MANIPAL ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

Question Paper - Report

26-Sep-2025 12:37:00
TIRTHAM BHAVESH BAFNA .
Logout

Question Paper

Back

MANIPAL ACADEMY OF HIGHER EDUCATION

Manipal Instiute of Technology
School of Computer Engineering
Mid Sem Examination - September 2025

DATA STRUCTURES [CSS 2101]

Marks: 30 **Duration: 90 mins.**

MCQ

Answer all the questions. Section Duration: 20 mins

Answer all the questions

1) What is the output of the following code snippet? (0.5)

```
#include < stdio.h>

int main() {

int a = 5, b = 10;

int *p = &a;

int *q = &b;

int **r = &p;

*p = *q + **r;

*q = **r - *p;

p = q;

**r = *p + *q;

printf("%d %d\n", a, b);

return 0;

}
```

15, 0   15, 15   15, -5   20,20

2) Consider the following C program: (0.5)

```
#include < stdio.h>
```

```
int main() {
int a[] = {2, 4, 6, 8};
int *p = a;
*p += *(p + 1);
p += 2;
*(p - 1) = *p - a[0];
*(p + 1) = a[1] + *p;
p = a + 1;
*p = a[3] + *(p - 1);
printf("%d %d %d %d\n", a[0], a[1], a[2], a[3]);
return 0;
}
```

What is the output?

6,12,6,6   6,0,6, 6   8,12,6,6   6,12,0,6

3) What is the output of the following code snippet? (0.5)

```
#include < stdio.h>
int f(int n) {
if (n == 0) return 0;
return (n % 2) + f(n / 2);
}
int main() {
printf("%d\n", f(13));
return 0;
}
```

2   3   4   5

4) What is the output of the following code snippet? (0.5)

```
#include < stdio.h>
void display(int n) {
if (n < = 0) return;
printf("%d ", n);

display(n - 2);

printf("%d ", n);

}

int main() {

display(5);

return 0;}
```

5 4 3 2 1 0 5 3 1 3 5 5 3 1 1 3 5 1 3 5 5 3 1

5) Consider a singly linked list with nodes storing integers. A function is written to reverse the list by changing links. After reversal, which statement is correct? (0.5)

Every node's next pointer remains unchanged.

Only the head pointer needs to be updated to the original last node

The order of nodes remains the same.

It is not possible to reverse a singly linked list without using an array.

6) In a singly linked list, if you are given only a pointer to a node p (but not the head or previous node), how can you delete that node (not for the last node)? (0.5)

Copy the data from the next node into p and adjust p->next to skip the next node

Deletion is not possible in any case.

The entire list will be deleted

The list will become circular automatically.

7) Given a pointer ptr to a node in a doubly linked list, which of the following statements are always true about navigation between nodes? (0.5)

i. ptr == ptr -> llink -> rlink

ii. ptr == ptr -> rlink -> llink

iii. ptr == ptr -> llink -> llink

iv. ptr == ptr -> rlink -> rlink

i only i & ii only iii & iv only i, ii, iii & iv

8) In a circular doubly linked organization, insertion of a node involves the modification of (0.5)

zero pointers. 2 pointers 4 pointers 6 pointers

9) A program attempts to generate as many permutations as possible of the string “abcd” by pushing the character a,b,c,d in the same order onto a stack, but it may (0.5)

pop off the top character at any time. Which one of the following a strings CANNOT be generated using this program?

abcd dcba cbad cabd

10) Consider the usual algorithm for determining whether a sequence of parentheses is balanced. The maximum number of parentheses that appear on the stack AT ANY ONE TIME when the algorithm analyzes: (()(())(()))? (0.5)

1 2 3 4

DESCRIPTIVE

Answer all the questions.

Answer all the questions

11) You are developing a feature for a math learning app that allows users to add polynomial expressions. Each polynomial is stored as a singly linked list, where each node contains a coefficient and exponent. The terms are sorted in descending order of exponents. Based on above, answer the following questions,

1. Explain why a linked list is a suitable data structure for representing polynomials in this system.
2. Write a code logic to add two polynomials represented as linked lists. (4)
    a. Describe how your logic handles matching and non-matching exponents.
    b. Identify one potential issue that could arise during addition and suggest how to handle it in code.

12) A university maintains details of students for academic records. Each student has the following information: - Name (string, up to 50 characters), Roll Number (integer), Marks in three subjects (array of integers), Average Marks (float, to be calculated). The university wants a program to:

1. Read details of N students (name, roll number, and marks in 3 subjects).
2. Calculate the average marks of each student. (3)
3. Identify and display the topper (student with the highest average).

Write a C program using an array of structures and without using pointer to implement the above scenario. Write functions for each task.

13) A movie streaming app allows users to manage their watchlist. Users can: Move to the next or previous movie, Insert a new movie between two existing ones, Remove any movie from the list. The app must maintain smooth navigation and quick updates. Answer below question based on above scenario: (3)

a. Suggest a suitable data structure for this system.

b. Justify your choice based on the operations.

c. Write a C function to handle the deletion of movie from the middle of the watchlist.

14) A circular linked list contains the elements: 10 → 20 → 30 → 40 → 50 → (back to 10)

You are asked to rotate this list clockwise by 2 positions.

Answer the following:

(3)

1. Illustrate with a diagram of the list after rotation, showing how the elements are linked.

2. Write a function to implement the given task using the pointer to the last node.

15) You are required to read and process a sequence of numbers (both positive and negative). Whenever a negative number is encountered, output the five numbers that appeared immediately before it in reverse order (most recent first), then discard the negative number and continue processing the remaining input.

- If fewer than five numbers exist before a negative number, display error message and terminate.
- Repeat the same process for every negative number encountered until the input ends.

Design and implement a solution to this problem using most suitable data structure. Give all necessary functions. (3)

Sample Input/Output:

Input: 5 10 20 30 40 50 -1 60 70 80 90 100 -2 110 120 -3

Output:

50 40 30 20 10 100 90 80 70 60

Error: fewer than 5 numbers before the negative number: –3.

16) Write the algorithm to convert a given infix expression into its equivalent prefix expression using stack. Note: The infix expression may include ONLY the following operators (listed in the order of their precedence): (3)

- ^ (exponentiation) → **right associative**
- * (multiplication), / (division) → **left associative**
- + (addition), - (subtraction) → **left associative**

Also, convert the infix expression, A ^ B ^ C * D / E to prefix expression showing the step-by-step conversion using the table given below:

Token	**Stack**	**Top**	**Output**

	[0] [1] [2] …		

.

17) A library has a set of books arranged on shelves. Each shelf has books numbered in sequence (e.g., 1 to *n*). A librarian wants to find the sum of book numbers on a shelf using a recursive approach instead of a loop. Write a recursive function in C to calculate the sum of book numbers from 1 to *n*, where *n* is the number of books on a shelf (For example, if the librarian has 5 books on a shelf numbered 1 to 5, the program should calculate 1 + 2 + 3 + 4 + 5 = 15) . (2)

18) A university stores student records for a course using a singly linked list. Each node contains: - Roll Number (integer), Name (string up to 50 characters), Pointer to the next student. Write C function to delete a student by roll number from the list. (2)

19) Implement the given functions of stack data structure using Singly Linked List. Use the following function prototypes: (2)

```
void Push(Nodeptr *top, int item);

int Pop(Nodeptr *top);
```

where Nodeptr is defined as follows:

```
typedef struct node *Nodeptr;

struct node{

int data;

Nodeptr next;

}; .
```

^ Top