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: **(3)**

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

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)**

Reg. No.										

MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

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

SUBJECT: DATA STRUCTURES [CSE 2103]

REVISED CREDIT SYSTEM

(24/11/2018)

Time: 3 Hours MAX. MARKS: 50

Instructions to Candidates:

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

1A. Write a program that creates the structure shown in Figure 1A, and reads data into a and b using the pointers x and y. The program then multiplies the value of a by b and stores the result in c using the pointers x, y, and z. Finally, it prints all the three variables using pointers x, y, and z.

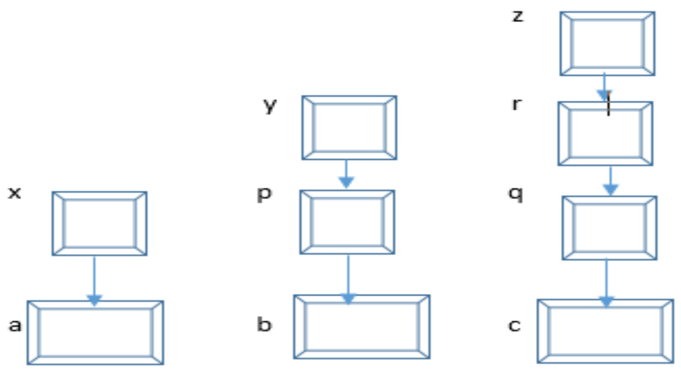


Figure 1A. **(3)**

1B. Write a recursive function to solve Tower of Hanoi problem. Use the prototype given below.

void Tower(int NoOfDisks, char src, char aux, char dest);

Also write call tree for function call ***Tower(3, 'A', 'B' , 'C');*** **(4)**

1C. Briefly explain how the Tagged and Type-defined structures are created and used in C with syntax and an example for each. **(3)**

2A. Write a complete menu driven C program to implement 'n' stacks using a single 1D array containing 'm' locations with following operations,

i) push (int i, int item, STACK * S); //pushing an item on ith stack

ii) pop(int i, STACK *S);//poping an item from ith stack

iii) display(STACK * S); // displaying all n stack contents

For the STACK structure, the members boundary[i] and top[i] represents boundary and top respectively for the ith stack along with element. If a stack is full during push operation and the space is available in 'm' locations of array then shift neighbouring stacks so that space is allocated to the full stack. Incorporate the proper validation checks wherever required. **(5)**

2B. Write an algorithm to convert a prefix expression to postfix expression. Trace the algorithm to convert the prefix expression: **- - / * +** ***ABCD*EF*GH*** to postfix expression by clearly showing the intermediate steps in the form of table shown below

Current symbol scanned	Action Taken (push/pop etc)	Stack Content

(3)

2C. Define queue? What is the disadvantage of ordinary queue and how to overcome it? **(2)**

3A. Write a complete C program consisting of a function kAltReverse(. . .) to reverse every alternate k nodes in a singly linked list, where k is read from the keyboard. Include functions to insert the nodes in the front of the list and also to display the nodes in the list. **(4)**

3B. Write a complete C program consisting of a function sortedInsert(. . .) to insert an integer into a sorted doubly linked list, such that after insertion the list remains sorted. Also include function to display the contents of the list. **(4)**

3C. Given a pointer to the first node of a doubly linked list. Write a function to display the number of nodes in the list and also free all the nodes of the linked list. **(2)**

4A. Write an iterative function BST_Delete(NODE root, int ele) to delete an element from binary search tree without connecting the left subtree of a node to be deleted to the inorder successor of the node to be deleted. **(5)**

4B. How to convert a binary tree into a threaded binary tree? Write a function to print the inorder sequence of a threaded binary tree. **(3)**

4C. Construct a binary tree for given preorder: **A, B, D, E, H, I, C, F, G** and inorder: **D, B, E, I, H, A, C, G, F** traversals of a tree. **(2)**

5A. Check whether the following trees(shown in Figure 5A (a)(b)(c)) are B-trees are not and also give justification to your answers

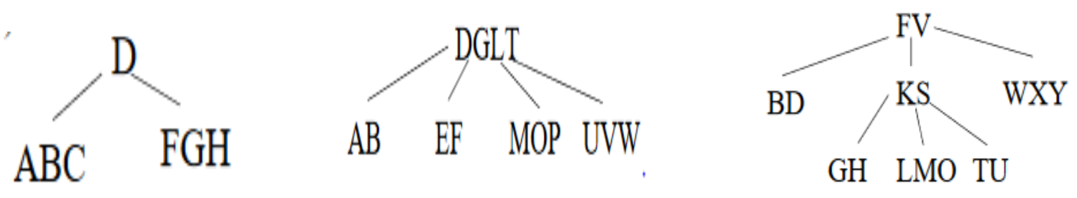


Figure 5A. (a) (b) (c) **(3)**

5B. Write a function to sort a given list of integers using quicksort. **(4)**

5C. List and explain any two methods used to represent the graphs. Using the same methods, represent the graph given below in Figure 5C.

