Interview Questions

C language

Q1. what is return type of printf and scanf function explain.

Ans: printf() - printf() returns the number of characters successfully written on the output. It is used to simply print data in the output.

scanf() - It returns the number of data items that have been entered successfully.

Q2. What are static variables and functions?

Ans: The variables and functions that are declared using the keyword Static are considered as Static Variable and Static Functions. The variables declared using Static keyword will have their scope restricted to the function in which they are declared.

Q3. Differentiate between calloc() and malloc()

Ans: calloc() and malloc() are memory dynamic memory allocating functions. The only difference between them is that calloc() will load all the assigned memory locations with value 0 but malloc() will not.

Q4. Can a C program be compiled or executed in the absence of a main()?

Ans: The program will be compiled but will not be executed. To execute any C program, main() is required.

Q5. What do you mean by a Nested Structure?

Ans: When a data member of one structure is referred by the data member of another function, then the structure is called a Nested Structure.

C++

Q1. Is deconstructor overloading possible? If yes then explain and if no then why?

Ans - No destructor overloading is not possible. Destructors take no arguments, so there’s only one way to destroy an object. That’s the reason destructor overloading is not possible.

Q2. What is call by value and call by reference in C++?

Ans - In the call by value method, you pass the copies of actual parameters to the function's formal parameters. This means if there is any change in the values inside the function, then that change will not affect the actual values.

In the call-by-reference method, the reference or address of actual parameters is sent to the function's formal parameters. This means any change in values inside the function will be reflected in the actual values.

Q3. What is an inline function?

Ans - An inline function when called expands in line. When you call this function, the whole code of the inline function gets inserted or substituted at the inline function call.

Syntax:

```
Inline return-type function-name(parameters)
{
 }
```

Q4. How do you allocate and deallocate memory in C++?

Ans -The new operator is used for memory allocation and deletes operator is used for memory deallocation in C++.

For example-

```
int value=new int;                  //allocates memory for storing 1 integer
delete value;               // deallocates memory taken by value

int *arr=new int[10];       //allocates memory for storing 10 int
delete []arr;               // deallocates memory occupied by arr
```

Q5. What are void pointers?

Ans - A void pointer is a pointer which is having no datatype associated with it. It can hold addresses of any type.

For example-

```
void *ptr;
char *str;
p=str;          // no error
str=p;          // error because of type mismatch
```

We can assign a pointer of any type to a void pointer but the reverse is not true unless you typecast it as

```
str=(char*) ptr;
```

Core Java

Q1. Can you override the static method in Java? if I create the same method in the subclass is it a compile-time error?

No, you can not override the static method in Java but it's not a compile-time error to declare the exact same method in a subclass, That is called method hiding in Java.

Q2. What is the importance of finally block in exception handling?

Answer

A *finally* block will always be executed, whether or not an exception is actually thrown. Even in the case where the catch statement is missing and an exception is thrown, the finally block will still be executed. Last thing to mention is that the finally block is used to release resources like I/O buffers, database connections, etc.

Q3. What is classloader?

Classloader is a subsystem of JVM which is used to load class files. Whenever we run the java program, it is loaded first by the classloader. There are three built-in classloaders in Java.

1. Bootstrap ClassLoader: This is the first classloader which is the superclass of Extension classloader. It loads the *rt.jar* file which contains all class files of

Java Standard Edition like java.lang package classes, java.net package classes, java.util package classes, java.io package classes, java.sql package classes, etc.

2. Extension ClassLoader: This is the child classloader of Bootstrap and parent classloader of System classloader. It loads the jar files located inside *$JAVA_HOME/jre/lib/ext* directory.
3. System/Application ClassLoader: This is the child classloader of Extension classloader. It loads the class files from the classpath. By default, the classpath is set to the current directory. You can change the classpath using "-cp" or "-classpath" switch. It is also known as Application classloader.

Q4. What do System.gc() and Runtime.gc() methods do?

Ans These suggest to JVM that garbage collection can be started; however, the JVM may finally run the garbage collector at a later time. System.gc() is a class method and Runtime.gc() is an instance method.

Q5. Distinguish between throw and throws in Java

The *throw* keyword is used within a method to raise an exception explicitly. This keyword is followed by an instance. The *throws* keyword declares an exception, precedes a class name, and works in a way that the try-catch block does. This keyword is used with the method signature.

