**ANALYSIS**

**(A)** **Compare the time complexity of linear and binary search algorithms.**

**LINEAR SEARCH:**

Time complexity of Best case is O(1).

The time complexity of average case is O(n/2) ≈ O(n)

The time complexity of worst case O(n),that means element found in the end or not found

The Space Complexity is O(1),no extra space is used.

Linear Search works by scanning each element one by one until the required element is found

**BINARY SEARCH:**

Best case is O(1), when element is found at the middle index.

The average or worst case is O(log n).

It works by dividing a sorted array into halves, and eliminates the other half.

Their comparison:

|  |  |  |
| --- | --- | --- |
| Feature | LINEAR SEARCH | BINARY SEARCH |
| Requires Sorting | NO | YES |
| Performance | Slow | Fast |
| Worst case | O(n) | O(log n) |

**(B) Discuss which algorithm is more suitable for your platform and why**.

-> For an e-commerce platform with a large and frequently searched product catalog, **Binary Search is the better choice than Linear Search** as long as the data is kept sorted.  
It gives **logarithmic time complexity**, improving performance significantly over linear search.