

T

TCS Ninja Interview Questions

[Download As PDF](#)



TCS Ninja Interview Questions

What is TCS Ninja?

TCS Ninja is a program offered by Tata Consultancy Services (TCS), one of the largest Indian multinational IT services company. TCS Ninja is a campus hiring program that is aimed at recruiting fresh graduates and experienced professionals for a wide range of IT and business roles. The program typically involves a series of written tests, technical and behavioural interviews, and other assessments to evaluate the candidate's skills, knowledge, and overall fit for the company. Successful candidates who pass the TCS Ninja program are offered employment with TCS and receive comprehensive training and development to help them succeed in their roles.

TCS Ninja Selection Process

The TCS Ninja recruitment process typically includes the following steps:

1. **Online Application:** The first step is to submit an online application, which includes your resume and other relevant information.
2. **Written Test:** The next step is to take an aptitude test, which is designed to assess your cognitive and problem-solving abilities.
3. **Technical Interview:** If you pass the aptitude test, you will be called for a technical interview, which is focused on assessing your technical skills and knowledge in your area of expertise.
4. **HR round:** This is the final round, where the candidate's overall fit for the company is evaluated. The interviewer may ask about the candidate's salary expectations, their long-term career goals, and any other concerns or questions they may have.

UGC NET/SET Course Online by

LIVE Access Live Classes, Mocks, PYP & Notes for 375+ Exams!



Get UGC NET/SET SuperCoaching @ just

~~₹26999~~ **₹8749**

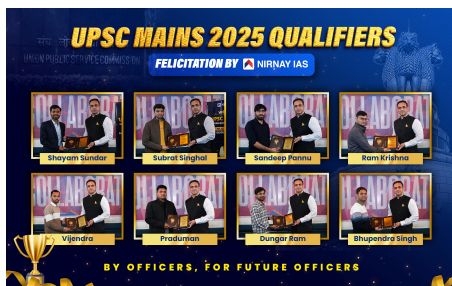
Your Total Savings ₹18250

Explore SuperCoaching

Want to know more about this Super Coaching ?

Download Brochure

People also like



**Nirnay IAS 2026 - संकल्प
Batch (Hindi)**

~~₹100000~~ (21% OFF)

₹79999 (Valid for 24 Months)

Explore this Supercoaching

TCS Ninja Technical Interview Questions



Access Live Classes, Mocks, PYP & Notes for 375+ Exams!

A linear data structure is a type of data structure that stores data in a linear fashion, meaning that the data is organized in a sequence. The most common examples of linear data structures are arrays, linked lists, and stacks.

In an array, data is stored in a contiguous block of memory, with each element of the array being assigned an index. Arrays are commonly used to store a large number of similar items.

In a linked list, each element of the list is called a node, and each node contains a value and a reference to the next node in the list. Linked lists are commonly used to store a large number of items that may change frequently.

A stack is a linear data structure that follows the Last-In-First-Out (LIFO) principle. This means that the element that is added last to the stack is the first one to be removed. Stacks are commonly used to store a set of items that need to be kept track of, such as function calls in a program.

In summary, linear data structures are data structures that store data in a linear fashion and are commonly used to store a large number of similar items, items that may change frequently, or items that need to be kept track of.

Q2. What is DHCP?

DHCP stands for Dynamic Host Configuration Protocol. It is a network protocol used to automatically assign IP addresses to devices connected to a network.

When a device connects to a network, it sends a broadcast message asking for an IP address. A DHCP server on the network listens for this message and assigns an available IP address to the device, along with other network configuration information such as the subnet mask, default gateway, and DNS server.

This process eliminates the need for manually configuring IP addresses on each device, making it easier to add or remove devices from the network. DHCP also helps to avoid IP address conflicts by keeping track of which IP addresses are currently in use.

DHCP is typically used on networks with a large number of devices, such as in home networks, small businesses, and enterprise networks. It is a widely used protocol, and most devices such as computers, smartphones, and IoT devices have built-in DHCP clients.

In summary, DHCP is a network protocol that automatically assigns IP addresses to devices connected to a network, making it easier to add or remove devices and avoiding IP address conflicts. It is widely used in home, small business and enterprise networks.

Q3. What do you mean by SDLC?

SDLC stands for Software Development Life Cycle. It is a methodology that is used to plan, design, build, test, and deploy software applications. SDLC defines a set of procedures and guidelines that

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

There are several different SDLC models, each with their own unique set of steps, but most SDLC models include the following general phases:

1. **Planning:** In this phase, the project team identifies the problem or opportunity, defines the project scope and objectives, and creates a project plan.
2. **Analysis:** In this phase, the project team gathers requirements, identifies the system's constraints, and defines the system's boundaries.
3. **Design:** In this phase, the project team creates a detailed design of the system, including the architecture, interfaces, and data structures.
4. **Implementation (or Development):** In this phase, the project team writes and tests the code for the system.
5. **Testing:** In this phase, the project team tests the system to ensure that it meets the requirements and is free of defects.
6. **Deployment:** In this phase, the system is deployed to the production environment and made available to the end-users.
7. **Maintenance:** In this phase, the project team provides ongoing support and makes any necessary updates or modifications to the system.

SDLC is an iterative process, and these phases may be repeated multiple times throughout the development process. The choice of SDLC model depends on the project's goals, size, complexity and the organization's culture. Some common SDLC models include Waterfall, Agile, Scrum, and DevOps.

In summary, SDLC is a methodology that defines a set of procedures and guidelines that are followed throughout the software development process to ensure that the end product meets the requirements of the user and is of high quality. There are several different SDLC models, each with their own unique set of steps, that can be chosen depending on the project's goals, size, complexity and the organization's culture.

