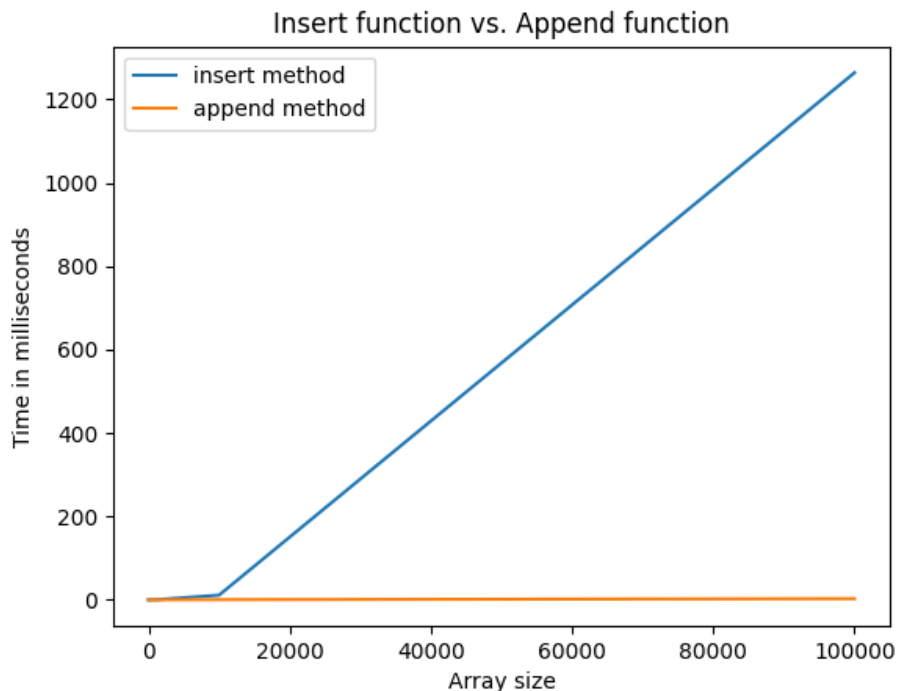


Unit-7 assessment part 1 Notes



Results for the tinyArray

insert 24.833 μ s
append 65.5 μ s

Results for the smallArray

insert 13.5 μ s
append 14.25 μ s

Results for the mediumArray

insert 151.75 μ s
append 53.583 μ s

Results for the largeArray

insert 11.184875 ms
append 467.25 μ s

Results for the extraLargeArray

insert 1.263898708 s
append 2.7075 ms

Using the results from the runtime.js file, I created a python file and used matplotlib to create a line graph showing how these two methods scale. As the graph shows, the insert method increases in runtime exponentially as the array entered grows in size, but the append method is constant. This shows that the append (.push) method is an $O(1)$ complexity and the insert (.unshift) method is an $O(n)$ complexity.

The append method scales better because when you add an item to the array, it only has to increase the size of the array by one and insert the new item into the newly created slot at the end of the array. The insert method has to go through each item of the array and move it over to create a space at the beginning of the array for the new item that is being added. This means that as the array grows the amount of operations grows with it because there are more items that have to be moved over before the new item can be inserted into the first position.