ROLL NO:	

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY UNIVERSITY, BENGALURU

SCHOOL OF ENGINEERING

Weightage: 20 % Max Marks: 40 Max Time: 1 hr. Monday, 24th September, 2018

TEST – 1

Odd Semester 2018-19 Course: **CSE 201 Data structure** III Sem. CSE

Instruction:

(i) Read the question properly and answer accordingly.

(ii) Question paper consists of 3 parts.

Part A

(4 Q x 4 M = 16 Marks)

1. Define a pointer, write the syntax to declare an integer pointer, write the output of the following code

```
main () {
double  *ptr = NULL;
printf("The value of ptr is : %x\n", ptr  );  }
```

2. Identify the type of argument passing used in the following code, and justify your answer, write the code for implementing the function max

```
main () {
int a = 100,  b = 200;
int  ret = max(a, b);
printf( "Max value is : %d\n", ret ); }
```

3. Match the following

Dynamic Memory Allocation	LIFO
user defined data structures	Big O
Asymptotic notation	malloc()
stack	LIST

4. Write the output of the following code

```
#include<stdio.h>
#include<stdlib.h>
void solution(int arr[], int n)
{
int *left = (int *)malloc(sizeof(int)*n);
int *right = (int *)malloc(sizeof(int)*n);
int *ans = (int *)malloc(sizeof(int)*n);
int i, j;
left[0] = 1;
```

```
right[n-1] = 1;
for(i = 1; i < n; i++)
    left[i] = arr[i-1]*left[i-1];
for(j = n-2; j >=0; j--)
    right[j] = arr[j+1]*right[j+1];
for (i=0; i<n; i++)
    ans[i] = left[i] * right[i];
for (i=0; i<n; i++)
    printf("%d ", ans[i]);
}
int main()
{
int arr[] = {10, 3, 5, 6, 2};
int n = sizeof(arr);
printf("The output array is: ");
solution(arr, n);
getchar();
}
```

Part B

(2 Q x 7 M = 14 Marks)

5. Discuss the methods for analyzing the efficiency of an algorithm, analyze the following code and represent it in order notation, write the output of the code considering N value as 3

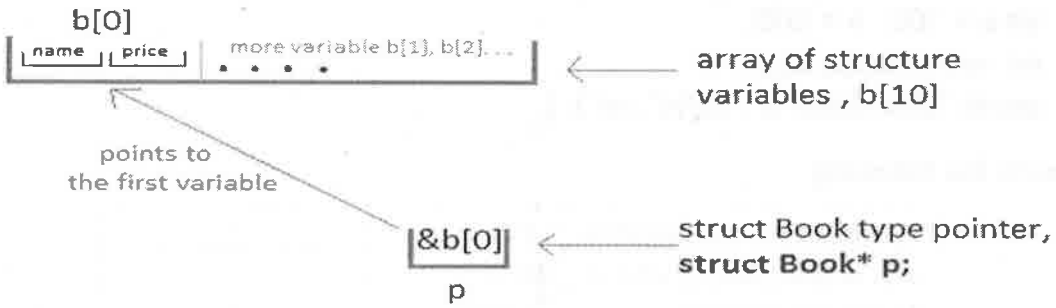
```
int count = 0;

for (int i = 0; i < N; i++)

        for (int j = 0; j < i; j++)

                count++;
```

6. Write the necessary **C statements** to simulate the following situation



Part C

(1 Q x 10 M = 10 Marks)

7. Read the following scenario and recommend a suitable data structure and justify your answer.

There is a single lane parking space for 7 cars with one gate, how many entry and exit required to park all seven cars and exit of 4^{th} car.

Write the pseudo-code/algorithm for implementing the entry and exit functions.

ROLL NO:

PRESIDENCY UNIVERSITY, BENGALURU

SCHOOL OF ENGINEERING

TEST 2

Odd Semester: 2018-19 | **Date**: 27 November 2018

Course Code: CSE 201 | **Time**: 1 Hour

Course Name: Data Structure | **Max Marks**: 40

Branch & Sem: CSE & III Sem | **Weightage**: 20%

Instructions: ***Answer all questions sequentially***

Part A

Answer **all** the Questions. **Each** question carries **four** marks. (4x4=16)

1. Using array implementation, write the conditions for the following
   a) full linear queue
   b) full circular queue
   c) empty linear queue
   d) empty circular queue
2. Starting from an initially empty circular queue of size 2, show the contents of the queue with the value of front and rear indexes, after each of the following operations in sequence.
   a) Enqueue A b) Enqueue B c) Dequeue d) Enqueue C
3. List any two differences between stack and queue data structures.
4. With a neat diagram, illustrate the structure of a node in a singly linked list and in a doubly linked list. Explain each field.

