Assignment # 6

Abhilash I

March 24, 2009

1 Design the data structure

1.1 Description

Design a data structure that supports the following operations.

Operation	Complexity
Insert(x)	$O(log(n))$
ExtractMin()	$O(1)$
ExtractMax()	$O(1)$
DeleteMin()	$O(log(n))$
DeleteMax()	$O(log(n))$

1.2 Input Format:

A number N followed by N instructions each on a line.

1.3 Output Format:

Ouput must be printed only for the operations ExtractMin() and ExtractMax().

1.4 Sample Input

```
3
Insert(1)
Insert(25)
Insert(28)
ExtractMin()
DeleteMin()
ExtractMin()
DeleteMax()
ExtractMax()
```

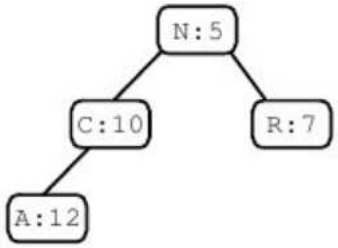
1.5 Sample Output:

```
1
25
25
```

2 Treaps

2.1 Description

Treaps = Trees + Heaps. Treaps are a hybird of Trees and Heaps. Treap is a binary search tree that orders the nodes by adding a priority attribute to a node, as well as a key. The nodes are ordered so that the keys form a binary search tree and the priorities obey the max heap order property



The treap was first described by Cecilia R. Aragon and Raimund G. Seidel in 1989.

- If v is a left descendant of u, then $key[v] \leq key[u]$;
- If v is a right descendant of u, then $key[v] > key[u]$;
- If v is a child of u, then $priority[v] \leq priority[u]$;

2.2 Input Format:

The first line indicates a number N. Following N lines will have $(key, priority)$ pairs.

2.3 Output Format:

You must output the depth of the tree and then the number of leaf nodes in the tree.

2.4 Sample Input

```
4
(A, 12)
(C, 10)
(N, 5)
(R,7)
```

2.5 Sample Output:

```
3 2
```

3 Heap Sort

3.1 Description

In certain time critical applications Heap Sort comes in handy becaues it's complexity is always $O(nlog(n))$. Comparison with quick sort: Quick sort has complexity $O(nlog(n))$ in the average case and it could go upto $O(n^2)$.

In this problem given a set of numbers you have to output the sorted sequence using the heap sort algorithm.

Extra credit: Implement quick sort, merge sort, heap sort and compare the time complexity on the data set published at course website.

3.2 Input Format:

N followed by N integers on following lines.

3.3 Output Format:

Sorted sequence.

3.4 Sample Input

3.5 Sample Output:

4 Travelling Sales Man Problem

4.1 Description

You are given a directed acyclic graph in which each node represents a city and an edge between two cities indicates the existence of a one way path between them. A travelling sales man has to visit all cities without visiting the same edge twice. Your task is to output the directed hamiltonian path.

Note that this graph has unique topological ordering.

4.2 Input Format:

The vertex set V is labelled $0, \ldots |V| - 1$. The edge set is E. Now the input starts with two integers $|V|$ followed by $|E|$ and the following $|E|$ lines consist of pairs of vertices between which the edges exist.

4.3 Output Format:

Output the directed hamiltonian path. The vertices should be seperated by spaces.

4.4 Sample Input

```
3 3
0 1
1 2
0 2
```

4.5 Sample Output:

```
0 1 2
```

5 Sorting

5.1 Description

You are given an input of Numbers . Output the Ascending order using Least significant digit radix sort (Refer Wikipedia). Now analyse the time and space complexity of the Implemented Algorithm.

5.2 Input Format:

The First line consists of Number of Test Cases (N). Each Test Case begins with number (K) followed by K numbers. Input (I) will be such that $0 \leq I \leq 10^{30}$.

5.3 Output Format:

Output the Numbers in ascending Orderd.

5.4 Sample Input

```
1
3
3
7
1
```

5.5 Sample Output:

```
1
3
7
```