Adv Java

1. **What are Collection related features in Java 8?**
   Java 8 has brought major changes in the Collection API. Some of the changes are:
   1. Java Stream API for collection classes for supporting sequential as well as parallel processing

2. Iterable interface is extended with forEach() default method that we can use to iterate over a collection. It is very helpful when used with lambda expressions because its argument Consumer is a function interface.
3. Miscellaneous Collection API improvements such as forEachRemaining(Consumer action) method in Iterator interface, Map replaceAll(), compute(), merge() methods.

2. **What is difference between Enumeration and Iterator interface?** Enumeration is twice as fast as Iterator and uses very little memory. Enumeration is very basic and fits basic needs. But the Iterator is much safer as compared to Enumeration because it always denies other threads to modify the collection object which is being iterated by it. Iterator takes the place of Enumeration in the Java Collections Framework. Iterators allow the caller to remove elements from the underlying collection that is not possible with Enumeration. Iterator method names have been improved to make its functionality clear.

What do you understand by iterator fail-fast property?

Iterator fail-fast property checks for any modification in the structure of the underlying collection everytime we try to get the next element. If there are any modifications found, it throws ConcurrentModificationException. All the implementations of Iterator in Collection classes are fail-fast by design except the concurrent collection classes like ConcurrentHashMap and CopyOnWriteArrayList

3. What are the two ways of implementing thread in Java?

There are basically two ways of implementing thread in java as given below:

- Extending the Thread class
- Implementing Runnable interface in Java

Q4. What's the difference between class lock and object lock?

Class Lock: In java, each and every class has a unique lock usually referred to as a class level lock. These locks are achieved using the keyword ‘static synchronized’ and can be used to make static data thread-safe. It is generally used when one wants to prevent multiple threads from entering a synchronized block.

Example:

```
public class ClassLevelLockExample
{
 public void classLevelLockMethod()
 {
   synchronized (ClassLevelLockExample.class)
     {
         //DO your stuff here
     }
 }
}
```

Object Lock: In java, each and every object has a unique lock usually referred to as an object-level lock. These locks are achieved using the keyword ‘synchronized’ and can be used to protect non-static data. It is generally used when one wants to synchronize a non-static method or block so that only the thread will be able to execute the code block on a given instance of the class.

Example:

```
public class ObjectLevelLockExample

{

 public void objectLevelLockMethod()

 {

    synchronized (this)

      {

           //DO your stuff here

      }

 }

}
```

DS

Q1. What are the advantages of binary search over a linear search?

In sorted List

A binary search is more efficient than a linear search because we perform **fewer comparisons**. With linear search, we can only eliminate one element per comparison each time we fail to find the value we are looking for, but with the binary search, we **eliminate half** the set with each comparison.

Binary search runs in **O(log n)** time compared to linear search's **O(n)** time. This means that the more of the elements present in the search array, the faster is binary search compared to a linear search.

Q2. Are linked lists considered linear or non-linear Data Structures?

Linked lists are considered **both** linear and non-linear data structures depending upon the application they are used for. When used for access strategies, it is considered as a linear data-structure. When used for data storage, it is considered a non-linear data structure.

Q3. What is Stack Along With Its Applications.

A stack is a data structure in which elements are placed one on top of the other and additions and deletions can occur only at the top. Stacks can be used in the following applications:

- A stack can be used in applications where users are enabled to backtrack on previous operations by one step or move forward with a new operation.
- Converting infix expressions into postfix expressions.
- The process of reversing the characters in a string can be completed by placing them in a stack and using the pop operation.

Q4. What is an AVL tree?

An AVL tree is a type of binary search tree that is always in a state of partially balanced. The balance is measured as a difference between the heights of the subtrees from the root. This self-balancing tree was known to be the first data structure to be designed as such.

Q5. How graph represented in memory?

A graph can be represented using 3 data structures- **adjacency matrix, adjacency list and adjacency set.**

An **adjacency matrix** can be thought of as a table with rows and columns. The row labels and column labels represent the nodes of a graph.

Spring Boot

Q1. What are starter dependencies?

Spring boot starter is a maven template that contains a collection of all the relevant transitive dependencies that are needed to start a particular functionality.

Like we need to import spring-boot-starter-web dependency for creating a web application.

```
<dependency>
<groupId> org.springframework.boot</groupId>
<artifactId> spring-boot-starter-web </artifactId>
</dependency>
```

Q2. What is Spring Initializer?

Spring Initializer is a web application that helps you to create an initial spring boot project structure and provides a maven or gradle file to build your code. It solves the problem of setting up a framework when you are starting a project from scratch.

