

Assignment 4  
Min-Heap

Due Oct 19

A min-heap is a heap in which the root has the smallest value in the heap. Normally one would implement a heap using a single array with the indexing indicating. In this assignment we will use either a vector or a map to represent. In our min-heap each node has a value and 0, 1 or 2 children heaps. The root node of a min-heap contains the smallest value in the heap. If you pick any node in the heap it has a value that is equal to or less than all the nodes in either the left or right sub-heaps of the node. See the figure 1 below for an example. To add a new value to the heap add it to the top of the heap. If the heap is empty the new value becomes the root. If heap is not empty then the smaller of the two values (current value in heap and the new value) is kept in the root. The larger of the two values is added the sub-heap with the smallest height. If both sub-heaps have the same height then the larger of the two values is added to the left heap. The process is repeated until a value is added to a new node at the bottom of the heap. (There are better ways of adding to a min-heap, but this is the way you are to use in this assignment.) Figure 2 is the result of adding 50 to the heap in figure 1. This of course is not the normal way of adding elements to a min-heap. This algorithm is not as efficient as the normal algorithm, so you would never implement it this way in real life. We are using this algorithm to reduce the chances you can find the solution on the internet.

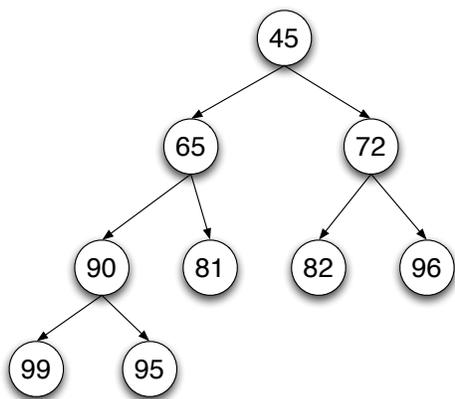


Figure 1

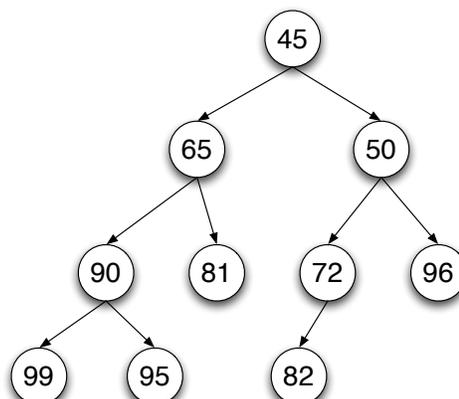


Figure 2

Your code is to have the following functionality:

1. Add elements to your min-heap. You can assume that the programmer will not mix different types in a single heap. There is no need to implement delete.
2. Print out the values in the heap in preorder (root, the left sub-heap, right sub-heap).

3. Print out the strings that end in "ing" in the heap in preorder.

Do not flatten your heap to traverse it.

## Grading

Item	Points
Working Code	30 points
Unit Tests	10 points
Comments	10 points
Quality of Code	10 points

**Working Code.** How well your code meets the functional requirements listed above accounts for 50% of the grade for the assignment.

**Unit Tests.** Having unit tests that cover the functionality of your code accounts for 1/6 of your grade. The unit tests are to be scalable. Tests using just print statements are not scalable.

### Turning in your Assignment

Zip up your project, which is assumed to have multiple files, and turn it in on-line.