**Object-Oriented Programming Questions:**

* What is Object-Oriented Programming (OOP)?
* What is a Class in Python?
* What is an Object in Python?
* What is Inheritance?
* What is Encapsulation?
* What is Polymorphism?
* What is the difference between \_\_init\_\_ and \_\_new\_\_?
* What are @classmethod and @staticmethod?
* What is Method Overriding?
* What is Multiple Inheritance?

**Difference-Based Questions:**

* What is the difference between a Class and an Object?
* What is the difference between a Method and a Function in Python?
* What is the difference between Inheritance and Composition?
* What is the difference between Method Overloading and Method Overriding?
* What is the difference between a Class Variable and an Instance Variable?
* What is the difference between Abstract Classes and Interfaces?

**Algorithm-Related Questions**:

* What is the difference between Linear Search and Binary Search?
* How do you find the time complexity of an algorithm?
* What is the difference between a Stack and a Queue?
* Explain the concept of recursion and provide an example.
* What is the difference between Merge Sort and Quick Sort?
* How do you implement a linked list in Python?
* What is Dynamic Programming, and how does it differ from Divide and Conquer?
* How would you detect a cycle in a linked list?
* What is the Big-O notation, and why is it important?
* How do you reverse a linked list?
* Explain the difference between Depth-First Search (DFS) and Breadth-First Search (BFS).
* What is a hash table, and how does it work?

**ANSWERS**

**Object-Oriented Programming Questions:**

1. What is Object-Oriented Programming (OOP)?

OOP is a programming paradigm that uses objects and classes to organize code, promoting modularity, reuse, and encapsulation.

1. What is a Class in Python?

A class is a blueprint for creating objects, defining their attributes and methods.

1. What is an Object in Python?

An object is an instance of a class that contains data and methods defined by the class.

1. What is Inheritance?

Inheritance allows a class to derive properties and methods from another class.

1. What is Encapsulation?

Encapsulation is the bundling of data and methods that operate on the data, restricting direct access to some components.

1. What is Polymorphism?

Polymorphism allows methods to behave differently based on the object invoking them.

1. What is the difference between \_\_init\_\_ and \_\_new\_\_?

\_\_init\_\_ initializes an object after it’s created; \_\_new\_\_ creates the object itself.

1. What are @classmethod and @staticmethod?

@classmethod takes the class as its first argument; @staticmethod doesn’t take any special first argument and is not bound to an instance.

1. What is Method Overriding?

Method overriding occurs when a subclass provides a specific implementation of a method already defined in its superclass.

1. What is Multiple Inheritance?

Multiple inheritance allows a class to inherit from more than one parent class.

**Difference-Based Questions:**

1. What is the difference between a Class and an Object?

A class is a blueprint; an object is an instance of that blueprint.

1. What is the difference between a Method and a Function in Python?

A method is a function defined within a class and operates on instances; a function is independent and not associated with a class.

1. What is the difference between Inheritance and Composition?

Inheritance derives from a parent class (is-a relationship); composition contains objects from other classes (has-a relationship).

1. What is the difference between Method Overloading and Method Overriding?

Method overloading refers to multiple methods with the same name but different parameters (not directly supported in Python); method overriding allows a subclass to change the behavior of a method inherited from its superclass.

1. What is the difference between a Class Variable and an Instance Variable?

A class variable is shared among all instances of the class; an instance variable is unique to each object.

1. What is the difference between Abstract Classes and Interfaces?

In Python, abstract classes provide some implementation and can have abstract methods; Python doesn't have true interfaces, but abstract classes can serve a similar role by defining methods without implementation.

**Algorithm-Related Questions:**

1. What is the difference between Linear Search and Binary Search?

Linear search checks each element sequentially; binary search divides the sorted list and eliminates half in each step.

1. How do you find the time complexity of an algorithm?

Analyze the number of operations relative to the input size, expressed using Big-O notation.

1. What is the difference between a Stack and a Queue?

A stack follows LIFO (Last In, First Out); a queue follows FIFO (First In, First Out).

1. Explain the concept of recursion and provide an example.

Recursion is a function that calls itself to solve smaller instances of a problem. Example: Calculating factorial (factorial(n) = n \* factorial(n-1)).

1. What is the difference between Merge Sort and Quick Sort?

Merge Sort divides the array into halves and merges sorted halves; Quick Sort selects a pivot and partitions the array around the pivot.

1. How do you implement a linked list in Python?

Use a class for nodes with attributes for data and the next node, and a class for the linked list to manage nodes.

1. What is Dynamic Programming, and how does it differ from Divide and Conquer?

Dynamic Programming solves overlapping subproblems using memorization; Divide and Conquer solves independent subproblems.

1. How would you detect a cycle in a linked list?

Use Floyd’s Cycle Detection algorithm (Tortoise and Hare) where two pointers move at different speeds.

1. What is the Big-O notation, and why is it important?

Big-O notation describes the upper bound of an algorithm’s time complexity, helping evaluate efficiency.

1. How do you reverse a linked list?

Iterate through the list, reversing the next pointers of the nodes.

1. Explain the difference between Depth-First Search (DFS) and Breadth-First Search (BFS).

DFS explores as far as possible along a branch before backtracking; BFS explores all neighbors at the current depth before moving to the next level.

1. What is a hash table, and how does it work?

A hash table stores key-value pairs, using a hash function to compute an index where each value is stored for fast retrieval.