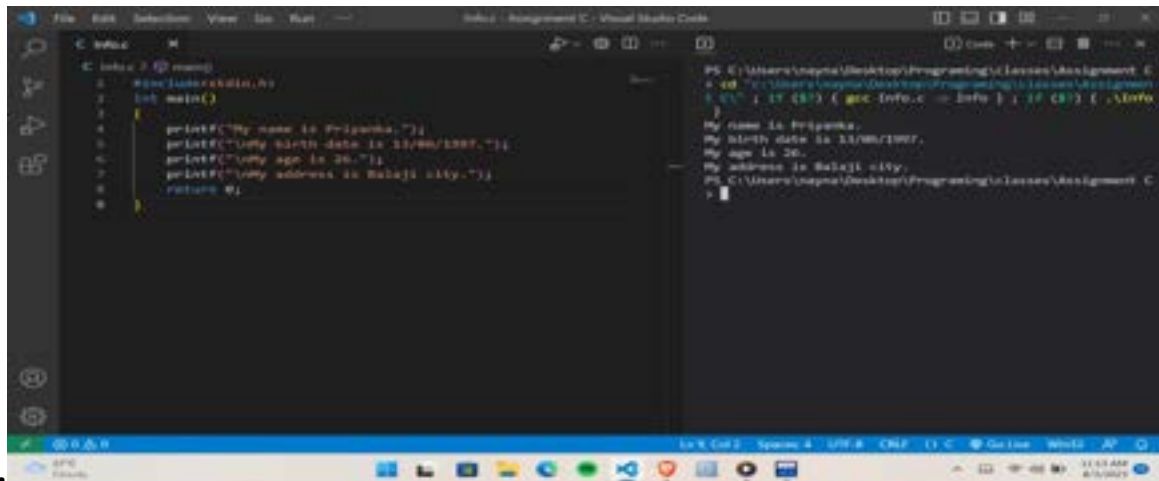


Assignment C

Module 3.1 --> C Language Fundamental

1> Display this information using printf.

1. Your name, 2. Your Birth Date, 3. Your Age, 4. Your Address.

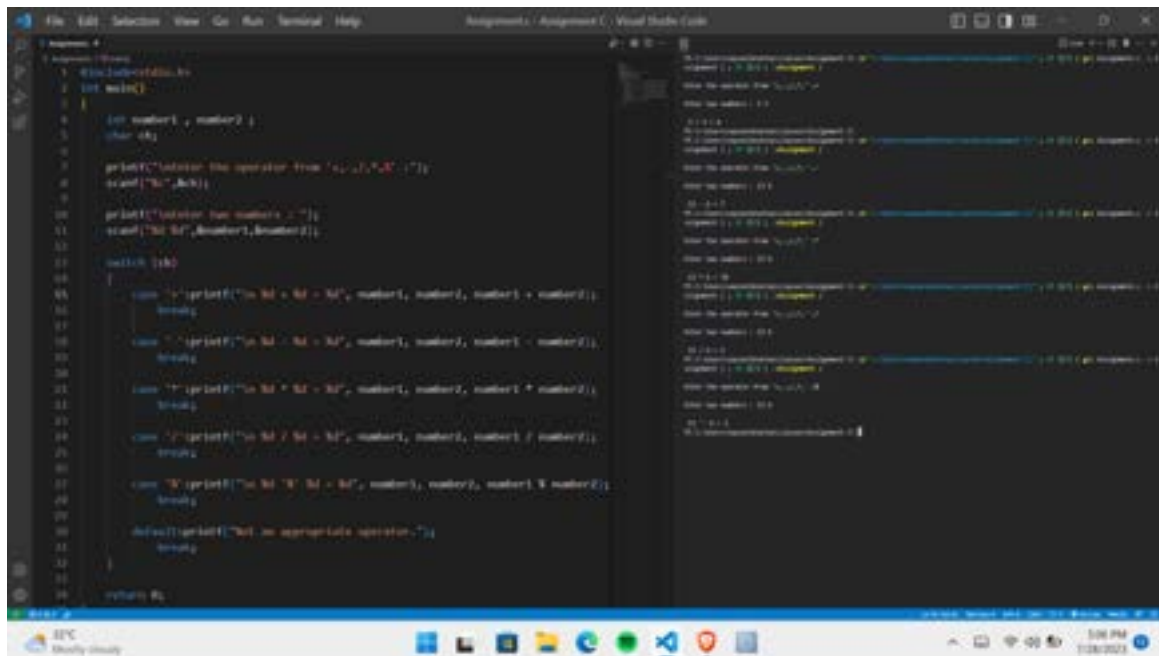


```
1 //info.c
2 #include <stdio.h>
3 int main()
4 {
5     printf("My name is Priyanka.");
6     printf("My birth date is 11/06/1997.");
7     printf("My age is 26.");
8     printf("My address is Balaji city.");
9     return 0;
10 }
```

The screenshot shows the source code on the left and the output on the right. The output displays the name, birth date, age, and address as specified in the program.

Ans.

2> Write a program to make Simple calculator (to make addition, subtraction, multiplication, division and modulo).

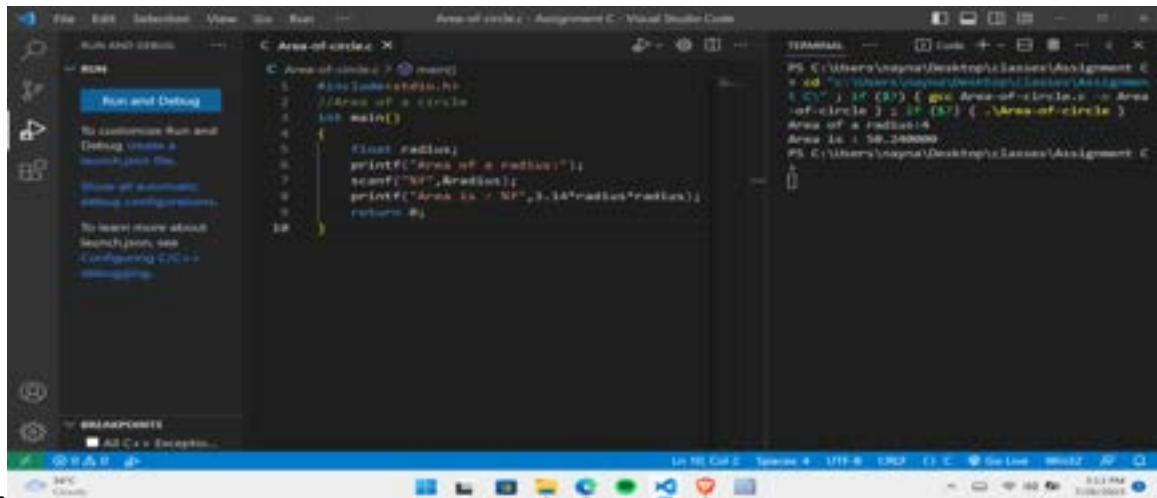


```
1 //SimpleCalculator.c
2 #include <stdio.h>
3 int main()
4 {
5     int number1, number2;
6     char op;
7     printf("Enter the operator from '+', '-', '*', '/', '%': ");
8     scanf("%c", &op);
9
10    printf("Enter two numbers: ");
11    scanf("%d %d", &number1, &number2);
12
13    switch (op)
14    {
15        case '+': printf("The sum is %d", number1 + number2); break;
16        case '-': printf("The difference is %d", number1 - number2); break;
17        case '*': printf("The product is %d", number1 * number2); break;
18        case '/': printf("The division is %d", number1 / number2); break;
19        case '%': printf("The modulo is %d", number1 % number2); break;
20        default: printf("Not an appropriate operator."); break;
21    }
22    return 0;
23 }
```

The screenshot shows the source code on the left and the output on the right. The output demonstrates the calculator's functionality for addition, subtraction, multiplication, division, and modulo operations.

Ans.

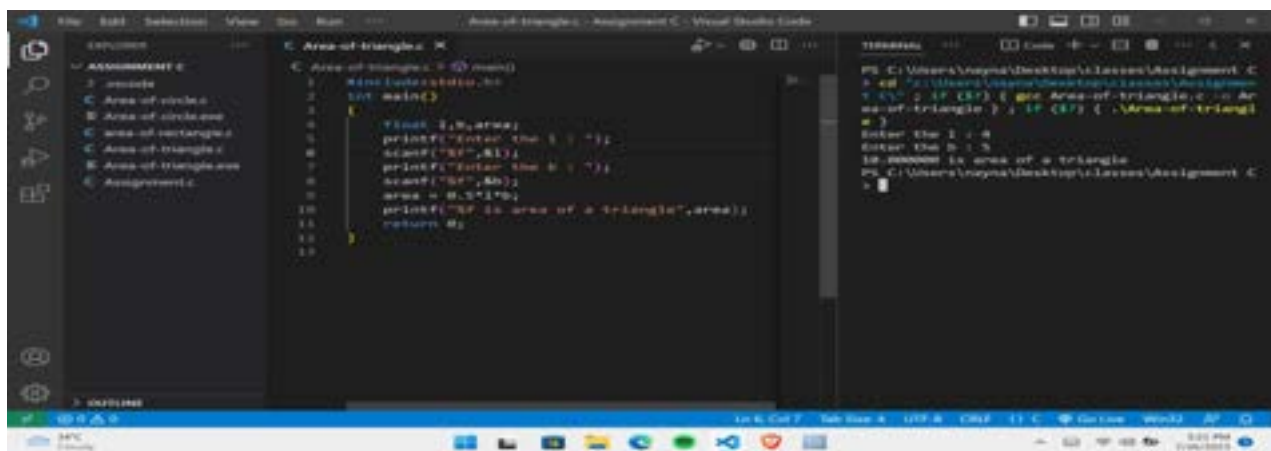
3> Write a program to find Area of circle, rectangle, and triangle.



The screenshot shows the Visual Studio Code interface with a C program titled "Area of circle.c". The program includes a header file "Area-of-circle.h" and a main function that prompts the user for a radius, calculates the area using the formula $A = \pi r^2$, and prints the result. The output in the terminal shows the area of a circle with a radius of 4 is 50.2654.

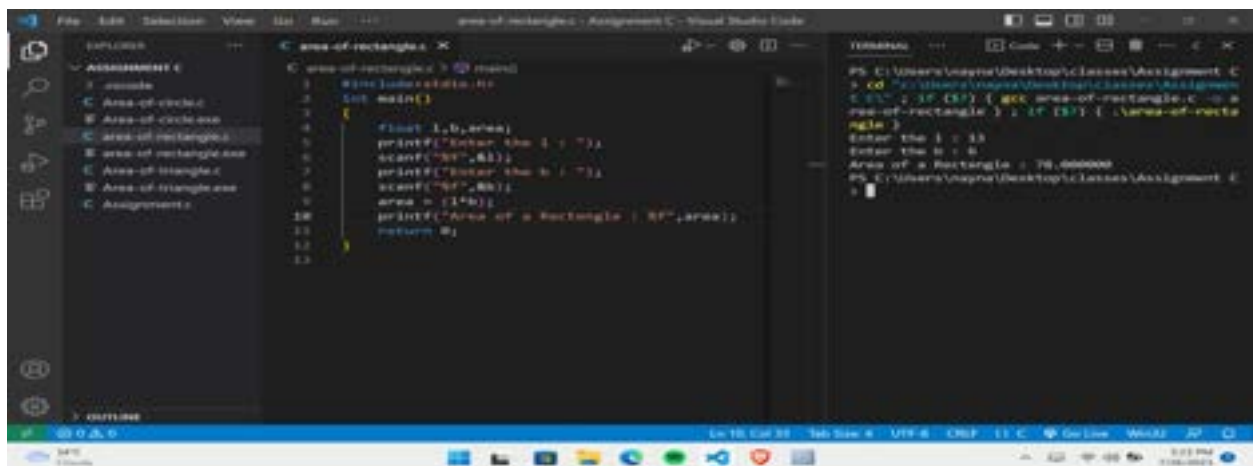
```
#include <stdio.h>
#include "Area-of-circle.h"
int main()
{
    float radius;
    printf("Area of a radius:");
    scanf("%f",&radius);
    printf("Area is = %f",3.14*radius*radius);
    return 0;
}
```

Ans.