Q4. What is an IP Address?

An IP address is a numerical label assigned to each device connected to a computer network that uses the Internet Protocol for communication. It serves two main functions: identifying the host or network interface, and providing the location of the host in the network. IP addresses are typically written in the form of four sets of numbers separated by periods, such as "192.168.1.1".

Q5. How do you differentiate between IPV4 and IPV6?

IPv4 and IPv6 are the two versions of the Internet Protocol (IP) that are currently in use.

IPv4 addresses are 32-bit numbers, typically written in the form of four sets of numbers separated by periods, such as "192.168.1.1". They can support a maximum of 4.3 billion unique addresses.



Access Live Classes, Mocks, PYP & Notes for 375+ Exams!

support a virtually unlimited number of unique addresses.

Another key difference between IPv4 and IPv6 is the way they are represented. IPv4 addresses are represented in a dotted-decimal notation, while IPv6 addresses are represented in a colon-separated hexadecimal notation.

Additionally, IPv6 has additional features such as support for multicast and anycast addressing, improved security and autoconfiguration.

Q6. Explain the TCP/IP model?

The TCP/IP model is a set of protocols used for communication over networks. It is the foundation of the internet and is divided into four layers: the application layer, the transport layer, the internet layer, and the link layer.

1. The application layer is the topmost layer and is responsible for providing interfaces to the user, such as HTTP, FTP, and DNS.
2. The transport layer provides end-to-end communication and is responsible for ensuring the reliability of data transfer. It includes protocols such as TCP and UDP.
3. The internet layer is responsible for routing packets of data across networks. It includes protocols such as IP and ICMP.
4. The link layer is responsible for communication between devices on the same network. It includes protocols such as Ethernet and WiFi.

TCP/IP is a standardized, open, and vendor-neutral communication protocol that connects different types of networks together and enables the communication between different types of devices.

Q7. What is an operating system?

An operating system (OS) is a program that acts as an intermediary between a computer's hardware and its software. It allows the hardware and software to interact and provides a way for users to interact with the computer. The operating system is responsible for managing the computer's resources, including memory, processors, and input/output devices. It also provides a way for software to access the computer's hardware and for users to access the software. Common examples of operating systems include Microsoft Windows, Apple macOS, and Linux.

Q8. What are the layers in an OSI model?

The OSI (Open Systems Interconnection) model is a framework that describes how communications should occur between different network devices. It defines seven layers, each with a specific function:

1. Physical Layer: responsible for transmitting raw bits over a communication channel.
2. Data Link Layer: responsible for creating a reliable link between two devices on a network.



Access Live Classes, Mocks, PYP & Notes for 375+ Exams!

5. Session Layer: responsible for setting up, managing, and terminating sessions between applications.
6. Presentation Layer: responsible for converting data into a format that can be understood by the application layer.
7. Application Layer: responsible for providing services to applications and end-users, such as email and file transfer.

Q9. What is polymorphism?

Polymorphism is a concept in object-oriented programming that refers to the ability of a single function or method to operate on multiple types of data. It allows for objects of different classes to be treated as objects of a common superclass, allowing for a more generic and flexible code.

There are two types of polymorphism:

1. Compile-time polymorphism, also known as static polymorphism, is when the type of the object is determined at compile-time, through function overloading and operator overloading.
2. Run-time polymorphism, also known as dynamic polymorphism, is when the type of the object is determined at runtime, through virtual functions and interfaces.

In other words, polymorphism allows objects of different types to be treated as objects of a common type. This enables you to write more flexible and generic code, by allowing objects of different types to be passed to the same function or method, and for that function or method to work correctly with any of those types.

Q10. What are LIFO and FIFO?

LIFO and FIFO are two types of data structures that are used to manage and organize data.

LIFO stands for Last-In, First-Out. It is a method of organizing and manipulating a data structure in which the last element added to the structure is the first one to be removed. LIFO is often implemented using a stack data structure, which is a linear collection of elements that follows the LIFO principle.

FIFO stands for First-In, First-Out. It is a method of organizing and manipulating a data structure in which the first element added to the structure is the first one to be removed. FIFO is often implemented using a queue data structure, which is a linear collection of elements that follows the FIFO principle.

In summary, LIFO is a data structure in which the last element added is the first one to be removed, and FIFO is a data structure in which the first element added is the first one to be removed.

11. Difference between execution and error?

Execution refers to the process of running a program or a piece of code, in which the instructions are executed in the order they were written. This process can be successful, meaning that the code

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

An error, also known as an exception, is an event that occurs during the execution of a program that disrupts the normal flow of instructions. Errors can be caused by a variety of things, such as a syntax error in the code, a logical error in the program's design, or an issue with the runtime environment.

Errors can be of different types, such as:

- Syntax errors: occur when the code is written in an incorrect way. For example, forgetting a semicolon or not closing a bracket.
- Logic errors: occur when the code is written correctly, but the algorithm or program logic is incorrect. This can result in unexpected results or infinite loops.
- Run-time errors: occur during the execution of the code, such as trying to divide by zero, trying to access an element out of an array's bounds, or trying to open a file that does not exist.

In summary, execution is the process of running a program, while error is a problem that occurs during the execution of the code, which can be caused by various reasons.

Q12. Given an array, how will you search an element in the most efficient manner?

The most efficient way to search for an element in an array would be to use a binary search algorithm. A binary search algorithm divides the array into two parts at each step and compares the target element with the middle element. If the target element matches the middle element, the search is successful. If the target element is less than the middle element, the search continues in the left half of the array. If the target element is greater than the middle element, the search continues in the right half of the array. This process is repeated until the target element is found or the search space is exhausted. The time complexity of a binary search algorithm is $O(\log n)$ making it more efficient than a linear search which has a time complexity of $O(n)$ where n is the size of the array.

