VIT
Vellore Institute of Technology

Final Assessment Test – April 2019

Course: ITE1004 - Data Structures and Algorithms
Class NBR(s): 4395/4401/4407/4413
Time: Three Hours

Slot: C1+TC1
Max. Marks: 100

Answer ALL Questions
(10 X 10 = 100 Marks)

1. a) Explain how stack ADT is used in recursion using Towers of Hanoi problem.(assume no of disks n=4) **[6]**

   b) Evaluate the following prefix expression and show the result. -+7*45+20 **[4]**

2. Write a program to implement 2 overlapping queues in an array of size 'N'. They are facing in opposite direction to each other. Give IsEmpty(i), Insert(i), Delete(i) and IsFull(i) routines for 'i'th queue. **[10]**

3. Given a Singly linked list with each node containing either 0, 1 or 2. Write appropriate code snippet to sort the list. **[10]**

   Input : 1 -> 1 -> 2 -> 0 -> 2 -> 0 -> 1 -> 0

   Output : 0 -> 0 -> 0 -> 1 -> 1 -> 1 -> 2 -> 2

4. a) What does the function mentioned below do? Derive its recurrence relation and hence deduce its time complexity. **[4]**

```
void recursion (int list[], int lo, int hi, int key)
{   int mid;
    if (lo > hi)
    {       printf("Key not found\n");
       return;   }
    mid = (lo + hi) / 2;
    if (list[mid] == key)
    {       printf("Key found\n");    }
    else if (list[mid] > key)
    {       recursion(list, lo, mid - 1, key);    }
    else if (list[mid] < key)
    {       recursion(list, mid + 1, hi, key);  }
}}
```

   b) **[3]**

```
int y=0;
for(int j=1; j*j<=n; j++)
y++;
```

   c) **[3]**

```
int a = 0;
int k = n*n;
while(k > 1){
for (int j=0; j<n*n; j++)
{ a++; }
k = k/2;
}
```

5. Use Quick sort technique to sort the array of elements given below. **[6]**

   900, 200, 1000, 300, 700, 100, 800, 600, 400, 500

   Derive the recurrence relations and hence the time complexity of sorting these elements if

   i) 300 is taken as the pivot **[2]**

   ii) 500 is taken as the pivot **[2]**

   Why do the two time complexities differ? Explain.

6. Insert the following numbers, in the order given, into

a) an ordinary, unbalanced binary search tree [3]

b) an AVL tree. For the AVL tree, indicate at which points rotations occur to restore the balance of the tree. [5]

94, 33, 50, 76, 96, 67, 56, 65, 83, 34

c) For the two trees from question 6(a), show that the in-order traversals are the same, so that both trees serve equally well for sorting a list into ascending order. [2]

7. a) Write a function that takes an input as a pointer to the head of a linked list and determines whether the list is circular or if the list has an ending node. If the linked list is circular then your function should return true, otherwise your function should return false if the linked list is not circular. [5]

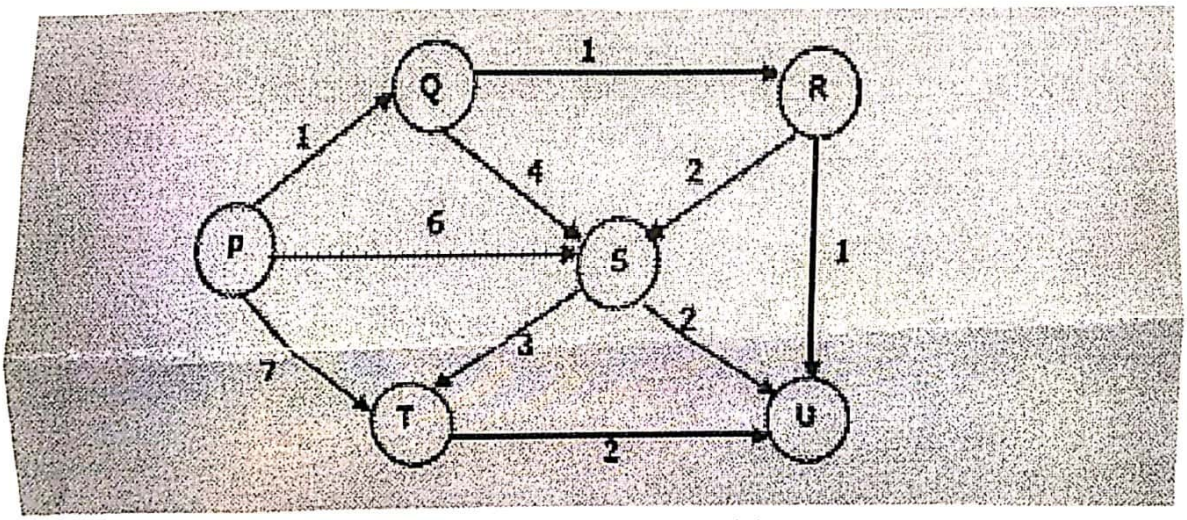
b) Propose an algorithm which helps to find cycle in a graph. Explain it with an example. [5]

8. a) Explain the applications of data structures in digital era. [5]

b) Propose a sorting algorithm which is best suited for sorted or pre-sorted input and implement it with suitable algorithm. [5]

9. Explain in detail about collision resolution techniques. [10]

10. a) [5]



Find the shortest path using Dijkstra's algorithm by assuming 'P' as source.

b) Find out the recurrence relation for the above code snippet and analyse the time complexity. [5]

```
void test (int n)
 {
   If (n>0)
   {        for(i=1; i<n; i++)
     {            printf("%d", i);
     }
   test(n-1);
   }
 }
```

$T(n) = T(n-1) + O(n)$ recurrence relation

Final answer: $T(n) = O(n^2)$

⇔⇔⇔