Fiona Liu

Fxl154

Final Project: Progress Report

1. IntArrayBag and IntLinkedBag data structures from HW3 were modified to be sorted in ascending order. The run times of 3 methods were analyzed – add, addAll, and remove. The procedure involved rewriting the add method to include sorting the target element into ascending order. The addAll method is then modified to implement the merge sort algorithm. The remove method implementation is the same from that of HW3. A main method is added to the IntArrayBag and IntLinkedBag class to demonstrate the implementation of these revised methods. The run times of each method are recorded, and then graphed with respect to its add, remove, and addAll implementation of a bag of length N. In our demonstration, N increments to 5000.
2. Big O run times and experimental graphs
   1. add() method:

For IntArrayBag, the time complexity is O(N) because the method needs to traverse through some length of an array before pushing the other elements to the next position.

For IntLinkedBag, the time complexity is O(N) because the method needs to find the value in the bag that is greater than the target element, which could be all the elements in the bag.

Graphs of experimental results:

* 1. addAll() method:

For IntArrayBag, the time complexity is O[(N1+N2)log(N1+N2)] where N1 elements are added to the list and N2 elements are already in the bag. N1 elements are first added to the end of the bag, before merge sort is called to sort the final array of (N1 + N2).

For IntLinkedList, the time complexity is O[([(N1+N2)log(N1+N2)] as the linked list elements (N1) are added to the start of the array. Similarly, merge sort is called to sort the final array.

However, there is a Stack Overflow exception when using merge sort with IntLinkedBag for N>3000. Need to further work on this to figure it out.

Graphs of experimental results:

* 1. remove() method:

For IntArrayBag, the time complexity is O(N) because the element is found, removed, and then all the elements before it are shifted back one position.

For IntLinkedBag, the time complexity is O(N) because the element is found, removed, and then all the elements before it are shifted one position.

Graphs of experimental results:

1. We have managed to compare the run times for add and remove methods after modifying the bag classes from HW3. We are having trouble with the implementation of addAll for the linked bag, due to a stac overflow exception. We will work on this implementation more to figure out how to go around this problem.
2. Fiona worked on the progress report, she also implemented the three methods for the IntArrayBag class. I worked on the presentation and implemented the three methods for IntLinkedBag. Once we have our mergesort algorithm done for IntLinkedBag, and then get the results for experimentation we will be done.