

Amazon Interview | Set 21

- Difficulty Level : [Medium](#)
- Last Updated : 14 Jun, 2019

Recently I have gone through 10 rounds of interviews of my dream company Amazon for SDE1 and I got selected. My only resource and library for preparation was GeeksforGeeks, which is the best website for tech preparation, if you aim for big shots like Amazon, Microsoft, etc. I was not too confident to attend interviews. The interview experiences blog gave me courage and coding experience to crack all rounds. Please do follow this site for every update, and suggest your friends to follow if they try for job change and if they aim for big companies.

I am sharing my interview experience, which will help others.

Total rounds : 10

1 online written test + 4 telephonic + 5 F2F screening rounds.

In F2F rounds, for every problem complete executable code + algorithm will be expected.

In all rounds , best solution will be expected (with less complexity)

Online Written Test:

1. Find maximum frequent numbers in an array. If there are more numbers with maximum frequency, they display all numbers in ascending order. Ascending order is important.

2. Two numbers represented as linked lists. (Bigger than integers). [Return a linked list which is the product of the given two linked lists.](#)

1->2->3->Null (123)

2->3-Null (23)

O/P: 2->8->2->9->Null

1st Phone Screening (45min):

1. [Find a linked list has circle in it, If it has loop, find origin of the loop.](#)

2. [In Linked list, Node has two pointers, one points to next node, other points to arbitrary node in the linked list. Write a function to return a new list which is clone of the given linked list.](#)

3. [An array is sorted and rotated by k times. Find an element in an array.](#) (efficient and logarithmic time solution is expected)

2nd Phone Screening(45min):

1. [Two strings s1,s2 are given as input. Remove characters present in s1 which are there in s2.](#)

2. How to find number of subsets in a set.

3. [Searching an element in 2D matrix which is sorted in row wise and column wise.](#)

1 2 3 4

2 3 4 5

3 4 5 6

4 5 6 7

5 6 7 8 Find an element in it.

4. Difference between [merge sort](#) and [quick sort](#). How do you improve quicksort (think about in selecting pivot element).

5. Give a scenario for quick sort in worst case.

6. How to store a set in memory, what ADTs do we use and what are tradeoffs for each ADT.

3rd Phone Screening(35min):

1. [N-Petrol bunk problem](#): There are n petrol bunks located in a circle. We have a truck which runs 1 km per 1 liter (mileage 1kmpl). Two arrays are given. The distances between petrol bunks are given in one array. Other array contains the no of liters available at each petrol bunk. We have to find the starting point such that if we start at that point, you would be able to visit entire circle without running out of fuel. Initially truck has no fuel.

2. [Reverse linked list in groups of size k.](#)

I/P: 1->2->3->4->5->6->7->8->Null k = 3

O/P: 3->2->1->6->5->4->8->7->Null

4th Round(35min):

1. Algorithm to construct a tree given [Pre](#) and [In order](#) traversals.

2. [Inorder successor of a tree.](#)

3. Threaded binary tree(inorder without recursion)

F2F Round 1 (with Hiring manager 60+ min):

1. Tell me about yourself and Projects you worked.

2. About the most critical situation in the project you went through. How you did it. (he needs complete explanation of the scenario)

3. If he gives the same scenario as an interview question, how will you improve code quality and its complexity.

4. About SDLC you followed. Which one do you like and why.

5. Do you have any questions to ask (very important one- ask something about projects they work, etc. Good sign)

F2F Round 2 (with Developer 60 min):

1. Tell me about yourself.

2. [Zigzag traversal of binary tree.](#) (more optimal solution is expected from you). Complete code should be written and they will check later.

3. A robot is there in 2D space, which can move to its left direction. You are given with an array which are moves of robot, which starts from origin(0,0). Find the rectangle covered by it.

I/P: { 2,3,4,5,6,1,3,5,5} starts at (0,0)

O/P: rectangle points: (-4,4) to (4,-2)

4. Casual discussion about hiring process.

F2F Round 3 (with Developer 50min):

1. Data structure which does insertion, deleting latest item, find min, find max in $O(1)$ time. (Gave hash, 2-D, linked list, many .. He impressed lot here)

2. [Vertical sum of a tree.](#) (Column wise sum \xe2\x80\x93 can find same one in geeksforgeeks)

3. Find n-th digit in the continues sorted stream of data.

I/P: {1,2,3,4,5,6,7,8,9,1,0,1,1,1,2,1,3,1,4,1,5,1,6,1,7,1,8\xe2\x80\xa6\xe2\x80\xa6\xe2\x80\xa6 infinite} n =28

O/P: 28th digit

Complexity analysis of all the above.

F2F Round 4 (with Developer 45 min):

1. [Print matrix spirally.](#)

1 2 3 4 5

6 7 8 9 10

11 12 13 14 15

16 17 18 19 20

21 22 23 24 25

O/P: 1 2 3 4 5 10 15 20 25 24 23 22 21 16 11 6 6 8 9 14 19 18 17 12 13 18

2. [Write a function to check syntax of opening and closing braces whether they are proper or not.](#)

3. Same question if you have k types of braces(\xe2\x80\x98(\xe2\x80\x98

\xe2\x80\x99[\xe2\x80\x98 \xe2\x80\x98{\xe2\x80\x98 ,\xe2\x80\xa6. K types) All are of same priority.

4. Same question if you have k types of braces(\xe2\x80\x98(\xe2\x80\x98

\xe2\x80\x99[\xe2\x80\x98 \xe2\x80\x98{\xe2\x80\x98 ,\xe2\x80\xa6. K types) All are of diff priority.

5. Print all valid combinations of k number of pairs of braces.

6. Return count of above combinations without using algorithm for printing them.

7. Memoziation \xe2\x80\x93do you know about it. Explain me.

F2F Round5 (Bar Raiser round) 60min:

1. Leader ship principles followed by Amazon

2. About project.

3. Why you are leaving prev company, What will stop you there.

4. Set of strings are given in a dictionary order. The problem here is order is not as our alphabetical. It may be different. C may come before a,b, x may come before d,c. etc. You will have to find the order of characters by using given input. ([topological sort](#) \xe2\x80\x93 complete code is required to write)

5. [Binary search tree into Sorted doubly linked list](#) (Expected Inplace algorithm)

Things to keep in mind:

1.For every problem, give one simple solution first (may have more time complexity) and think for optimal solution.

2.Write a code in clear manner. It should be understandable without your explanation.

3.In a position to tell complexity for code you are going to write.

4.First tell the algorithm or approach and proceed with writing code.

5.Do not hesitate to ask for clarification. They will impress.

That's all from my side. Best of luck.

Thanks again for GeeksforGeeks, a lovable website for techies.

This article is compiled by **Ranganath**. Many Many congratulations to Ranganath for his selection. 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