BITS Pilani, Pilani Campus
2nd Sem. 2017-18
CS F211 Data Structures & Algorithms
==========
Lab IX - (26th Mar. to 31st Mar.)
==========

Topics: N-ary trees and Trees with Arbitrary Branching; Tree Traversals;

Programming Environment: C on Linux

Exercise 1: **[Expected Time: 20 minutes.]**

a) Define an ADT Iterator (a la Java) with the following interface:
   - Boolean hasMoreElements(Iterator it)
   - Element getNextElement(Iterator it)

b) Provide separate implementations of the ADT Iterator using: (i) a dynamically allocated array and (ii) a linked list.

Exercise 2: [Expected Time: 20 minutes]

a) Define a new type for a (general) tree node containing an integer value and an Iterator (where each Element in the iterator is a pointer to a tree node) and a type for a pointer to a tree node.

b) Implement the following operations on general trees:
   a. Tree_node createTree(int no_of_children)
   b. Boolean isEmptyTree(Tree t)
   c. Element getRootVal(Tree t)
   d. Iterator getChildren(Tree t)

   The createTree function builds the tree recursively. The argument to createTree function must be a generated random number between 0 and 10. The value at each node can be anything of your choice, except that all the nodes must get unique numbers. Ensure that maximum height of the tree doesn't cross a predefined height say *Max_Height* = 5.

Exercise 3: [Expected Time: 75 minutes]

a) Write a generic treeTraverse procedure:
   - Maintain a currentSet of nodes (in the tree) to be explored
   - Use a selectNextNode operation to choose the next node to be visited
   - Use an identifyMoreNodes operation to identify the next set of nodes to be explored.
   - Use an addMoreNodes operation to add more nodes to be explored into the currentSet.
   - Use the following (rough) structure:

```
repeat {
        n = selectNextNode(currentSet);
        visitNode(n);
        addMoreNodes(currentSet, identifyMoreNodes(n));
} (until currentSet is empty);
```

b) Implement the abstract operations used in the treeTraverse procedure in two different ways to obtain DFT and BFT. While visiting each node, print the integer value present at it.

Exercise 4: [Expected Time: 60 minutes]

a) Adapt the abstract operations used in the treeTraverse procedure suitably to implement a DFT on a Linux filesystem given a root node. [Hint: Use fstat, struct dirent, and readdir to obtain / access files and directory information. End of Hint.]