The screenshot shows the Visual Studio Code interface with a C program titled "Area of triangle.c". The program includes a header file "Area-of-triangle.h" and a main function that prompts the user for the base and height of a triangle, calculates the area using the formula $A = 0.5 * b * h$, and prints the result. The output in the terminal shows the area of a triangle with base 4 and height 5 is 10.0000.

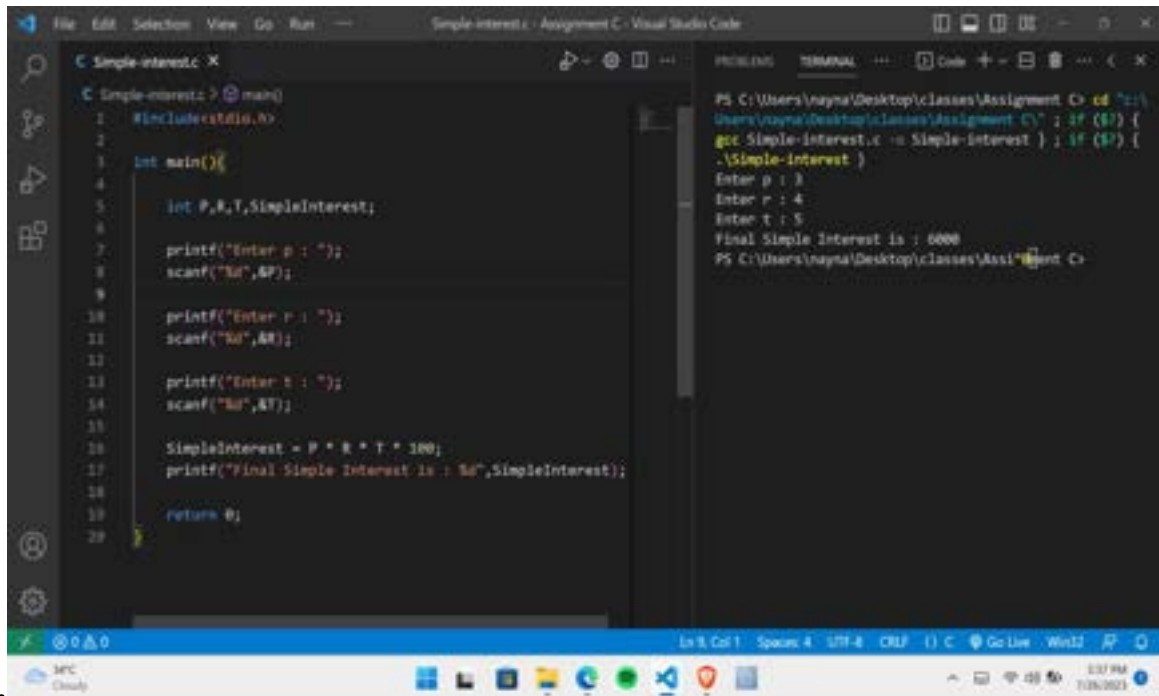
```
#include <stdio.h>
#include "Area-of-triangle.h"
int main()
{
    float l,b,area;
    printf("Enter the l : ");
    scanf("%f",&l);
    printf("Enter the b : ");
    scanf("%f",&b);
    area = 0.5*l*b;
    printf("Area of a triangle : %f",area);
    return 0;
}
```



The screenshot shows the Visual Studio Code interface with a C program titled "area-of-rectangle.c". The program includes a header file "area-of-rectangle.h" and a main function that prompts the user for the length and breadth of a rectangle, calculates the area using the formula $A = l * b$, and prints the result. The output in the terminal shows the area of a rectangle with length 10 and breadth 8 is 80.0000.

```
#include <stdio.h>
#include "area-of-rectangle.h"
int main()
{
    float l,b,area;
    printf("Enter the l : ");
    scanf("%f",&l);
    printf("Enter the b : ");
    scanf("%f",&b);
    area = (l*b);
    printf("Area of a Rectangle : %f",area);
    return 0;
}
```

4> Write a program to find simple interest.



The screenshot shows a Visual Studio Code editor with a C program for calculating simple interest. The code is as follows:

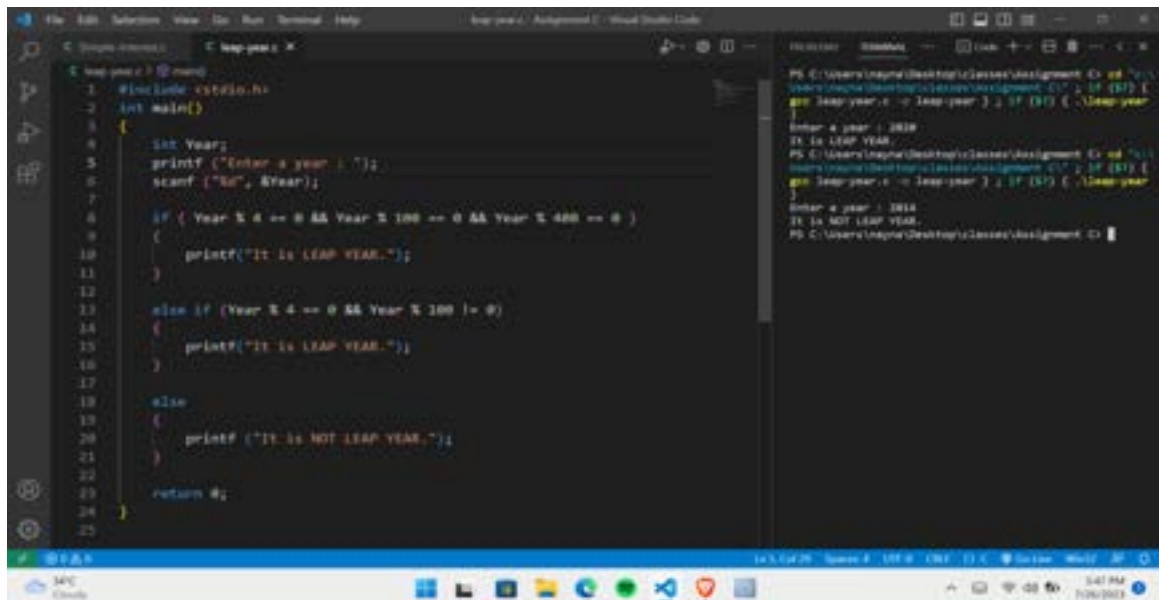
```
1 #include <stdio.h>
2
3 int main()
4 {
5     int P, R, T, SimpleInterest;
6
7     printf("Enter p : ");
8     scanf("%d", &P);
9
10    printf("Enter r : ");
11    scanf("%d", &R);
12
13    printf("Enter t : ");
14    scanf("%d", &T);
15
16    SimpleInterest = P * R * T * 100;
17    printf("Final Simple Interest is : %d", SimpleInterest);
18
19    return 0;
20 }
```

The terminal output shows the program being compiled and executed with the following input and output:

```
PS C:\Users\nayna\Desktop\classes\Assignment C> cd "C:\Users\nayna\Desktop\classes\Assignment C" & if ($?) {
gcc Simple-interest.c -o Simple-interest } & if ($?) {
./Simple-interest
Enter p : 3
Enter r : 4
Enter t : 5
Final Simple Interest is : 6000
PS C:\Users\nayna\Desktop\classes\Assignment C>
```

Ans.

5> Write a program to check if the given year is a leap year or not.



The screenshot shows a Visual Studio Code editor with a C program for checking leap years. The code is as follows:

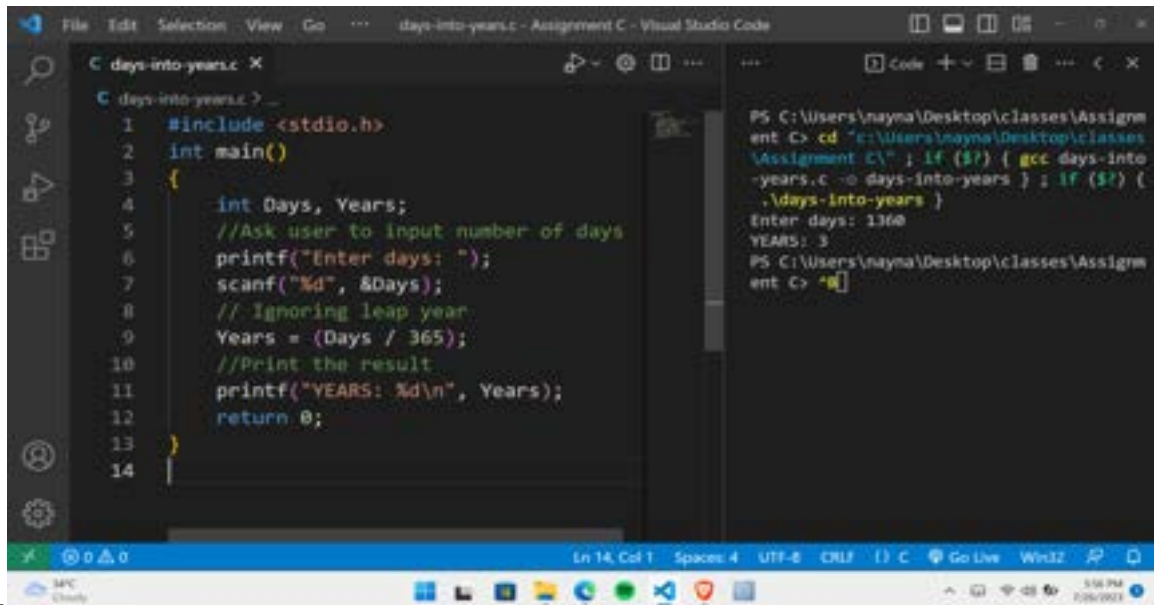
```
1 #include <stdio.h>
2 int main()
3 {
4     int Year;
5     printf("Enter a year : ");
6     scanf("%d", &Year);
7
8     if (Year % 4 == 0 && Year % 100 != 0 && Year % 400 == 0)
9     {
10        printf("It is LEAP YEAR.");
11    }
12
13    else if (Year % 4 == 0 && Year % 100 != 0)
14    {
15        printf("It is LEAP YEAR.");
16    }
17
18    else
19    {
20        printf("It is NOT LEAP YEAR.");
21    }
22
23    return 0;
24 }
```

The terminal output shows the program being compiled and executed with the following input and output:

```
PS C:\Users\nayna\Desktop\classes\Assignment C> cd "C:\Users\nayna\Desktop\classes\Assignment C" & if ($?) {
gcc leap-year.c -o leap-year } & if ($?) {
./leap-year
Enter a year : 2020
It is LEAP YEAR.
PS C:\Users\nayna\Desktop\classes\Assignment C> cd "C:\Users\nayna\Desktop\classes\Assignment C" & if ($?) {
gcc leap-year.c -o leap-year } & if ($?) {
./leap-year
Enter a year : 2014
It is NOT LEAP YEAR.
PS C:\Users\nayna\Desktop\classes\Assignment C>
```

Ans.

6> Write a program to convert years into days and days into years.



