Reg. No.										

MANIPAL INSTITUTE OF TECHNOLOGY

MANIPAL

(A constituent unit of MAHE, Manipal)

III SEMESTER B.TECH. (COMPUTER SCIENCE & ENGINEERING)
END SEMESTER MAKEUP EXAMINATIONS, DECEMBER 2018

SUBJECT: DATA STRUCTURES [CSE 2103]

REVISED CREDIT SYSTEM

(27/12/2018)

Time: 3 Hours MAX. MARKS: 50

Instructions to Candidates:

- Answer **ALL** questions.
- Missing data may be suitably assumed.

1A. Explain different dynamic memory allocation and de-allocation functions with prototype and example to each. **(4)**

1B. Write a complete C program to illustrate passing and returning structures to and from functions through pointers by to defining a structure **FRACTION** with numerator and denominator (integers) as its data members. Write the functions with following prototypes. Use type defined structure.

void getFr(FRACTION *);
void printFr(FRACTION *) ;
FRACTION * multiFr(FRACTION *, FRACTION *);. **(4)**

1C. Explain the functionality of the following recursive function. **(2)**

```
int foo(int x, int y)
{
  if(x == 0)   return y;
else
   return foo(x – 1,x +y);
}
```

2A. Write a complete C program to perform the following operations on a queue of integers using only standard queue operations,

i) Insertq(x): Add an item x to queue.
ii) Deleteq() : Remove an item from queue.
iii) Display(): Displaying queue elements
iv) Reverse() : Contents of queue are reversed using only standard queue operations. **(4)**

2B. Write a complete C program to implement push, pop and display operations of a stack using dynamic array to hold 5 integers. If the stack is full when the push operation is called, it must increase the size of the stack by 5 more integers. **(3)**

2C. Write an algorithm to convert an infix expression to postfix expression. Trace the algorithm for the infix expression: ***((A+B)*D)*((E-F)-G)*** by filling the table given below:

Current symbol scanned	Action Taken(push/pop etc)	Content of the stack	Intermediate result

(3)

3A. Write a function to add two polynomials, polynomial A, and polynomial B, represented as singly linked lists. The function should accept pointers to linked lists representing two polynomials and return a pointer to the linked list representing the sum. **(4)**

3B. Given a singly linked list, write a complete C program to find and display the middle element of the linked list. If there are even number nodes, display the second middle element. **(4)**

3C. Write a C function to invert a singly linked list. The function should accept a pointer to the given list and return a pointer to the inverted list. **(2)**

4A. Write a complete C program to do the following,

i) Create a binary tree

ii) Convert the created binary tree into binary search tree without changing structure of the tree.

iii) Traverse the tree in preorder **(4)**

4B. Write a function to construct an expression tree for the given postfix expression. Using the same, draw expression tree for the postorder: ABC*+DE/- by considering each letter as a single operand. **(3)**

4C. Construct a binary search tree for the given set of numbers {100, 80, 90, 88, 200, 150, 179, 300, 400} in the order they are read from left to right (100 as root). Display the postorder traversal sequence of the constructed tree. **(3)**

5A. Derive an expression for finding the total cost of a BST (including both successful and unsuccessful searches) for a set of elements. What is the relation of this expression with optimal BST? Assume the root is at level 1. **(4)**

5B. Define B-tree of order m and also mention its properties. What do you mean by 2-3-4 tree, explain with an example? **(3)**

5C. Given input list (26, 5, 77, 1, 61, 11, 59, 15, 48, 19). Show the working of merge sort by showing the contents of the array after each pass. **(3)**