Q13. Explain quick sort with algorithm and example?

Quick sort is a divide-and-conquer algorithm that sorts an array by selecting a "pivot" element and partitioning the array into two parts: one part with elements less than the pivot, and one part with elements greater than the pivot. The pivot is then placed in its correct position in the sorted array. The process is then repeated recursively for the two partitioned arrays until the entire array is sorted.

Here is an example of the quick sort algorithm using the following array: [5, 3, 8, 4, 9, 1, 6, 2, 7]

1. Select a pivot element, in this case we will use the first element (5)
2. Partition the array into two parts: [3, 4, 1, 2] and [8, 9, 6, 7]
3. Recursively repeat step 1 and 2 for the partitioned arrays
4. Once all partitioned arrays are sorted, the entire array is sorted

The final sorted array would be [1, 2, 3, 4, 5, 6, 7, 8, 9]

```
if low < high
    pivot_index = partition(arr, low, high)
    QuickSort(arr, low, pivot_index)
    QuickSort(arr, pivot_index + 1, high)

partition(arr, low, high)
    pivot = arr[low]
    left = low + 1
    right = high
    done = false
    while not done
        while left <= right and arr[left] < pivot
            left = left + 1
        while arr[right] > pivot and right >= left
            right = right - 1
        if right < left
            done = true
        else
            swap arr[left], arr[right]
    swap arr[low], arr[right]
    return right
```

Q14. What is an IoT device?

IoT (Internet of Things) devices are physical devices that are connected to the internet and can collect, transmit, and act on data. These devices can range from simple sensors and actuators to complex systems such as industrial machinery and vehicles. IoT devices can be found in a wide range of applications, including smart homes, industrial automation, healthcare, transportation, and agriculture.

IoT devices typically have a few key components:

1. **Sensors:** These are used to collect data from the environment, such as temperature, humidity, and motion.
2. **Connectivity:** This allows the device to connect to the internet and transmit data. This can be done through wired or wireless connections, such as Wi-Fi, Bluetooth, or cellular.
3. **Processing:** This allows the device to analyze and act on the data it collects. This can be done through a microcontroller or an embedded computer.
4. **Power supply:** IoT devices run on batteries or they have to be plugged into an outlet.

Examples of IoT devices include smart thermostats, security cameras, smart locks, wearable fitness devices, and industrial sensors. With the increasing availability of low-cost hardware and the

development of new technologies, more and more devices are becoming IoT-enabled, leading to the creation of a vast interconnected network of devices.

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

C and C++ are both programming languages, but they have some important differences.

1. C++ is an extension of C: C++ was developed as an extension of the C programming language, and it includes all the features of C. C++ also includes additional features such as classes, objects, and templates that are not found in C.
2. Object-Oriented Programming: C++ is an object-oriented programming (OOP) language, while C is not. This means that C++ allows for the creation of classes and objects, and it supports inheritance, polymorphism, and encapsulation. C, on the other hand, is a procedural programming language, which means that it focuses on procedures, or functions, rather than objects.
3. Standard Template Library (STL): C++ has a Standard Template Library (STL) that provides a set of common classes and functions for tasks such as input/output, strings, and containers. C does not have this library.
4. Exception Handling: C++ includes built-in support for exception handling, which allows for the creation of try-catch blocks to handle errors or exceptions. C does not have this feature.
5. Templates: C++ has a feature called templates which allows for generic programming. Templates allow you to write code that works with different types, and it's used to define generic classes and functions. C does not have this feature.

Overall, C++ is a more powerful and versatile language than C, but it is also more complex. C is simpler and easier to learn, but it has fewer features and capabilities than C++.

Q16. What is a constructor in java and its types?

In Java, a constructor is a special method that is used to initialize an object when it is created. It is automatically called by the Java runtime when an object is instantiated using the "new" keyword, and it has the same name as the class.

A constructor typically has the following characteristics:

- It does not have a return type, not even void.
- It has the same name as the class.

Java has two types of constructors:

1. Default Constructor: A default constructor is a constructor that takes no arguments. It is provided by the compiler if the class does not define any constructors.
2. Parameterized Constructor: A parameterized constructor is a constructor that takes one or more parameters. It is used to initialize the object with specific values when it is created.

Here is an example of a class with a default and a parameterized constructor:

```
class MyClass {
```

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

```
    MyClass() {  
        x = 0;  
        y = 0;  
    }  
    // Parameterized constructor  
    MyClass(int x, int y) {  
        this.x = x;  
        this.y = y;  
    }  
}
```

In this example, the class `MyClass` has two constructors. The default constructor initializes the values of `x` and `y` to 0, and the parameterized constructor initializes the values of `x` and `y` to the values passed as arguments. An object can be created by using both constructors

```
MyClass obj1 = newMyClass(); MyClass obj2 = newMyClass(10,20);
```

In addition, Java also provides a special constructor called copy constructor, which creates a new object and initializes it with the values of an existing object. However, this is not a default constructor provided by the Java language, it's a user-defined constructor. It's implemented in the class by the developer.

Q17. What are collections in JAVA?

In Java, the Collections Framework is a set of classes and interfaces that provide a standard way to manage and manipulate collections of objects. The framework defines a set of common interfaces and classes, such as `List`, `Set`, and `Map`, that can be used to store and manipulate collections of data.