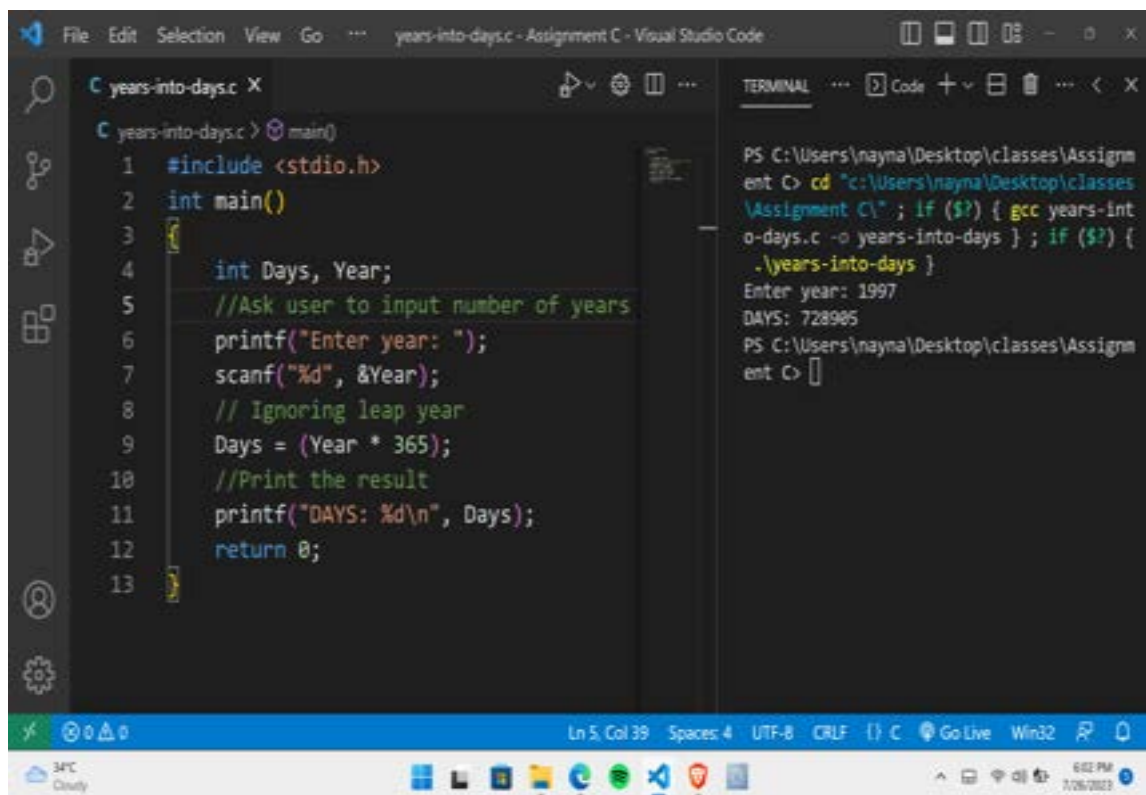
The screenshot shows the Visual Studio Code editor with a C program named 'days-into-years.c'. The code is as follows:

```
1 #include <stdio.h>
2 int main()
3 {
4     int Days, Years;
5     //Ask user to input number of days
6     printf("Enter days: ");
7     scanf("%d", &Days);
8     // Ignoring leap year
9     Years = (Days / 365);
10    //Print the result
11    printf("YEARS: %d\n", Years);
12    return 0;
13 }
14
```

The terminal on the right shows the execution of the program:

```
PS C:\Users\nayna\Desktop\classes\Assignment C> cd "c:\Users\nayna\Desktop\classes\Assignment C\"; if ($?) { gcc days-into-years.c -o days-into-years }; if ($?) { .\days-into-years }
Enter days: 1360
YEARS: 3
PS C:\Users\nayna\Desktop\classes\Assignment C>
```

Ans.



The screenshot shows the Visual Studio Code editor with a C program named 'years-into-days.c'. The code is as follows:

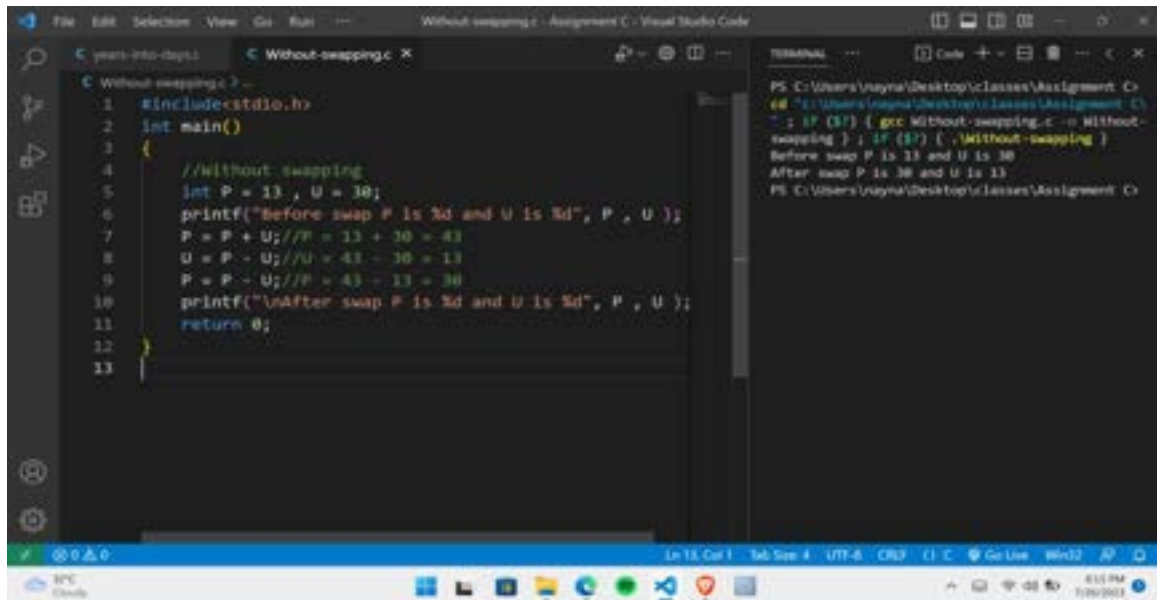
```
1 #include <stdio.h>
2 int main()
3 {
4     int Days, Year;
5     //Ask user to input number of years
6     printf("Enter year: ");
7     scanf("%d", &Year);
8     // Ignoring leap year
9     Days = (Year * 365);
10    //Print the result
11    printf("DAYS: %d\n", Days);
12    return 0;
13 }
```

The terminal on the right shows the execution of the program:

```
PS C:\Users\nayna\Desktop\classes\Assignment C> cd "c:\Users\nayna\Desktop\classes\Assignment C\"; if ($?) { gcc years-into-days.c -o years-into-days }; if ($?) { .\years-into-days }
Enter year: 1997
DAYS: 728905
PS C:\Users\nayna\Desktop\classes\Assignment C>
```

Module 3.2 --> C Language Programming with C

1> Write a program to swap two numbers without using Third variable.

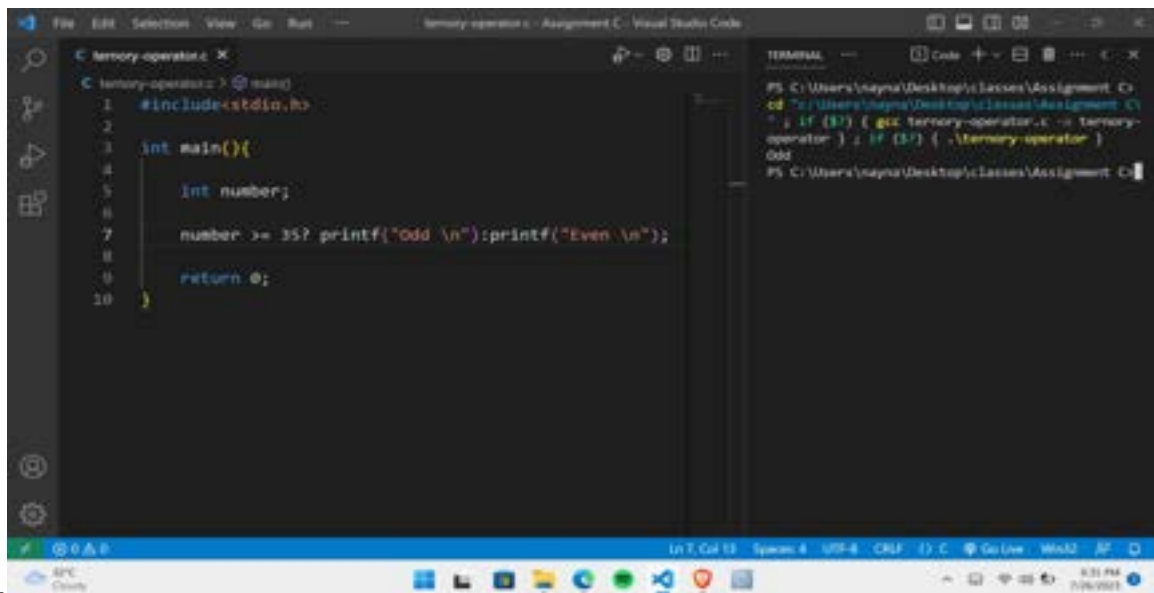


```
Without-swapping.c - Assignment C - Visual Studio Code
C Without-swapping.c >
1 #include<stdio.h>
2 int main()
3 {
4     //Without swapping
5     int P = 13, U = 30;
6     printf("Before swap P is %d and U is %d", P, U);
7     P = P + U; //P = 13 + 30 = 43
8     U = P - U; //U = 43 - 30 = 13
9     P = P - U; //P = 43 - 13 = 30
10    printf("\nAfter swap P is %d and U is %d", P, U);
11    return 0;
12 }
13
```

```
PS C:\Users\Nayna\Desktop\classes\Assignment C>
cd "C:\Users\Nayna\Desktop\classes\Assignment C"
; if ($?) { gcc Without-swapping.c -o Without-swapping } ; if ($?) { .\Without-swapping }
Before swap P is 13 and U is 30
After swap P is 30 and U is 13
PS C:\Users\Nayna\Desktop\classes\Assignment C>
```

Ans.

2> Write a program to find number is even or odd using ternary operator.



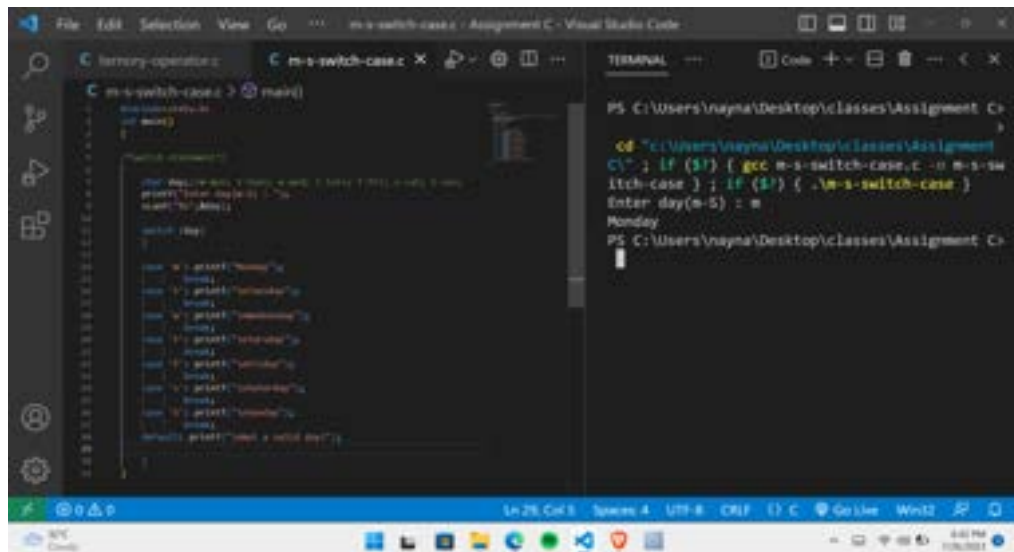
```
ternary-operator.c - Assignment C - Visual Studio Code
C ternary-operator.c >
1 #include<stdio.h>
2
3 int main(){
4
5     int number;
6
7     number >= 35? printf("Odd \n"):printf("Even \n");
8
9     return 0;
10 }
```

```
PS C:\Users\Nayna\Desktop\classes\Assignment C>
cd "C:\Users\Nayna\Desktop\classes\Assignment C"
; if ($?) { gcc ternary-operator.c -o ternary-operator } ; if ($?) { .\ternary-operator }
Odd
PS C:\Users\Nayna\Desktop\classes\Assignment C>
```

Ans.

