Append results
extraLarge array = 4.107039 ms faster
tinyArray   = 89.508 µs
Smallarray = 105.783 µs
mediumArray = 242.831 faster
largeArray = 638.606 µs faster

Insert results
extraLarge array = 1.053090267
tinyArray   = 49.215 µs.   Faster
Small array = 54.952 µs. Faster
mediumArray =248.481
largeArray = 6.812538 ms

Initially the double insert function was faster than the double append function.
Then after passing the medium array for both, they were very much near each other but, that's when the append function was faster just by a very little amount. But after that the append function was much faster when passing in the large and extra large arrays. So overall the append function is much faster and more efficient as you work with larger data.

Extra credit

I think the reason why the push method is faster than the unshift method is because of how data works behind the scenes in the memory. When you push an item to an array in memory the previous array memory aren't affected but when you unshift an item to an array it is added to the beginning of the stack and pushes all the items in the array by one, which makes the computer work more hence takes a longer time to run the code