

Top Most asked Amazon Interview Questions

Once the CEO of Amazon, **Jeff Bezos** said, "I'd rather interview 50 people and not hire anyone than hire the wrong person."

You can understand the quality of the recruitment process of Amazon by the statement of their CEO. It doesn't mean Amazon wants you to fail. Instead, it simply means they only want to hire people who prove themselves worthy.

In this article, we will explain the interview process of Amazon, the technical topics you have to prepare for discussion, and how to answer the interview questions asked by the interviewer.



Amazon's Interview Process

There will be a total of 5 rounds (1 online coding test + 4 interview rounds) in the whole process.

Round 1 (Online coding test Round)

This test contains two coding questions for which you have to be given 2 hours to solve. You also have to submit the approach used for solving in words and your algorithm's time and space complexity.

Round 2 (Technical Round)

You will get 2 to 4 coding questions that focus on basic problem solving and data structures in this round. The questions may vary according to your experience level. The less experienced you are, the more number of coding rounds you will face.

Round 3 (Design Interview Round)

In this round, the candidate may ask a question on high-level design architecture for real-life products and OOPS-based design of components. The company may skip this round for entry-level openings for the software engineering role.

Round 4 (Hiring Manager Round)

This hiring manager round is used to test for cultural fit based on attitude and previous work experience. In this round, the candidate is asked about the general HR questions and the previous company's experience. The interviewer may discuss your projects in the previous company and ask some behavioral questions based on those experiences.

Round 5 (Bar Raiser Round)

This is the last interview round called the optional bar raiser round, which combines all of the above. Here, a senior engineering manager can ask about your previous works and again ask some behavioral questions. The idea is to judge if you are technically better than an average person in a particular Amazon team.

Technical topics you must prepare for Amazon's Interview.

Programming Language

Amazon doesn't ask for specialization in any specific programming language before attending an interview for a tech position. But you must be familiar with the syntax of languages such as C/C++, Java, Python, C# or Ruby. You should also know how memory management works or the most commonly used collections, libraries, etc.

Data Structures

This is the most important technical topic you should prepare for. Most of the work Amazon do involve storing and providing access to data in efficient ways. So you must have a good knowledge and understanding of the inner workings of common data structures and compare and contrast their use in various applications.

Algorithms

You must have the knowledge and a good understanding of the most common algorithms such as traversals, divide and conquer, breadth-first search vs. depth-first search etc. and make sure you understand the trade-offs for each. Please focus on the learning of implementation strategies of different classes of algorithms rather than memorizing them.

Coding

You may be asked to write syntactically correct code rather than pseudocode. Try to write code without an IDE. It would be best if you practiced coding with a pen and paper. It will be helpful at the time of the interview. The company prefers the candidate who writes scalable, robust and well-tested code. These are the main evaluation criteria for your code.

Object-oriented Design

Object-oriented design is the best practice to write good software because good software design is critical to success. Your software needs to be extensible and maintainable. So, you should have a working knowledge of a few common and useful design patterns and know how to write software in an object-oriented way.

Databases

The company prefers candidates who have a good knowledge of the non-relational database. Amazon has developed Amazon Web Services such as DynamoDB so that their developer community can easily leverage the benefits of non-relational databases. So, it is good if you know relational and non-relational databases.

Operating Systems

The candidate must be familiar with some Operating System topics to enhance the performance of the code. (e.g. memory management, processes, threads, synchronization, paging and multithreading).

Internet Topics

You must know the fundamentals of the internet, such as how browsers function at a high level, from DNS lookups and TCP/IP to socket connections.

Basic Machine learning and Artificial Intelligence

You must have basic knowledge of machine learning and artificial intelligence such as data-driven modeling, train/test protocols, error analysis and statistical significance. It would help if you visited your Machine Learning and Artificial Intelligence textbooks to prepare for these topics.

Some examples of technical Interview Questions

There are two integer arrays A and B, of size N given. There are N gas stations along a circular route, where the amount of gas at station i is A[i]. You have a car with an unlimited gas tank, and it costs B[i] of gas to travel from station i to its next station (i+1). You begin the journey with an empty tank at one of the gas stations.

