Luke Renton 2540440

Lab 2 write-up

Testing practices

Test Setup

The test was conducted on lists of distinct random integers. The sizes of the lists were chosen to cover a wide range of values, from 100 to 1,000,000 elements.

For consistency, the same array was used in each *n* step for all three cases (best, average, and worst). This ensures that variations in array content do not affect the timing results, allowing for a more accurate comparison across different cases.

To prevent any anomalies in the timing results, each search operation was repeated five times on the same array, and the average time was recorded. This repetition helps to smooth out any variations in system performance that might occur during individual runs.

Best Case

The best-case scenario was tested by placing the search key at the first position of the array. Since the linear search algorithm finds the key on the first comparison, the performance for the best case is expected to be constant, $O(1)$.

Average Case

The average case scenario was tested by selecting 10 random keys from the array and measuring the time taken to search for each key. The times were then averaged to get a representative value for the average case. This method ensures that we account for the fact that the key could be located anywhere in the array, thus providing a more accurate measure of the algorithm's typical performance. We still expect the algorithm to behave in a linear time complexity, $O(n)$ – although the average case has a smaller constant factor than in the worst case.

Worst Case

The worst-case scenario was tested by placing the search key at the last position of the array (this case is arbitrarily identical to the case where the key is not in the list). In this case, the linear search algorithm must traverse the entire array to find the key (or not find the key), resulting in a linear time complexity, $O(n)$.