Q3. What are the Spring Boot Annotations?

The @RestController is a stereotype annotation. It adds @Controller and @ResponseBody annotations to the class. We need to import org.springframework.web.bind.annotation package in our file, in order to implement it.

Q4. What is Spring Boot dependency management?

Spring Boot manages dependencies and configuration automatically. You don't need to specify version for any of that dependencies.

Spring Boot upgrades all dependencies automatically when you upgrade Spring Boot.

Q5. What are the Spring Boot key components?

Below are the four key components of spring-boot:

- Spring Boot auto-configuration.
- Spring Boot CLI.
- Spring Boot starter POMs.
- Spring Boot Actuators.

OS

Q1. Explain demand paging?

Demand paging is a method that loads pages into memory on demand. This method is mostly used in virtual memory. In this, a page is only brought into memory when a location on that particular page is referenced during execution. The following steps are generally followed:

- Attempt to access the page.
- If the page is valid (in memory) then continue processing instructions as normal.
- If a page is invalid then a page-fault trap occurs.
- Check if the memory reference is a valid reference to a location on secondary memory. If not, the process is terminated (illegal memory access). Otherwise, we have to page in the required page.
- Schedule disk operation to read the desired page into main memory.
- Restart the instruction that was interrupted by the operating system trap.

Q2. What are daemons?

A daemon is a computer program that runs as a background process to provide functions that might not be available in the base Operating System. Daemons are usually used to run services in the background without directly being in control of interactive users. The purpose of Daemons are to handle periodic requests and then forward the requests to appropriate programs for execution.

Q3. Explain grep command.

Grep stands for Global Regular Expression Print. The grep command is used to search for a text in a file by pattern matching based on regular expression.

Syntax: grep [options] pattern [files]

Example:

$ grep -c "linux" interview.txt

This command will print the count of the word “linux” in the “interview.txt” file.

Q4. What is zombie process?

Zombie process, referred to as a defunct process, is basically a process that is terminated or completed but the whole process control block is not cleaned up from the main memory because it still has an entry in the process table to report to its parent process. It does not consume any of the resources and is dead, but it still exists. It also shows that resources are held by process and are not free.

Q5. How to terminate a running process in Linux?

Every process has a unique process id. To terminate the process, we first need to find the process id. The ps command will list all the running processes along with the process id. And then we use the kill command to terminate the process.

The command for listing down all the processes

$ ps

Suppose the process id of the process you want to terminate is 3849, then you will have to terminate it like this:

$ kill 3849

DBT

Q1. Can you elaborate on BLOB and TEXT in MySQL?

BLOB

BLOB(Binary Large Object) is used to hold a variable amount of data and holds up to 65,535 bytes of data. The following are the four types of BLOB.

- TINYBLOB
- BLOB
- MEDIUMBLOB
- LONGBLOB

TEXT

TEXT is used to store string values and holds up to a maximum length of 65,535 characters. The following are the four types of TEXT

- TINYTEXT
- TEXT
- MEDIUMTEXT
- LONGTEXT

Q2. How can you test for NULL values in a database?

A NULL value is a field with no value present in that particular field. Since the NULL value cannot be compared to any other NULL values, you cannot use the comparison operators such as =, <, or <>. To compare the fields with NULL values, you have to use the IS NULL and IS NOT NULL operator.

Refer below for Syntax of IS NULL and IS NOT NULL.

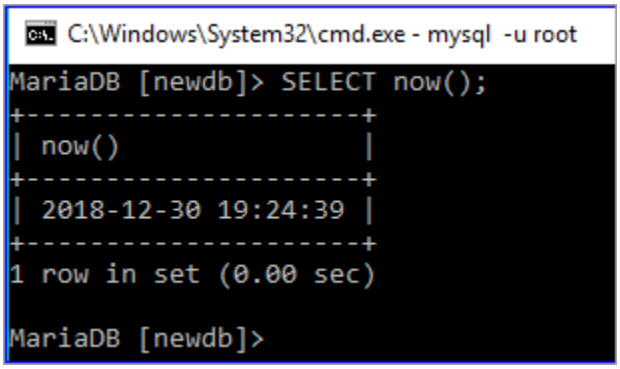
1 SELECT column_names FROM table_name WHERE column_name IS NULL;