3> Write a program to show

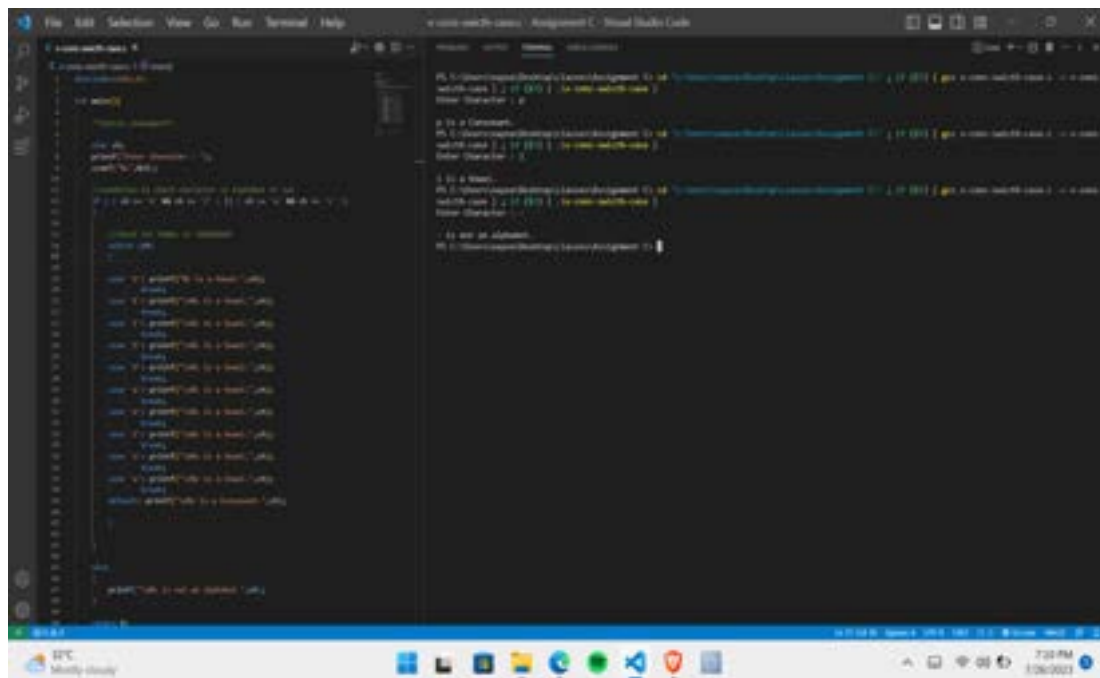
1. Monday to Sunday using switch case.
2. Vowel or Consonant using switch case.



The screenshot shows a Visual Studio Code editor with a C program named `m-s-switch-case.c`. The program uses a `switch` statement to print the days of the week based on an input integer `m` (ranging from 1 to 7). The terminal window shows the command to compile and run the program, and the output displays "Monday" for the input 1.

```
PS C:\Users\vaanya\Desktop\classes\Assignment C> cd "C:\Users\vaanya\Desktop\classes\Assignment C\"; if ($?) { gcc m-s-switch-case.c -o m-s-switch-case }; if ($?) { .\m-s-switch-case }  
Enter day(m-S) : 1  
Monday  
PS C:\Users\vaanya\Desktop\classes\Assignment C>
```

Ans. (1) Ans.



The screenshot shows a Visual Studio Code editor with a C program named `m-s-switch-case.c`. The program uses a `switch` statement to classify a character as a vowel or consonant based on its ASCII value. The terminal window shows the command to compile and run the program, and the output displays "Vowel" for the input 'a'.

```
PS C:\Users\vaanya\Desktop\classes\Assignment C> cd "C:\Users\vaanya\Desktop\classes\Assignment C\"; if ($?) { gcc m-s-switch-case.c -o m-s-switch-case }; if ($?) { .\m-s-switch-case }  
Enter Character : a  
a is a Vowel.  
PS C:\Users\vaanya\Desktop\classes\Assignment C> cd "C:\Users\vaanya\Desktop\classes\Assignment C\"; if ($?) { gcc m-s-switch-case.c -o m-s-switch-case }; if ($?) { .\m-s-switch-case }  
Enter Character : 1  
1 is not an alphabet.  
PS C:\Users\vaanya\Desktop\classes\Assignment C>
```

(2) Ans.

4> Looping programs:

1. Write a program to print 972 to 897 using for loop.
2. Write a program to take 10 no. Input from user and find out
How many Even numbers are there?
How many odd numbers are there?
Sum of even numbers.
Sum of odd numbers.

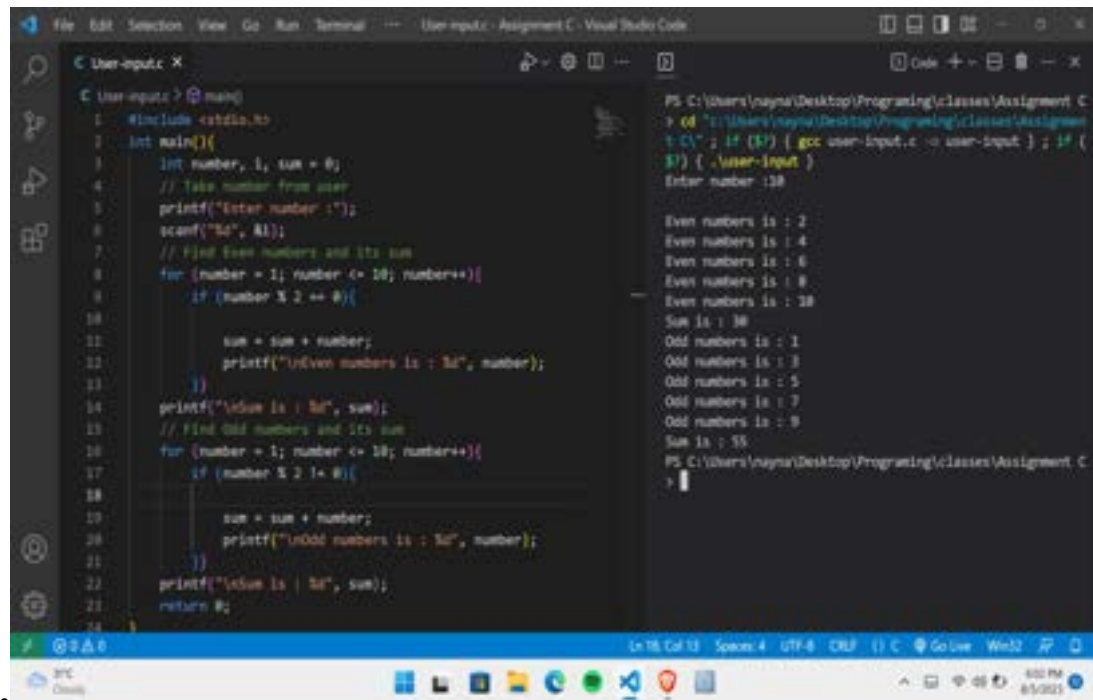
```
#include <iostream>
using namespace std;

int main()
{
    for (int i = 972; i >= 897; i--)
        cout << i << endl;
    return 0;
}
```

Ans. (1) Ans.

```
#include <iostream>
using namespace std;

int main()
{
    int even = 0, odd = 0, sumEven = 0, sumOdd = 0;
    for (int i = 0; i < 10; i++)
    {
        int n;
        cin >> n;
        if (n % 2 == 0)
        {
            even++;
            sumEven += n;
        }
        else
        {
            odd++;
            sumOdd += n;
        }
    }
    cout << "Even numbers: " << even << endl;
    cout << "Odd numbers: " << odd << endl;
    cout << "Sum of even numbers: " << sumEven << endl;
    cout << "Sum of odd numbers: " << sumOdd << endl;
    return 0;
}
```



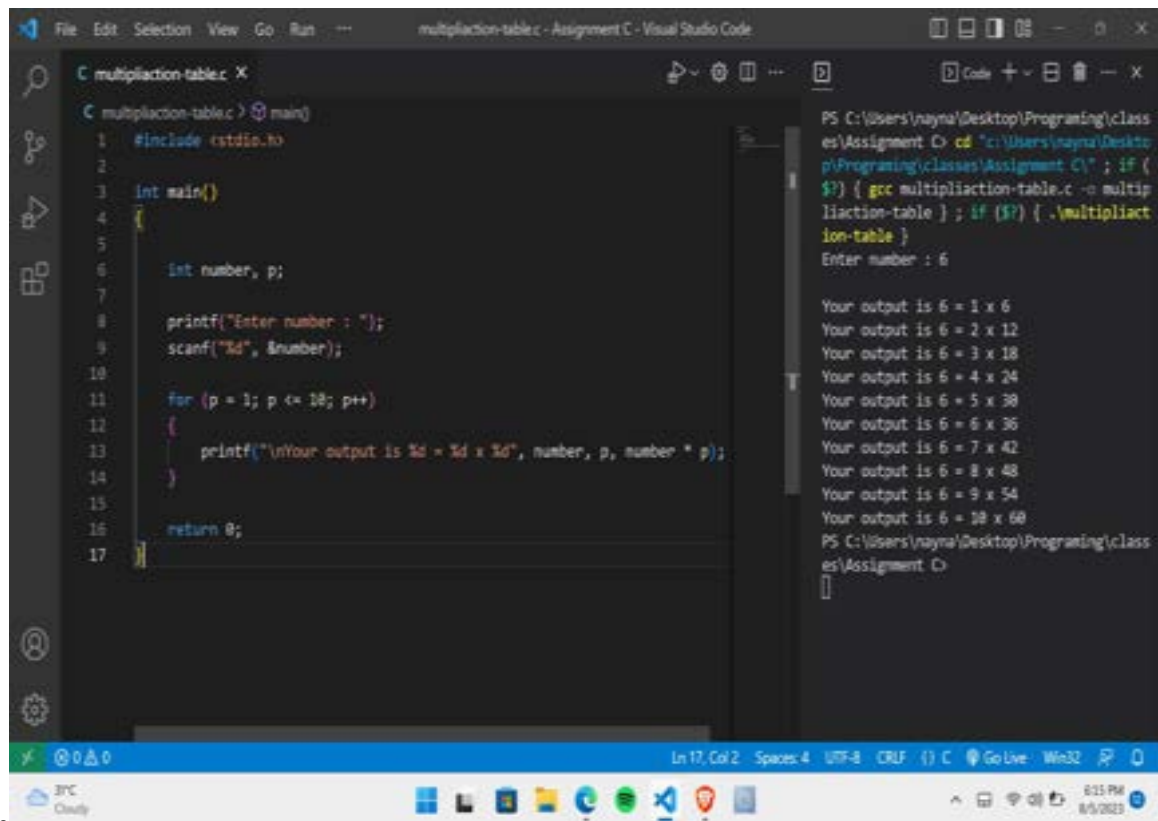
```
C User-input.c X
C User-input.c > main()
1 #include <stdio.h>
2 int main()
3 {
4     int number, i, sum = 0;
5     // Take number from user
6     printf("Enter number :");
7     scanf("%d", &i);
8     // Find Even numbers and its sum
9     for (number = 1; number <= i; number++){
10         if (number % 2 == 0){
11             sum = sum + number;
12             printf("\nEven numbers is : %d", number);
13         }
14     }
15     printf("\nSum is : %d", sum);
16     // Find Odd numbers and its sum
17     for (number = 1; number <= i; number++){
18         if (number % 2 != 0){
19             sum = sum + number;
20             printf("\nOdd numbers is : %d", number);
21         }
22     }
23     printf("\nSum is : %d", sum);
24     return 0;
25 }
```

PS C:\Users\nayna\Desktop\Programming\classes\Assignment C> cd "C:\Users\nayna\Desktop\Programming\classes\Assignment C\"; if (\$?) { gcc user-input.c -o user-input }; if (\$?) { .\user-input }
Enter number 10

Even numbers is : 2
Even numbers is : 4
Even numbers is : 6
Even numbers is : 8
Even numbers is : 10
Sum is : 38
Odd numbers is : 1
Odd numbers is : 3
Odd numbers is : 5
Odd numbers is : 7
Odd numbers is : 9
Sum is : 55
PS C:\Users\nayna\Desktop\Programming\classes\Assignment C>

(2) Ans.

5> Write a program to print the table of the given number.