Part B

Answer **all** the Questions. **Each** question carries **seven** marks. (2x7=14)

5. Using the infix to postfix algorithm, convert the infix expression A * B – (C+D) + E to postfix. Show the results of each step with the following details in a table format as given below

Infix character read	Stack contents	Postfix string

6) Define a function **reverse(),** which takes the address of the first node (head) of a doubly linked list and prints the list in reverse order. The function prototype is given below
void reverse(struct node *head);

Part C

Answer the following Question. Question carries **ten** marks. (1x10=10)

7. A singly linked list is used to store the mark obtained by each student in a class of 5 students. Define a function **find_highest(),** which takes the address of the first node(head) and returns the highest score in the class. Assume there are no duplicate values in the list. Use the function prototype given below

   float find_highest(struct node *head);

****

Roll No.												

PRESIDENCY UNIVERSITY
BENGALURU

SCHOOL OF ENGINEERING

END TERM FINAL EXAMINATION

Odd Semester: 2018-19 | **Date**: 27 December 2018

Course Code: CSE 201 | **Time**: 2 Hours

Course Name: Data Structures | **Max Marks**: 80

Programme & Sem: CSE & III Sem | **Weightage**: 40%

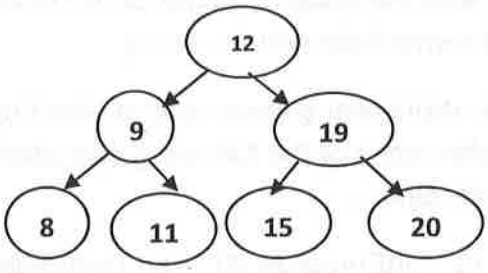
Instructions:

(i) ***Answer All Questions Sequentially***

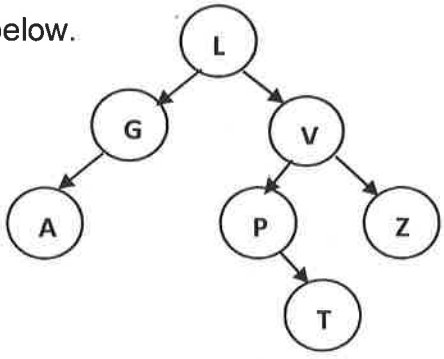
Part A

Answer **all** the Questions. **Each** question carries **five** marks. (4Qx5M=20)

1. List all the tree traversal methods. Write the result of all traversals on the binary tree given below.



2. Define the function **int count(struct node *head, int key),** which counts and returns the number of occurrences of a given key, in a doubly linked list. Assume the linked list is already created.
3. What are expression trees? Give an example.
4. Define balance factor of a node in an AVL tree. Calculate the balance factor of all nodes in the tree given below.



Part B

Answer **all** the Questions. **Each** question carries **ten** marks. (3Qx10M=30)

5. Draw the binary tree from its traversals given below. Show the partially constructed tree in each step.

   In-order : 4 8 2 5 1 6 3 7

   Post-order: 8 4 5 2 6 7 3 1

6. What are binary search trees? Construct a binary search by inserting nodes in this order 100, 50, 200, 25, 90, 80, 150, 300 and 180. What is the height of the final tree?
7. Construct a directed graph with V = {1,2,3,4,5} E = { (1,1), (1,2), (1,4), (2,3), (3,5), (4,5)}. Draw the adjacency list representation of this graph. Write the result of BFS of this graph.

Part C

Answer **all** the Questions. **Each** question carries **fifteen** marks. (2Qx15M=30)

8. Insert 14, 17, 11, 7, 53, 4, 13 and 12 into an initially empty AVL tree in this order. Draw the tree after each insertion with the balance factor of all nodes. Also name the rotations done in each step with the intermediate results, if any.

9. A singly linked list is used to represent a linear queue. Starting from an initially empty queue, draw the linked list after each of the following operations. Clearly indicate the front and rear pointers in each case.

i) Enqueue 15 ii) Enqueue 13 iii)Enqueue 20 iv) Dequeue v) Dequeue