

- Results for the extraLargeArray  
**insert:** 552.3149 ms  
**append:** 2.701 ms
- Results for the largeArray  
insert 7.0887 ms  
append 490.7 µs
- Results for the mediumArray  
insert 110 µs  
append 108.1 µs
- Results for the smallArray  
insert 37.7 µs  
append 78.2 µs
- Results for the tinyArray  
insert 25.6 µs  
append 64.1 µs

Array	Using .unshift()	Using .push()
EXA	<b>insert:</b> 552.3149 ms	<b>append:</b> 2.701 ms
LA	insert 7.0887 ms	append 490.7 µs
MA	insert 110 µs	append 108.1 µs
SA	insert 37.7 µs	append 78.2 µs
TA	insert 25.6 µs	append 64.1 µs

Because of the nature of how .push() and .unshift() functions work I see that as the size of array grows the runtime of the function that uses .unshift() becomes larger than the runtime of the function that uses .push(). Initially when the size of the array is small, .push() methods have longer runtimes. When the size of an array is medium, the runtime of .push() is equal to the runtime of .unshift(). As the array size gets larger, the runtime of .unshift() starts increasing dramatically making it 4 times longer than .push() runtime in ms. The .push() is preferred because it scales better no matter the size of the array.

Function that uses `.unshift()` has to move all the elements that are already present in the Array. So it would have to move the element at position  $N$  to position  $N+1$ ,  $N1$  to  $N1+1$ ,  $N2$  to  $N2+1$  etc. But, the function that uses `.push()` has to insert an element at the end of the array, so none of the Array elements' index has to change. That's why it is more efficient to use `.push()` to get better runtime.

To sum this up, `.unshift()` has worse time complexity because for each new insertion each element in the array has to be relocated (shifted) to make space for the new element and that becomes a lot of operations (expensive) as the array gets larger. On the other hand, `.push()` knows exactly where to locate a new element which is to the end of the array and does the same amount of work regardless of the size of the array.