Helwan University
Faculty of Computers and Information

Department of Computer Science
Data Structures
Spring 2007

Sheet 5

1 Introduction

Please implement the data structure BST by writing the file `BST.cpp`, which includes all of the functions studied in lectures. Make sure that among these functions are the two functions `TraverseInorderIterative` and `TraverseInorderRecursive`. For simplicity, choose the `ElementType` to be `float`.

Hint: you will need to use the linked stack, as well, to implement the iterative function.

2 Main Program

The main program has two purposes; the first is to test the basic functions coded in the implementation level. The second, is to compare the execution time of the two functions `TraverseInorderIterative` and `TraverseInorderRecursive` at different sizes of the tree. To achieve these two purposes, please design the program that generates the following menu.

1. Destroy the existing tree.
2. Generate a random BST of size n.
3. Traverse the tree `Inorder` recursively.
4. Traverse the tree `Inorder` iteratively.
5. Exit.

The program should start by creating a tree, and whenever you need to create a new tree you should choose the first option first. When you choose the second option, the program should prompt you to enter the required size n of the new tree. Then, the program should generate n random numbers between 0 and 1, displays them on screen, and insert them in an empty tree. You can read about the number random generator in the help of the C language. The numbers generated randomly are distributed uniformly between 0 and 1; hence, the tree will be almost balanced. The third and fourth options should exactly produces the same output.

The second purpose (mentioned above at the beginning of this section) is achieved by the following. When you choose the third or the fourth option, you should call a function that get the cpu-time before and after the calling. You should subtract the two values and print the time consumed in Traversing the tree. On a piece of paper, record the time consumed by both the iterative and recursive function. Repeat this for many values of n. Then, on Microsoft Excel plot the time of the iterative function and the time of the recursive function on the same graph versus the size of the tree n. Let's discuss the result in the next meeting.