Name: **Roll No:**

National Institute of Technology Calicut

Department of Computer Science and Engineering

Second Midterm Examination, March 2013

CS2005 Data Structures and Algorithms

Time : 1 Hour **Maximum Marks: 20**

Instructions:

Write the answers for all the questions in the corresponding space provided in the question paper. Answers written elsewhere will not be evaluated. Additional sheet may be used for rough work only.

PART A

1. Write a recursive algorithm for finding the number of leaves in a tree T? [2 Marks]

```
Count-leaves(T)
    print(leaves(T.root))
```

```
leaves(x)
  if x=nil then return 0
     else if x.left != nil or x.right != nil
              then return(leaves(x.left) + leaves(x.right))
     else return(1)
```

2. Which binary tree is not a tree? [0.5 Marks]

The null tree which has 0 nodes

3. List all binary trees whose nodes are traversed in exactly the same order when either an inorder or a postorder traversal is executed. Use formal notations and a generic 'n' for the number of nodes, as far as possible. [1.5 Marks]

All binary trees having n nodes($n \geq 0$) such that each node has its right child as nil.

4. Convert the following Prefix expression to postfix. [2 MARKS]

   -*A^+BCD/EF

ABC+D^EF/- or ABC+D^*EF/-*

5. Prove by induction that in a binary tree [2 MARKS]

   Number of internal nodes = Number of edges – Number of leaves +1

Let Ii= no.of internal nodes in i node binary tree

Ei= no.of iedges in i node binary tree

Li = no.of leaf nodes in i node binary tree

For i=1, $I_1=0, E_1=0, L_1=1 \longrightarrow 0=0-1+1$ *true*

For i=2, $I_2=1, E_2=1, L_2=1 \longrightarrow 1=1-1+1$ *true*

Assume true for 1,2…n-1.

Then $I_{n-1} = E_{n-1} - L_{n-1}+1$ …. (1)

If a new(nth)node is added, it can be a leaf node, adding 1 link, either as a child of a leaf node or an internal node.

$I_n=I_{n-1}$, $E_n = E_{n-1}+1$, $L_n = L_{n-1}+1$ or $I_n = I_{n-1}$ 1 , $E_n = E_{n-1}+1$, $L_n = L_{n-1}$

In either case $I_n = E_n - L_n+1$

PART B

6. In a tree there are n_1 nodes of degree one, n_2 nodes of degree two…… n_m nodes of degree m. Find the number of leaf nodes in the tree? [2 Marks]

n_L= no.of leaves

No. leaves= $n_1x1 + n_2x2 +…….n_mxm +n_lx0 = E$

No.of nodes = $E+1= n_1 +2n_2+….+mn_m$

i.e $n_1 +n_2 +…n_m+n_L = n_1 + 2n_2 + mn_m$

$n_L = n_2+2n_3 +….+(M-1)n_m$

7. Write a recursive function kthMax(T,k) which finds the k^th largest element of a BST that runs in ϴ(n) steps? [2 Marks]

```
Kthmax(T,K)
1.  m= minimum(T)
2.   print successor(T,m,k)

Printsuccessor(T,x,k)
1. If (k=1)then print(x)
2.        else
3.        printsuccessor
4.         (successor(T,x),n-k-1)
```

8. Write a recursive function for converting a max_heap into a min_heap which runs in Θ(n) time, where n represents the number of elements in the heap? [2 marks]

Build min heap (Done in class, Refer text Book)

PART C

9. Consider the given iterative algorithm. The input and the steps in the algorithm are specified.

   a. State the output of the algorithm in the space provided, without any ambiguity. [2]
   b. Write a recursive algorithm to achieve the same output, in the space provided. [3]
   c. Name the fundamental operation performed by this algorithm [1]

Input: A binary tree node x

a.Output: *The number of nodes having degree 1 in the subtree rooted at x*

Iterative_unknown_algorithm (x)

```
Initialize_Stack(S)
data=0
If x≠nil
    then push(S,x)
        while (not(stack_empty(S)))
            do x=pop(S)
               if (x.right ≠ nil)
                   then push(S,x.right)
                       if x.left = nil then data=data+1
               If(x.left ≠ nil)
                   then push(S, x.left)
                       if x.right= nil then data=data+1

return (data)
```

b.Algorithm:

Recursive(x)

```
if x=nil then data=0
    else data =recursive(x.left) +recursive(x.right)
        if((x.left!=nil) or(x.right!=nil))
            then data=data+1
return(data)
```

d. Post order traversal