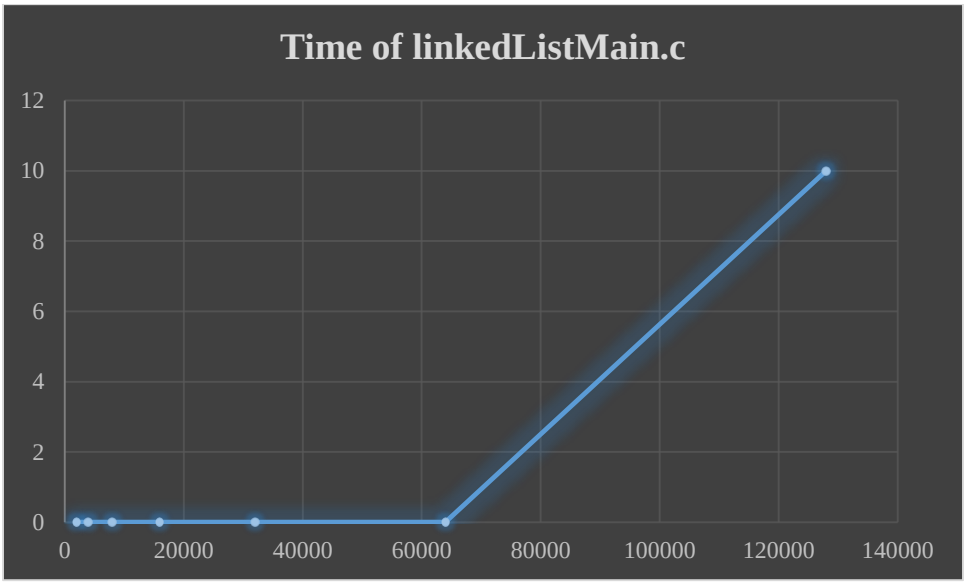
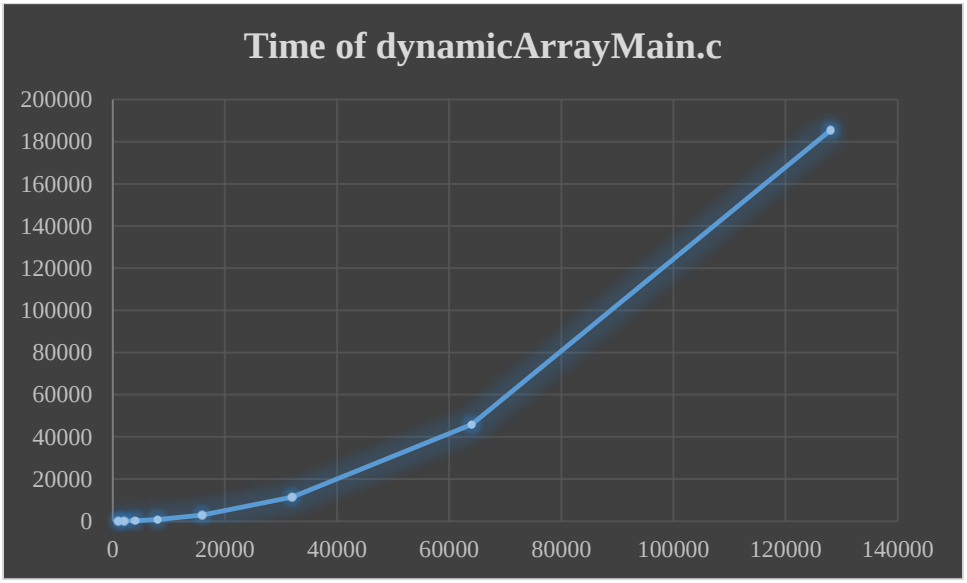
Rhea Mae Edwards
Part 2: Comparison

Time Comparison:

x-axis = Number of Elements
y-axis = Run Time (ms)

