P. K. Gupta

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

TEST-2 EXAMINATION (APR 2018)

B-Tech (2nd SEM)

Course Code: 10B11CI211 Max. Marks: 25

Course Name: DATA STRUCTURES Max. Time: 1 HRS 30 MIN.

Course Credit: 4

Note: All questions are compulsory. Skip syntax error if there any.

SECTION A (8x1=8 marks)

Q1.

a) Suppose a circular queue of capacity (n – 1) elements is implemented with an array of n elements. Assume that the insertion and deletion operation are carried out using REAR and FRONT as array index variables, respectively. Initially, REAR = FRONT = 0. Write the condition(s) to detect whether queue is full and queue is empty?

b) Find the worst case time complexity for the number of comparisons needed to search a singly linked list of length **n** for a given element.

c) Consider the following pseudo code. Assume that **IntQueue** is an integer queue. What will be the output of following given function **dst**?

```
void dst(int n)
{
    IntQueue q = new IntQueue();
    q.enqueue(0);
    q.enqueue(1);
    for (int i = 0; i < n; i++)
    {
        int a = q.dequeue();
        int b = q.dequeue();
        q.enqueue(b);
        q.enqueue(a + b);
        print(a);
    }
}
```

d) Define the height of a perfect binary tree and binary tree.

e) How would you do the concatenation of two linked lists in O(1) time? Mention only the type of linked list.

f) What is the worst case time complexity of $\sum_{i=1}^{n} \log i$? Show the calculation steps.

g) Why we do not move head pointer from first node to others for performing operations like search, insert, delete etc. on linked lists?

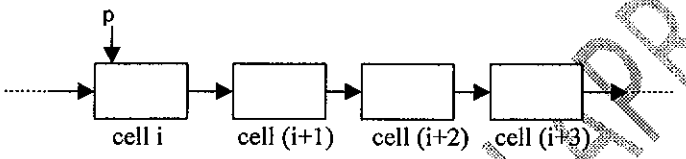
h) Differentiate between Deque and Dequeue.

Section B (4x2=8 marks)

Q2. Consider an empty stack of integers. Let the numbers 1,2,3,4,5,6 be pushed on to this stack in the order they appear from left to right. Let **S** indicate a push and ***X*** indicate a pop operation. Can they be permuted in to the order 3 2 5 6 4 1 (output) and order 1 5 4 6 2 3? Justify your answer.

Q3. Given an integer *k* and a queue of integers, write an algorithm for reversing the order of the first *k* elements of the queue, leaving the other elements in the same relative order? For example, if *k*=4 and queue has the elements [10, 20, 30, 40, 50, 60, 70, 80, 90]; the output should be [40, 30, 20, 10, 50, 60, 70, 80, 90].

Q4. Let p be a pointer as shown in the figure in a singly linked list.



What do the following assignment statements achieve?

q: = p→ next
p → next: = q→ next
q → next: = (q → next) → next
(p → next) →next: = q

Q5. A **K-ary** tree is such that every node has either K sons or no sons. If L and I are the number of leaves and internal nodes respectively, then express L in terms of K and I.

Section C (3x3=9 marks)

Q6. What is balanced binary tree? Insert the following keys one by one into a binary search tree in order specified.

15, 32, 20, 9, 3, 25, 12, 1

Show the *binary search tree* and *balanced binary search tree* after insertions. Also represent the *binary search tree* and *balanced binary search tree* after deleting 15 from it. **[Marks - 1+0.5+0.5+0.5+0.5]**

OR

Write and discuss the detailed algorithm for creating the Binary search tree.

Q7. How can you implement a queue using two stacks? Write algorithm(s) and explain them. Analyze the running time of stack.

Q8. Convert the following given expressions:

a) *Infix to Prefix* expression using stack conversion method e^d-a*b^f/g+h*c/i+j-k

b) *Postfix to Infix*: A B C < C D > || ! && ! C E < ||