**WEEK1\_ALGORITHMS\_DATA\_STRUCTURES**

**EXERCISE-6**

**Understand Search Algorithms**

**Linear Search:**

* **Algorithm:** Checks each item in the list sequentially until the target is found or the end of the list is reached.
* **Time Complexity:** O(n), where n is the number of elements in the list.
* **When to Use:** Suitable for small or unsorted datasets where binary search is not applicable.

**Binary Search:**

* **Algorithm:** Divides the sorted list into halves and checks the middle element. If the target is less than the middle element, it searches the left half, otherwise, it searches the right half.
* **Time Complexity:** O(log n), where n is the number of elements in the list.
* **When to Use:** Suitable for large, sorted datasets.

**Analysis**

**Time Complexity:**

* **Linear Search:** O(n) - Scans each element in the list. Suitable for small or unsorted lists.
* **Binary Search:** O(log n) - Requires sorted data but much faster for large datasets compared to linear search.

**When to Use Each Algorithm:**

* **Linear Search:** Use when the dataset is small or unsorted. Easy to implement and does not require sorting.
* **Binary Search:** Use when the dataset is large and sorted. Provides faster search times but requires pre-sorting.