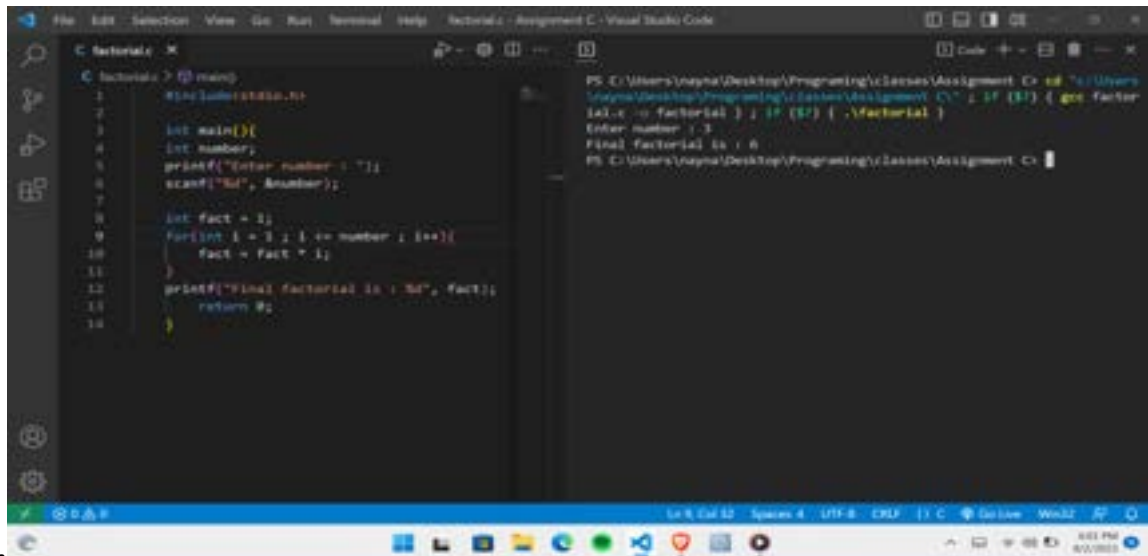
```
C multiplication-table.c X
C multiplication-table.c > main()
1 #include <stdio.h>
2
3 int main()
4 {
5
6     int number, p;
7
8     printf("Enter number : ");
9     scanf("%d", &number);
10
11     for (p = 1; p <= 10; p++)
12     {
13         printf("\nYour output is %d = %d x %d", number, p, number * p);
14     }
15
16     return 0;
17 }
```

PS C:\Users\nayna\Desktop\Programming\classes\Assignment C> cd "C:\Users\nayna\Desktop\Programming\classes\Assignment C\"; if (\$?) { gcc multiplication-table.c -o multiplication-table }; if (\$?) { .\multiplication-table }
Enter number : 6

Your output is 6 = 1 x 6
Your output is 6 = 2 x 12
Your output is 6 = 3 x 18
Your output is 6 = 4 x 24
Your output is 6 = 5 x 30
Your output is 6 = 6 x 36
Your output is 6 = 7 x 42
Your output is 6 = 8 x 48
Your output is 6 = 9 x 54
Your output is 6 = 10 x 60
PS C:\Users\nayna\Desktop\Programming\classes\Assignment C>

Ans.

6> Write a program to print factorial of given number.



The screenshot shows a C program in Visual Studio Code. The code is as follows:

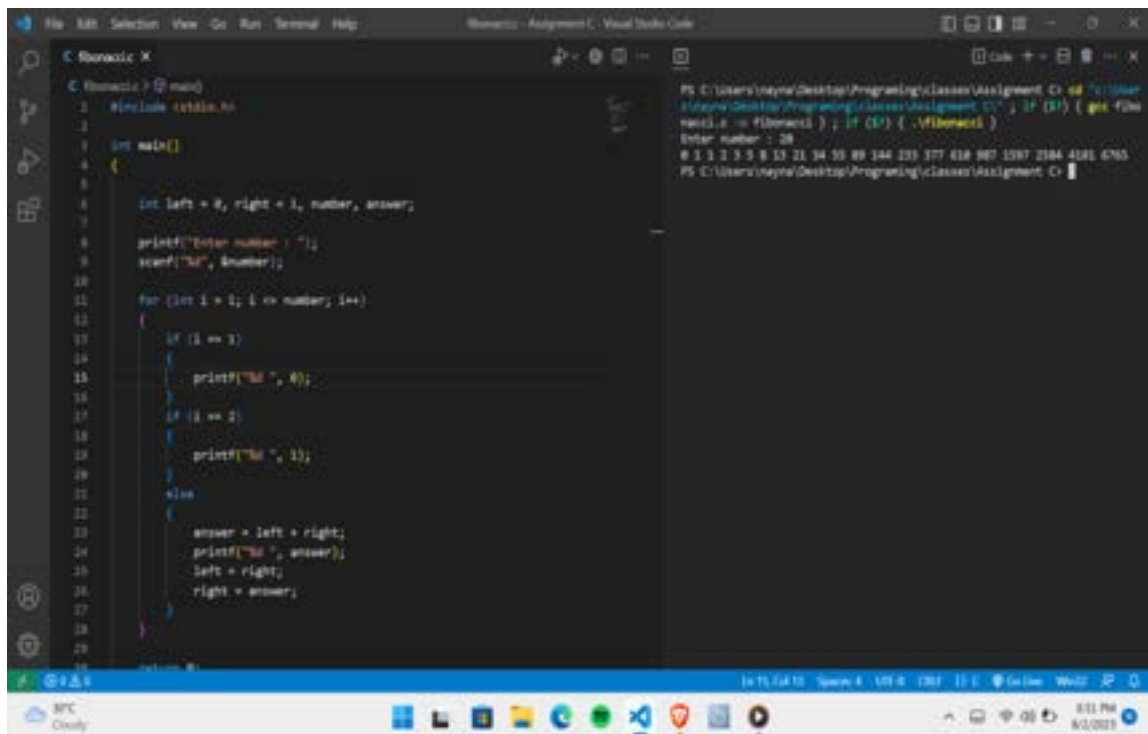
```
1 #include <stdio.h>
2
3 int main()
4 {
5     int number;
6     printf("Enter number : ");
7     scanf("%d", &number);
8
9     int fact = 1;
10    for(int i = 1; i <= number; i++){
11        fact = fact * i;
12    }
13    printf("Final factorial is : %d", fact);
14    return 0;
15 }
```

The terminal output shows the program execution:

```
PS C:\Users\neha\Desktop\Programming\classes\Assignment C> cd "C:\Users\neha\Desktop\Programming\classes\Assignment C" & if ($?) { gcc factor
al.c -o factorial } & if ($?) { ./factorial }
Enter number : 3
Final factorial is : 6
PS C:\Users\neha\Desktop\Programming\classes\Assignment C>
```

Ans.

7> Write a program to print Fibonacci series up to given numbers.



The screenshot shows a C program in Visual Studio Code. The code is as follows:

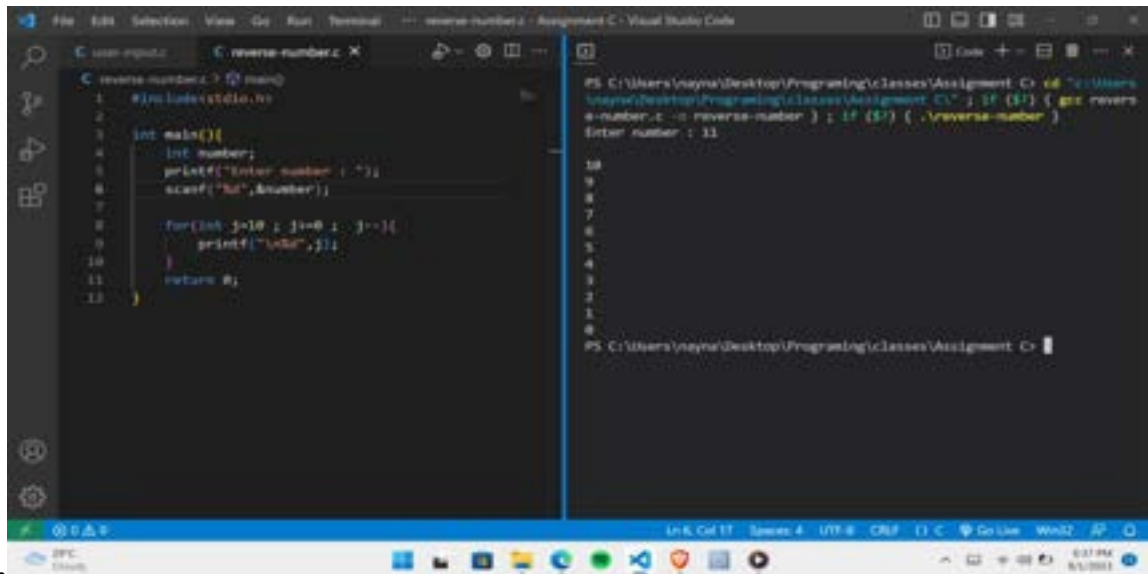
```
1 #include <stdio.h>
2
3 int main()
4 {
5     int left = 0, right = 1, number, answer;
6
7     printf("Enter number : ");
8     scanf("%d", &number);
9
10    for (int i = 1; i <= number; i++)
11    {
12        if (i == 1)
13        {
14            printf("%d ", 0);
15        }
16        if (i == 2)
17        {
18            printf("%d ", 1);
19        }
20        else
21        {
22            answer = left + right;
23            printf("%d ", answer);
24            left = right;
25            right = answer;
26        }
27    }
28    return 0;
29 }
```

The terminal output shows the program execution:

```
PS C:\Users\neha\Desktop\Programming\classes\Assignment C> cd "C:\Users\neha\Desktop\Programming\classes\Assignment C" & if ($?) { gcc fibo
nacci.c -o fibonacci } & if ($?) { ./fibonacci }
Enter number : 20
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 618 987 1597 2584 4181 6765
PS C:\Users\neha\Desktop\Programming\classes\Assignment C>
```

Ans.

8> Write a program to print numbers in reverse order.

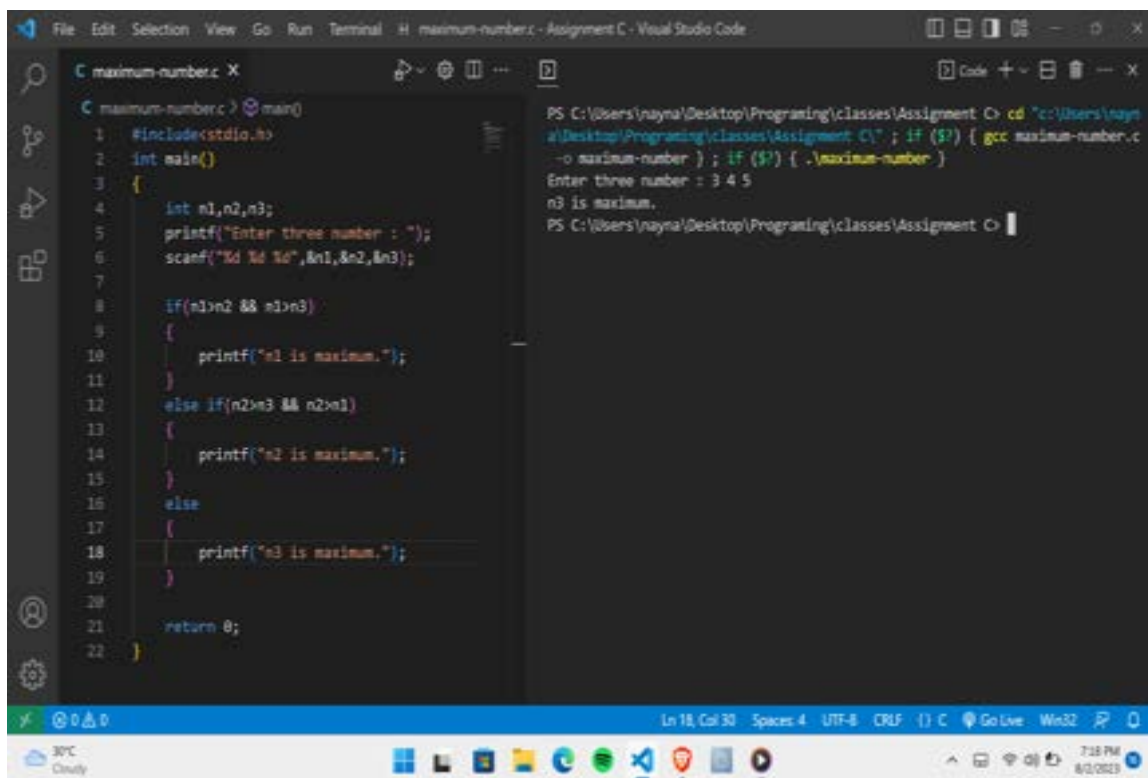


```
C reverse-number.c > main()
1 #include<stdio.h>
2
3 int main(){
4     int number;
5     printf("Enter number : ");
6     scanf("%d",&number);
7
8     for(int i=number; i>0; i--){
9         printf("%d\n",i);
10    }
11    return 0;
12 }
```

```
PS C:\Users\nayna\Desktop\Programming\classes\Assignment C> cd "c:\Users\nayna\Desktop\Programming\classes\Assignment C" ; if ($?) { gcc reverse-number.c -o reverse-number } ; if ($?) { .\reverse-number }
Enter number : 11
11
10
9
8
7
6
5
4
3
2
1
PS C:\Users\nayna\Desktop\Programming\classes\Assignment C>
```

Ans.