The main interfaces in the Collections Framework are:

- **Collection:** This is the root interface in the Collections Framework. It represents a group of objects known as elements.
- **List:** A `List` is a collection that can hold duplicate elements. It maintains the order of elements and allows positional access to elements. Examples of classes that implement `List` are `ArrayList` and `LinkedList`.
- **Set:** A `Set` is a collection that cannot hold duplicate elements. It does not maintain the order of elements. Examples of classes that implement `Set` are `HashSet` and `TreeSet`.
- **Map:** A `Map` is a collection that stores key-value pairs. It does not maintain the order of elements. Examples of classes that implement `Map` are `HashMap` and `TreeMap`.

The Collections Framework also provides several utility classes, such as `Collections` and `Arrays`, that provide useful methods for working with collections, such as sorting and searching.

In addition to the basic collection interfaces, the framework also provides advanced collections such as Queue, Deque, and SortedSet, which have specialized behaviors and additional methods.

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

sets, and maps, and provides a wide range of methods for working with those collections.

Q18. Explain ACID properties in DBMS?

ACID properties are a set of properties that ensure that database transactions are processed reliably. The term ACID stands for Atomicity, Consistency, Isolation, and Durability. These properties ensure that a database transaction is treated as a single, indivisible unit of work, and that the integrity of the data is preserved even in the event of errors or failures.

1. Atomicity: This property ensures that a transaction is atomic, meaning that it is treated as a single, indivisible unit of work. If a transaction is atomic, then either all of its operations are executed, or none of them are. This means that if any part of the transaction fails, the entire transaction is rolled back, and the data is left in its original state.
2. Consistency: This property ensures that a transaction brings the database from one valid state to another. A transaction should not violate any integrity constraints defined on the database.
3. Isolation: This property ensures that concurrent execution of transactions does not result in inconsistencies. Each transaction is executed in isolation, meaning that it cannot see or be affected by any other transactions that are executing concurrently.
4. Durability: This property ensures that once a transaction is committed, its effects are permanent and will survive any subsequent failures. This is achieved by writing the transaction's changes to non-volatile storage, such as disk.

ACID properties are a fundamental part of any DBMS and are essential for maintaining the integrity and consistency of a database. These properties allow multiple users to interact with the same data concurrently, while ensuring that the data remains consistent and accurate at all times.

Q19. What is the DDL command in MySql?

In MySQL, DDL (Data Definition Language) commands are used to define the database schema. It simply deals with descriptions of the database schema and is used to create and modify the structure of database objects in the database. Some examples of DDL commands in MySQL include:

1. CREATE: This command is used to create a new table, view, index, or other database object.

```
CREATETABLE Employee ( EmployeeID INTPRIMARY KEY, FirstName VARCHAR(255),  
LastName VARCHAR(255), Salary INT );
```

2. ALTER: This command is used to alter the structure of an existing table. It can be used to add, modify, or delete columns, constraints, and indexes.

```
ALTER TABLE Employee ADD Address VARCHAR(255);
```



Access Live Classes, Mocks, PYP & Notes for 375+ Exams!

```
DROPTABLE Employee;
```

4. TRUNCATE: This command is used to delete all data from a table, but keep the table structure intact.

```
TRUNCATETABLE Employee;
```

5. RENAME: This command is used to rename an existing table.

```
RENAME TABLE Employee TO Employee_backup;
```

These are some of the commonly used DDL commands in MySQL. It is important to be careful when using these commands, as they can permanently modify or delete the structure of your database.

Q20. explain Create and Alter command?

The CREATE and ALTER commands are used in SQL to create and modify database objects such as tables, views, and indexes.

The CREATE command is used to create a new database object. For example, the following SQL statement creates a new table named "employees" with columns "id", "name", and "salary":

```
CREATE TABLE employees (  
    id INT PRIMARY KEY,  
    name VARCHAR(255),  
    salary DECIMAL(10,2)  
);
```

The ALTER command is used to modify an existing database object. For example, the following SQL statement adds a new column named "address" to the "employees" table:

```
ALTER TABLE employees  
ADD COLUMN address VARCHAR(255);
```

It can also be used to rename or drop columns, change the data type of columns, add or drop constraints, and perform other modifications.

```
ALTER TABLE employees
```

```
MODIFY COLUMN emp_id INT(4);
```

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

```
ALTER TABLE employees
```

```
RENAME TO emp;
```

```
ALTER TABLE employees
```

```
DROP COLUMN address;
```

It is important to be careful when using the ALTER command, as it can permanently modify the structure of your table and its data.

Q21. What is the Drop and Truncate command in SQL?

1. DROP: The DROP command is used to delete an existing table, view, index, or another database object in MySQL. The basic syntax for dropping a table is as follows:

```
DROPTABLE table_name;
```

For example, to delete the "Employee" table, you can use the following command:

```
DROPTABLE Employee;
```

It is important to be careful when using the DROP command, as it will permanently delete the table and all data stored in it.

2. TRUNCATE: The TRUNCATE command is used to delete all data from a table, but keep the table structure intact. The basic syntax for truncating a table is as follows:

```
TRUNCATETABLE table_name;
```

For example, to delete all data from the "Employee" table, you can use the following command:

```
TRUNCATETABLE Employee;
```

The TRUNCATE command is generally faster than a DELETE command with a WHERE clause that deletes all rows, because it does not generate undo logs, it resets the table, auto-increment counter to zero, and not fire triggers, thus it is more efficient.

It is also important to be careful when using the TRUNCATE command, as it will permanently delete all data stored in the table but the table structure will remain intact.

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

JIT (Just-In-Time) compiler is a feature of the Java Virtual Machine (JVM) that improves the performance of Java applications by compiling bytecode into native machine code at runtime. The JIT compiler is responsible for converting the bytecode, which is the machine-independent code that the JVM executes, into machine code that can be executed directly by the computer's processor.