Return the minimum starting gas station's index if you can travel around the circuit once, otherwise return -1.

You can only travel in one direction. i to i+1, i+2, ... n-1, 0, 1, 2.. Completing the circuit means starting at i and ending up at i again.

The question says that we have two arrays of int. The first one is cost, and another one is gas. We have a car and need to travel to "n" places where n is the above two arrays' length.

Now, according to the question, we can start from a point. Let's say 3, so that means to reach the next point, we need gas greater than the cost at 3.

See the example:

Suppose the gas array is: {1, 2, 3, 4, 5}

And the cost arrays is: {3, 4, 5, 1, 2}

Here, we can see that the 3rd term from gas is: 4 (Index starts from 0).

And the same for the cost is 1, so to go for the 5th position our gas should be greater than the cost. As we can see in this case, it is as $4 > 1$.

Here you see that this much gas is used to go to the next point. Even after reaching there, you will leave with 3 gas as $4-1=3$.

So, at the next point, you will initially have 3 units of gas.

Now, the question is asking from which point we should start so that we can traverse all the points.

Now, let's take the gas array and cost array as following:

gas = {1,2}

cost= {2,1}

Now, let's start from the first element that is index 0.

So, the gas in our car after first travel will be = $\text{gas}[0] - \text{cost}[0] = 1 - 2 = -1$.

Since it would be a negative number, that means we can't traverse using the 0 index.

Now, let's start from the second element, that is the index 1;

The gas in our car after first travel will be = $\text{gas}[1] - \text{cost}[1] = 2 - 1 = 1$.

We have left with 1 unit of gas, after that we will come to the 0th index so we will be having = $1 + \text{gas}[0] - \text{cost}[0] = 1 + 1 - 2 = 0$

So, that means we will be able to complete our journey from the 1 index.

So, there are two ways to solve this problem:

The first solution would be to use two loops and check whether the index exists but will take higher time and $O(n^2)$ time complexity.

Another way to solve this problem is to follow the steps given below:

- Take a variable **start** and initialize it to 0, and another variable **end** and then initialize it to 1.
- We will loop from start to end and keep increment end until it reaches the same value as the start. For example, if our array is of length 5 then: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 0$. In that case, we will know that start is the index from where the answer comes from.
- For storing the current gas in our car, we will use another variable curr and then if at any point of time, if curr becomes negative, that means that index cannot be the starting point. In that case, we will increment the start by 1.
- Here, the main problem is that we know how to move forward in an array. I mean, we know how to increment the current pointer. But how do we get back? Here, the pointer goes like this: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \dots$ and so on. It won't come back like: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 1$.
- To solve this problem, we have to use a simple trick. Instead of increment the value like : $n = n + 1$, we will use : $n = (n + 1) \% \text{length}$.
- In that case, when the pointer gets to the end, it will again start from zero.

See the complete code for the solution of the above problem:

```
public int canCompleteCircuit(int[] gas, int[] cost) {
    if(gas.length==0){
        return -1;
    }
    if(gas.length==1){
        return gas[0]-cost[0]<0?-1:0;
    }
    //First we will declare the variables which were start, end
    int start = 0;
    int end = 1;
    //Initially the first value of curr will be the cost incur to go to //the next location
    int curr= gas[0]-cost[0];
```

```

//Running a loop till start meets end, meaning till our cycle gets //completed
while(start!=end) {
//If the curr value is negative, then keep incrementing the start //till it became positive
    while(curr<0 && start!=end) {
//We need to remove the value of current start that we have added //before
        curr= curr- (gas[start]-cost[start]);
        start = (start+1)%gas.length;
//If we again come back to start, that means no solution exist and //we return -1
        if(start==0)
            return -1;
    }
//Adding the gas that we got after visiting the current index
    curr+= gas[end]-cost[end];
    end = (end+1)%gas.length;
}
//If in the end, curr became negative that means it was impossible //to traverse all the location and hence return -1
if(curr<0)
    return -1;
//If that's not the case, then we got the answer.
return start;
}

```

Find the K largest elements from a big file or array. / Write an efficient program for printing k largest elements in an array. Elements in the array can be in any order.

