



Software Development

[Articles](#) [Ebooks](#) [Free Practice Tests](#) [On-demand Webinars](#) [Video Tutorials](#) [Live Webinars](#)

[Home](#) [Resources](#) [Software Development](#) [Top 40 Coding Interview Questions You Should Know](#)

Top 40 Coding Interview Questions You Should Know

By [Taha Sufiyan](#)

Last updated on Jun 30, 2022

60359



Table of Contents

[Coding Interview Questions On Conceptual Understanding](#)

[Programming Interview Questions](#)

[Get Ahead of the Curve and Master Programming Today](#)



Programming questions are an integral part of an interview for the developer's position. No matter which **programming language** you master, familiarity with fundamental concepts of programming is something that is always expected from you.

Coding skills are always the deciding factor in any programming interview. In this coding interview questions article, we will discuss the top 40 Coding Interview Questions you should absolutely know to crack those interviews and get your dream job.

So without further ado, let's get started!

The coding interview questions addresses in this article are grouped into 2 categories (as below) to make your learning simpler.

1. Conceptual Interview Questions
2. Programming Interview Questions

We will now look at the first category of coding interview questions.

Caltech Coding Bootcamp

Become a full stack developer in 6 months

ENROLL NOW

Coding Interview Questions On Conceptual Understanding

This section covers some coding interview questions that test the conceptual understanding of the candidate.

1. What is a Data Structure?

- A data structure is a storage format that defines the way data is stored, organized, and



manipulated.

- Some popular data structures are Arrays, Trees, and Graphs.

2. What is an Array?

- An array is commonly referred to as a collection of items stored at contiguous memory locations.
- Items stored are of the same type.
- It organizes data so that a related set of values can be easily sorted or searched.

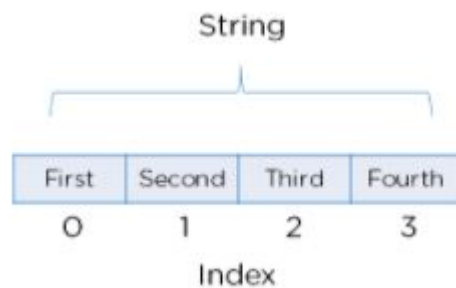


Fig: Array

3. What is a Linked List?

- Like an array, a linked list refers to a linear data structure in which the elements are not necessarily stored in a contiguous manner.
- It is basically a sequence of nodes, each node points towards the next node forming a chain-like structure.

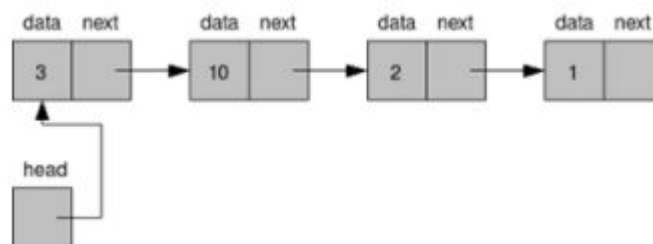


Fig: Linked List

4. What is LIFO?

- LIFO is an abbreviation for Last In First Out



- It is a way of accessing, storing and retrieving data.
- It extracts the data that was stored last first.

5. What is a Stack?

- A stack refers to a linear data structure performing operations in a LIFO (Last In First Out) order.
- In a stack, elements can only be accessed, starting from the topmost to the bottom element.

6. What is FIFO?

- FIFO stands for First In First Out.
- It is a way of accessing, storing and retrieving data.
- The data that was stored first is extracted first.

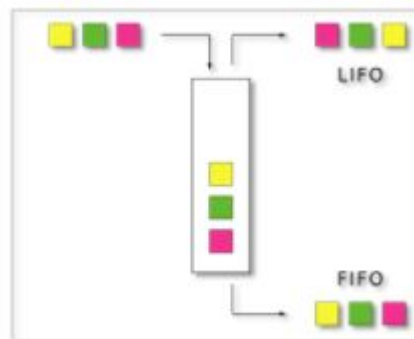


Fig: LIFO, FIFO

Till now, you've covered some very fundamental coding interview questions. Going ahead you will dive deeper into the subject.