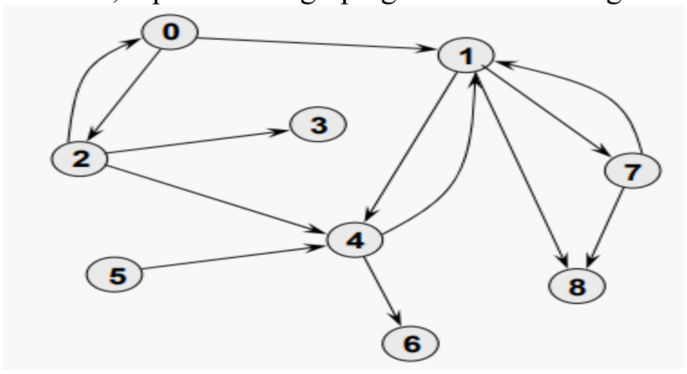


Figure 5C. **(3)**

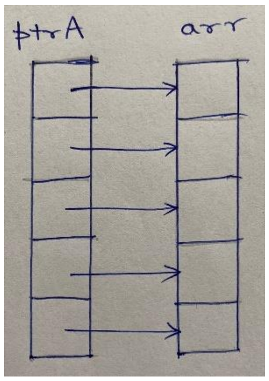
MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

III SEMESTER B. TECH (COMPUTER SCIENCE & ENGINEERING)
MID SEMESTER EXAMINATION, OCTOBER 2020
SUBJECT: DATA STRUCTURES & APPLICATIONS (CSE 2152)
REVISED CREDIT SYSTEM

Date of Exam: 22/10/2020 **Time: 90 Minutes** **MAX. MARKS: 20**

Note: Answer ALL the questions.

1. Write a complete C program to do the following. Read an integer 'n' from the keyboard. Read 'n' integers & store in an array 'arr'. Create an array of pointers to integers, 'ptrA'. Each pointer in 'ptrA' should point to the corresponding element in 'arr' as shown in figure. Write function ***display()*** to display the elements of 'arr'. Also write a function ***rotate()***, which rotates the elements in 'arr' one position to the right without changing the position of the elements in the array, that is, only the pointers have to be rearranged. For both the functions, the only parameters that may be passed are 'ptrA' and 'n'.
Example:- Suppose array 'arr' contains the elements 1, 2, 3, 4, and 5. First time when display() is called the output should be "1 2 3 4 5". When display() is called, after rotate(), the output should be "2 3 4 5 1". **3**



2. Consider the following program. How many times will the word "fibonaci" be printed? What will be printed in the last line of the output? **2**

```
int fix(int n){
    if(n==0) return 0;
    if(n==1) return 1;
    else{
        printf("fibonaci\n");
        return (fix(n-1) + fix(n-1));
    }
}
```

```
void main(){
    int num, fnum;
    num=5;
    fnum=fix(num);
    printf("\n%d", fnum);
}
```

3. Convert the infix expression **((A+B)*C-(D-E))^(F+G)** to its equivalent Prefix expression by filling the structure given below(^ is exponentiation operator). **3**

Scanned symbol	**action taken (push, pop, add to prefix exprn)**	**Stack contents**	**Current Prefix exprn**

4. Write C functions to implement following operations of multiple stacks (number of stacks is 'n') using a single 1-D array having 'm' locations with following prototypes, **2**
   i) **void *push(int i, int item, STACK *S)*;** //push an item on ith stack
   ii) **int *pop(int i, STACK *S)*;** //pop an item from ith stack

Use the following ***STACK*** structure:

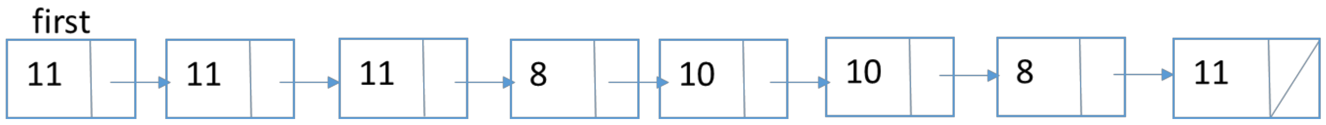
```
typedef struct {
    int stackArr[m];
    int boundary[n+1];
    int top[n+1];
}STACK;
```

Here, *boundary[i]* and *top[i]* represents boundary and top respectively for the i^th stack. Necessary validation check has to be done for stack overflow and underflow situations.

