

1. In your notes document, take note of the timing result for the **extraLargeArray** results—comparing when the **extraLargeArray** is passed to **doublerAppend** and **doublerInsert**.

Timing result is

-Insert: 2.6400132 seconds

-Append: 8.536097 milliseconds

2. Notate these in your document in some kind table table so that you can easily compare the different values for the timers in relation to the size of the array that was passed into each function.

extraLargeArray	largeArray	mediumArray	smallArray	tinyArray
insert 3.297068076 s	insert 16.977776 ms	insert 447.922 µs	insert 22.45 µs	insert 31.951 µs
append 6.652627 ms	611.961 µs	append 1.955345 ms	append 15.997 µs	append 5.997 µs

3. Read over the results, and write a paragraph that explains the pattern you see. How does each function “scale”? Which of the two functions scales better? How can you tell?

The pattern that i am seeing is the smaller the array get the faster the functions append and insert. However, there is an edgecase for largeArray, it is very fast compared to the others. It does look like the insert array scaled better in the sense of getting consistently smaller. I can tell by running down the values from each size of array that was passed into the function. It functionally ran better and was able to get faster.