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

Ans:-Big O notation describes the upper bound of an algorithm’s growth rate in the worst case as a function of input size n.

Example:-

O(1) Constant time

O(log n) Logarithmic time

O(n) Linear time

2)Describe the best, average, and worst-case scenarios for search operations.

Ans:-

For Linear Search

Best Case:-O(1)

Average Case:-O(n)

Worst Case:-O(n)

For Binary Search

Best Case;-O(1)

Average Case:-O(log n)

Worst Case:-O(log n)

4) Analysis of our question is that

The time complexity of Linear search in worst case is O(n)

The time Complexity of Binary Search in Worst case O(log n)

I would prefer to use Binary search for our platform because Binary search takes O(log n) time which is much faster than linear search which takes O(n) in worst case scenario. Since the search functionality of ecommerce platform need to be optimized and fast for large amount of data we should use the Binary search algorithm.with increase in product the values of n will increase and linear search timming increases drastically and hence I prefer binary search under such condition.