ADVICE FOR ONE DAY PRE-INTERVIEW

Abhishek Shrivastava :

My interview was on day 2. The day before that I had faced a rejection from a software company on day 1. Naturally I was disheartened and didn't do much on the previous day. But it would be wise to go through your CV and prepare what to say about each of your projects. One might also read a bit about the company and it's recent projects. It might increase your chances.

Don't panic or think too much. No need to revise everything again. Just be calm and confident, and believe in your skills and preparation.

No special advice. Just the usual. Do a dry run of your system twice or thrice before your interview. Look presentable and greet the panelists. Don't be nervous or scared of them. They are there to help you. This is more than enough in my opinion.

Shreyas Kowshik :

CDC internship is quite a hectic procedure. I would suggest you to be just mentally prepared a day before your day 1 interview and worst come worst for day 2 as well. For a scenario let's suppose that you are going to have your interview for GS on day 1. You didn't have a good sleep the night before your interview and GS interview starts from 6 in the morning and ends upto 5 to 7 in the evening as they conduct 7-8 rounds of interview. Now there might be a possibility that even after all these rounds you were not selected and Day2 companies like INTUIT conducts their coding round in the evening of day 1 itself, so you have to be mentally prepared for this situation that in the worst case scenario you will have to give some coding rounds in the evening as well and then again the next day is going to be the same as DAY1. There will be some issues of sleep deprivation on the weekends due to this.

Shubham Mishra:

On 4th September, the shortlists for Day 1 came out. I was thrilled to see myself getting shortlisted for DE Shaw and Quadeye. At the same time, I was a little disappointed for not getting selected in Microsoft and Uber. I shaved my beard and took out my best formals. The stage was set. Interviews would start at 7 am. I placed DE Shaw over Quadeye in my preference. Laying down in my bed after dinner, I started reading up C++ internals, for example, how do virtual functions, constructors etc. work, as these were some of the key questions DE Shaw is known to ask. I slept really tensed. I woke up the next day at 6 am. I got ready. My interview was scheduled at 7 am sharp. I was greeted by two interviewers.

For more details read this blog : https://grapheo12.in/life/cdc-intern/?fbclid=IwAR2cZ4Z5iVlrEKiQek6vjIQmIButHkwukKjA2xtolg29fH6Dg_m-j9LV91E

Prathmesh Deshmukh:

I had 2 days time for interview after shortlist was announced. I spent 2 days reading about past interview experiences from GFG and Glassdoor. Read about company history and other stuff. This is very important for some companies and asked consistently, but may not be relevant for other companies. Revised my resume and prepared for possible questions. Brushed up little for topics like OS, DBMS. I also talked to senior who was selected last year to know about his experience. Be confident in your interview.

Praagy Rastogi :

interviews can start as soon as 5 am. Most important thing is to get enough sleep! Just sit and relax before your round. Make sure you are in all the correct whatsapp groups. Placecom would be helping you through that. Make sure lighting is fine. Have a backup internet connection.

Parting advice:

1. Revise your strengths and try to get rid of weaknesses do not try to learn anything new.
2. Try to practice thinking loud. Practice speaking out any relevant information of the question in your thinking process.
3. Ask your mates to take your mock interview.
4. Try to talk to people who have appeared for that specific company, interview experience varies from company to company.
5. Try to wear formal clothes, suit if possible

For the last few days this blog from **Ishan Ranga** might be very useful :

After my last article last year regarding the preparation of competitive coding, this one is for some pointers to keep in mind just before the interviews. This post again will be a big one, so brace yourselves!!!
You should be somewhat confident in one or more languages, and you might have developed some intuition for solving the questions. Now, this is probably the time to brush up some of the topics which you fear the most and to revise the rest. But there is a lot more to the interviews. So let's start:

● Revise: Mostly, if you have done a lot of coding, then solutions will come intuitively, but what about some questions with unique answers. These answers could even be exclusive to just save on the space. So for these, go through interviewbit topics once and check if you remember those. If you don't, then don't panic as it is better to not remember now than in the interview itself.

● Data structures overview: Yes, you do remember that you can solve a problem using Set (in C++) but what is the data structure in it? Yes, the interviewer might ask this to understand whether you know how the data structures work. Don't worry about learning how a binary search tree is balanced, but rather just know how it works (no one will ask you to implement it, hopefully).

