## Step 1: Understand Asymptotic Notation

1. 1. What is Big O Notation?

Big O notation describes the upper bound of an algorithm’s runtime as input size grows. It gives a high-level understanding of how fast or slow an algorithm behaves in different scenarios.

1. 2. Best, Average, and Worst-Case Scenarios

* Best Case – Fastest execution time (e.g., first match in a search).
* Average Case– Expected time taken over multiple inputs (e.g., match in middle).
* Worst Case – Longest possible execution time (e.g., no match or last item).

## Step 4: Analysis

### Comparison of Search Algorithms:

|  |  |  |
| --- | --- | --- |
| Search Type | Time Complexity | Best Use Case |
| Linear Search | O(n) | Unsorted data, small arrays |
| Binary Search | O(log n) | Sorted arrays, large datasets |

Which algorithm is more suitable for your platform and why.

• For large and sorted product listings, I will prefer Binary Search for optimal speed.

• For small or unsorted datasets, I can use Linear Search for simplicity.