QN.7

State two data structures where unary operators are employed in their manipulation and traversal.
Study the following C code segments and answer questions that follow.

```
intmain(){
{
intnum1=2,num2=3;num3=4;
int rec1, rec2, rec3;
num 1tt;
--num3;
rec1 = num3 *--num3;
rec2 =++rec1+num3;
rec3 = rec2 -num3;
}
```

State the value of num1 on line 5
State the value of num3 on line 6
State the value of rec1 on line 7
State the value of rec2 on line 8
State the value of rec3 on line 9

Answers.

Unary operators are operators that operate on a single operand. Two data structures where unary operators are commonly employed in their manipulation and traversal are:
Linked Lists:
In linked lists, unary operators like ++ and -- are used to traverse the list from one node to another. For example, to move to the next node in a singly linked list, you would use the ++ operator, which increments the current node pointer to the next node.

Binary Trees:
In binary trees, unary operators like * and & are used to manipulate the nodes of the tree. For example, the * operator is used to dereference a pointer to a node, which allows you to access the value stored in that node. Similarly, the & operator is used to get the memory address of a node, which allows you to modify the value stored in that node.

State the value of num1 on line5 =2
State the value of num3 on line6 =3
State the value of rec1 on line7 =6
State the value of rec2 on line8 =10
State the value of rec3 on line9 =6

QN.6

Define the following C program function terminologies;
Function definition
Arithmetic operators
Storage class
Local variables
Global variables

(b)

Study the C program below and answer questions that follow; 1.

```
#include<stdio.h>
void addition(void);
void multiply (int , int);
 void addition (void);
{
int num1, num2, sum;
printf "Enter any two numbers...\n");
scanf("%d%d", &num1, &num2);
sum = num1 + num2;
printf(" In Sum of %d and %d is %d", num2 ,sum)
multiply ( num1, num2);
}
Void multiply (int num1 , int num2)
{
Int product = num1* num2;
Printf ("\n The product of %d and %d` is %d", num2 ,product);
}
Int main(void)
{
addition();
return 0;
```

List the lines that show the function prototypes.
List the names of the user defined functions (UDFs).
List the lines that show function calls of the UDFs.
State the purpose of #include directive in the above code
State one function of UDFs.

Ans;

Function definition:
A function definition in is a block of code that specifies the actions or tasks to be performed by a function.
Arithmetic operators:
Arithmetic operators are operators that are used to perform mathematical operations. These operators include addition (+), subtraction (-), multiplication (*), division (/), and modulus
Storage class:
A storage class in C programming defines the scope, visibility, and lifetime of variables or functions within a program.
Local variables:
A local variable is a variable that is declared within a function and has a limited scope. It can only be accessed within the function in which it is defined. Local variables are usually used to store temporary values or intermediate results within a function.
Global variables:
A global variable is a variable that is declared outside of any function and has a global scope. It can be accessed from any part of the program, including any function within the program. Global variables are usually used to store values that need to be accessed and modified by multiple functions within a program.

List the lines that show the function prototypes.
```
void addition(void);
void multiply(int, int);
```

List the names of the user defined functions (UDFs).
```
void addition(void);
void multiply(int, int);
```

List the lines that show function calls of the UDFs.
```
addition();
multiply(num1, num2);
```

State the purpose of #include directive in the above code
The purpose of the #include directive in the above code is to include the stdio.h header file, which contains declarations for the standard input/output functions like printf and scanf.

State one function of UDFs.
One function of the UDFs is multiply. This function takes two integer arguments, multiplies them together, and prints the result.

QN.5

Define the meaning of the following disk input/output file modes;
r+
w+
a+

Write a C code that prints numbers from 2 to 10 and calculates their sum.

Ans;

r+: Open for reading and writing. The file must exist, otherwise the operation fails. The file pointer is positioned at the beginning of the file.

w+: Open for reading and writing. If the file exists, its contents are truncated to 0 bytes and the file is treated as a new empty file. If the file does not exist, it is created. The file pointer is positioned at the beginning of the file.

a+: Open for reading and writing, appending to the end of file. If the file exists, the data is written at the end of the file. If the file does not exist, it is created. The file pointer is positioned at the end of the file.

Write a C code that prints numbers from 2 to 10 and calculates their sum.

```
#include <stdio.h>

int main() {
    int sum = 0;

    for (int i = 2; i <= 10; i++) {
        printf("%d ", i);
        sum += i;
    }

    printf("\nThe sum is: %d\n", sum);

    return 0;
}
```

QN.4

Define the following terms;
Structure
Union
State the difference between structure instance and union instance.
A lecturer has been requested to write a C computer code to capture two (2) course work marks and the final examination marks for a student. The course works are out of twenty (20) and the examination is out of one hundred (100). The lecturer must also capture the student's name of at most twenty (20) characters long. Course work accounts for 40% and examination accounts for 60% of the final mark. Using C structures develop a solution for the lecturer and print the final examination.

Ans.
Structure:
A structure is a user-defined data type in C programming language that enables you to group multiple variables of different data types into a single unit. The individual variables inside the structure are called members.

Union:
A union is a user-defined data type in C programming language that enables you to store different data types in the same memory location. Only one member of the union can be accessed at a time.

Difference between structure instance and union instance:
A structure instance holds values for all members of the structure, while a union instance holds values for only one member at a time, sharing the same memory location for all members.
Solution using C structures:

```
#include <stdio.h>
#define MAX_NAME_LENGTH 20

struct student {
   char name[MAX_NAME_LENGTH];
   int coursework1;
   int coursework2;
   int final_exam;
};

int main() {
  struct student my_student;

   printf("Enter the student's name (at most %d characters): ",
MAX_NAME_LENGTH);
   scanf("%s", my_student.name);

   printf("Enter coursework mark 1 out of 20: ");
   scanf("%d", &my_student.coursework1);
   printf("Enter coursework mark 2 out of 20: ");
   scanf("%d", &my_student.coursework2);

   printf("Enter final exam mark out of 100: ");
   scanf("%d", &my_student.final_exam);

   float final_mark = 0.4 * ((my_student.coursework1 + my_student.coursework2) /
2.0) + 0.6 * my_student.final_exam;

   printf("Final exam mark for %s is %d\n", my_student.name,
my_student.final_exam);

   return 0;
}
```