

## RESOURCES FOR INTERVIEW PREPARATION

1. InterviewBit
2. Cracking the Coding Interview
3. Codeforces
4. Hackerrank
5. Hackerearth

To crack a great job that you have been dreaming of, the most important step is to nail the Interview. In Software Engineering, the interview procedure is pretty much standardized across most companies and comprises of the following phases:

1. Coding Test: the purpose of the Coding Test is to filter out the bad candidates who certainly do not have much idea of Coding or general Computer Science.
2. Interview Round 1( Technical round) : the objective of the Interview Round 1 is to assess the basic programming and problem-solving skills of the candidate.
3. Interview Round 2( HR round) : this is the final round to confirm that the candidate is hired.

## CODING TEST

Important topics to be covered to clear coding test

1. Arrays/ vectors/ lists
2. DP
3. Trees and graphs

Programming Language preferable : C++/ Java

For Coding Tests, keep the following points in mind:

1. Right at the beginning, skim through all the problems quickly to get an idea of the easy ones and difficult ones.
2. Start with the easiest problems first. Solving it will boost your confidence and you'd be able to do much better in other problems.
3. If you get stuck in a particular problem for more than 5 - 10 minutes, skip it and move to other problems.
4. Use the library functions heavily to arrive at a solution faster. No need to rewrite something which is already available as a part of the programming language.

Note : If you are finding difficulty in optimizing your solution, always write your code in Brute Force first. Some companies have very weak test cases which passes with simple Brute Force solution.

## TECHNICAL ROUND

In interview rounds, you are usually sitting 1 - on - 1 with an interviewer who would be asking you questions related to various topics. Questions could range from your likings and dislikings to the projects on your resume. You should, therefore, be thoroughly prepared for the same. Aim to prepare the following:

1. Projects on your resume: the interviewer may be interested in certain projects on your resume and they would certainly want to know more about the project. In particular, they would be interested in knowing your contributions to the project. Therefore, you should be absolutely comfortable explaining your projects. The interviewer may ask you about the tools and technologies you used and you should also be familiar with them. For instance, if you did a Java-based project, the interviewer may ask you general questions about Java like Object Oriented Programming in Java, garbage collection in Java, etc.
2. Interview problems: be prepared with 1 - 2 medium to tough level problems in which you would be asked to find a solution, explain it to the interviewer and implement it in the programming language of your choice. Most interviewers don't really care which programming language you are using for the implementation, even if they are not very familiar with the language. However, C, C++, Python and Java are the recommended ones since most interviewers are familiar with at least one of these languages and so, can correlate better. Remember - your relative performance matters far more than your absolute performance. Most companies compare candidates and then choose the best one, rather than creating an absolute benchmark.
3. General questions: some interviewers may ask you general questions related to you, your family background, your hobbies, likes and dislikes, etc. Be frank and open and do not hesitate. The objective of such questions is not to judge you, but rather to give you an opportunity to stand out of other candidates. For instance, if you talk about something interesting that the interviewer remembers, it would be a good plus point for you.

Often at the end of the interview, you may be asked a simple question - 'do you have any questions to ask?' Now, this is often quite tricky to answer. Here are some recommended answers:

1. May I know your inputs about the areas where I could improve myself?
2. What are some of the most challenging problems on which you work?
3. What tools and technologies do you use at the company to perform X?

Generally, it is a bad practice to ask the interviewer 'how did I perform in the interview?'. Remember, most companies evaluate you relative to other candidates and so, the question itself is irrelevant.