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

**Scenario:**

You are working on the search functionality of an e-commerce platform. The search needs to be optimized for fast performance.

**1. Understand Asymptotic Notation**

**What is Big O Notation?**

* **Big O notation** describes the **upper bound** of an algorithm's runtime as the input size grows.
* It helps evaluate the **scalability** and **efficiency** of algorithms.

**Best, Average, and Worst Cases for Search:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Best Case** | **Average Case** | **Worst Case** |
| **Linear Search** | **O(1) — first item** | **O(n/2) → O(n)** | **O(n) — last or not found** |
| **Binary Search** | **O(1) — middle item** | **O(log n)** | **O(log n)** |

Binary Search is significantly faster, but requires the array to be sorted.

**Analysis:**

**Time Complexity Comparison**

|  |  |  |  |
| --- | --- | --- | --- |
| **Algorithm** | **Best Case** | **Average Case** | **Worst Case** |
| **Linear Search** | **O(1)** | **O(n/2)** | **O(n)** |
| **Binary Search** | **O(1)** | **O(log n)** | **O(log n)** |

**When to Use Which?**

|  |  |
| --- | --- |
| **Situation** | **Use** |
| **List is small or unsorted** | **Linear Search (no sorting needed)** |
| **List is large and sorted** | **Binary Search** |
| **Frequent insertions or updates** | **Linear preferred, unless sorting is re-applied** |
| **Frequent queries/searches** | **Binary Search (after sort once)** |

**For an e-commerce platform with thousands of products and frequent user searches, Binary Search is more suitable — if the data is kept sorted.**