Suppose we have an array [10, 13, 2, 69, 40, 29, 70] and we have to retrieve the largest 3 elements, i.e., k = 3, then our program should print 70, 69 and 40.

There are several methods to solve this problem:

Method 1 (Using Sorting Method)

- First, we sort the elements in descending order in $O(n \log n)$
- Now, print the first k numbers of the sorted array $O(k)$.

See the complete Java code for the solution of the above problem:

```

import java.util.Arrays;
import java.util.Collections;
class GFG {
    public static void kLargest(Integer[] arr, int k)
    {
        // Sort the given array arr in reverse order
        // This method doesn't work with primitive data
        // types. So, instead of int, Integer type
        // array will be used
        Arrays.sort(arr, Collections.reverseOrder());
        // Print the first kth largest elements
        for (int i = 0; i < k; i++)
            System.out.print(arr[i] + " ");
    }
    public static void main(String[] args)
    {
        Integer arr[] = new Integer[] { 10, 13, 2, 69, 40, 29, 70 };
        int k = 3;
        kLargest(arr, k);
    }
}

```

```
}
}
```

Output:

```
70 69 40
```

Python code for the above problem:

```
def kLargest(arr, k):
    # Sort the given array arr in reverse
    # order.
    arr.sort(reverse = True)
    # Print the first kth largest elements
    for i in range(k):
        print (arr[i], end = " ")
# Driver program
arr = [1, 23, 12, 9, 30, 2, 50]
# n = len(arr)
k = 3
kLargest(arr, k)
```

Time complexity: $O(n \log n)$

Method 2 (Using Bubble sort k times)

- Modify Bubble Sort to run the outer loop at most k times.
- Print the last k elements of the array obtained in step 1.

Time Complexity: $O(nk)$

Note: Like Bubble sort, we can use other sorting algorithms like Selection Sort to get the k largest elements.

Method 3 (Using temporary array) K largest elements from arr[0..n-1]

- Store the first k elements in a temporary array temp[0..k-1].
- Find the smallest element in temp[], let the smallest element be min.
- For each element x in arr[k] to arr[n-1]. $O(n-k)$
- If x is greater than the min, then remove min from temp[] and insert x.
- Then, determine the new min from temp[]. $O(k)$
- Print final k elements of temp[]

Time Complexity: $O((n-k)*k)$. For sorted output, it will be $O((n-k)*k + k \log k)$.

Method 4 (Using Max Heap)

- Build a Max Heap tree in $O(n)$.
- Use Extract Max k times to get k maximum elements from the Max Heap $O(k \log n)$.

Time complexity: $O(n + k \log n)$

Method 5 (Using Min Heap)

- Build a Min Heap MH of the first k elements (arr[0] to arr[k-1]) of the given array. $O(k)$
- For each element, after the kth element (arr[k] to arr[n-1]), compare it with the root of MH.
- If the element is greater than the root, then make it root and call heapify for MH else ignore it. // The step 2 is $O((n-k) \log k)$

- Finally, MH has k largest elements and root of the MH is the kth largest element.

Time Complexity: $O(k + (n-k)\log k)$ without sorted output. If sorted output is needed then $O(k + (n-k)\log k + k\log k)$