9> Write a program to find out the max from the given number.

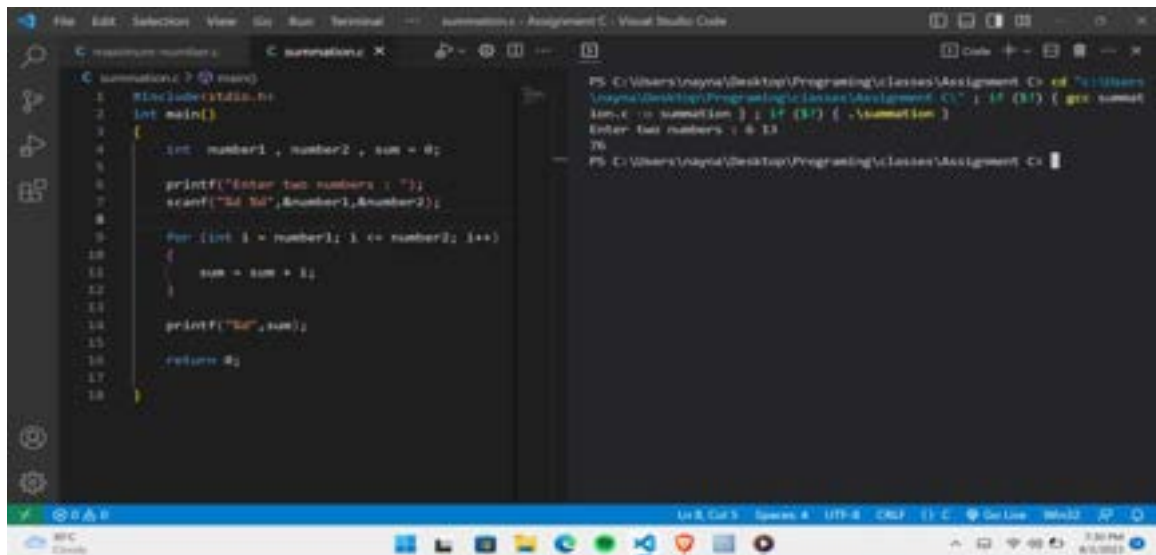


```
C maximum-number.c > main()
1 #include<stdio.h>
2 int main()
3 {
4     int n1,n2,n3;
5     printf("Enter three number : ");
6     scanf("%d %d %d",&n1,&n2,&n3);
7
8     if(n1>n2 && n1>n3)
9     {
10        printf("n1 is maximum.");
11    }
12    else if(n2>n3 && n2>n1)
13    {
14        printf("n2 is maximum.");
15    }
16    else
17    {
18        printf("n3 is maximum.");
19    }
20
21    return 0;
22 }
```

```
PS C:\Users\nayna\Desktop\Programming\classes\Assignment C> cd "c:\Users\nayna\Desktop\Programming\classes\Assignment C" ; if ($?) { gcc maximum-number.c -o maximum-number } ; if ($?) { .\maximum-number }
Enter three number : 3 4 5
n3 is maximum.
PS C:\Users\nayna\Desktop\Programming\classes\Assignment C>
```

Ans.

10> Write a program and make a summation of the given number.

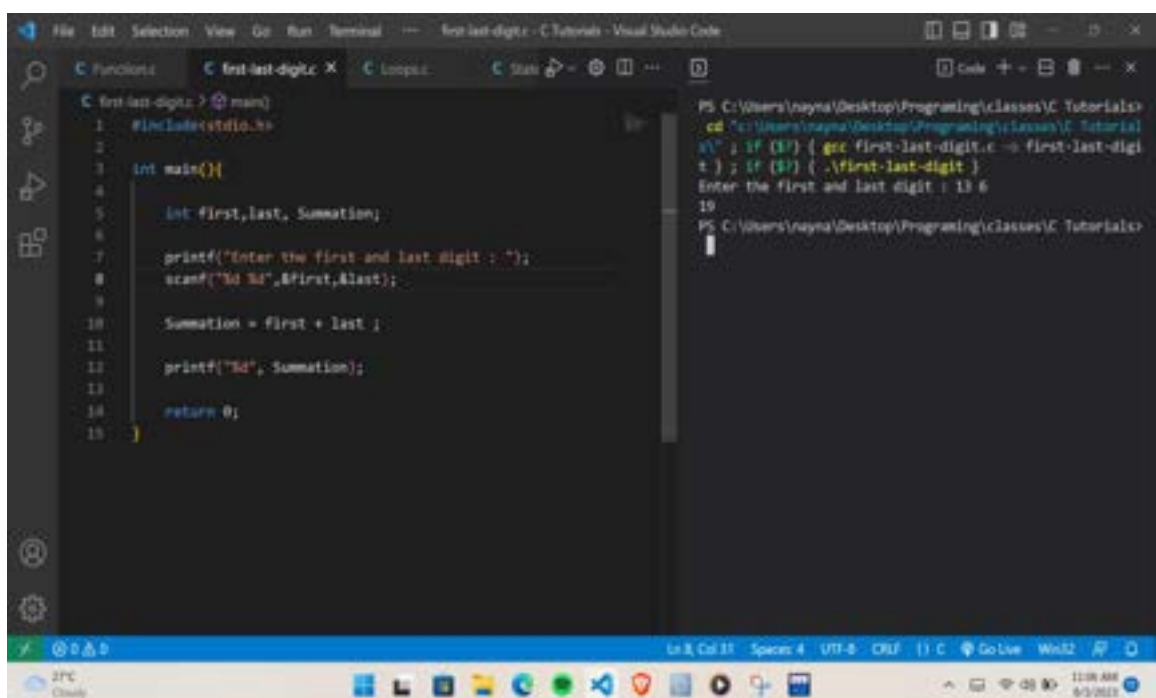


```
1 #include<stdio.h>
2 int main()
3 {
4     int number1 , number2 , sum = 0;
5
6     printf("Enter two numbers : ");
7     scanf("%d %d",&number1,&number2);
8
9     for (int i = number1; i <= number2; i++)
10     {
11         sum = sum + i;
12     }
13
14     printf("Sum",sum);
15
16     return 0;
17 }
```

```
PS C:\Users\Nayna\Desktop\Programming\classes\Assignment C> cd "C:\Users\Nayna\Desktop\Programming\classes\Assignment C" & if ($?) { gcc summation.c -o summation } & if ($?) { .\summation }
Enter two numbers : 4 13
27
PS C:\Users\Nayna\Desktop\Programming\classes\Assignment C>
```

Ans.

11> Write a program you must make a summation of first and last Digit.



```
1 #include<stdio.h>
2
3 int main(){
4
5     int first,last, Summation;
6
7     printf("Enter the first and last digit : ");
8     scanf("%d %d",&first,&last);
9
10    Summation = first + last ;
11
12    printf("%d", Summation);
13
14    return 0;
15 }
```

```
PS C:\Users\Nayna\Desktop\Programming\classes\C Tutorials> cd "C:\Users\Nayna\Desktop\Programming\classes\C Tutorials" & if ($?) { gcc first-last-digit.c -o first-last-digit } & if ($?) { .\first-last-digit }
Enter the first and last digit : 13 6
19
PS C:\Users\Nayna\Desktop\Programming\classes\C Tutorials>
```

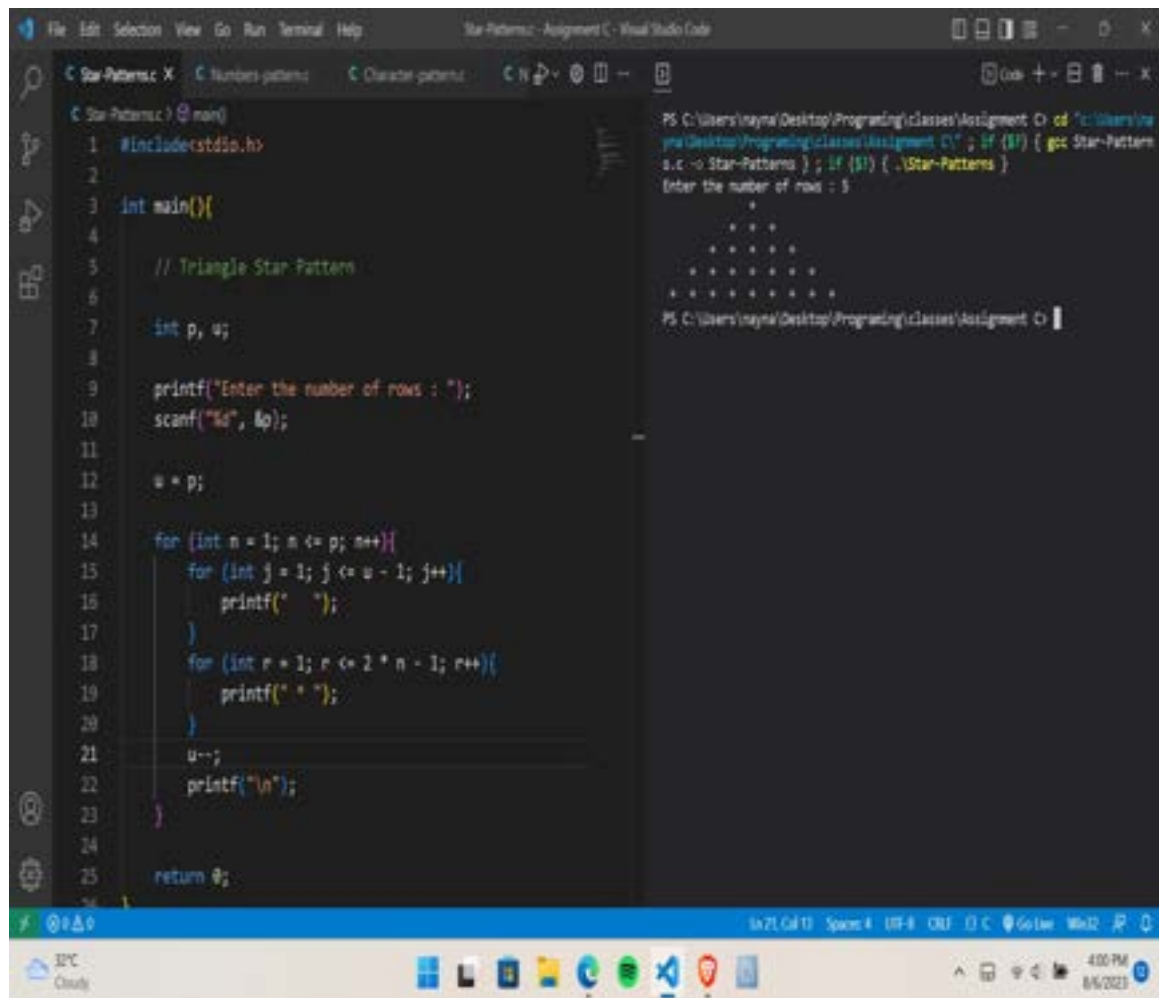
Ans.