7. What is a Queue?

- A queue refers to a linear data structure that performs operations in a FIFO order.
- In a queue, the least recently added elements are removed first as opposed to a stack.



Fig: Queue

8. What are Binary Trees?

- A binary tree is an extension of the linked list structure where each node has at most two children.
- A binary tree has two nodes at all times, a left node and a right node.

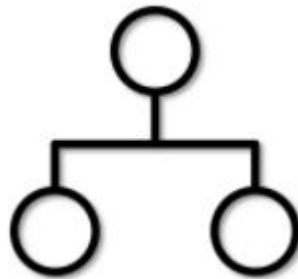


Fig: Binary Trees

9. What is Recursion?

- Recursion refers to a function calling itself based on a terminating condition.
- It uses LIFO and therefore makes use of the stack data structure.

The next couple of coding interview questions will explore your knowledge of OOPs.

10. What is the OOPs concept?

OOPs stands for **Object-Oriented Programming** System, a paradigm that provides concepts such as objects, classes, and inheritance.

11. What are the concepts introduced in OOPs?

Following are the concepts introduced in OOPs:

- Object - A real-world entity having a particular state and behavior. We can define it as an instance of a class.

- **Class** - A logical entity that defines the blueprint from which an object can be created or instantiated.
- **Inheritance** - A concept that refers to an object gaining all the properties and behaviors of a parent object. It provides code reusability.
- **Polymorphism** - A concept that allows a task to be performed in different ways. In Java, we use method overloading and method overriding to achieve polymorphism.
- **Abstraction** - A concept that hides the internal details of an application and only shows the functionality. In Java, we use abstract class and interface to achieve abstraction.
- **Encapsulation** - A concept that refers to the wrapping of code and data together into a single unit.

This is one of the very common coding interview questions, that often allows the interviewer to branch out into related topics based on the candidate's answers

12. Explain what a Binary Search Tree is.

- A **binary search tree** is used to store data in a manner that it can be retrieved very efficiently.
- The left sub-tree contains nodes whose keys are less than the node's key value.
- The right sub-tree contains nodes whose keys are greater than or equal to the node's key value

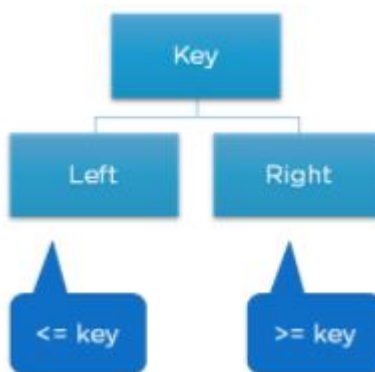


Fig: Binary Search Tree

13. Explain Doubly Linked Lists?

- Doubly linked lists are categorized as a special type of linked list in which traversal across the data elements can be done in both directions.
- This is made possible by the presence of two links in every node, one that links to the node next

to it and another that connects to the node before it.

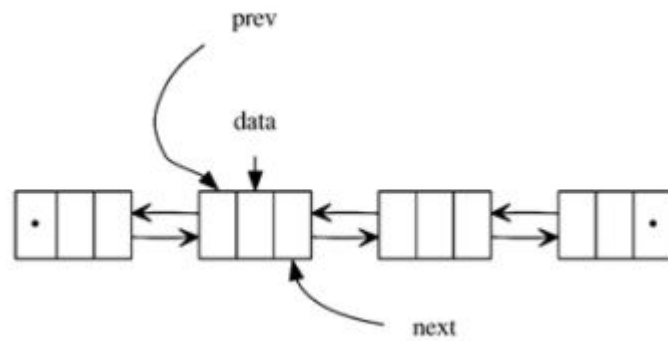


Fig: Doubly Linked List

14. What is a Graph?

- A graph is a particular type of data structure that contains a set of ordered pairs.
- The ordered pairs in a graph are also known as edges or arcs and are most commonly used to connect nodes where the data can be stored and retrieved.

