Intern 411

Updated: 26/06/2018

Disclaimer: This document was made and maintained by the members of the Web Enthusiasts' Club, and the prep advice is more inclined towards software tech companies.

Coding Prep

- InterviewBit - Highly recommended, a go-to site for preparation and will single handedly fulfill all your preparation needs. Contains various Coding/DSA questions asked in previous interviews to candidates. Allows you to get a feel for how well you know your stuff. Requires basic DSA knowledge, recommended to read a bit of the initial few basic chapters of Cracking the Coding Interview before attempting this. Really well organised website as each question comes with a set of hints, solution approaches and code incase you are stuck.

- GeeksForGeeks - Contains almost every question ever asked in an interview. Guides you through the solutions and helps you see how to optimize solutions. Use this after solving a question on InterviewBit to have a look at alternate solutions. It also has articles on various DSA/OS concepts in case you're not very clear on certain parts. A suggestion from us - **Do not try to mug up the coding solutions from this website, but just use it as a guide.** Memorising solutions from this website might cost you a lot as the interviewers are well experienced in

detecting if you have heard the question before and also, it's very likely that you miss critical parts of the solution while coding it. Interviewers are also known to twist the questions before asking them to you.

- Cracking The Coding Interview - Not a website, but a top-notch book. InterviewBit basically contains most of the questions in this book. In addition, it has a lot of helpful tips and guides in approaching interviews. Helps a lot if you're starting from scratch.
- HackerRank/HackerEarth - Helps you get used to the feel of the coding tests, and allows you to gauge how comfortable you are coding. We wouldn't recommend you "study" or "practice" from these sites though, they're just the major platforms used by companies.
- The below two links are to sample tests from hackerearth and hackerrank respectively. These platforms ARE used by companies. So make sure you're familiar with the format. The questions in the sample test are similar in level to those asked in the first round, so try to do your best!
    - https://www.hackerearth.com/challenge/test/programming-practice-challenge/
    - https://www.hackerrank.com/tests/sample

Recruitment Format

- First round - The first round of a company is a coding/aptitude round which will have either DSA oriented questions or MCQ based aptitude/OS questions or both. Typically, this round will have 1 - 3 coding questions. The coding questions will generally have the highest priority, so try to solve those first, if possible.
- Intermediate round - Some companies have a group discussion round or code-on-paper round
- Final round(s) - The final round(s) are the technical interviews. These will consist largely of DSA questions like the ones found on InterviewBit. There may also be a few OS questions on topics like threading, virtual memory and deadlocks. Be prepared to answer the typical HR/semi-HR questions like "Why should we hire

you?", "Tell me the most complex bug you've encountered", etc. Some companies have a seperate HR interview round before they make the final selection.

Rules and Recommendations

- Once you get an offer from a company, you are not allowed to apply/sit for any other companies on-campus. Sometimes this leads to situations where you'll have to choose between two companies to sit for their test.
- If you sign up for a company, you MUST attend the test and pre-rec talk.
- Be on campus by the first week of the fifth semester, and keep the whole month free, lots of companies will be appearing at random times.
- Keep a lot of hard copies of your resume along with formal wear ready at all times.

Resume Tips

- Your resume must contain your name, college, branch, degree, and year of graduation. It must not contain your picture or the college's emblem.
- Mention at least one project that you've done. You should know this project in-depth as the interviewers may ask very detailed questions on this.
- Recruiters prefer unique resumes. This does not mean you have to make it colourful and stylish. A black and white resume can still look unique if formatted appropriately.
- Try to use a standard format. As a final resort, just use a template. You can find templates on sites like LinkedIn, ShareLatex or Google Docs. The most popular template is the one by Gayle Laakmann McDowell (Author of the Cracking the Coding Interview) : https://careercup.com/resume.
- Make sure everything you write is honest.

List of important DSA topics for interviews

1) Asymptotic Analysis - best case, worst case and average case analysis
2) Linked Lists, Stacks and Queues
3) Sorting
    a) Bubble Sort, selection sort, insertion sort
    b) Quick Sort
    c) Merge Sort
    d) Heap Sort
4) Searching
    a) Sequential
    b) Binary Search
5) Graphs
    a) Basics - definition of graph, vertex, edge, degree, path, cycle, tree.
    b) Representation - adjacency lists and adjacency matrix.
    c) Traversals - BFS and DFS (very important)
    d) Minimum Spanning Tree - Prim's and Kruskal's
    e) Shortest Paths Algorithms - Dijkstra's, Bellman Ford, Floyd Warshall
    f) Binary Tree
        i) Definition
        ii) Traversals - preorder, inorder, postorder
    g) Binary Search Tree
        i) Insertion
        ii) Deletion
        iii) Balanced vs Unbalanced
        iv) Balanced - AVL , basic idea of Red Black tree can help
6) Disjoint set
    a) Basics
    b) Union by Rank
    c) Path Compression
7) Heaps
    a) Types - min heap, max heap
    b) Uses
        i) Sorting

   ii) Priority Queues
   c) Implementation
8) String Matching
   i) Naive
   ii) KMP
9) Bit Manipulation
   a) Basics bit operation - complement, AND, OR, XOR
   b) Using bits to save memory
   c) Going over all subsets using bit manipulation
10) Problem Solving Paradigms - VERY IMPORTANT
   a) Dynamic Programming - top-down and bottom-up;
   b) Greedy
   c) Backtracking

NOTE : For Operating Systems, Computer Communication Networks, and important C concepts like pointers, memory layout(heap and stack particularly) refer to GeeksForGeeks.

Additional Pointers

1) Language: MOST companies don't restrict candidates to a particular language. However, candidates with proficiency in C/C++/Java might have an edge over others. C++ users with proficiency in STL particularly have an added advantage. Refer to topcoder tutorials for a better grip in this aspect. https://www.topcoder.com/community/data-science/data-science-tutorials/power-up-c-with-the-standard-template-library-part-1/

Frequently Asked Questions (FAQs)

<Will be updated as and when a query is answered on the google form: https://goo.gl/forms/H3OzQWoyMwGU3dqU2>