Lab06-Heaps and BST

VE281 - Data Structures and Algorithms, Xiaofeng Gao, TA: Li Ma, Autumn 2019

* Please upload your assignment to website. Contact webmaster for any questions.
* Name:_____ Student ID:_____ Email: _____

1. **D-ary Heap.** D-ary heap is similar to binary heap but (with one possible exception) each non-leaf node of d-ary heap has d children, not just 2 children.

   (a) How to represent a d-ary heap in an array?

   (b) What is the height of the d-ary heap with n elements? Please use n and d to show.

   (c) Please give the implementation of insertion on the min heap of d-ary heap, and show the time complexity with n and d.

```
// Input: an integer k
// Output: null
void enqueue(int k)
{
    // TODO;
}
```

2. **Median Maintenance.** Input a sequence of numbers $x_1, x_2..., x_n$, one-by-one. At each time step i, output the median of $x_1, x_2..., x_i$. How to do this with $O(\log i)$ time at each step i? Show the implementation.

3. **BST**. Two elements of a binary search tree are swapped by mistake. Recover the tree without changing its structure. Implement with a constant space.

```
/**
 * Definition for binary tree
 * struct TreeNode {
 *     int val;
 *     TreeNode *left;
 *     TreeNode *right;
 *     TreeNode(int x) : val(x), left(NULL), right(NULL) {}
 * };
 */
void recoverTree(TreeNode *root)
{
    // TODO;
}
```

4. **BST**. Input an integer array, then determine whether the array is the result of the post-order traversal of a binary search tree. If yes, return Yes; otherwise, return No. Suppose that any two numbers of the input array are different from each other. Show the implementation.

```
// Input: an integer array
// Output: yes or no
bool verifySquenceOfBST(vector<int> sequence)
{
    // TODO;
}
```