15. Differentiate between linear and non-linear data structure?

Linear data structure	Non-linear data structure
It is a structure in which data elements are adjacent to each other	It is a structure in which each data element can connect to over two adjacent data elements
Examples of linear data structure include linked lists, arrays, queues, and stacks	Examples of nonlinear data structure include graphs and trees

16. What is a Deque?

- A deque is a double-ended queue.



- This is a structure in which elements can be inserted or removed from either end.

17. What's the difference between Stack and Array?

Stack	Array
Stack follows a Last In First Out (LIFO) pattern. What this means is that data access necessarily follows a particular sequence where the last data to be stored is the first one that will be extracted.	On the other hand, Arrays do not follow a specific order, but instead can be accessed or called by referring to the indexed element within the array.

18. Which sorting algorithm is the best?

- There are many types of sorting algorithms: bubble sort, quick sort, balloon sort, merge sort, radix sort, and more.
- No algorithm can be considered as the best or fastest because they have designed each for a specific type of data structure where it performs the best

19. How does variable declaration affect memory?

- The amount of memory that is to be reserved or allocated depends on the data type being stored in that variable.
- For example, if a variable is declared to be "integer type", 32 bits of memory storage will then be reserved for that particular variable.

20. What are dynamic data structures?

Dynamic data structures have the feature where they expand and contract as a program runs. It provides a very flexible method of data manipulation because adjusts based on the size of the data to be manipulated.

These 20 coding interview questions that test the conceptual understanding of the candidates give the interview a clear idea on how strong the candidate's fundamentals are

Full Stack Web Developer Course

To become an expert in MEAN Stack

[VIEW COURSE](#)

Programming Interview Questions

The next set of coding interview questions focus tests the programming expertise of the candidates and dives deep into various related aspects.

The code screenshots given along with the below coding interview questions helps you provide the answer to the question, with clarity.

21. How do you reverse a string in Java?

- Declare a string.
- Take out the length of that string.
- Loop through the characters of the string.
- Add the characters in reverse order in the new string.

```
String str = "hello";
```

```
String reverse = "";
```

```
int length = str.length();
```

```
for (int i = 0; i < length; i++) {
```



```
reverse = str.charAt(i) + reverse;  
  
}
```

```
System.out.println(reverse);
```

22. How do you determine if a string is a palindrome?

- A string is a palindrome when it stays the same on reversing the order of characters in that string.
- It can be achieved by reversing the original string first and then checking if the reversed string is equal to the original string.

```
if (str.equals(reverse)) {  
  
    System.out.println("Palindrome");  
  
} else {  
  
    System.out.println("Not Palindrome");  
  
}
```

23. Find the number of occurrences of a character in a String?

To find the number of occurrences, loop through the string and search for that character at every iteration; whenever it is found, it will update the count.

```
int count = 0;  
  
char search = 'a';  
  
for (int i = 0; i < length; i++) {  
  
    if (str.charAt(i) == search) {  
  
        count++;  

```



```
}
```

```
}
```

```
System.out.println(count);
```

24. How to find out if the given two strings are anagrams or not?

Two strings are anagrams if they contain a similar group of characters in a varied sequence.

- Declare a boolean variable that tells at the end of the two strings are anagrams or not.
- First, check if the length of both strings is the same, if not, they cannot be anagrams.
- Convert both the strings to character arrays and then sort them.
- Check if the sorted arrays are equal. If they are equal, print anagrams, otherwise not anagrams.

```
boolean anagrmstat = false;
```

```
if (str.length() != reverse.length()) {
```

```
    System.out.println(str + " and " + reverse + " not anagrams string");
```

```
} else {
```

```
    char[] anagram1 = str.toCharArray();
```

```
    char[] anagram2 = reverse.toCharArray();
```

```
    Arrays.sort(anagram1);
```

```
    Arrays.sort(anagram2);
```

```
    anagrmstat = Arrays.equals(anagram1, anagram2);
```

```
}
```



```
if (anagramstat == true) {  
  
    System.out.println(" anagrams string");  
  
} else {  
  
    System.out.println(" not anagrams string");  
  
}
```