2 SELECT column_names FROM table_name WHERE column_name IS NOT NULL;

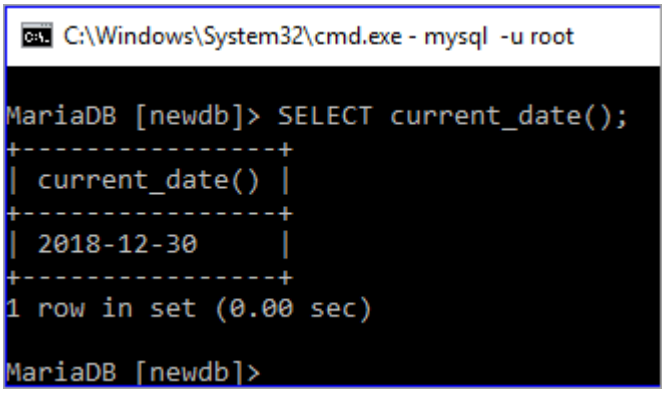
Q3. What is the difference between NOW() and CURRENT_DATE()?

Answer: Both NOW() and CURRENT_DATE() are built-in MySQL methods. NOW() is used to show the current date and time of the server and CURRENT_DATE() is used to show only the date of the server.

SELECT now();



SELECT current_date();



Q4. How can you change the name of any existing table by using the SQL statement?

Answer: The following SQL command is used to rename an existing table of the database.

RENAME TABLE table_name TO new_name

Q5. Explain the difference between DELETE and TRUNCATE.

1. DELETE command is used to delete a single or multiple or all the records from the table. The TRUNCATE command is used to delete all the records from the table or make the table empty.
2. When DELETE command is used to delete all the records from the table then it doesn't re-initialize the table. So, the AUTO_INCREMENT field does not count from one when the user inserts any record. But when all the records of any table are deleted by using TRUNCATE command then it re-initializes the table and a new record will start from one for the AUTO_INCREMENT field.

HTML

Q1 What is the difference between span and div?

Answer

- div is a block element
- span is inline element

For bonus points, you could point out that it's illegal to place a block element inside an inline element, and that while div can have a p tag, and a p tag can have a span, it is not possible for span to have a div or p tag inside.

Q2. Why do we use HTML5?

Answer: HTML5 supports animation, drawing, audio, video, etc and it easily embeds a video on the web page. It does not require any additional software like Flash for watching videos.

Some of the important reasons to use HTML5 are given below:

- Legacy and cross-browser support

- Better interactions
- Smarter storage
- Cleaner code

Q3. In how many ways can we specify the CSS styles for the HTML element?

There are three ways in which we can specify the styles for HTML elements:

- Inline: Here we use the ‘style’ attribute inside the HTML element.
- Internal: Here we use the <style> tag inside the <head> tag. To apply the style we bind the elements using ‘id’ or ‘class’ attributes.
- External: Here we use the <link> tag inside <head> tag to reference the CSS file into our HTML code. Again the binding between elements and styles is done using ‘id’ or ‘class’ attributes.

Q4. Is the <datalist> tag and <select> tag same?

No. The <datalist> tag and <select> tag are different. In the case of <select> tag a user will have to choose from a list of options, whereas <datalist> when used along with the <input> tag provides a suggestion that the user selects one of the options given or can enter some entirely different value.

Q.5 Explain the concept of web storage in HTML5.

This web storage helps in storing some of the static data in the local storage of the browser so that we do not need to fetch it from the server every time we need it. There is a size limit based on different browsers. This helps in decreasing the load time and a smooth user experience. There are two types of web storage that are used to store data locally in HTML5:

- Local Storage - This helps in storing data that will be retained even though the user reopens the browser. It is stored for each webapp on different browsers.
- Session Storage - This is used for one session only. After the user closes the browser this gets deleted.

Q6. Difference between SVG and Canvas HTML5 element?

SVG	Canvas
SVG is a vector based i.e., composed of shapes.	It is Raster based i.e., composed of pixels.
SVG works better with a larger surface.	Canvas works better with a smaller surface.
SVG can be modified using CSS and scripts.	Canvas can only be modified using scripts.
SVG is highly scalable. So we can print at high quality with high resolution.	It is less scalable.

ReactJS

Q1. What are props in React?

The props in React are the inputs to a component of React. They can be single-valued or objects having a set of values that will be passed to components of React during creation by using a naming convention that almost looks similar to HTML-tag attributes. We can say that props are the data passed from a parent component into a child component.

