Qubercomm Technologies

Drive date : 11/06/2018

Company Location : Taramani, Chennai

Package : Full time 3.6 to 5.5 LPA (fixed on your intern performance) + 3 month of intern with stipend

About Company : It mainly deals with IOT, Cloud computing, IEEE protocols etc... providing real time solutions for various problems in Industrial and Commercial sectors. Their well-known projects are NMESH, LOCATUM, Qubercast, Indoor mapping, IOT security. The company was started in 2015. Being a Start-up company, it has a strength of around 50 employees. The CEO is “Sudarshan Vasudevan” and company has two branches one in Chennai and other in USA.

Recruitment Process

Nearly 3 rounds where conducted as follows:

<table>
<thead>
<tr>
<th>Round 1</th>
<th>Aptitude (General, C, Java)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Round 2</td>
<td>Coding round (system or written)</td>
</tr>
<tr>
<td>Round 3</td>
<td>HR Round</td>
</tr>
</tbody>
</table>

Round 1 – Aptitude:

General - 20 questions
C - 20 questions
Java - 15 questions
**General Aptitude topics:**

Logical reasoning, logical thinking, odd one out, profit and loss, cubes, shapes, puzzles, population, fractions, percentages, distance, time etc…

**C questions:**

Mostly reasoning questions where aroused much from C. So, we were to write 2 to 3 lines of reasons for each question asked. Some topics are: Pointers, MACRO, Structure, String, snippets, static keyword, Union etc…

**JAVA questions:**

JAVA questions are multiple choice questions. It was easiest one among all three. The questions where from thread, JAVA keywords, OOPS concepts, JAVA classes, JAVA functionality etc…

**Round 2-Written coding round:**

**Instructions:** Since there was a lack of time. The second round was changed from Online-coding to written coding round. We were asked to write a clear code on a paper satisfying all test cases, edge cases... Though it was a language independent round, mostly preference was given to C language.

**Write code for the following inbuilt functions:**

<table>
<thead>
<tr>
<th>Functions</th>
<th>Parameters</th>
<th>Return type</th>
</tr>
</thead>
</table>
| Strstr(); | 1. mainString.  
2. substring. | In C, pointer to the string.  
In Java, index of the substring |
| Strrev(); | String | String |
| Strcmp(); | 1. mainString  
2. substring | Boolean |
Round 3 - HR round:

Instructions: Total of around three HR panel are been conducted subsequently. It took around 2 hours. Each panel contains question from your core subject, JAVA, C, Area of Interest and your first-round aptitude result.

The following question are asked in HR round:

1. Tell about Java’s inbuilt garbage collector and how it works? Can we invoke it explicitly? If three objects are created: one is recently created, second is older than the third, third is older than first. All three references are made null, which objects will be garbage collected?
2. How to implement all the functionality of doubly linked list in a singly linked list like, travelling in the reverse direction without using an extra pointer?
3. Do you know about Priority Queue? How does it work? How it is different from the normal queue? How its priority is being assigned?
4. Find the next number in the series: 1, 11, 21, 2111, ____?
5. Values for the Hexadecimals: FF, FC etc…
6. Use of transient keyword in Java. Where it is used?
7. Difference between Serialization and synchronization?
8. Difference between big endian and little-endian machine?
9. Program to find whether a machine is a big endian or little-endian machine?
10. How can we delete a node in a singly linked list when to pointer to that node is alone given without the head pointer of the list?
11. Given K, write a single line code to reset the kth bit in integer x.
12. State about Serial communication and parallel communication
13. Which communication protocol do you prefer for very close transaction of data?
14. When we have multiple master and multiple slaves which protocol do you prefer and why?
15. What is JVM, JRE, JDK? Which of these interpret with the generated byte code or .class file to convert it to machine level language?
16. What is Thread, multithreading, Thread life cycle, resume(), sleep()?
17. What is the time complexity of HashMap while searching elements? (best/average/worst)
18. How can we improve the time complexity of HashMap to O(log n) and up to O(1) even in the worst case?
19. On considering the data Structures like Arrays, LinkedList, Tree. Which do you prefer of your application and why?
20. Tell me about yourself?
21. Tell me about your project?
22. Imagine you walked towards the centre of a four-road intersection where you found a broken four-way indication board, how will you find your way out to your destination?
23. Difference between Structure and Union in C? if a data is stored in any variables in both, how do they differ from one another?

**Tips:**

1. The Interview process will take too long. So, have sufficient diet and drink adequate water before each round. (For my batch, the second round started at evening 6 pm and it took around mid-night 1 am to complete all 3 rounds and the results were announced around 1:30 am. So be ready to test your endurance).
2. Be Bold and Express your thoughts always.
3. Don’t panic. It will be much easier than you thought.
4. Even when you don’t perform well with any of the HR panel. Don’t worry bring your best in the remaining panels.
5. Be sure about your field of interest because most questions where aroused from it.
6. Explain clearly about your project you stated in your Resume.
7. Make sure you give a unique Self-Intro.
8. Whenever a question is asked in HR panel. First discuss about it clearly with the HR. Then try to solve it.
9. If you get stuck in middle of a solution discuss with the HR. Sometimes, he slips a clue to the solution. You can then proceed with your clues.
10. Use paper and pen thoroughly in HR panel.
11. During the written coding round, make sure you right clearly on your paper. Don’t strike and write your code. Make it a crisp and clear code.
12. Make sure you meet all the test cases, corner cases, edge cases while writing a code. Make it optimized and clear solution.