25. How do you calculate the number of vowels and consonants in a String?

- Loop through the string.
- Increase the vowel variable by one whenever the character is found to be a vowel, using the if condition. Otherwise, increment the consonant variable.
- Print the values of both the vowel and the consonant count.

```
int vowels = 0;
```

```
int consonants = 0;
```

```
for (int k = 0; k < str.length(); k++) {
```

```
    char c = str.charAt(k);
```

```
    if (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u')
```

```
        vowels++;
```

```
    else
```

```
        consonants++;
```

```
}
```

```
System.out.println("Vowel count is " + vowels);
```



```
System.out.println("Consonant count is: " + consonants);
```

26. How do you get the matching elements in an integer array?

- Declare an array.
- Nest a couple of loops to compare the numbers with other numbers in the array.
- Print the matching elements if found.

```
int[] a = { 1, 2, 3, 4, 5, 1, 2, 6, 7 };
```

```
for (int m = 0; m < a.length; m++) {
```

```
    for (int n = m + 1; n < a.length; n++) {
```

```
        if (a[m] == a[n])
```

```
            System.out.print(a[m]);
```

```
    }
```

```
}
```

27. How would you implement the bubble sort algorithm?

- Declare an array.
- Nest a couple of loops to compare the numbers in the array.
- The array will be sorted in ascending order by replacing the elements if found in any other order.

```
int[] a = { 1, 2, 7, 6, 4, 9, 12 };
```

```
for (int k = 0; k < a.length; k++) {
```

```
    for (int l = 0; l < a.length - l - 1; l++) {
```

```
        if (a[l] > a[l + 1]) {
```



```
    int t = a[l];

    a[l] = a[l + 1];

    a[l + 1] = t;

}

}

}
```

Learn the top skills in-demand including Angular, Spring Boot, JSPs, and SOA to build highly web scalable apps with the [Full Stack Java Developer Masters Program](#).

28. How would you implement the insertion sort algorithm?

- We assume the first element in the array to be sorted. The second element is stored separately in the key. This sorts the first two elements. You can then take the third element and do a comparison with the ones on the left of it. This process will go on until a point where we sort the array.

```
int[] a = { 1, 2, 7, 6, 4, 9, 12 };

for (int m = 1; m < a.length; m++) {

    int n = m;

    while (n > 0 && a[n - 1] > a[n]) {

        int k = a[n];

        a[n] = a[n - 1];

        a[n - 1] = k;

        n--;
```



```
}  
  
}
```

29. How do you reverse an array?

- Loop till the half-length of the array.
- Replace the numbers corresponding to the indexes from the starting and the end.

```
int[] a = { 1, 2, 7, 6, 4, 9, 12 };
```

```
for (int t = 0; t < a.length / 2; t++) {
```

```
    int tmp = a[t];
```

```
    a[t] = a[a.length - t - 1];
```

```
    a[a.length - t - 1] = tmp;
```

```
}
```

30. How would you swap two numbers without using a third variable?

- Declare two variables and initialize them with values.
- Make b the sum of both numbers.
- Then subtract the sum (b) from a, so a is now swapped.
- Lastly, subtract a from the sum (b), so b is also swapped.

```
int a = 10;
```

```
int b = 20;
```

```
b = b + a; // now b is sum of both the numbers
```

```
a = b - a; // b - a = (b + a) - a = b (a is swapped)
```



```
b = b - a; // (b + a) - b = a (b is swapped)
```

31. Print a Fibonacci series using recursion?

- The Fibonacci numbers are the numbers in the following integer sequence:

0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144,

- We can calculate them using the mathematical formula used in the Fibonacci recursive function.

```
public static int fibonacci(int n) {  
  
    if (n <= 1)  
  
        return n;  
  
    return fibonacci(n - 1) + fibonacci(n - 2);  
  
}  
  
public static void main(String args[]) {  
  
    int n = 10;  
  
    System.out.println(fibonacci(n));  
  
}
```