The main purpose of props is to provide different component functionalities such as:

- Passing custom data to the React component.
- Using through this.props.reactProp inside render() method of the component.
- Triggering state changes.

For example, consider we are creating an element with reactProp property as given below: <Element reactProp = "1" />

This reactProp name will be considered as a property attached to the native props object of React which already exists on each component created with the help of React library: props.reactProp;.

Q2. What is Jest?

Jest is a JavaScript unit testing framework created by Facebook based on Jasmine. It offers automated mock creation and a jsdom environment. It is also used as a testing component.

Q3. How to create components in ReactJS?

Answer

There are two possible ways to create ReactJS Components.

1. Functional components: This is the simplest way to create ReactJS components. It accepts props as an Object and returns ReactJS elements. We call it as “functional” because those are pure JavaScript functions.

```
function Greeting(props) {
  return <h1> Hello, {props.message}</h1>
}
```

2. Class components: You can also use Es6 class to define component. The above functional component can be written as below,

```
class Greeting extends React.Component {
      render() {
            return <h1>Hello, {this.props.message}</h1>;
        }
      }
```

Q4. What is useState() in React?

The useState() is a built-in React Hook that allows you for having state variables in functional components. It should be used when the DOM has something that is dynamically manipulating/controlling.

In the below-given example code, The useState(0) will return a tuple where the count is the first parameter that represents the counter's current state and the second parameter setCounter method will allow us to update the state of the counter.

```
...
const [count, setCounter] = useState(0);
const [otherStuffs, setOtherStuffs] = useState(...);
...
const setCount = () => {
  setCounter(count + 1);
  setOtherStuffs(...);
  ...
};
```

We can make use of setCounter() method for updating the state of count anywhere. In this example, we are using setCounter() inside the setCount function where various other things can also be done. The idea with the usage of hooks is that we will be able to keep our code more functional and avoid class-based components if they are not required.

Q5 What is *Reconciliation*?

Answer

When a component's props or state change, React decides whether an actual DOM update is necessary by comparing the newly returned element with the previously rendered one. When they are not equal, React will update the DOM. This process is called reconciliation.

NodeJS

Q1 What do you mean by Asynchronous API?

Answer

All APIs of Node.js library are aynchronous that is non-blocking. It essentially means a Node.js based server never waits for a API to return data. Server moves to

next API after calling it and a notification mechanism of Events of Node.js helps server to get response from the previous API call.

Q2. What is typically the first argument passed to a Node.js callback handler?
Answer
Node.js core modules, as well as most of the community-published ones, follow a pattern whereby the first argument to any callback handler is an optional error object. If there is no error, the argument will be null or undefined.

A typical callback handler could therefore perform error handling as follows:

```
function callback(err, results) {
   // usually we'll check for the error before handling results
   if(err) {
      // handle error somehow and return
   }
   // no error, perform standard callback handling
}
```

Q3. What is the cluster?

Cluster is a process to handle thread execution load while working with multi-core systems.

Q4. Name the types of API functions in Node.js?

There are two types of API functions in Node.js:

- Asynchronous, Non-blocking functions
- Synchronous, Blocking functions

1. Blocking functions

In a blocking operation, all other code is blocked from executing until an I/O event that is being waited on occurs. Blocking functions execute synchronously.

Example:

```
const fs = require('fs');
const data = fs.readFileSync('/file.md'); // blocks here until file is read
console.log(data);
// moreWork(); will run after console.log
```

The second line of code blocks the execution of additional JavaScript until the entire file is read. moreWork () will only be called after Console.log

2. Non-blocking functions

In a non-blocking operation, multiple I/O calls can be performed without the execution of the program being halted. Non-blocking functions execute asynchronously.

Example:

```
const fs = require('fs');
fs.readFile('/file.md', (err, data) => {
  if (err) throw err;
  console.log(data);
});
// moreWork(); will run before console.log
```

Since fs.readFile() is non-blocking, moreWork() does not have to wait for the file read to complete before being called. This allows for higher throughput.

Q5. What is the role of assert in Node.js?

The Node.js Assert is a way to write tests. It provides no feedback when running your test unless one fails. The assert module provides a simple set of assertion tests that can be used to test invariants. The module is intended for internal use by Node.js, but can be used in application code via require ('assert'). For example:

```
1. var assert = require('assert');
2. function add (a, b) {
3.    return a + b;
```

```
4. }
5. var expected = add(1,2);
6. assert( expected === 3, 'one plus two is three');
```