1. **Understand Asymptotic Notation:**
   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 as a function of the input size. It helps in analyzing and comparing the efficiency of different algorithms by focusing on their growth rates and scalability.

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

**Best-case:** The element is found immediately.

**Average-case:** The element is found after examining half the data on average.

**Worst-case:** The element is not found or is the last element.

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

**Linear Search:** Best case-> O(1) , Average case-> O(n) & Worst case-> O(n)

**Binary Search:** Best case-> O(1) , Average case-> O(log n) & Worst case-> O(log n)

* 1. **Discuss which algorithm is more suitable for your platform and why.**

Binary search is generally better than linear search for large sorted datasets as it is faster due to its divide-and-conquer approach.