Java code for the above problem:

```
import java.io.*;
import java.util.*;
class GFG{
public static void FirstKelements(int arr[], int size, int k)
{
    // Creating Min Heap for given
    // array with only k elements
    // Create min heap with priority queue
    PriorityQueue<Integer> minHeap = new PriorityQueue<>();
    for(int i = 0; i < k; i++)
    {
        minHeap.add(arr[i]);
    }
    // Loop For each element in array
    // after the kth element
    for(int i = k; i < size; i++)
    {
        // If current element is smaller
        // than minimum ((top element of
        // the minHeap) element, do nothing
        // and continue to next element
        if (minHeap.peek() > arr[i])
            continue;
        // Otherwise Change minimum element
        // (top element of the minHeap) to
        // current element by polling out
        // the top element of the minHeap
        else
        {
            minHeap.poll();
            minHeap.add(arr[i]);
        }
    }
    // Now min heap contains k maximum
    // elements, Iterate and print
    Iterator iterator = minHeap.iterator();
    while (iterator.hasNext())
    {
        System.out.print(iterator.next() + " ");
    }
}
// Driver code
public static void main (String[] args)
{
    int arr[] = {10, 13, 2, 69, 40, 29, 70};
    int size = arr.length;
    // Size of Min Heap
    int k = 3;
    FirstKelements(arr, size, k);
}
```

Output:

70 69 40

Most asked HR questions at Amazon

1) Why do you want to work at Amazon? / Why have you chosen Amazon?

This is one of the first standard questions you will be asked at the HR interview session. But most people don't have a great answer. We know that there is no right answer to this question, but the candidate must avoid general statements like "this is a great company." If you want to say that, it is okay, but first, you have a great explanation to prove your point.

Here, we are providing some standard answers according to different career perspectives. Following are some best possible answers:

An Engineer answer:

I would like to work at Amazon because I think I can help the company open a new business opportunity in the area of home automation. (Here, you can specify the more specific niche exactly what you want to apply in home automation). Since the Alexa division is working on home automation, I can extend my Master's research project regarding home automation, smart meters and Big Data. [Here, it would be better if you explain your specific area of work and the idea you have in this area.]

A Finance Manager's Answer:

I would like to work at Amazon because it is one of the leading global online retailing companies with 136 billion USD Net Sales and is 12th on the Fortune 500. It is a fast-growing company and a pioneer in the internet retail sector, which has doubled its sales in the past four years. This is a place I would like to work because I know I would have a chance to utilize my capabilities. [Here, you can specify the most challenging part of your previous job and something you know you will have to do in the new job.]

A Sales Manager's or Product Manager's Answer:**I want to work for Amazon for the following reasons:**

- The first and most important reason I want to work for Amazon is that I see the possibilities for creating and marketing products that have a huge impact on society. I have great experience in the consumer goods industry working in leadership roles in sales and marketing, so I can see that Amazon is a very interesting place to work where the product quality and the delivery speed is exceptional. They have a great chance of succeeding their competitors.
- I always enjoy diving deep and staying connected to details at work. I see Amazon is a very data-driven company, so I could use data if I worked there.
- In my previous FMCG organization, I have developed a passion and curiosity to learn and understand consumer behavior and creating marketing strategies to delight customers. It is really about putting yourself in the shoes of the consumer to win their trust and confidence. So I believe Amazon will provide me a great chance to apply my skills on a broad scale.

2) Do you know our CEO? How do you pronounce his name?

This question is asked to check whether you have researched this company well or not. Generally, people know about tech giants like Amazon and their CEOs, but this question is asked to check how the candidate pronounces the name. The CEO of Amazon is **Jeff Bezos** since 1996. It is pronounced as "Bay-zohs," not "Bee-zos".

3) How would you solve problems if you were from Mars?

This question is asked to check your creativity. The interviewer wants to see if you can think out-of-the-box. You can answer this fancy question interestingly, develop a creative solution to a customer problem, improve an internal process, or make a sale via an innovative strategy.

4) Tell the story of the last time you had to apologize to someone.

This question is asked to check whether you admit your mistakes? Are you a team player? This is a very important clue that a recruiter wants to know. Most of us have made several mistakes, but pick one where you apologized and managed to the right the wrong successfully, and let that be the focus of your story. Never say you have not had to apologize yet. This shows your highhanded attitude.

5) What is the most difficult situation you have ever faced in your life? How did you handle it?

By asking this question, the interviewer wants to know your inner views. Sometimes candidates open up and share their personal stories. This is a bad practice, indeed. You should stick to a professional anecdote and avoid sharing a personal story. Your positive attitude should be recognized during your narration. It tells the interviewer that you are resilient and willing to work around a problem. Seeing a project through on a tight budget, getting a project back on track when a deadline is missed, handling a difficult client, etc., are some situations you can talk about.

