

Runtime Results

Array	Doubler Insert ($O(n^2)$)	Doubler Append ($O(n)$)
Extra large array (100,000)	775.83 ms	2.10 ms
Large array (10,000)	6.34 ms	434.46 μ s
Medium array (1000)	127.88 μ s	95.80 μ s
Small array (100)	30.00 μ s	68.67 μ s
Tiny array (10)	25.33 μ s	62.00 μ s

ms = millisecond (1000 of a sec) = 1000 microseconds

μ s = microseconds (1 millionth of a sec)

Doubler insert requires less time for smaller inputs.

Insert time is greater than append time. As the input increases the time increases where unshift and push in Doubler Insert take the most time for the function to complete. The Doubler Insert function requires a loop through an array and then each element in array has to increment in order to push new elements in the front. Whereas, doubler append requires one loop in an array and new elements are pushed into the end of the array not causing the elements to move. Therefore, doubler append requires less time as input increases compared to doubler insert giving it a better runtime complexity.