GCIS 123. Sect 600 Activity 3

Group 6 (Mohamed Nagla, Munzier, Hishaam)

Mohammed Nagla's Repository link - https://github.com/mohamed19874/Mohamed-nagla

Hishaam's Repository link - https://github.com/hb7464/hb7464

Munzir's Repository link - https://github.com/munzir1910/GCIS123

**Phase 4**

|  |  |  |
| --- | --- | --- |
| Test Cases | Screenshots | Which is faster? |
| Case 1: Size 10 |  | Linear |
| Case 2: Size 100 |  | Binary |
| Case 3: Size 1000 |  | Binary |
| Case 4: Size 10000 |  | Binary |
| Case 5: Size 100000 |  | Binary |

Additional Question Answer

Sorting algorithms can affect the total complexity of the program, as certain search algorithms like binary search aren’t viable, which is a problem for larger datasets as unsorted data leads to inefficient searching. Thus, we need to have an effective sorting algorithm as well to maximize efficiency in searching as well.

Binary search is better than linear search because it divides the array in half through each iteration, meaning that there is less data to sort through, but in linear search it iterates through each element one by one. This results in binary search having a time complexity of O(log n) and linear search having a time complexity of O(n).

Efficient sorting methods enhance performance and, sorting is a pre-cursor to binary search and is required, and makes the whole program faster by reducing the time complexity.