| Array name | Append | Insert |
| --- | --- | --- |
| extraLargeArray | 3.8635 ms | 1.1542053 s |
| largeArray | 859.9 μs | 11.9054 ms |
| mediumArray | 209.5 μs | 251.7 μs |
| smallArray | 150 μs | 66.1 μs |
| tinyArray | 138.2 μs | 52.1 μs |

One thing that I notice about the pattern the two arrays follow is, of course as the arrays get bigger the runtime gets longer. The insert function seems to scale way worse than the append function. While the append array stays in the microseconds up until the largeArray, the insert array is almost a thousand times slower. Even with the extraLargeArray appending is still about three hundred times faster, so I believe the append function scales better.

Bonus: The reason that the insert scales worse than the append is due to the fact that the insert function uses unshift while the append function uses push. Unshift is worse for scaling because while it is adding it to the front, for every subsequent addition to the new array needs to push everything else back. While push from the append function adds to the end so nothing needs to be shifted around allowing it to scale better.