6) Who was your most difficult customer?

By asking this question, the interviewer wants to know your previous experience of dealing with customers. Here, you can share your story where you managed to successfully solve a highhanded customer. It would be best if you emphasized how you stayed calm and diffused the situation.

7) How do you handle a missed deadline/productivity target?

This question is asked to check the candidate's honesty and the skill of presenting things. Here, you have to straight forward and tell the interviewer what he or she wants to hear. The interviewer doesn't want to hear about the details of the project you were working on. They don't care. They want to know why you missed the deadline and how you handled the repercussions.

You have to very careful with the example you want to share. It would be best if you chose a relevant story with facts that allow you to present it positively. There is nothing good about missing a deadline, and if you try to justify it with any means, you will look foolish. To err is human. Everyone makes mistakes, and the interviewers also know this fact. So, you should accept your mistake and focus on explaining how you changed and learned from that mistake. People love honesty, and it is even more refreshing when employers hear that their employees are taking accountability for their mistakes and working to keep them from happening again.

The best practice is to be honest, and explain the situation. The STAR method is best for answering these behavioral interview questions. It can be explained as follows:

ST- situation or task

A- Action

R- Result

The STAR method emphasizes the following things:

- The situation of the issue or the task you were given.
- Describe the actions you took to resolve the situation or complete the task.
- Explain what you learned from the experience.

Explain the Positive side: In some cases, missing a deadline is not in the employee's control. If that is true of your situation, you can use it as your advantage and emphasize that the only reason it happened was because of an outdoor event. However, that is not always possible. In these types of cases, you must highlight the result or what you learned and how you improved.

If you are solely responsible for the deadline and have described the mistake you made, don't leave your interviewer with a negative image. Instead, make your answer work for you by ending your response with a very positive, clear message about what steps you took to improve and where you are on reaching your goals.

Be Confident: If the interviewer asks you to share a story about your failure, don't lose your focus and confidence. Remember, everyone has made small or big mistakes. Explain how you move on from that incident by learning something new.

Example of a good sample answer:

Once I was given a deadline to write an article for a client on a short turnaround time. I believed I could handle the article in addition to the workload I already had, but I miscalculated how long it would take me to write it. The morning the article was due, I realized I would not make it in time and contacted my boss to explain the situation. I apologized, explained what

happened and asked for an extension, which he granted. I learned that I need to be honest with myself about the workload I can handle each day. I also learned that when accepting assignments, I need to include a time buffer to ensure that even if unforeseen events arise, I can meet my deadlines.

8) What would you do if you found out that your closest friend at work was stealing?

This question is asked to check your loyalty towards the company as well as your friend. You should consider the nature of the offense before deciding how to proceed.

According to the question, your friend's act is unethical, so you should tell the company about your friend's act. You can give a chance to your friend and warn him earlier.

9) If your direct manager instructed you to do something you disagreed with, how would you handle it?

By asking this question, your interviewer wants to know how you handle disagreement. This is also helpful to check your negotiation skills. You can explain your communicational flexibility and convey to the interviewer that you aren't averse to speaking up. You believe in communicating till you find a solution.

10) What would you do if you saw someone being unsafe at work?

You can tell the interviewer that you have a duty of care to yourself and your colleagues. If you find these types of circumstances, you will warn the person that he is working can lead to an accident. If you know a safer way to do the job, you will suggest it. Otherwise, you will call your supervisor.

Interview Tips	Job/HR Interview Questions
Company Interview Questions & Procedure	JavaScript Interview Questions
Java Basics Interview Questions	Java OOPs Interview Questions
Servlet Interview Questions	JSP Interview Questions
Spring Interview Questions	Hibernate Interview Questions
PL/SQL Interview Questions	SQL Interview Questions
Oracle Interview Questions	Android Interview Questions
jQuery Interview Questions	MySQL Interview Questions