1. **Explain Big O notation and how it helps in analyzing algorithms.**

Big O notation is a mathematical notation used to describe the upper bound of an algorithm's time complexity. It provides a way to express how the runtime of an algorithm grows relative to the size of the input. It helps in analysing and comparing the efficiency of different algorithms.

* **O(1)**: Constant time complexity. The runtime does not change with the size of the input.
* **O(n)**: Linear time complexity. The runtime grows linearly with the size of the input.
* **O(log n)**: Logarithmic time complexity. The runtime grows logarithmically with the size of the input.
* **O(n^2)**: Quadratic time complexity. The runtime grows proportionally to the square of the input size.

1. **Describe the best, average, and worst-case scenarios for search operations.**

* **Best Case**: The search operation finds the target element in the first position or at a position where it is most accessible.
* **Average Case**: The search operation finds the target element somewhere in the middle of the search space.
* **Worst Case**: The search operation has to go through the entire search space before finding the target element or concludes it is not present.

1. **Compare the time complexity of linear and binary search algorithms**

**Time Complexity**

1. **Linear Search**:
   * **Best Case**: O(1) – Target found at the first position.
   * **Average Case**: O(n) – Target found somewhere in the middle.
   * **Worst Case**: O(n) – Target not found or at the end.
2. **Binary Search**:
   * **Best Case**: O(1) – Target found at the middle.
   * **Average Case**: O(log n) – Target found after multiple divisions of the array.
   * **Worst Case**: O(log n) – Target not found after the complete division process.
3. **Compare the time complexity of linear and binary search algorithms**

* **Binary Search** is more suitable when the data is sorted and search operations are frequent, as it provides logarithmic time complexity, which is much faster than linear search for large datasets.
* **Linear Search** is simpler but less efficient for large datasets where sorted data and frequent searches make binary search advantageous.