Lab 09 Report: Min Heap of Integers

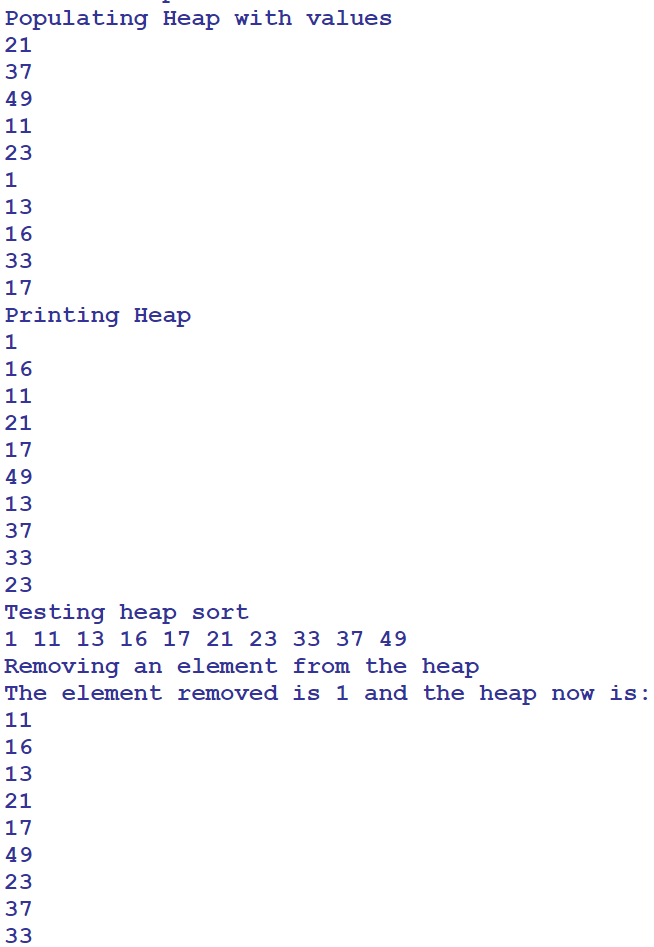
Problem

Implement a minimum (min) heap of integers, and write a driver to demonstrate its functionality. Min heaps work the same way as maximum (max) heaps except the a parent is always smaller than its children.

Proposed Solution

1. Make a class min heap of integers
   1. Insert: add a new element to the heap, and bubble up as needed.
   2. Delete: remove the first element from the heap and then reheapify
   3. heapsort: make a copy of the heap, the remove and print each element from the clone. Make sure this doesn’t modify the current heap.
   4. Print: prints the elements in the heap in breadth order

Tests and Results



Problems Encountered

This lab was very straight forward and the only issue encountered was making the Max heap of integers given in class into a min heap by changing the comparison signs within the bubbleUp and bubbleDown methods.

Conclusions and Discussion

This lab was straight forward and based on examples and code which was shown in class. This gets students prepared and familiar with min heap and their functionalities. Traversal of heaps is now understood, and the structure of them as well. This is shown in the insert method on how the insert method decides where to put the new data when comparing it to the existing tree.

Additional Questions

1. Demonstrate each step of inserting the elements 21, 27, 49, 11, 23, 1, 13, 16, 33, and 17 into this min heap.

Lets say its an int array like int[] arr = {21, 27, 49, 11, 23, 1, 13, 16, 33, 17}| which is 10 ints and for the length of it, heap.insert(arr[i]); this would then use its comparison methods inside to decide how the numbers go down into the heap.

1. Demonstrate each step of inserting the same elements in a max heap.

It would be the same process, but instead of making sure that the numbers that is being inserted into the array is less than the numbers it is being compared to, it would have to be greater than. Greatest number at top, any number greater than it on its right and any number less on the left, and it goes like this for each branch.

1. Show each step when deleting two elements from the min heap.

If you are deleting from the heap it checks to see where the value is and it the tree needs to be resorted. It does this by checking the sister/brother to the element being removed and the parents.