Results for the extraLargeArray
insert 1.5047154 s
append 3.8844 ms

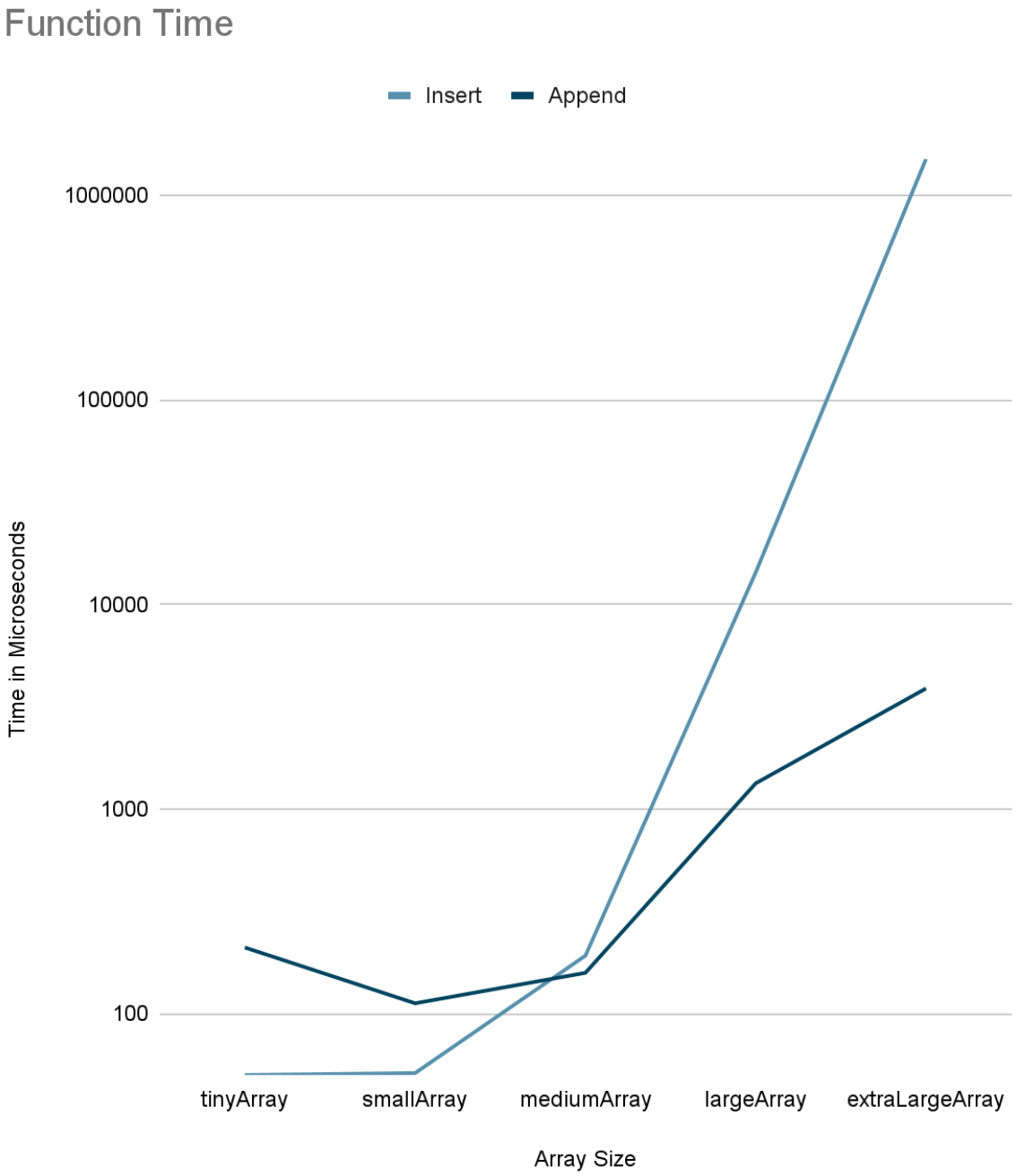
Results for the largeArray
insert 14.3329 ms
append 1.3353 ms

Results for the mediumArray
insert 191.9 µs
append 158.2 µs

Results for the smallArray
insert 51.2 µs
append 112.3 µs

Results for the tinyArray
insert 50.1 µs
append 210.4 µs

From looking at the results of the tests, the doublerInsert function scales much faster than the doublerAppend function. At the beginning of the tests with the beginning with tinyArray, the doublerAppend function takes 4 times as long to produce the results than the doublerInsert function. From there, the doublerInsert function's time exponentially grows, being roughly 395 times the rate of the time it takes for the doublerAppend function to produce the results of the extraLargeArray. From the beginning and according to the results, it would seem the doublerInsert function is faster at producing, though as the size of the array grows larger, that isn't the case. The result of this could be due to the functionality of the .unshift array method. This method is having to shift each index in the array over so that it can insert the given numbers along with returning the length of that shifted array. In return, this causes the time to produce results to grow as the length of the array grows. In conclusion, the doublerAppend function is much faster with larger arrays compared to the doublerInsert function.


Function Time
Insert
Append
1000000
100000
10000
1000
100
Time in Microseconds
tinyArray
smallArray
mediumArray
largeArray
extraLargeArray
Array Size