Results for the extraLargeArray
insert.unshift - 439.9488 ms
append.push - 2.5377 ms

Results for the largeArray
insert.unshift - 4.6728 ms
append.push - 677.5 μs

Results for the mediumArray
insert.unshift - 182.6 μs
append.push - 152.8 μs

Results for the smallArray
insert.unshift - 48.8 μs
append.push - 86 μs

Results for the tinyArray
insert.unshift - 35.4 μs
append.push - 84.7 μs

The Insert Function definitely begins taking much longer with larger arrays, this is especially noteable after the mediumArray. The Append Function handles the Time Complexity much smoother when dealing with larger arrays. However, there is slight to no difference in Time with the Append Function with the smaller arrays and in fact, takes longer the the Insert Function on those smaller Arrays.

From my understanding, the Insert Function is taking so much longer with the larger array due to the fact the data is being insereted with .unshift, which adds the data to the beginning of the array, therefore changing the index of every item in the array. Whereas the Append function is only pushing the data to the end of the array, where only the data is being considered instead of each item in the array.

.unshift = O(n)
.push = O(1)