32. How do you find the factorial of an integer?

- A factorial is a function that multiplies a number by every number below it. For example, $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$.
- Recursive function multiplies the numbers until it reaches 1.

```
public static long factorial(long n) {  
  
    if (n == 1)
```




```
    return 1;

else

    return (n * factorial(n - 1));

}
```

33. How do you reverse a Linked List?

- Declare a linked list.
- Add elements to that linked list.
- Apply the descending iterator method to the linked list.
- This reverses the order of elements in the linked list.

```
LinkedList<Integer> ll = new LinkedList<>();
```

```
ll.add(1);
```

```
ll.add(2);
```

```
ll.add(3);
```

```
System.out.println(ll);
```

```
LinkedList<Integer> ll1 = new LinkedList<>();
```

```
ll.descendingIterator().forEachRemaining(ll1::add);
```

```
System.out.println(ll1);
```

34. How would you implement Binary Search?

- Binary search divides the array into half in every iteration step until it finds the element.
- It works on the sorted arrays since it compares the values of adjacent elements and then



calculates the mid number.

- If the value of low becomes greater than high at any point, it means the element is not present in the list.

```
int mid = (low + high) / 2;
```

```
while (low <= high) {
```

```
    if (arr[mid] < key) {
```

```
        low = mid + 1;
```

```
    } else if (arr[mid] == key) {
```

```
        return mid;
```

```
    } else {
```

```
        high = mid - 1;
```

```
    }
```

```
    mid = (low + high) / 2;
```

```
}
```

```
if (low > high) {
```

```
    return -1;
```

```
}
```

```
return -1;
```

35. How would you find the second largest number in an array?

- Loop through the array.



- If the value of *i* is greater than the highest, store the value of *i* in highest, and store the value of highest in the second-highest variable.

```
private static int findSecondHighest(int[] array) {  
  
    int highest = Integer.MIN_VALUE;  
  
    int secondHighest = Integer.MIN_VALUE;  
  
    for (int i : array) {  
  
        if (i > highest) {  
  
            secondHighest = highest;  
  
            highest = i;  
  
        } else if (i > secondHighest) {  
  
            secondHighest = i;  
  
        }  
  
    }  
  
    return secondHighest;  
  
}
```

36. How do you remove all occurrences of a given character from the input string?

- Use the built-in string method “replace” to replace a character with any other character, including symbols and white spaces.

```
String str1 = "Australia";  
  
str1 = str1.replace("a", "");
```



```
System.out.println(str1); // ustrli
```

37. Showcase Inheritance with the help of a program?

- The class Cat inherits the property color from the class Animal by extending the parent class (Animal).
- This way a class Cat can have more parent classes if it wishes to inherit their properties.

```
class Animal {  
  
    String color;  
  
}  
  
class Cat extends Animal {  
  
    void meow() {  
  
        System.out.println("Meow");  
  
    }  
  
}
```

38. Explain overloading and overriding with the help of a program?

Overloading:

When a class has two or more methods with the same name, they are called overloaded methods.

```
class Foo {  
  
    void print(String s) {  
  
        System.out.println(s);  
  
    }  
  
}
```



```
void print(String s, int count) {  
  
    while (count > 0) {  
  
        System.out.println(s);  
  
        count--;  
  
    }  
  
}  
  
}
```

Overriding:

When a superclass method is also implemented in the child class, it's a case of overriding.

```
class Base {  
  
    void printName() {  
  
        System.out.println("Base Class");  
  
    }  
  
}
```

```
class Child extends Base {  
  
    @Override  
  
    void printName() {  
  
        System.out.println("Child Class");  
  
    }  
  
}
```



```
}
```