The JIT compiler works by analyzing the bytecode as it is executed and identifying the frequently executed code, also known as "hot spots." These hot spots are then compiled into native machine code, which can be executed much more quickly than the interpreted bytecode. This process of compiling hot spots on-the-fly is what gives the JIT compiler its name.

The JIT compiler can be beneficial for Java applications that have performance bottlenecks, as it can significantly speed up the execution of frequently executed code. However, it can also introduce some overhead, as the JIT compiler needs to analyze and compile the code at runtime. Additionally, the JIT compiler's ability to optimize code will depend on the specific JVM implementation and the characteristics of the code being executed.

Q23. What is the OOPs concept?

OOP (Object-Oriented Programming) is a programming paradigm that uses objects and their interactions to design applications and computer programs. It is based on the concept of "objects", which can contain data and code to manipulate that data.

The main concepts of OOP are:

1. **Encapsulation:** The process of wrapping up data and methods that operate on that data within a single unit, or object. This protects the data from outside access and modification, and allows the object to control how its data is used.
2. **Inheritance:** The mechanism by which one class can inherit properties and methods from a parent class. This allows for a hierarchy of classes and the reuse of common properties and methods.
3. **Polymorphism:** The ability of objects of different classes to be treated as objects of a common superclass. This allows for the use of a single interface to represent multiple types of objects.
4. **Abstraction:** The process of hiding the internal details of an object and showing only the necessary information to the outside world. This allows for a simplified interface and the ability to change the internal implementation without affecting the rest of the system.

By using these concepts, OOP helps developers to create more modular, reusable, and maintainable code. It also allows for the representation of real-world objects and the modeling of complex systems in a natural and intuitive way.

Q24. What is Abstraction in OOPS?

Abstraction in object-oriented programming (OOP) refers to the process of hiding the implementation details of a class or an object, and exposing only the necessary information to the

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

Q25. What are inheritance and its types in Java?

Inheritance in Java is a mechanism that allows a class to inherit the properties and methods of another class. It enables the creation of a new class (derived class) that reuses, extends, or modifies the behavior defined in an existing class (base class).

There are two types of inheritance in Java:

1. Single Inheritance: When a class inherits from a single parent class, it is known as single inheritance.
2. Multiple Inheritance: When a class inherits from more than one parent class, it is known as multiple inheritance. Java does not support multiple inheritance directly, but it can be achieved through the use of interfaces.

Java also supports multi-level inheritance, where a class can inherit from a class that inherits from another class, and so on.

Java also supports hierarchical inheritance, where multiple classes inherit from a single base class.

And lastly, Java also supports Hybrid Inheritance, where a combination of more than one types of inheritance used.

Q26. What is Method Overloading?

Method overloading in Java is a feature that allows a class to have multiple methods with the same name, but with different parameters. This is also known as "compile-time polymorphism" or "static polymorphism" because the decision of which method to call is made at compile-time based on the number, type, and order of the parameters passed to the method.

When a method is called, Java uses the type and/or number of arguments to determine which method implementation should be executed. If a method with the same name but different parameters is defined in the same class, the method with the most specific parameter type will be called.

Method overloading can be achieved in two ways:

1. By changing the number of arguments of the method.
2. By changing the data type of the arguments.

It is important to note that, Method overloading is not possible by just changing the return type of the method.

Q27. Explain the Insertion sort?

Insertion sort is a simple sorting algorithm that builds the final sorted list one item at a time. It repeatedly takes the next unsorted element and inserts it in the correct position within the already

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

part, and the unsorted part. Initially, the sorted part contains only the first element of the list, and the unsorted part contains the rest of the elements.

The algorithm iterates through the unsorted part of the list, and for each element, it compares it with the elements in the sorted part and finds the correct position to insert it.

Here are the steps of the Insertion sort algorithm:

1. Start with the first element of the list, and consider it as the sorted part.
2. Pick the next element from the unsorted part of the list.
3. Compare this element with the elements in the sorted part of the list.
4. Find the correct position to insert this element in the sorted part.
5. Move the elements in the sorted part, if necessary, to make room for the new element.
6. Insert the element in the correct position.
7. Repeat steps 2-6 for all elements in the unsorted part of the list.

Insertion sort is an in-place sorting algorithm, which means it does not require any extra memory space. It has a time complexity of $O(n^2)$ in worst and average case and $O(n)$ in best case when the input list is already sorted.

It is useful when the input list is almost sorted or when the number of elements is small.

Q28. What is a linked list explain with an example?

A linked list is a data structure that consists of a sequence of elements, called nodes, each of which contains an element of the list (the data) and a reference (pointer) to the next node in the list. The first node is called the head, and the last node has a reference to null, which indicates the end of the list.

Here is an example of a singly linked list:

```
class Node {
    int data;
    Node next;
}

class LinkedList {
    Node head;
    // other methods to add/delete/update nodes
}
```

In the above example, the Node class has two fields: data, which stores the element of the list, and next, which is a reference to the next node in the list. The LinkedList class has a head field that

points to the first node of the list.

To add a new element to the list, a new node is created, with the new element as the data and the



Access Live Classes, Mocks, PYP & Notes for 375+ Exams!

To remove an element from the list, the reference to the previous node is updated to point to the next node, effectively removing the current node from the list.

Linked lists have some advantages over arrays, such as dynamic size and efficient insertion and deletion of elements. However, they have a slower access time for elements and require more memory to store the references.

It's also worth mentioning that there are different types of linked lists: singly linked list, doubly linked list and circular linked list. In singly linked list, only next pointer is present whereas in doubly linked list, both next and previous pointer are present. In circular linked list, the last node points to the first node, creating a loop.

Q29. What is a firewall? How does it work?

A firewall is a security system that controls the incoming and outgoing network traffic based on a set of rules and regulations. It is implemented as a software or hardware solution, and it acts as a barrier between a private internal network and the public Internet.