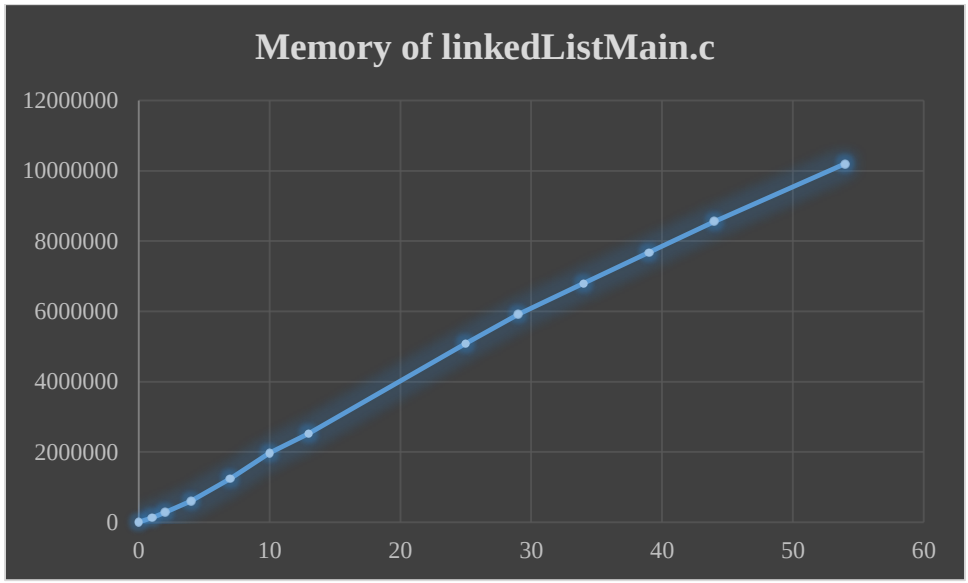


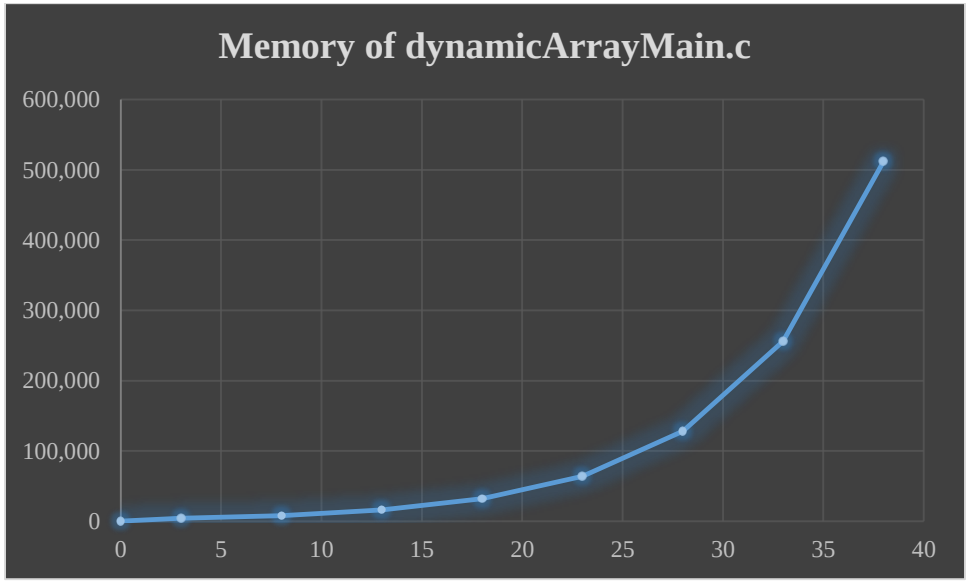


Rhea Mae Edwards
Part 2: Comparison

Memory Comparison:

x-axis = Number of Detailed Snapshot
y-axis = Amount of Memory (B)





Rhea Mae Edwards
Part 2: Comparison

Questions:

1. Which of the implementations uses more memory? Explain why.

linkedListMain.c used more memory, because you have extra characteristics such as pointers along with all the other aspects needed in order to make a linked list when compared to making a dynamic array.

2. Which of the implementations is the fastest? Explain why.

linkedListMain.c was the fastest because all you have to do is add single elements on either ends of the list with no need of copying or greatly increasing size by obtaining a great amount of space in memory for adding elements such as in a dynamic array. There is no capacity in a linked list that you have to worry about such as in a dynamic array.

3. Would you expect anything to change if the loop performed remove() instead of contains()? If so, what?

The difference that I would expect is the loop performed remove() instead of contains() is that the amount of memory for both programs would decrease, both switching back and forth to a constant maximum and constant minimum space in memory once an element is added and then removed. Also it may take a little bit longer for both programs to perform since they are both removing elements rather than just looking for a certain element.