**Timing Results for Extra Large Array:**
Insert: 1.6715499999999999 s
Append: 4.8542 ms

**TIming Results for Differently Sized Arrays:**

| **Methods** | Tiny | Small | Medium | Large | Extra Large |
|---|---|---|---|---|---|
| Insert (.unshift ) | 53.3 µs | 20.8 µs | 324.6 µs | 29.7556 ms | 1.7174252 s |
| Append (.push) | 130.6 µs | 21 µs | 60.2 µs | 964.9 µs | 6.7187 ms |

**Observation**
The patterns that I have observed are that the unshift method which is used in the doublerInsert function runs faster than the push method which is used in the doublerAppend function only if the array is very small. The smaller the array, the higher the probability of the unshift method being faster than the .push. As the arrays grow in size, the .push method becomes faster than the .unshift method.
Using unshift is only better when dealing with extremely small arrays.
Overall using push is more effective when dealing with arrays that contain more than one hundred numbers. This makes the push method the better of the two since it has far more range and can handle a larger load.