**Kathmandu University**

Department of Computer Science and Engineering

Dhulikhel, Kavre



Algorithms and Complexity (COMP 314) – Lab 1

Submitted To:

Dr. Rajani Chulyadyo

Department of Computer Science and Engineering

Submitted By:

Mani Dumaru (15)

CE-2019 Year/Sem: 3rd Year/2nd Sem

Submission Date: 12th April, 2022

Implementation, Testing and Performance measurement of Linear Search and Binary Search Algorithms.

**Implementation**

Linear Search Algorithm:

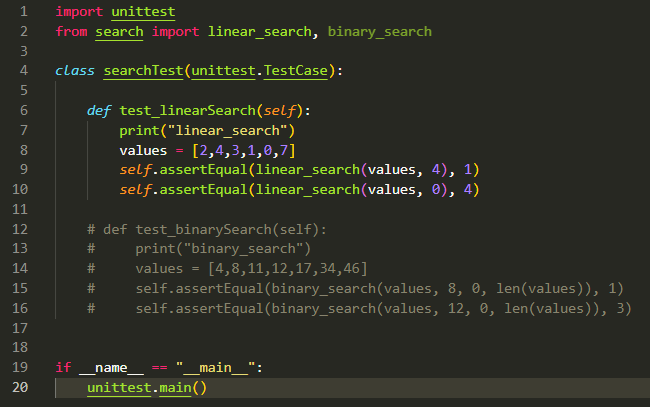


Binary Search Algorithm:

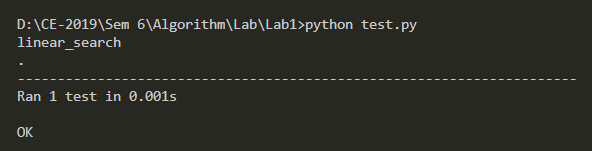


**Testing**

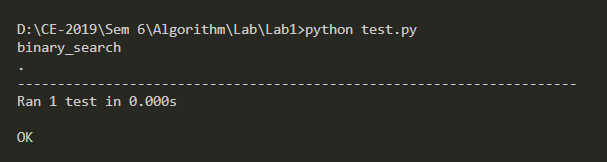
*Testing of Linear Search Algorithm:*



*Output of Testing Linear Search*

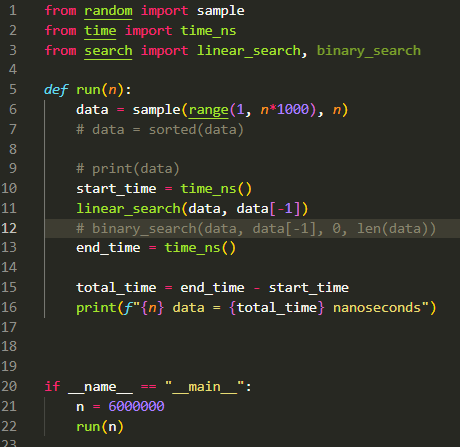


*Output of Testing Binary Search*



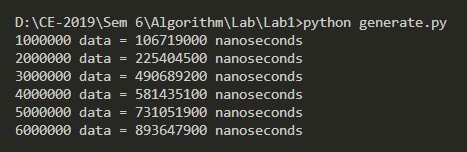
**Performance**

Code for generating random data and applying Linear and Binary Search algorithm to check the time required.

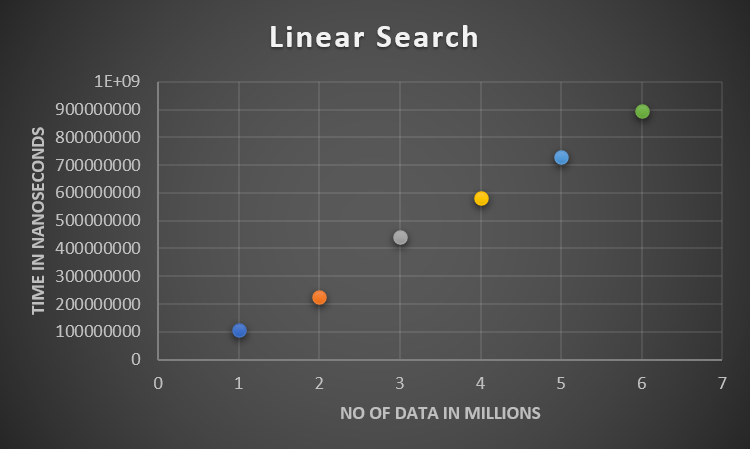


This code generates given number of random data and stores it in an array “data”. This array is then passed to both linear\_search and binary\_search algorithm to search for the very last element of array. Total time for searching is calculated using the time\_ns function from time. Time vs Number of data graph is then plotted to observe the curve.

***Output for Linear Search:***

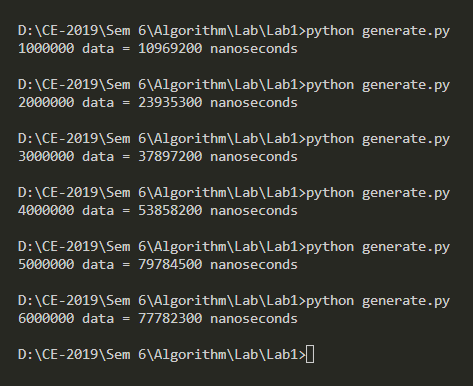


***Graph for Linear Search:***

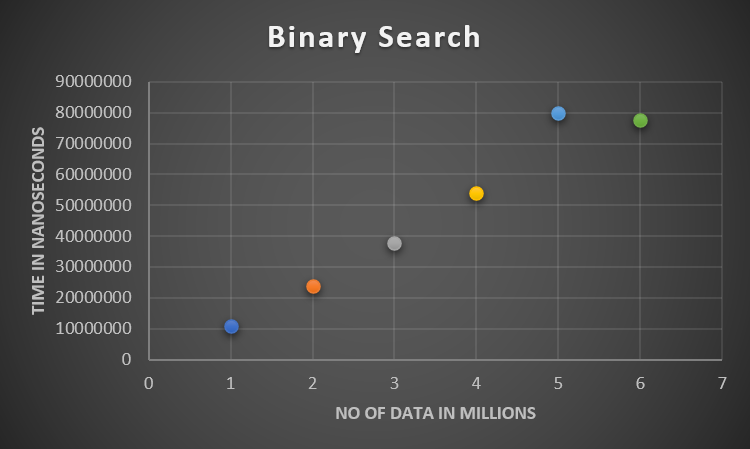


From this graph, we can observe that the linear search algorithm gives a linear path while plotting no. of data vs time graph. This means that the time complexity for linear graph is O(n).

***Output for Binary Search:***



***Graph for Binary Search:***

******

From this graph, we can see that the time complexity for binary search is O(log n).

Conclusion:

Hence, algorithm for linear and binary search was implemented and tested if they gave the appropriate results. Each of their performance was also tested based on the time they took to search the provided data from the list of millions of randomly generated data. Time complexity of Linear Search algorithm was found to be O(n) and O(log n) for Binary Search.