

b'

Amazon Interview | Set 111 (On-Campus)

- Difficulty Level : \nMedium
- Last Updated : \n20 Jun, 2019

Amazon visited our campus, these are the questions that I faced.

Online Round (90 minutes)

20 MCQs(Data structure, C, C++, OS, Probability).

2 programs :

1. [Given an array of integers. Segregate all the non-zero numbers at the beginning. Print the number of non-zero integers](#) and the minimum number of swaps required for these operations.

Eg. : I/p : 1, 0, 0, -6, 2, 0

o/p : Number of non-zero integers : 3

Minimum number of swaps : 2

2. [Given a text txt\[0..n-1\] and a pattern pat\[0..m-1\], write a function search\(char pat\[\], char txt\[\]\) that prints all occurrences of pat\[\] and its permutations \(or anagrams\) in txt\[\]. You may assume that n > m.](#)

E.g. : I/p : txt[] = \xe2\x80\x9cBACDGABCD\x9d pat[] = \xe2\x80\x9cABCD\x9d

o/p :0,5,6

1st PI

1. About 10 minute discussion on my project on Cloud Platforms and Big Data Analysis in Yahoo SDC. I asked him some doubts and he answered them clearly.
2. [Given a string of digits, output all the dictionary words they can represent.](#) (Basically smart dial algorithm of an android phone). Each digit represents the corresponding characters on a cellphone as follows.

1	ABC 2	DEF 3
GHI 4	JKL 5	MNO 6
PQRS 7	TUV 8	WXYZ 9
*	0	#

e.g. : I/p : 3323 O/p : DEAD , DEAFFirst, he asked me how I would store the dictionary. When he was satisfied by my approach, he asked me to write a recursive function that would do the job if a dictionary is given as an input.

3. Then he gave me a class design. He went on changing his requirements and I was asked to add support for them in my implementation. At first, his requirement was to draw a rectangle and a

circle. Then he went on adding requirements and finally it was to draw different specializations (like curved-edges, skewed-edges, dashed-lines) of rectangles, triangles and circle. He gave a constraint that support for a new specialization can be added easily in my design.

2nd PI

1. [Given, the starting and ending time of different meetings, what is the minimum number of conference rooms that will be enough to accommodate all of them.](#) He asked me to code it. E.g,
:

I/p : Meeting I \xe2\x80\x93 8:30 \xe2\x80\x93 12:50

Meeting II \xe2\x80\x93 10:15 \xe2\x80\x93 11:30

Meeting III \xe2\x80\x93 11:45 \xe2\x80\x93 1:30

O/p : 2

2. [Implementation of a doubly-linked list using a single pointer.](#) I didn't know the answer at that time. He gave me a few hints and I was able to solve it. Then he asked me to code it.

3rd PI

1. Given an array of integers. This array denotes our own ascending order of the elements. So if the array is {2,3,1,4}, by mathematics we can say that $2 < 3 < 1 < 4$. Given another array, sort this new array in our ascending order. Let's say the new array is {1,2,4,3,5,4,9,2}, output will be {2,2,3,1,4,4,5,9}. Note that since 5 and 9 do not occur, they are sorted by actual ascending order at the end.
2. Return the [left-view](#) and [right-view](#) of an n-ary tree in a single traversal.

4th PI

1. 10 minute discussion about my internship in Cloud Platforms team in Yahoo SDC. He asked me a few HR questions like what do you think was your biggest screw-up in college life, etc.
2. Find power(a, n) iteratively without extra space in $O(\log n)$ time. He gave me a lot of hints. But unfortunately I couldn't solve it.
3. Pre-order traversal of a binary tree without using recursion.

Fortunately, I was selected as an FTE in Amazon.

Suggestions

I would like to give a few suggestions to all the upcoming candidates :-

1. Firstly, mugging up of codes will never take you far. If you find a problem that is solved in geeksforgeeks, rather than blindly mugging up the code, read the description properly. Try to understand what is being done and why is it being done. Try to understand each and every line of code. It is naïve to expect common known questions in interviews. But, if you can understand the approach of the codes solved in geeksforgeeks, you can use it to solve a plethora of problems.
2. In an interview, always think out loud. Explain what you are doing and why you are doing it. If you solve a problem within a few minutes without explaining your steps, the interviewer will assume (perhaps not incorrectly), that you have mugged up the code. This creates a very bad impression.
3. Follow proper etiquettes. Shake the interviewer's hands while entering and while

leaving the room. Also, try to ask proper questions to the interviewer when he gives you the chance to do so. These are the small things that set you apart from the other interviewees.

4. Lastly, never be too tensed. Always remember, rejection is a part of life. Luck matters a lot in job interviews. So don't be disheartened if you are rejected. You can always apply later.

All the best of luck to all of you for your upcoming interviews.

If you like GeeksforGeeks and would like to contribute, you can also write an article and mail your article to contribute@geeksforgeeks.org. See your article appearing on the GeeksforGeeks main page and help other Geeks.

[All Practice Problems for Amazon !](#)

My Personal Notes

Add your personal notes here

Save