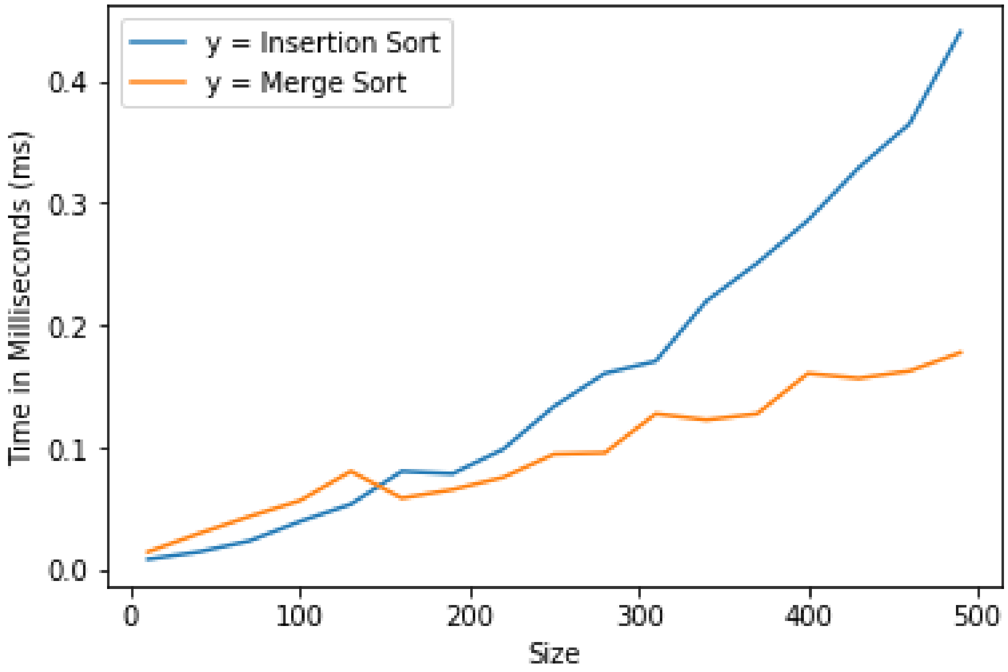
Experimentation

Insertion and Merge sort Analysis



The above graph indicates the cross over range at 160 to 180 sizes. As the size increases insertion sort takes more time than merge sort

This indicates that for smaller inputs insertion sorts works better than merge sort

For best case scenario i.e when array is already sorted we get O(n) complexity. For merge sort using divide conquer in best case we can get O(n)

Machines specifications

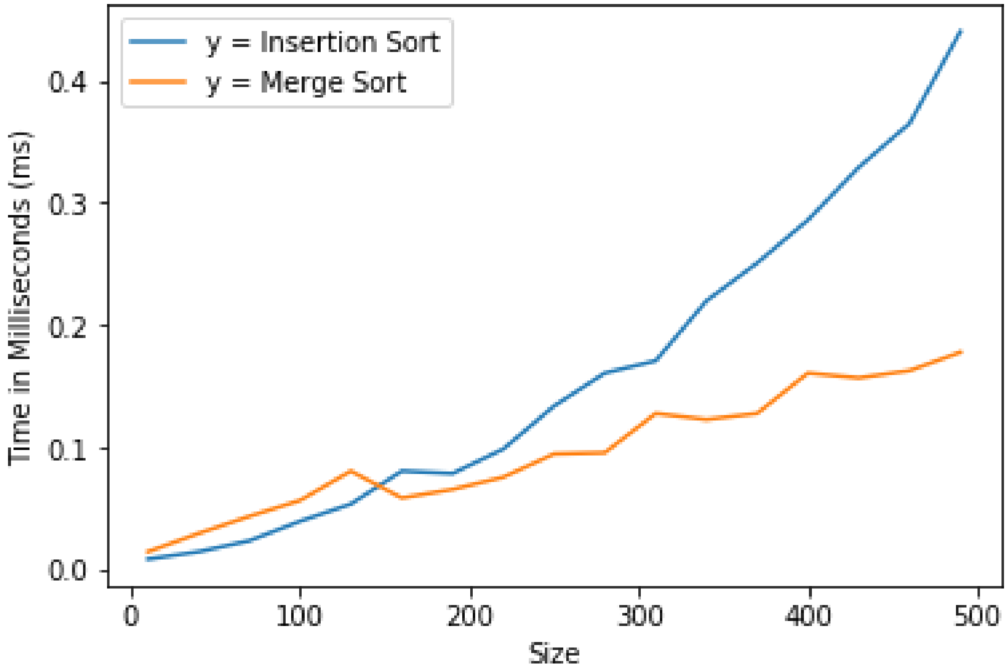
- Memory – 200GB
- Processor – Intel Core i7 – 3632OM CPU @2.20 GHz * 8
- Os Type – 64 bit

Finding maximum element

In finding maximum element in an array we first sort the array in increasing array and find the last element in the array

Sorting for insertion works faster for lower inputs unlike merge sort

Once sorting is done it takes constant amount of time to fetch the last element of the array



The yellow line above indicates the time curve for merge sort and the blue line indicates the time curve for insertion sort. The cross over is indicated around 150 size

As the input size increases the merge sort becomes more efficient than insertion sort

Machines specifications

- Memory – 200GB
- Processor – Intel Core i7 – 3632OM CPU @2.20 GHz * 8
- Os Type – 64 bit