AUTUMN END SEMESTER EXAMINATION-2015

3rd Semester B.Tech & B.Tech Dual Degree

DATA STRUCTURE AND ALGORITHM (CS-2001/CS-301)

(Regular-2014 & Back of Previous Admitted Batches)

Full Marks: 60 **Time: 3 Hours**

Answer any SIX questions including Question No.1 which is compulsory.

The figures in the margin indicate full marks.

Candidates are required to give their answers in their own words as far as practicable and all parts of a question should be answered at one place only.

1. a) Find the time complexity of the following code segment in Big-Oh notation, in terms of the variable n. [2 × 10

```
int i, j, k, sum = 0;
for (i = 1; i<= n; i++) {
for (j = 1; j <= 1000; j += 2) {
    sum++; } }
for (k = -50; k <= -1; k++) {
    sum++; }
printf("%d", sum);
```

b) What is the output of following function for *start* pointing to first node of the linked list: 1->2->3->4->5->6 ?

```
voidfunc(struct node* start){
        if(start==NULL)return;
        printf("%d ", start->data);
  if(start->next != NULL )
      func(start->next->next);
  printf("%d ", start->data);
}
```

c) The following is a C like pseudo code of a function that takes a number as an argument, and uses a stack S to do processing. What does the function display?

```
void func(int n){
        Stack S;
        while (n > 0)  {
      push(S, n%2);
            n = n/2;
   }
       while (!isEmpty(S))
            printf("%d ", pop(S));
}
```

d) The preorder traversal sequence of a binary search tree is 30, 20, 10, 15, 25, 23, 39, 35, 42. Find the postorder traversal sequence of the same tree.

e) The postfix expression with single digit operands is evaluated using a stack: 8 2 3 ^ / 2 3 * + 5 1 * -. Note that ^ is the exponentiation operator. Find the top two elements of the stack after the first * is evaluated.

f) Write a recursive algorithm to count number of nodes in a single linked list.

g) Show that there are n+1 NULL pointers in a binary tree of n nodes.

h) Let a Binary Search Tree(BST) consists of *n* number of nodes having values from 1 to n. A pointer ***ptr*** is pointing to the right child of the root node. Write an algorithm to find the value of root node(if the root node is not known).

i) What are the different ways to represent a graph in computer?

j) Construct a binary tree with the following sequence of traversal given. The English alphabets represent the node information.

Preorder : A,B,D,H,I,E,J,K,C,F,G

Inorder : H,D,I,B,J,E,K,A,F,C,G

2. a) Describe 3-tuple representation of sparse matrix. Write an algorithm to add two sparse matrices. [4

b) Design a method for keeping two stacks within a single linear array S in such a way that neither stack overflows until the entire array is used. Write C functions push1(), push2(), pop1(), and pop2() to manipulate the two stacks. (Hints: Two stacks grow towards each other) [4

3. a) What is dqueue? Implement an input restricted dqueue. [4

b) Given an array *A[1.........n]*. Each element of the array represents a vote in the election. Assume that each vote is given as an integer representing the *ID* of the chosen candidate. Devise an algorithm to determine who wins the election? [4

4. a) Write a C function to delete a node containing a given value in a double linked list. [4

b) Write a C function to delete a node from Binary Search Tree (BST). [4

5. a) Given two sorted arrays *A* and *B*. Array *A* is full whereas array *B* is partially empty and number of empty slots are just enough to accommodate all elements of *A*. Give an algorithm to merge the two sorted arrays to fill the array *B*. [4

b) Write depth first search algorithm. Explain with suitable example. [4

6. a) Define strictly binary tree, complete binary tree and fully complete binary tree. Suppose the height of a fully complete binary tree is h. Find out the following: [4

- Total number of nodes present
- Total number of leaf nodes present

b) Write the recursive binary search algorithm. Explain each step of algorithm with suitable example. [4

7. a) Write the algorithm for insertion sort. Discuss the time complexity of best-case and worst-case situation. [4

b) In the context of hashing, define the term collision. How do we solve collision in hashing using chaining? Explain it with suitable example. [4

8. Write Short Notes on following. [2 × 4

a) Time Complexity

b) Divide-and-conquer principle

c) Threaded Binary Tree

d) Adjacency Matrix

X X X X X