The main function of a firewall is to block unauthorized access while allowing authorized communication. It can be configured to allow or deny traffic based on a variety of criteria, such as the source and destination IP addresses, ports, and protocols.

There are two common types of firewall: network firewalls and host-based firewalls.

A network firewall is typically a standalone device that sits between the internal network and the Internet. It inspects all incoming and outgoing network traffic and blocks or allows it based on the configured rules.

A host-based firewall is installed on a specific host or device, such as a computer or a mobile phone. It controls the incoming and outgoing network traffic for that specific host or device.

Firewalls use different techniques to control the network traffic. These include:

- Packet filtering: This is the most basic form of firewall protection. It examines the header of each incoming and outgoing packet and compares it against a set of rules. If the packet meets the rules, it is allowed to pass through the firewall, otherwise, it is blocked.
- Stateful inspection: This technique keeps track of the state of the network connection. It examines not only the packet header but also the context of the connection. This allows the firewall to block malicious traffic that could not be detected by packet filtering alone.
- Application-layer filtering: This technique examines the payload of the packet, which is the actual data being transmitted. It is used to block specific applications or protocols, such as BitTorrent or Skype.

- Deep packet inspection: This technique examines the entire packet, including the payload, and compares it against a set of rules. It can detect malicious traffic that is hidden in the payload,

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

Overall, a firewall is a fundamental component of network security that controls the flow of data in and out of a network, protecting it from unauthorized access, hacking attempts and other malicious activities, it helps to protect both the internal and external network and the devices connected to it.

Q30. What is the stack data structure and its operations?

A stack is a linear data structure that follows the Last In First Out (LIFO) principle. It is used to store a collection of elements and provides two main operations: push and pop.

- Push: This operation adds an element to the top of the stack. The element becomes the new top of the stack, and any elements previously on the top are pushed down.
- Pop: This operation removes the top element from the stack. The element that was previously beneath the top element becomes the new top.
- Peek: This operation is used to return the top element of the stack without removing it.
- IsEmpty: This operation is used to check if the stack is empty or not.

Here is an example of how a stack is implemented in Java:

```
class Stack {
    private int[] data;
    private int top;

    public Stack(int size) {
        data = new int[size];
        top = -1;
    }

    public void push(int element) {
        if (top == data.length - 1) {
            throw new StackOverflowError();
        }
        data[++top] = element;
    }

    public int pop() {
        if (top == -1) {
            throw new EmptyStackException();
        }
        return data[top--];
    }
}
```

```
}
```

 Access Live Classes, Mocks, PYP & Notes for 375+ Exams!

```
    }  
    return data[top];  
}  
  
public boolean isEmpty() {  
    return top == -1;  
}  
}
```

In the above example, the Stack class has a private data array that stores the elements of the stack and a private top variable that keeps track of the index of the top element. The push() method adds an element to the top of the stack, the pop() method removes the top element, the peek() method returns the top element, and the isEmpty() method checks if the stack is empty or not.

Stacks are used in a variety of applications such as parsing expressions, backtracking, undo-redo operations, memory management and more. Stacks are also used in many algorithms such as Depth First Search (DFS) and in some problems like the Tower of Hanoi.

Q31. What is cloud computing?

Cloud computing is a model for delivering IT services over the internet. It provides on-demand access to computing resources, such as servers, storage, databases, networking, software, analytics, and intelligence, without the need for users to have direct control over the underlying infrastructure.

Cloud computing can be classified into three main service models:

- Infrastructure as a Service (IaaS): This is the most basic form of cloud computing, where users can rent virtualized computing resources such as servers, storage, and networks on a pay-per-use basis. IaaS is often used for hosting web applications and services, as well as for running virtual machines.
- Platform as a Service (PaaS): This is a higher-level service than IaaS, where users can deploy and run applications without having to manage the underlying infrastructure. PaaS providers typically offer a development environment, runtime, and middleware, such as databases, web servers, and load balancers.
- Software as a Service (SaaS): This is the highest-level service of cloud computing, where users can access and use software applications over the internet. SaaS providers typically offer a wide range of applications, including email, customer relationship management, and enterprise resource planning.

There are also different deployment models of cloud computing:

- Public Cloud: This is where the cloud resources are owned and operated by a third-party provider and made available to the public.



Access Live Classes, Mocks, PYP & Notes for 375+ Exams!

- Hybrid Cloud: This is where an organization uses a combination of public and private clouds to meet its specific needs.

Cloud computing provides many benefits, including cost savings, scalability, flexibility, and improved disaster recovery. It also allows for greater collaboration and innovation, as well as the ability to access powerful technology without the need to invest in expensive hardware and software.

Overall, cloud computing is a way to access and use computing resources over the internet, it can be used for many different purposes, and it offers many benefits over traditional IT models.

Q32. What is Machine Learning?

Machine Learning (ML) is a field of artificial intelligence that enables computers to learn from data, without being explicitly programmed. It involves developing algorithms and statistical models that allow computers to learn from and make predictions or decisions without human intervention.

There are three main types of machine learning:

- Supervised Learning: In this type of learning, the computer is provided with labeled training data, which includes input and the corresponding correct output. The goal is to learn a mapping from the input to the output. This is the most common type of machine learning and is used for tasks such as image classification, speech recognition, and natural language processing.
- Unsupervised Learning: In this type of learning, the computer is provided with unlabeled data, and the goal is to find structure or patterns in the data. This is used for tasks such as clustering, anomaly detection, and dimensionality reduction.
- Reinforcement Learning: In this type of learning, the computer learns to make decisions by interacting with an environment and receiving feedback in the form of rewards or penalties. This is used for tasks such as game playing, robotics, and decision making.