12>Patterns Programs:

(1)

```

          *
        * * *
      * * * * *
    * * * * * *
  * * * * * * *
* * * * * * * *
```



```
Star-Patterns.c: Assignment C - Visual Studio Code
File Edit Selection View Go Run Terminal Help
C Star-Patterns.c X C Numbers-patterns C Character-patterns C
C Star-Patterns.c (main)
1 #include<stdio.h>
2
3 int main(){
4
5     // Triangle Star-Pattern
6
7     int p, u;
8
9     printf("Enter the number of rows : ");
10    scanf("%d", &p);
11
12    u = p;
13
14    for (int n = 1; n <= p; n++){
15        for (int j = 1; j <= u - 1; j++){
16            printf(" ");
17        }
18        for (int r = 1; r <= 2 * n - 1; r++){
19            printf(" * ");
20        }
21        u--;
22        printf("\n");
23    }
24
25    return 0;

```

```
PS C:\Users\inayna\Desktop\Programming\Classes\Assignment C> cd "C:\Users\inayna\Desktop\Programming\Classes\Assignment C" & if ($?) { gcc Star-Patterns.c -o Star-Patterns } & if ($?) { .\Star-Patterns }
Enter the number of rows : 5
      *
    * *
  * * *
* * * *
* * * * *
```

Ans.

```

*
*      *
*      *      *
*      *      *      *
*      *      *      *      *
*      *      *      *      *      *
*      *      *      *      *      *
*      *      *      *
*      *
*
```

The screenshot displays a Windows 10 desktop environment. The primary focus is the Visual Studio Code (VS Code) application, which is open with a C++ source file named 'Cross-Triagle Star-Patterns.c'. The code is as follows:

```

1 #include <iostream>
2
3 int main()
4 {
5     // Cross triangle Star Pattern
6
7     int p, n = 5;
8
9     printf("Enter the number of columns : ");
10    scanf("%d", &n);
11
12    for (int i = 1; i <= n; i++)
13    {
14        for (int j = 1; j <= n; j++)
15        {
16            printf("* ");
17        }
18        printf("\n");
19    }
20
21    for (int i = p + 1; i <= n; i++)
22    {
23        for (int j = 1; j <= n; j++)
24        {
25            printf(" ");
26        }
27        printf("\n");
28    }
29
30    return 0;
31 }

```

The output of the program is shown in the VS Code terminal, indicating that the number of columns is 5 and displaying a 5x5 grid of stars with a cross pattern:

```

PS C:\Users\mayne\Desktop\Programming\Classes\Assignment C> cd "C:\Users\mayne\Desktop\Programming\Classes\Assignment C" & if ($?) { gcc Cross-Triagle-Star-Patterns.c -o Cross-Triagle-Star-Pattern } & if ($?) { .\Cross-Triagle-Star-Pattern }
Enter the number of columns : 5
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

```

The Windows taskbar at the bottom of the screen shows the system clock as 4:05 PM on 8/6/2021, along with various background applications and the network status.

Ans.

(3)

1				
2	3			
4	5	6		
7	8	9	10	
11	12	13	14	15

```

C Number-Pattern.c (main)
1 #include <stdio.h>
2
3 int main()
4 {
5     // Number Pattern
6
7     int rows, i, j, number = 1;
8
9     printf("Enter the number of rows: ");
10    scanf("%d", &rows);
11
12    for (i = 1; i <= rows; i++)
13    {
14        for (j = 1; j <= i; j++)
15        {
16            printf("%d ", number);
17            ++number;
18        }
19        printf("\n");
20    }
21
22    return 0;
23 }

```

```

PS C:\Users\neyna\Desktop\Programming\classes\Assignment C> cd "c:\Users\neyna\Desktop\Programming\classes\Assignment C"; if ($?) { gcc Number-Pattern.c -o Number-Pattern }; if ($?) { .\Number-Pattern }
Enter the number of rows: 5
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
PS C:\Users\neyna\Desktop\Programming\classes\Assignment C>

```

Ans.

(4)

1				
1	0			
1	0	1		
1	0	1	0	
1	0	1	0	1

```

1 #include <stdio.h>
2
3 void pattern(int a);
4
5 int main()
6 {
7     pattern(5);
8
9     return 0;
10 }
11
12 void pattern(int a)
13 {
14     for (int d = 1; d <= a; d++)
15     {
16         printf("\n");
17         for (int e = 1; e <= d; e++)
18         {
19             int ans = (e % 2);
20             printf("%d ", ans);
21         }
22     }
23 }
  
```

Terminal Output:

```

PS C:\Users\mayna\Desktop\Programming\classes\Assignment C> cd "c:\Users\mayna\Desktop\Programming\classes\Assignment C\" ; if ($?) { gcc Numbers-pattern.c -o Numbers-pattern } ; if ($?) { .\Numbers-pattern }
1
1 0
1 0 1
1 0 1 0
1 0 1 0 1
  
```

Ans.

(5) A

A B

A B C

A B C D

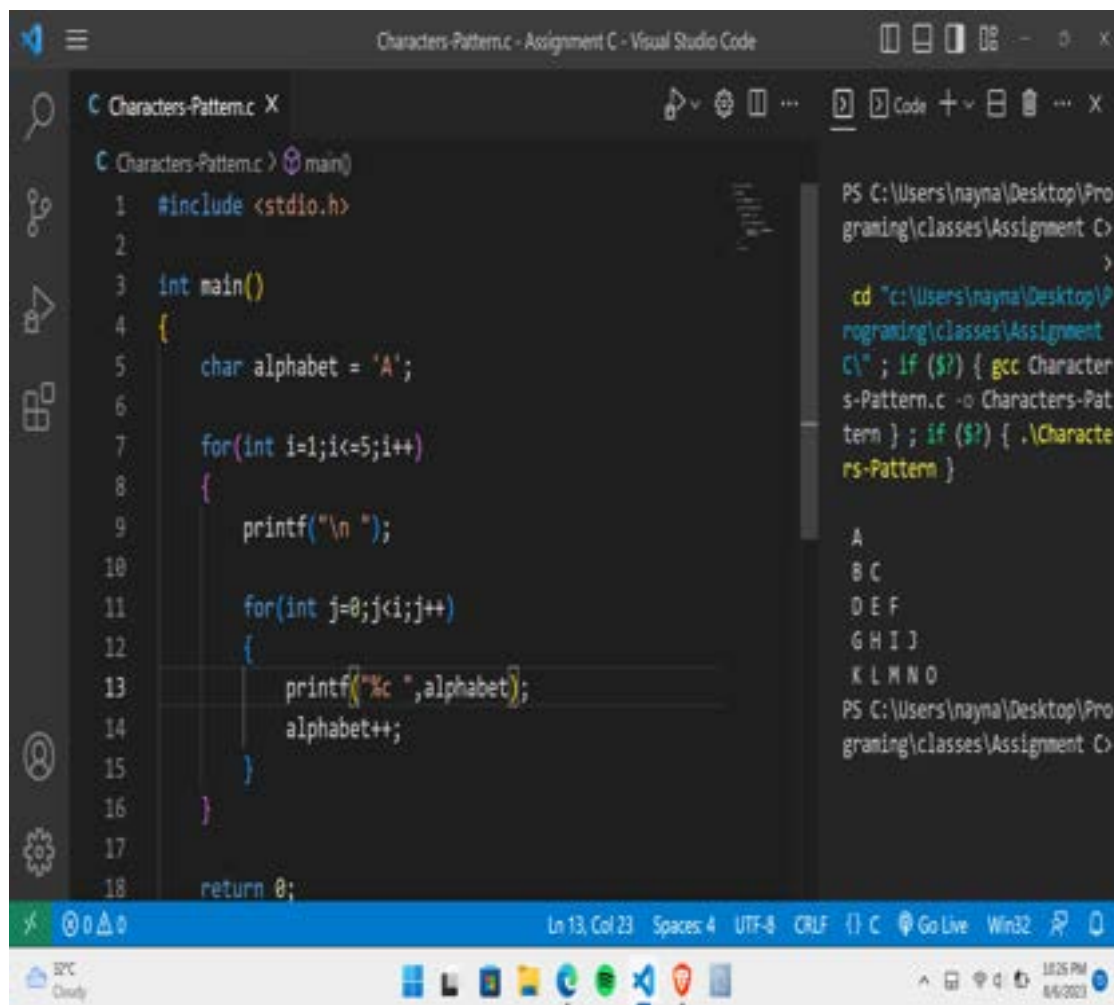
A B C D E

```
1 #include <stdio.h>
2
3 int main()
4 {
5     // Character Pattern
6
7     int n, x, y;
8
9     printf("Enter number of row to show character pattern: ");
10    scanf("%d", &n);
11
12    for (x = 1; x <= n; x++)
13    {
14        for (y = 1; y <= x; y++)
15        {
16            printf("%c ", 'A' + y - 1);
17        }
18        printf("\n");
19    }
20
21    return 0;
22 }
```

PS C:\Users\neha\Desktop\Programming\Classes\Assignment C> cd "C:\Users\neha\Desktop\Programming\Classes\Assignment C" & if (\$?) { gcc Character-pattern.c -o Character-pattern } & if (\$?) { .\Character-pattern }
Enter number of row to show character pattern: 5
A
A B
A B C
A B C D
A B C D E
PS C:\Users\neha\Desktop\Programming\Classes\Assignment C>

Ans.

(6) A
B C
D E F
G H I J
K L M N O



```
C Characters-Pattern.c X
C Characters-Pattern.c > main()
1 #include <stdio.h>
2
3 int main()
4 {
5     char alphabet = 'A';
6
7     for(int i=1;i<=5;i++)
8     {
9         printf("\n ");
10
11         for(int j=0;j<i;j++)
12         {
13             printf("%c ",alphabet);
14             alphabet++;
15         }
16     }
17
18     return 0;
```

PS C:\Users\nayna\Desktop\Programming\classes\Assignment C>
cd "c:\Users\nayna\Desktop\Programming\classes\Assignment C\"; if (\$?) { gcc Characters-Pattern.c -o Characters-Pattern }; if (\$?) { .\Characters-Pattern }

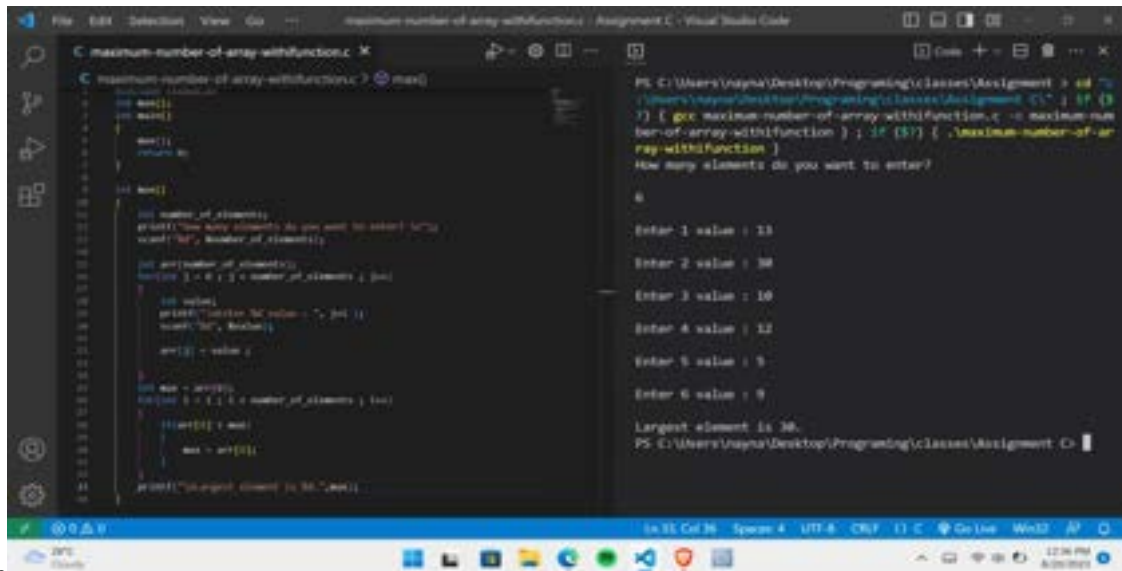
A
B C
D E F
G H I J
K L M N O
PS C:\Users\nayna\Desktop\Programming\classes\Assignment C>

Ln 13, Col 23 Spaces 4 UTF-8 CRLF {} C Go Live Win32 18:25 PM 8/8/2023

Ans.