● Pointers: What was the last time you implemented a linked list? Or trie? Or trees with Node* child? It is common not to know how to apply these, or maybe you forgot or worst case you are afraid of them. But relax, it is easy actually if you don't procrastinate. Give a day or two just to understand how this works and you will have that confidence back.

- CV: It is excellent mentioning all the courses in the CV and the projects that you did. BUT, can you make the interviewer believe its importance and all your efforts there? Some things in CV might be many months older, and you might not remember them correctly at first, so just go through it once again and be prepared for any questions regarding them.
- Communication: This is surprisingly, very important. You don't want to leave a wrong impression on the interviewer where you thought it was your confidence, but the interviewer took it as arrogance. Yes, it happens. Also, what if the interviewer doesn't get what you explained. Tough stuff! So for this, take some mock interviews and get an outsider view about your communication skills.
- Mock interviews: This is the final piece and a must. You will face a lot of pressure in the real interview, and it might result in you forgetting things and panicking. So, for this, you can simulate that experience with the mock interviews. These will also help you to understand your communication skills. And finally, you might get some fantastic ideas to improve your interview experience.

I guess these do cover all the major points needed for the pre-preparation for your interview. Now, coming to some tips for during the interview:

- Relax: Get proper sleep and food before the interview for sure. It will help you all the energy to give your best.

Validating question: Describe the question again in your words to make sure that you are on the same page as the interviewer, and you got the question right. You don't want to spend 10-20 minutes on the wrong question.

Ask for an example: Another step to be sure that you understood the question correctly is to ask for an example to the question.

- Validating inputs: Before jumping to solving the problem, think about the edge cases for the question, and do ask what input can you expect. At times expecting integer output might feel straightforward, but still, ask if it is okay to expect the output to be within the bounds of an integer. Ask whether the input array can be empty or not, and so on.
- Start with brute force: You might get the optimal solution at once, but they came to see your thought process. Start with a general basic solution to show how you improve that. Also, you don’t want

to show that you have just crammed up many solutions and this is one of them, so show them your thought process from start instead.

- Keep on optimizing: From your previous non-optimal solution, now start to move towards a solution with a better time and space complexity (yes, even space complexity matters!). Don't panic if you feel you are stuck, write some more examples, and try to see some connection between them. Feel free to take 5-10 minutes for that as you got at least 45 minutes.
- Give voice to your thoughts: Interviewer cannot read your mind and understand what trouble you are facing or what fantastic solution you will have. So, be transparent and tell them what you are thinking. It could be stupid too, but won't it be impressive that you finally got an excellent solution at the end while you were stuck at something stupid at the start? This will also help the interviewer to give you some hints if you are going in the wrong direction.
- Hints are not bad: The interviewer gave you a hint, so you failed! Not true at all. You are not expected to be perfect and have the most optimized answer for every question directly. So, take it constructively and try to use it to get the job done. It will not mean that you failed in any way.
- Explain in layman terms: You said that you will use a set as you code in C++, but the interviewer might not know it and might expect something like AVL tree or red-black tree (No, they won't ask you to implement them). Different languages might have different implementations, so try also to mention the fundamental data structure behind the inbuilt structures you are using. (if needed).
- Coding: You cleared the part to get to the solution, and now finally, you have to code. The interviewer will read the code, and they must understand what you are doing, and so the best coding practices should be followed. Start with writing a skeleton of the function for the question first and ask if it’s okay. Keep the code clean. Reuse the code. Have variable names that directly imply their use. Keep code properly indented.
- Mention the final time and space complexities below the code. (it looks good)
- Dry run on an example: Mention the interviewer that you will dry run on some example, so to check it works correctly. This helps you explain to the interviewer how your code works. And also you can find some edge cases you missed or some mistakes you made. You can also go a dry run on more examples to check for any edge cases left.

After all this, patiently wait to get an offer from the company 🙂

I hope I covered all the major points and that you all find it meaningful and helpful. Do mention if I missed something, and I will add it in the same post. Wishing you all the very best for the interviews.

- https://medium.com/@ish.ranga11/coding-preparation-endgame-e7d866958e6b