profile and participation in google kickstart/Code jam.

- Two interview rounds happened for all the candidates. Third round happened for some of the candidates for which Google thought they needed more information. 2 people were selected after 2 rounds and one after the 3 rounds.

● Interview Round 1

- 2 very simple questions, one related to bit manipulation and other graphs (vanilla dfs).
- Interviewer focused on small optimizations in standard solutions. He asked a lot of counter questions in vanilla dfs.
- The 2nd question was very simple, but was phrased in a weird way.
- It is okay if you do not know anything specific to language. For example I did not know the difference between map and unordered map in C++, but I did know the concept of hashing.

"I have not failed. I've just found 10,000 ways that won't work." -Edison

● **Interview Round 2**

- This was harder than the previous one (this is not the case for everybody).
- Both questions were related to binary trees.
- I took a lot of time in the first question (20-30 minute as I remember), it had a simpler solution but my solution was a little complicated.
- I could not give a concrete logic of the second question in time.

Interviewer gave me a hint in the end to use the solution of the previous question as a Blackbox, and then it struck me and I gave him the correct logic. I could not implement that in time.

Microsoft

● **Coding Round**

- 2 parts - 1 for SE role, 1 for ML role ○ SE
 - 3 easy questions were asked based on arrays, strings and math.
 - Multiple sets were there.
- ML
 - Multiple choice questions related to Probability and Statistical

Methods in AI were asked.

● **Group fly round**

- 2 questions were asked and were common to all people.
1 was regarding binary search and 1 regarding lowest common ancestor. It was kind of luck based because there were few people who solved both questions correctly.
- Around 60 people were shortlisted from this round

*“All you need in this life is ignorance and confidence, and then success is sure.” -
Mark Twain*

● **Interview Round 1**

- CP
- 1 question to implement the code to rotate the matrix spirally.
- 1 question to implement functioning of head and tail commands.

● **Interview Round 2**

- Resume + CP
- Interviewer asked about my interests in Computer Science, to which I replied 'Machine Learning'. Then he discussed a few of my projects in detail along with the related coursework.
- 1 question based on strings was asked. Question was quite controversial, and I discussed a lot with him to confirm what I understood was right. The interviewer was not from a CP background, what he wanted to judge here was if I am comfortable with functions in C++ or not.

● **Interview Round 3**

- CP
- 1 question based on 2-D sorted matrix search was asked.
- 1 question based on bit overflow.

“Our greatest weakness lies in giving up. The most certain way to succeed is always to try just one more time.” -Edison

“It's not that I'm so smart, it's just that I stay with problems longer.” -Albert Einstein

Some general tips and tricks

- Whenever you are asked “Tell me about yourself”, tell the things that are not mentioned on your resume. Personal opinion: Always start talking about your background, your journey in the field of interest. Slowly move on to the things relevant to the job, like if it’s related to machine learning, then explain how you get introduced to ML, and how it attracted you toward itself. Your answer must not be as short as a 10 seconds slogan, nor it should be long as 10 mins speech. Somewhere around 3-5 minutes will work
- **Do you have any questions for us?** - Mostly every second interviewer will ask you this question. You must have (or will) read from various sources, “Always ask a question, this shows your interest in their company”. You must know that the person sitting in front of you has also faced interviews, and must have interviewed many too. So, be yourself, no need to ask unnecessary questions. If you are comfortable (have no doubts), state clear to him/her about your thoughts and show how crystal clear you are about this role.
- When you mention competitive programming as your interests, it may sometimes become negative as the interviewer may not be a big fan of it. You’ll get to know if he’s enjoying the verse or not. It’s advisable to divert away from CP in case he/she doesn’t like it.
- Type of questions asked in the coding rounds are different from those asked in the interviews. It is therefore very important that you practice the standard interview problems. Even if you are proficient in competitive programming, I’ll recommend to go through the standard problems at least once before sitting in the interview.
- In case, you face rejection in any interview, note down and analyze the mistakes you made that day, and go through it before attending the next interview. It will surely give you confidence and you won’t make the same mistakes ever.
- In almost every technical round where you are asked a problem, you’ll be asked to code it down, that too on a piece of paper. Try to keep the code modular, and if possible, label comments over each function. What matters the most is the logic to solve that problem, but following this shall grant you some bonus points.

**Thank you for giving this a read. I believe it will prove to be helpful.
All the best!**

"Rejection does not mean you aren't good enough. It just means the other person failed to realize what you have to offer." -Mark Amend
