| **ArraySize** | **doublerAppend speed** | **doublerInsert speed** |  |  |
| --- | --- | --- | --- | --- |
| --------------- | ------------------- | ------------------- |  |  |
| extraLargeArray | 2.836 ms | 996.6941 ms |  |  |
| tinyArray | 36.6 μs | 88.5 μs |  |  |
| smallArray | 88.5 μs | 47.3 μs |  |  |
| mediumArray | 134.6 μs | 178.7 μs |  |  |
| largeArray | 493.9 μs | 8.5633 ms |  |  |

This analysis compares two different functions that both add numbers into an array but in their own unique way. DoublerAppend uses the .push() method which adds a new element into the end of the array while the doublerInsert uses the .unshift() method that adds the new element into the front of the array and shifts the other elements left. The results conclude that the .push() method is significantly faster. One thing to observe is that as the array size increases, the .unshift()’s speed time also increases – exponentially. Through research for an answer as to why, it is apparent the .unshift() method is slower because it needs to “unshift” all the elements to the left once the first element is added that requires an additional action.