

Assignment 2  
Max-Heap

Due Feb 6 23:59

A max-heap is a heap in which the root has the largest value in the heap. Normally one would implement a heap using an array. In this assignment you are to use pointers, not an array. In our max-heap each node has a value and 0, 1 or 2 children heaps. The root node of a max-heap contains the largest value in the heap. If you pick any node in the heap it has a value that is equal to or larger than all the nodes in either the left or right sub-heap 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 larger of the two values (current value in heap and the new value) is kept in the root. The smaller of the two values is added the sub-heap with the smallest height. If both sub-heaps have the same height then the smaller 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 max-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.

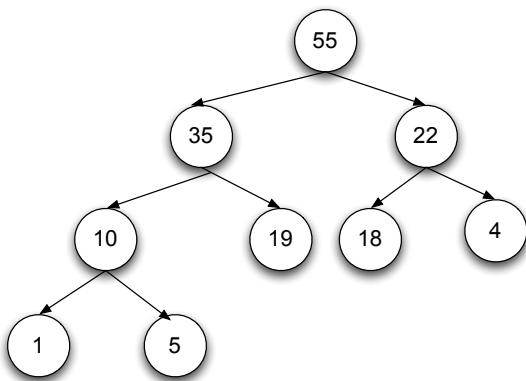


Figure 1

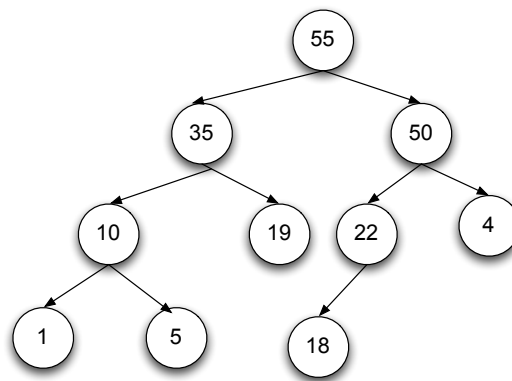


Figure 2

Your code is to have the following functionality:

1. Add numbers to your max-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 odd numbers in the heap in preorder.

Don't store the heap in an array or vector like structure. Do not convert the heap into an array or vector like structure to traverse the heap.

## Grading

Item	Percent of Grade
Working Code	30%
Unit Tests	20%
Comments	20%
Quality of Code	30%

**Working Code.** How well your code meets the functional requirements listed above accounts for 30% of the grade for the assignment.

**Unit Tests.** Having unit tests that cover the functionality of your code accounts for 20% of your grade. The unit tests are to be scalable. Tests using just print statements are not scalable.

### Turning in your Assignment

For the people not using Smalltalk. To turn in your assignment combine all your files for the assignment into a zip file. (No rar files) and upload them in the course portal (Under the assignments tag click on assignment 2).

For the people using Smalltalk. Place all your source code in a package called CS635Assignment2. Upload the package to your Smalltalk source control repository. If you do not know what this means contact me.