Full Stack Java Developer Course

In Partnership with HIRIST and HackerEarth

EXPLORE COURSE

39. How do you check if the given number is prime?

- Use if statements to check for each condition separately:
 - If the number is 0 or 1, it cannot be prime.
 - If the number is 2, it is prime number.
 - If the number is indivisible by other numbers, it is prime.

```
public static boolean isPrime(int n) {
```

```
    if (n == 0 || n == 1) {
```

```
        return false;
```

```
    }
```

```
    if (n == 2) {
```

```
        return true;
```

```
    }
```

```
    for (int i = 2; i <= n / 2; i++) {
```



```
        if (n % i == 0) {  
  
            return false;  
  
        }  
  
    }  
  
    return true;  
  
}
```

40. How do you sum all the elements in an array?

- Use for loop to iterate through the array and keep adding the elements in that array.

```
int[] array = { 1, 2, 3, 4, 5 };
```

```
int sum = 0;
```

```
for (int i : array)
```

```
    sum += i;
```

```
System.out.println(sum);
```

As you get prepared for your job interview, we hope that these Coding Interview Questions have provided more insight into what types of questions you are likely to come across.

Watch the video below that deals with real-time and corporate-level programming-based interview questions and answers.



Get Ahead of the Curve and Master Programming Today

Now that you are well-versed with the top Coding Interview Questions, you should be seeking out opportunities to gain the skills you need to leverage the immense popularity of [software development](#) and build a successful career in it. Well, you needn't look any further. We have got your back! Our comprehensive [Post Graduate Program in Full Stack Web Development](#) and the [Full Stack Developer Course](#) will help you with the necessary skills and more that you need and help you become career-ready upon completion.

If you are looking for more information, do check out our YouTube video that further explains these coding interview questions and answers, and helps you become more confident while appearing for the interview processes.

Do you have any questions for us? Well, feel free to mention them in the comments section of this "Top 40 Coding Interview Questions" article. Our subject matter experts will review and answer them for you soon.

Find our Full Stack Web Developer - MEAN Stack Online Bootcamp in top cities:

Name	Date	Place
Full Stack Web Developer - MEAN Stack	Cohort starts on 25th Jul 2022, Weekend batch	Your City
Full Stack Web Developer - MEAN Stack	Cohort starts on 9th Aug 2022, Weekend batch	Chicago

Full Stack Web Developer - MEAN Stack	Cohort starts on 30th Aug 2022, Weekend batch	Houston
---------------------------------------	---	---------

About the Author



Taha Sufiyan

Taha is a Research Analyst at Simplilearn. He is passionate about building great user interfaces and keeps himself updated on the world of Artificial Intelligence. Taha is also interested in gaming and ph...

Recommended Programs

Full Stack Web Developer - MEAN Stack

953 Learners

Lifetime
Access*

Full Stack Java Developer

4187 Learners

Lifetime
Access*

Post Graduate Program in Full Stack Web
Development

3368 Learners

Lifetime
Access*



*Lifetime access to high-quality, self-paced e-learning content.

Explore Category

Find Full Stack Web Developer - MEAN Stack Master's Program in these cities

Mean Stack Developer Course in Atlanta | Mean Stack Developer Course in Austin | Mean Stack Developer Course in Charlotte | Mean Stack Developer Course in Chicago | Mean Stack Developer Course in Dallas | Mean Stack Developer Course in Houston | Full Stack Web Developer - MEAN Stack in Los Angeles | Mean Stack Developer Course in New York | Mean Stack Developer Course in San Francisco | Mean Stack Developer Course in San Jose | Mean Stack Developer Course in Tampa | Mean Stack Developer Course in Washington

NEXT ARTICLE

Top R Programming Interview Questions and Answers

By Shardul Prakash Mohite

4378

Oct 11, 2021

Recommended Resources

Kubernetes Interview Guide

Facebook Interview Questions

0 Comments

Simplilearn

 Disqus' Privacy Policy Login ▾ Favorite 1 Tweet Share

Sort by Best ▾



Start the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS 

Be the first to comment.

 Subscribe  Add Disqus to your site  Add DisqusAdd  Do Not Sell My Data

© 2009 -2022- Simplilearn Solutions

Disclaimer

PMP, PMI, PMBOK, CAPM, PgMP, PfMP, ACP, PBA, RMP, SP, and OPM3 are registered marks of the Project Management Institute, Inc.

