American University of Armenia
CS121 Data Structures
PSS 10

Tree Traversal

1. Write a method for the AbstractBinaryTree class that returns the node with the minimum element using one of the traversal methods.

2. Design a preorderNext(p) methods for AbstractBinaryTree class, which given the position p will return the position visited after p in preorder traversal of tree T.

3. Let the rank of a position p during a traversal be defined such that the first element visited has rank 1, the second element visited has rank 2, and so on. For each position p in a tree T, let pre(p) be the rank of p in a preorder traversal of T and let post(p) be the rank of p in a postorder traversal of T. Write a method difference(p) that given a position p calculates the absolute difference between pre(p) and post(p).

Priority Queue

1. Given an array of n numbers add them to a Priority Queue so that the key value of the given number corresponds to its frequency in the array.

Comparators

1. Given b1 and b2 nonnegative integers write a comparator that determines order based on the number of 1's in each integer's binary expansion, so that b1 < b2 if the number of 1's in binary representation of b1 is less than the number of 1's in binary representation of b2.

Heaps

1. Let Maxheap be a heap where the object in the node is greater than or equal to its descendant objects and let Minheap be a heap where the relations is less than or equal. Given the following integer keys (with no values specified): 30, 65, 22, 40, 15, 70, 80, 60, 55, 10 draw the tree representation of the heap that results when all of the above elements are added to initially empty:
    - Maximum binary heap
    - Minimum binary heap

2. Given an array of n numbers implement a method that returns m smallest numbers using Heap.

3. Given a heap H and a key k, give a method for computing all the entries in H having a key less than or equal to k.

If time permits

1. Trace the steps of a heap sort on the following array: 20, 40, 30, 10, 90, 70