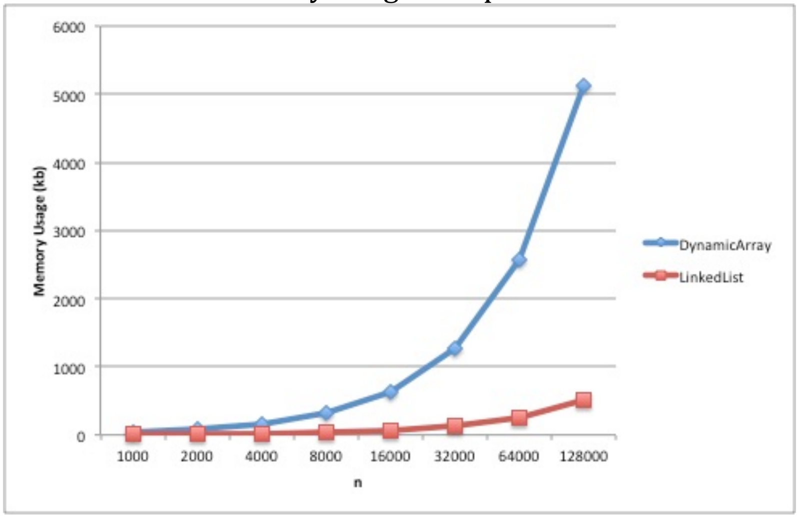
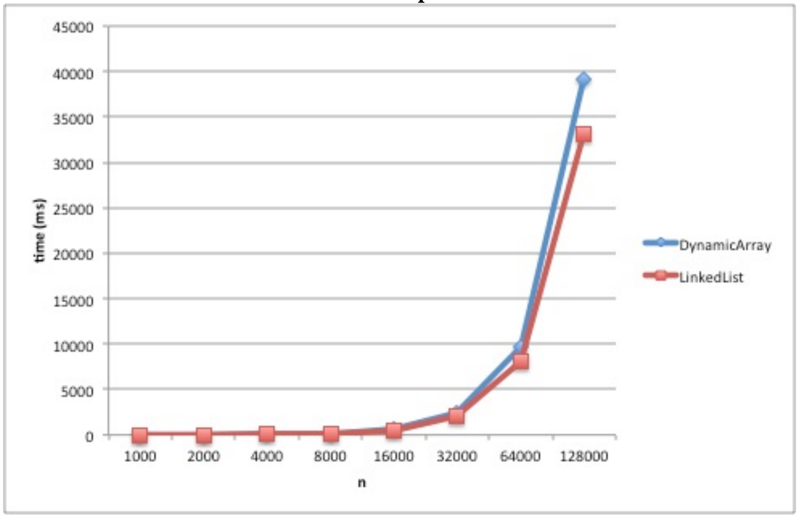
Pranav Ramesh & Sean Marty
CS 261-001
January 30, 2015
Assignment 3: Problem 2

Memory Usage Comparison



Time Comparison



Questions:

1. **Which of the implementations uses more memory? Explain why.**
   The linked list uses more memory because there is a whole collection of data for each node, where as there is only one piece of data for each node in the dynamic array. There is the fact that the size of the dynamic array doubles when there needs to be more space, but that is outweighed by each struct for a linked list node having 3 data pieces.

2. **Which of the implementations is the fastest? Explain why.**
   The dynamic array implementation is faster because accessing a random place in the array is O(1) where as the same operation in the linked list is O(n), which is less time efficient.

Pranav Ramesh & Sean Marty
CS 261-001
January 30, 2015
Assignment 3: Problem 2

3. **Would you expect anything to change if the loop performed remove() instead of contains()? If so, what? (Note, it's very easy to run this experiment given the code we've provided!)**
   The time to run would be faster for the linked list, because when you remove from a dynamic array it has to shift every item after the removed item back one (O(n) operation). The memory would skew slightly better in the linked list's direction (although maybe not make it more memory efficient) because when you remove from our current dynamic array it does not reduce the size of the array, where as removing from the linked list does free that piece of memory.