Machine learning is used in a wide range of applications such as natural language processing, computer vision, speech recognition, fraud detection, predictive maintenance, and many more. As the amount of data and the computing power available continues to increase, machine learning is becoming more powerful and is being used to solve more complex problems.

Q33. What is cloud computing, the application of cloud computing, SAAS?

Cloud computing is a model for delivering IT services over the internet. It provides on-demand access to computing resources, such as servers, storage, databases, networking, software, analytics, and intelligence, without the need for users to have direct control over the underlying infrastructure.

One of the main application of cloud computing is Software as a Service (SaaS). SaaS is the highest-level service of cloud computing, where users can access and use software applications over the

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

internet. This allows users to access the software from any location and on any device with an internet connection, without having to install it on their own devices.

SaaS applications are usually subscription-based, with users paying a monthly or annual fee to access the software. SaaS applications are often updated automatically by the provider, so users don't have to worry about maintaining the software or keeping it up to date. SaaS providers typically handle the security, scalability, and availability of the software, allowing customers to focus on their own business operations.

Examples of SaaS applications include:

- Email services such as Gmail and Microsoft 365
- CRM software such as Salesforce and Zoho CRM
- Project management software such as Asana and Trello
- Human resources software such as ADP and Workday
- E-commerce platforms such as Shopify and Magento
- Accounting software such as Xero and QuickBooks

Overall, SaaS is a way to access and use software over the internet, it can be used for many different purposes, and it offers many benefits over traditional software delivery models.

Q34. Do you know of any real-time applications of cloud computing?

Yes, there are many real-time applications of cloud computing. Some examples include:

1. Streaming Services: Cloud computing enables streaming services such as Netflix, Hulu, and Amazon Prime Video to deliver large amounts of data to users in real-time. The cloud provides the necessary storage and bandwidth to handle the large number of users and the large amount of data that is being streamed.
2. Online Gaming: Cloud computing is used to power online gaming platforms such as Xbox Live and PlayStation Network. The cloud provides the necessary resources to support the large number of users and the complex game environments.
3. Social Media: Social media platforms such as Facebook, Twitter, and Instagram rely on cloud computing to handle the large number of users and the large amount of data that is being shared in real-time.
4. E-commerce: E-commerce platforms such as Amazon and Alibaba use cloud computing to provide real-time inventory management, order processing, and delivery tracking.
5. IoT: Internet of Things (IoT) devices generate a large amount of data, and cloud computing provides the necessary resources to process and analyze this data in real-time. Examples include connected cars, smart homes, and industrial IoT.

6. Real-Time Analytics: Cloud computing allows businesses to process and analyze large amounts of data in real-time. This enables them to make more informed decisions, improve

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

fail, using sensor data and machine learning algorithms. The cloud-based system can then schedule maintenance and repair, avoiding downtime or emergency repairs.

These are just a few examples of the many real-time applications of cloud computing. The ability to scale resources on-demand and the ability to access data and services from anywhere with an internet connection makes cloud computing a powerful tool for real-time applications.

TCS Ninja Coding Questions

Q1. Write a Program in C to print the Half Pyramid Number Pattern?

Here is a C program that will print a half-pyramid number pattern of a specified height:

```
#include

void print_half_pyramid(int height) {
    for (int i = 1; i <= height; i++) {
        for (int j = 1; j <= i; j++) {
            printf("%d ", j);
        }
        printf("\n");
    }
}

int main() {
    int height = 5;
    print_half_pyramid(height);
    return 0;
}
```

This program defines a function called `print_half_pyramid` that takes an integer argument `height`, which represents the height of the pyramid. The function uses nested for loops to print the pyramid pattern. The outer loop iterates over the rows of the pyramid, and the inner loop iterates over the columns of each row, printing the current column value.

Here is an example of how the above program would output when the height is 5

```
1
1 2
```

```
1 2 3
1 2 3 4
1 2 3 4 5
```

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

Note that the `printf` function is used to print the numbers, and the `\n` is used to move to the next line after each row is printed. Also the function `int main()` is used to call the function `print_half_pyramid(height)` with the defined height variable.

Q2. Write a program in C to print the left triangle start pattern?

Here is an example of a program in C that will print a left triangle start pattern:

```
#include

int main() {
    int n = 5;

    // outer loop to handle number of rows
    for (int i = 0; i < n; i++) {
        // inner loop to handle number of columns
        // values changing acc. to outer loop
        for (int j = 0; j <= i; j++) {
            // printing stars
            printf("*");
        }
        // ending line after each row
        printf("\n");
    }
    return 0;
}
```

In this program, the outer loop (`for (int i = 0; i < n; i++)`) is used to iterate over the number of rows and the inner loop (`for (int j = 0; j <= i; j++)`) is used to iterate over the number of columns. The inner loop is nested inside the outer loop so that each row is printed one at a time. The `printf("*")` statement is used to print a single star for each column. The `printf("\n")` statement is used to move to the next line after each row is printed.

Q3. Write a program to check if a number is prime or not?

Here is an example of a program in C that will check if a number is prime or not:

```
#include
```

```
int main() {
    int n, i, flag = 0;
```

 Access Live Classes, Mocks, PYP & Notes for 375+ Exams!

```
    for(i = 2; i <= n/2; ++i) {
        // condition for nonprime number
        if(n % i == 0) {
            flag = 1;
            break;
        }
    }

    if (n == 1) {
        printf("1 is not a prime number.");
    } else if (flag == 0) {
        printf("%d is a prime number.", n);
    } else {
        printf("%d is not a prime number.", n);
    }

    return 0;
}
```

In this program, the user is prompted to enter a positive integer, which is stored in the variable `n`. The program then uses a for loop (`for(i = 2; i <= n/2; ++i)`) to iterate from 2 to `n/2`. For each value of `i`, the program checks if `n` is divisible by `i` using the modulus operator (`n % i == 0`). If `n` is divisible by `i`, then the variable `flag` is set to 1 and the loop is exited. If the loop completes and the value of `flag` is still 0, then the number is determined to be prime and the program prints "`n` is a prime number." Otherwise, the number is determined to be not prime and the program prints "`n` is not a prime number."