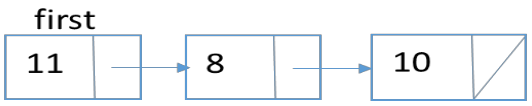
5. Given a nonempty unsorted singly linked list with a list_pointer ***first*** pointing to the first node in the list. Write a function ***void delete_multi(list_pointer *first)*** which deletes multiple occurrences of a node, keeping only the first occurrence in the list without creating a new list. Also, do not sort the list. Use the following definition to represent each node in the list: — 4

```
typedef struct list_node *list_pointer;
struct list_node {
    int data;
    list_pointer link;
};
```

Sample input:



Expected output for the above given input:



6. Given a circular singly linked list with a list_pointer ***first*** pointing to the first node in the list, write a function ***void reverse(list_pointer *first)*** which reverses the given circular linked list by changing links between the nodes. Use the following definition to represent each node in the list: — 2

```
typedef struct list_node *list_pointer;
struct list_node {
    int data;
    list_pointer link;
};
```

7. Write a complete C program to do the following: — 4

(i) Define a function ***Nodeptr CreateDLL(char str[])*** which takes a string as parameter and creates a Doubly Linked List of characters and returns the pointer to the first node.

(ii) Define a function ***int IsPalindrome(Nodeptr first)*** to check whether the string represented by the above doubly linked list pointed to by ***first***, is a palindrome or not and return 1/0 accordingly. Do not use any additional data structure.

Write a main function to read a string and create Doubly Linked of characters and check whether the string is a palindrome using above functions. Assume the following structure definition:

```
typedef struct NODE *Nodeptr;
struct NODE{
    char letter;
    Nodeptr llink, rlink;
};
```

Reg. No.										

MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL

A Constituent unit of MAHE, Manipal

III SEMESTER B.TECH. (COMPUTER SCIENCE & ENGINEERING)

END SEMESTER EXAMINATIONS, JAN 2021

SUBJECT: DATA STRUCTURES & APPLICATIONS [CSE 2152]

REVISED CREDIT SYSTEM
(05/03/2021)

Time: 3 Hours MAX. MARKS: 50

Instructions to Candidates:

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

1A.	Create a structure as shown: struct DISTANCE {int feet; float inch;}; In main(), create pointers to the above struct and allocate memory using dynamic memory allocation. Read in and store values of distances d1and d2. Find the sum of the two distance values using function: void addDist (struct DISTANCE *d1, struct DISTANCE *d2, struct DISTANCE *result); Display the value of result in main().	**5**
1B.	Write a complete program to check whether the given string is a palindrome using recursion.	**3**

1C.	Write the output of the following program:
	`#include<stdio.h>`
	`void main() {`
	`    int num[5] = {3, 4, 6, 2, 1};`
	`    int *p = num;`
	`    int* q = num+2;`
	`    int* r = &num[1];`
	`    printf("\n%d %d", num[2], *(num+2));`
	`    printf("\n%d %d", *p, *(p+1));`
	`    printf("\n%d %d", *q, *(q+1));`
	`    printf("\n%d %d", *r, *(r+1));`
	`}`
		2

2A.	Write a C program(Menu driven) to implement ‘n’ stacks using a single 1-D array containing ‘m’ locations with following prototypes,
	i) push(int i, int item, STACK *S); //pushing an item on ith stack
	ii) pop(int i, STACK *S);//poping an item from ith stack
	iii) display(S) // displaying all ‘n’ stack contents
	For the STACK structure, the members boundary[i] and top[i] represents boundary and top respectively for the ith stack along with element represented by other member. While pushing if the particular stack is full, and if there is space available elsewhere in the array of ‘m’ locations, it should shift the stacks so that space is allocated to the full stack.
		5

2B.	Write a program to convert the given prefix expression to postfix equivalent	**3**

2C.	Write C functions for the following operations on ordinary queue using array with proper validation: a) insert b) Delete	**2**
3A.	Write a function ***"poly_add (poly_pointer, poly_pointer)"*** to add two polynomials represented by two singly linked lists, A and B and return the new polynomial, C.	**5**
3B.	Give any six differences between Array and Linked List.	**3**
3C.	Write code to implement stack operations (push and pop) using singly linked list.	**2**
4A.	Consider all five possible binary search trees for the key set (a1, a2, a3) = (Creta, City, Punto). Find the optimal binary search tree (i) with equal probabilities, pi = qi = 1/7 for all i and j (ii) with p1=0.05, p2=0.5, p3=0.1, q0=0.05, q1=0.1, q2=0.15, q3=0.05	**5**
4B.	Given a prefix expression, write a function in C to create an expression tree.	**3**
4C.	Discuss with an example for each, the storage representations of binary trees using arrays and linked representations.	**2**
5A.	Write an iterative function in C to traverse a binary tree in post-order traversal showing the node structure of the stack.	**4**
5B.	Write function ***"struct node* findunion (struct node *LLOne, struct node *LLTwo)"*** which returns the union of two lists represented using singly linked list.	**4**
5C.	Convert the given infix expression, ***A + B - C * D / (E – F + G) *H*** to prefix, by showing the Scanned Symbol, Action Taken, Stack Contents and Current Prefix in the form of a table, as shown below: Scanned symbol | Action Taken | Stack Contents | Current Prefix	**2**