WEEK\_1\_ALGORITHMS\_DATA\_STRUCTURES

Exercise 2: **E-commerce Platform Search Function**

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

Big O notation describes the upper bound of an algorithm's time complexity, giving an idea of the worst-case scenario. It helps in understanding how the runtime of an algorithm grows with the input size. It basically tells us about how many iterations will an algorithm run in the worst case scenario. It helps us in analyzing algorithms since in that way we get to know that how much time a particular algorithm will take to run in the worst case. If it is really large, we can then change the algorithm.

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

Best Case: The condition where the algorithm performs the minimum number of operations. Ex- O(1).

Average Case: The expected number of operations considering all possible inputs. Ex- O(log n).

Worst Case: The condition where the algorithm performs the maximum number of operations. Ex- O(n).

Where n – size of the data structure.

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

|  |  |  |  |
| --- | --- | --- | --- |
|  | Best case | Average case | Worst case |
| Linear Search | O(1) | O(n) | O(n) |
| Binary Search | O(1) | O(log n) | O(log n) |

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

Binary search is more suitable for the e-commerce platform due to its superior performance on large datasets, despite the need for sorted data. The efficiency gained in search operations outweighs the cost of sorting, especially when search operations are frequent.