

Amazon Interview Experience | Set 227 (On-Campus for Internship and Full-Time)

- Difficulty Level : [Expert](#)
- Last Updated : 02 Jul, 2019

This year our college kicked off placement season with the Amazon. We got the job description(JD) before the placement process. We got good news that Amazon have increased total CTC. With this good news it also added tension and pressure.

Day 1: Coding and aptitude round

Coding round was at 5pm so from morning I had revised very basic things like sorting techniques, B Tree, AVL tree, finding the power set and some DP questions.

Online coding exam was of two hours and was on HackeRank. There were 20 multiple choice questions and two programs. Multiple choice questions were GATE type questions mainly from OS, data-structure , algorithm, database and C. My plan was to attempt these questions as early as possible and starts programming but questions were lengthy and it took 30 mins. After 30 mins there were 3 to 4 questions which I skipped and started programming.

Program 1: Anagram

[Two words are said to be anagram when one word can be converted into another word by shuffling/rearranging characters. Given a set of words find the number of anagram for each word in given set of words.](#)

Program 2: [Find diameter of binary search tree](#) in $O(n)$ complexity.

[Given the set of integer values, insert these values in the binary search tree and find diameter of the tree.](#)

Both the program were easy. I solved program 2 correctly in $O(n)$ and second program in $O(n^2)$. Result was on next day so all of us were in tension. I could sleep only 3 hours :p.

Day2: Interview process

Interview Round 1:

Question 1: First question in first round of interview was how to [insert node in binary search tree](#) using recursion and without recursion. He asked me to write the code of both.

Question 2: [Given a binary tree how to find the least common ancestor of two nodes in the tree.](#)

Question 3: [Given a binary search tree find the least common ancestor of the two nodes in tree. He asked me to write the code of this program as well.](#)

Question 4: He asked about projects it has taken almost 10-15 mins.

Time: 45 mins.

Interview Round 2:

Question 1: He asked me to tell about my self. During introduction I mentioned about my previous company's project. He asked next question about that project. I had worked on that project almost 2 years so I could explain him that project in detail. He asked to write the code for the Wait And Signal method of the counting semaphore.

Question 2: [Given an array with -ve as well as +ve elements find max continues sum.](#)

This question was easy.

Question 3: Given a singly circular link list containing both +ve as well as -ve numbers find max continues sum. I explained him $O(n^2)$ algorithm but he was asking $O(n)$. I was not getting answer of this question. I worked on this for 4 to 5 mins but I was not getting perfect answer so he gave hint by drawing all elements on the circle. As soon as I saw circle I got answer, I told to break the ring from most -ve number and then use normal algorithm which was in 1st question. He was not sure about this algorithm he tried to break this by using 4 to 5 example but he could not. He was happy after this answer because might be he had another solution in his mind.

Question 4: [Given prices of stocks per day find maximum profit you can earn by buying and selling shares.](#) You can make only one transaction.

This question is easy and from geeksforgeeks.

Question 5: [Given prices of stocks per day find maximum profit u can earn by buying and selling shares. You can make multiple transactions.](#) He asked me to write the code of this program.

Time: 1 hour 10 mins.

Interview Round 3:

Question 1: He saw my resume and asked me which project I find most difficult. I told same office project. He asked me which part of that project I find most difficult. I told part related to the scheduling. He asked me basic questions related the scheduling and synchronisation.

Question 2: What is copy constructor in CPP

Question 3: Explain the [LRU cache technique and how to implement it.](#) I gave various approaches like array, queue , stack, Link list. finally He agreed on Link List with hashing approach. All other approaches were $O(n^2)$ solution but Link List with hashing approach is $O(n)$. He was happy with that answer.

Link-List with Hashing approach:

Size of LinkList will be size of cache. Data in list represent page number of cache. Before running algorithm we will build Hash Table. Key of HashTable will be page Number and value will be address of the node in link list which contains key. One pointer to Link List represent the most recently used page(Front of Link List) and one pointer will point to least recently used page(End of Link List). When new page come then we will get node containing page number directly from hash table in $O(1)$. After getting that node move that node to the front of link list which represent most recently used page. If the new page not present in the Hash Table then update the value in page pointed by least recently used pointed and move that page to front. and also update key in the hash table.

Here we are building hash only one time and not modifying its values after that.

Time: 50 mins

Interview Round 4:

Question 1: He asked about projects. I explained him my project in details which took around 10-12 mins. During explanation he was asking like why u have used this approach why not other and like that.

Question 2: Given n Runways and m Planes. How you will efficiently design this scheduling problem so that there should not be collision. I explained this by giving view of CPU scheduling problem and semaphore.

Use priority based Scheduling where priority can be departure time and emergency. Each plane use runways as below:

```
\r\n      Plane Pi:\r\n      Wait(S)\r\n      Take off(M.E.)\r\n      Signal(S)\r\nS=n
```

Question 3: This was easy question. [Given matrix which is sorted, row wise as well as column wise like below:](#)

```
\r\n      01 04 07 09\r\n      11 23 45 46\r\n      51 56 59 63\r\n      70 73 75 79
```

and key. Find the position of key in given matrix efficiently. I explained the solution which is in $O(\log m)+O(\log n)$. He asked me write the code this problem.

Question 4: He asked very basic questions of database. After that he asked me what is eventual consistency. I didn't know answer of this so I told that I don't know, He again asked that do u know about NoSQL I said no.

Time: 45 mins

Thanks to my parents, teachers, friends and GeeksForGeeks.

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\

narrow_drop_up

Add your personal notes here

Save