Q4. Write a recursive function to print the factorial of a number?

Here is an example of a recursive function in C that will print the factorial of a number:

```
#include

int factorial(int n) {
    if (n == 0) {
        return 1;
    } else {
        return n * factorial(n-1);
    }
}
```



```
int main() {
    int n;
```

 Access Live Classes, Mocks, PYP & Notes for 375+ Exams!

```
    printf("Factorial of %d = %d\n", n, factorial(n));

    return 0;
}
```

In this program, the function `factorial(int n)` takes an integer input `n` and calculates the factorial of that number. The function uses a recursive approach to solve this problem by calling the function itself with the parameter `n-1` until the base case is met where `n==0` and the function returns 1. Then the function starts to return the value of the factorial step by step by multiplying it with the next value of `n`. The main function takes the input from the user and calls the `factorial(n)` function to print the factorial of the entered number.

Q5. Write a program to check if there exist a duplicate key in the array?

Here is an example of a program in C that will check if there exists a duplicate key in an array:

```
#include
#define MAX_SIZE 100

int duplicateExist(int arr[], int size) {
    int i, j;
    for(i = 0; i < size; i++) {
        for(j = i+1; j < size; j++) {
            if(arr[i] == arr[j]) {
                return 1;
            }
        }
    }
    return 0;
}

int main() {
    int arr[MAX_SIZE], size, i;
    printf("Enter size of the array: ");
    scanf("%d", &size);
    printf("Enter elements in the array: ");
    for(i=0; i
```

In this program, the function `duplicateExist(int arr[], int size)` takes an array of integers and its size as input, and uses two nested for loops (`for(i = 0; i < size; i++)` and `for(j = i+1; j < size; j++)`) to compare each element of the array with all the other

elements in the array. If any two elements are found to be equal, the function returns 1 indicating that a duplicate key exists in the array. In the main function, the program prompts the user to enter

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

Q6. Write a program to check if there exist a duplicate key in the array?

Here is an example of a program in C that will check if there exists a duplicate key in an array using a hash table:

```
#include
#include
#define MAX_SIZE 100

bool duplicateExist(int arr[], int size) {
    int i;
    bool hashTable[MAX_SIZE] = {false};
    for(i = 0; i < size; i++) {
        if(hashTable[arr[i]]) {
            return true;
        }
        hashTable[arr[i]] = true;
    }
    return false;
}

int main() {
    int arr[MAX_SIZE], size, i;
    printf("Enter size of the array: ");
    scanf("%d", &size);
    printf("Enter elements in the array: ");
    for(i=0; i
```

In this program, the function `duplicateExist(int arr[], int size)` takes an array of integers and its size as input, and uses a hash table (an array `hashTable[]` of size `MAX_SIZE`) to keep track of the elements of the input array.

The function iterates through the input array, checking if the current element has already been seen (i.e if the corresponding `hashTable[]` element is set to true). If it has, then the function returns true indicating that a duplicate key exists in the array. Otherwise, it sets the corresponding `hashTable[]` element to true for the current element and continues iterating through the array.

In the main function, the program prompts the user to enter the size of the array and the elements of the array. Then the `duplicateExist(arr, size)` is called to check if there exist duplicate key or

not. Based on the returned value from the function, the main program prints the appropriate message whether the duplicate key exists or not.

 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

here's a program in C that calculates the sum of the diagonal of a matrix:

```
#include

int main() {
    int n, i, j, sum = 0;
    printf("Enter the number of rows and columns: ");
    scanf("%d", &n);
    int mat[n][n];
    printf("Enter the matrix elements:\n");
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            scanf("%d", &mat[i][j]);
        }
    }
    for (i = 0; i < n; i++) {
        sum += mat[i][i];
    }
    printf("The sum of diagonal elements is: %d\n", sum);
    return 0;
}
```

Here's how the program works:

1. The program starts by prompting the user to enter the number of rows and columns of the matrix.
2. The program then declares a 2D array `mat` with `n` rows and `n` columns.
3. The user is then prompted to enter the matrix elements. The program reads the user input and stores the values in the `mat` array.
4. The program then loops through the diagonal elements of the matrix, i.e. `mat[0][0]`, `mat[1][1]`, `mat[2][2]`, etc., and adds their values to the variable `sum`.
5. Finally, the program prints the sum of the diagonal elements.

Note that in this program, we assume that the matrix is square (i.e. it has the same number of rows and columns). If the matrix is not square, this program will not work correctly.

TCS Ninja Interview Questions FAQ

How to prepare for TCS ninja interview?



 **Access Live Classes, Mocks, PYP & Notes for 375+ Exams!**

Is TCS Ninja interview easy?



[Report An Error](#)

Important Links

Overview

Paytm Interview Questions

REST API Interview Questions

[TCS Ninja Interview Questions](#)

E-mail Marketing Interview Questions

JavaScript Array Interview Questions

SEO Interview Questions For Freshers

JavaScript Closure Interview Question

JavaScript oops Interview Questions And Answers

JavaScript Hoisting Interview Questions

JavaScript Logical Interview Questions

Common JavaScript Promise Interview Questions

JavaScript Design Pattern Interview Questions

JavaScript UI Developer Interview Questions