B.Tech-3rd(CSE/IT)
Data Structures

Full Marks : 50

Time : $2\frac{1}{2}$ hours

Answer **all** questions

The figures in the right-hand margin indicate marks

Symbols carry usual meaning

1. Answer *all* questions : 2×5

(*a*) What is the time complexity of the following code snippet

```
int fun (int n)
{
    int count = 0 ;
    for (int i = n ; i > 0 ; i/=2)
        for (int j = 0; j < i ; j + +)
        count + = 1 ;
    return count ;
}
```

(*b*) Consider the following linked list :

I-> I->T->K->G->P

What is the output of the following function when it is called with the head of the list ?

```
void fun (struct node * start)
{
if (start = = NULL)
    return;
printf ("%c", start->data); //Considering
                    data is of 'char' type
if (start->next ! = NULL)
    fun (start->next->next);
printf("%c", start->data);
}
```

(*c*) (*i*) Given a binary tree T having 20 leave nodes. Calculate the number of nodes in *T* having two children.

(*ii*) Let post order traversal of a binary search tree (BST) is given by VSQTURP. If S < V < Q < P < T < R < U, then the pre-order traversal of the BST is : ——

(*d*) Differentiate between Stable and Unstable sorting and Give suitable examples.

(*e*) What is rehashing and when it is applied ?

2. (*a*) What is an ADT ? Discuss the various types of ADT. 4

(*b*) Discuss the best case, worst case, average case, and amortized time complexity of an algorithm. 4

Or

(*c*) Let A be a two-dimensional array declared as follows : 4

A : array [1 .... 10] [1 ..... 15] of integer ;

Assuming that each integer takes two byte memory locations and the first element of the array is stored at location 100, what is the address of the element A [6][8] if the array is stored in

(*i*) Row-major order

(*ii*) Column-major order.

(*d*) What is data structure ? Differentiate between linear and non-linear data structure and give an example of each one. 4

3. (*a*) Evaluate the following post-fix expression using stack. 4

7 5 3 * 5 1 ^ / + 3 2 – +

(*b*) Write an algorithm to delete a node from a circular linked list. 4

Or

(*c*) Write the algorithm to convert an infix expression to postfix expression. Using the algorithm convert the infix expression (A + B) * C + D/ (E + F * G) – H into a postfix expression. 4

(*d*) Write an algorithm to print all element present in a double ended queue. 4

4. (*a*) Show the result of inserting 3, 1, 4, 6, 9, 2, 5, 7 into an initially empty binary search tree. Also show the result of deleting the root. 4

(*b*) Write a non-recursive algorithm to perform in-order traversal on a binary tree using stack. 4

Or

(*c*) Write an algorithm to perform level order traversal on a binary tree. 4

(*d*) Construct the expression tree for the following post-fix expression : 4

a b c * + d e f + * +

5. (*a*) Find the element 10 in the given list 12, 15, 3, 32, 7, 19, 10, 35, 67 using Binary Search algorithm. Write the low, high and mid index at each step. How many more/ less comparisons are needed w.r.t to Linear Search. 4

(*b*) Given the following list of keys create the Maxheap and sort them using the Heap Sort technique. 4

8, 20, 9, 4, 15, 10, 7, 22, 3, 12

Or

(*c*) Write an algorithm to perform bubble sort on the following given elements :

64, 34, 25, 12, 22, 11, 90

The algorithm should take necessary

steps to print the array elements in each pass and stop sorting, if it is sorted at the beginning or atany intermediate pass. 4

(*d*) Differentiate internal sorting and external sorting with examples.

Consider the following array of size 10.

45, 3, 12, 89, 54, 15, 43, 78, 28, 10

Selection sort, bubble sort and insertion sort were applied to this array and the contents of the array after three passes in each sort are shown below :

(*i*) 3, 12, 15, 43, 45, 28, 10, 54, 78, 89

(*ii*) 3, 10, 12, 89, 54, 15, 43, 78, 28, 45

(*iii*) 3, 12, 45, 89, 54, 15, 43, 78, 28, 50

Identify the sorting technique used in each case. 4

6. (*a*) The keys 12, 18, 13, 2, 3, 23, 5 and 15 are inserted into an initially empty hash table of length 10 using open addressing with hash function $h(k) = k \bmod 10$ and linear probing. What is the resultant hash table ? 4

(*b*) Insert the following keys in an array of size 17 using the modulo division method. Use double hashing to resolve the collisions. Take $h'(k) = (\text{key } \%7) + 1$ as the second hash function. 4

94, 37, 29, 40, 84, 88, 102, 63, 67, 120, 122.

Or

(*c*) Explain the technique of 'Hashing' as an effective searching technique. What are 'Collisions' ? How can they be handled ? 4

(*Continued*)

(*d*) What are the different collision resolution techniques ? Apply chaining and linear probing to store the following values in a hash table of size 7 : 4

25, 42, 96, 101, 102, 162, 197.

———————