Module 3.3 --> File Handling and debugging

1> Write a program to find out the max number from given array using function.



The screenshot shows a Visual Studio Code editor with a C program named 'maximum-number-of-array-withfunction.c'. The program defines a function 'max()' that takes an array and its size as arguments, iterates through the array to find the maximum value, and returns it. The main function prompts the user for the number of elements, reads the values into an array, and then calls the 'max()' function to find the largest element. The output window shows the program's execution, where the user enters 6 elements, and the program correctly identifies 36 as the largest element.

```
C:\maximum-number-of-array-withfunction.c> max()
max()
return 0;

int max()
{
    int number_of_elements;
    printf("How many elements do you want to enter? ");
    scanf("%d", &number_of_elements);

    int arr[number_of_elements];
    for(int i = 0; i < number_of_elements; i++)
    {
        int value;
        printf("Enter value %d: ", i);
        scanf("%d", &value);
        arr[i] = value;
    }

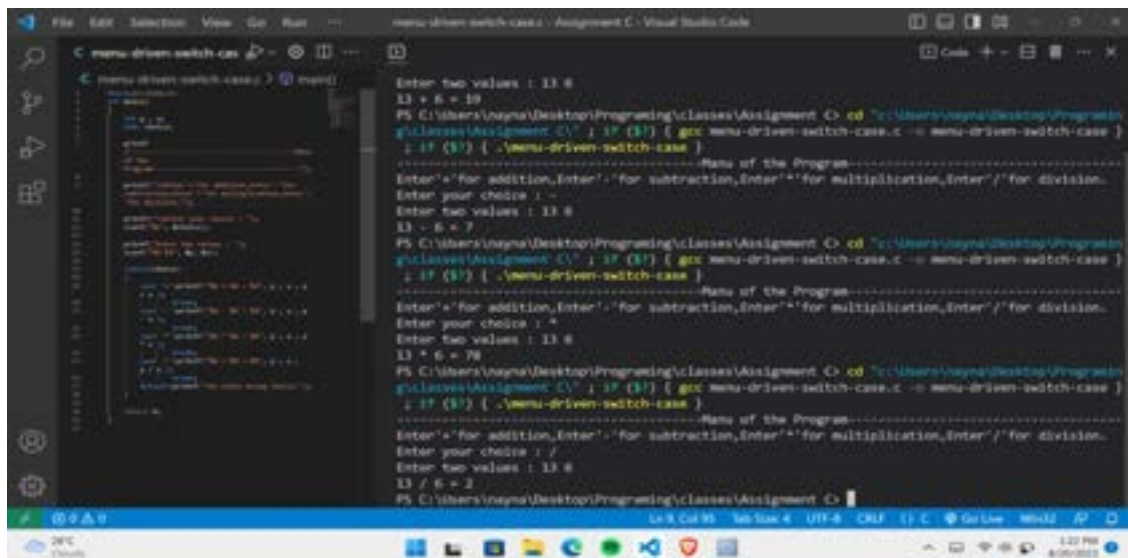
    int Max = arr[0];
    for(int i = 1; i < number_of_elements; i++)
    {
        if(arr[i] > Max)
        {
            Max = arr[i];
        }
    }

    printf("Largest element is: %d\n", Max);
}
```

PS C:\Users\Vaanya\Desktop\Programming\Classes\Assignment C> cd ~\Users\Vaanya\Desktop\Programming\Classes\Assignment C> gcc maximum-number-of-array-withfunction.c -o maximum-number-of-array-withfunction
How many elements do you want to enter?
6
Enter 1 value : 13
Enter 2 value : 36
Enter 3 value : 10
Enter 4 value : 12
Enter 5 value : 5
Enter 6 value : 9
Largest element is 36.
PS C:\Users\Vaanya\Desktop\Programming\Classes\Assignment C>

Ans.

2> Write a program of Addition, Subtraction, Multiplication and Division using Switch case. (Must Be Menu Driven).



The screenshot shows a Visual Studio Code editor with a C program named 'menu-driven-switch-case.c'. The program uses a switch case to perform arithmetic operations based on the user's choice. The menu options are: 1 for addition, 2 for subtraction, 3 for multiplication, and 4 for division. The program prompts the user for their choice and then for two values to perform the operation. The output window shows the program's execution for three different choices, correctly performing addition, subtraction, and multiplication.

```
C:\menu-driven-switch-case.c> menu()
menu()
return 0;

int menu()
{
    int choice;
    printf("Menu of the Program\n");
    printf("Enter 1 for addition, Enter 2 for subtraction, Enter 3 for multiplication, Enter 4 for division.\n");
    printf("Enter your choice : ");
    scanf("%d", &choice);

    int a, b;
    printf("Enter two values : ");
    scanf("%d %d", &a, &b);

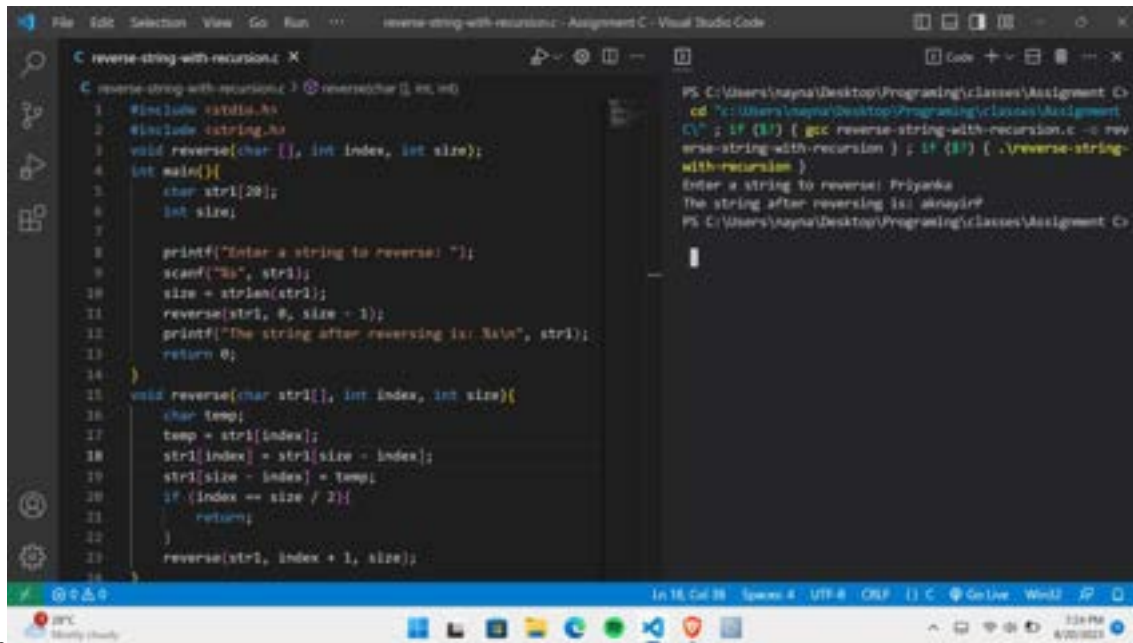
    switch(choice)
    {
        case 1:
            printf("%d + %d = %d\n", a, b, a + b);
            break;
        case 2:
            printf("%d - %d = %d\n", a, b, a - b);
            break;
        case 3:
            printf("%d * %d = %d\n", a, b, a * b);
            break;
        case 4:
            printf("%d / %d = %d\n", a, b, a / b);
            break;
    }

    return 0;
}
```

PS C:\Users\Vaanya\Desktop\Programming\Classes\Assignment C> cd ~\Users\Vaanya\Desktop\Programming\Classes\Assignment C> gcc menu-driven-switch-case.c -o menu-driven-switch-case
Menu of the Program
Enter 1 for addition, Enter 2 for subtraction, Enter 3 for multiplication, Enter 4 for division.
Enter your choice : 1
Enter two values : 13 8
13 + 8 = 21
PS C:\Users\Vaanya\Desktop\Programming\Classes\Assignment C> cd ~\Users\Vaanya\Desktop\Programming\Classes\Assignment C> gcc menu-driven-switch-case.c -o menu-driven-switch-case
Menu of the Program
Enter 1 for addition, Enter 2 for subtraction, Enter 3 for multiplication, Enter 4 for division.
Enter your choice : 2
Enter two values : 13 8
13 - 8 = 5
PS C:\Users\Vaanya\Desktop\Programming\Classes\Assignment C> cd ~\Users\Vaanya\Desktop\Programming\Classes\Assignment C> gcc menu-driven-switch-case.c -o menu-driven-switch-case
Menu of the Program
Enter 1 for addition, Enter 2 for subtraction, Enter 3 for multiplication, Enter 4 for division.
Enter your choice : 3
Enter two values : 13 8
13 * 8 = 104
PS C:\Users\Vaanya\Desktop\Programming\Classes\Assignment C>

Ans.

3> Write a program to find the reverse of string using recursion.

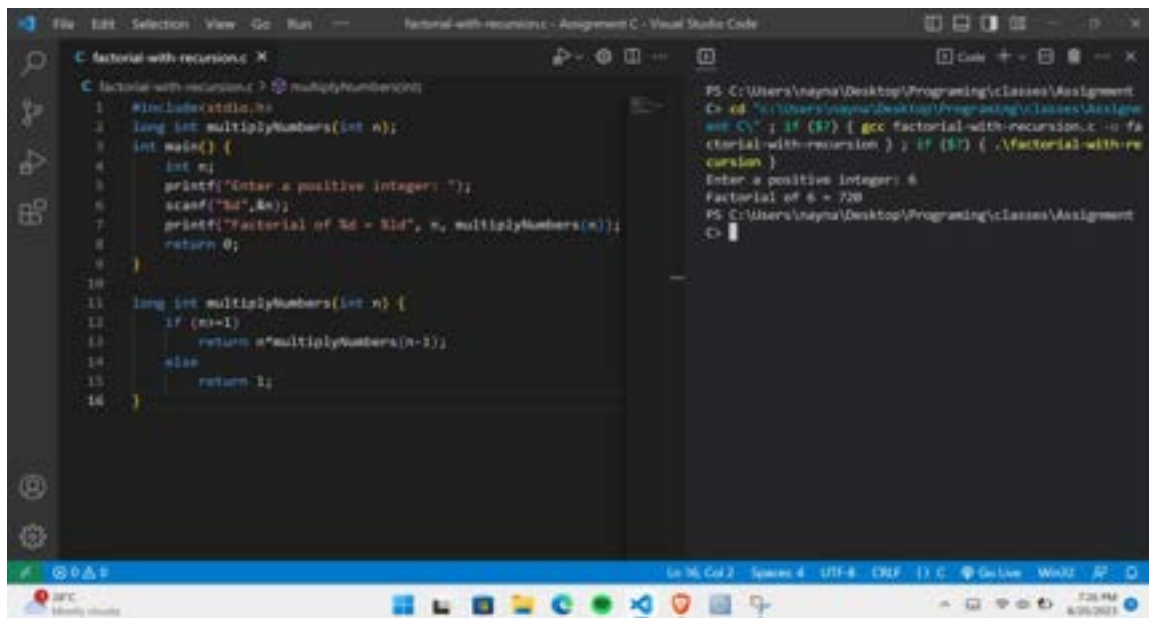


```
reverse-string-with-recursion.c
1 #include <stdio.h>
2 #include <string.h>
3 void reverse(char [], int index, int size);
4 int main()
5 {
6     char str[20];
7     int size;
8
9     printf("Enter a string to reverse: ");
10    scanf("%s", str);
11    size = strlen(str);
12    reverse(str, 0, size - 1);
13    printf("The string after reversing is: %s\n", str);
14    return 0;
15
16    void reverse(char str[], int index, int size){
17        char temp;
18        temp = str[index];
19        str[index] = str[size - index];
20        str[size - index] = temp;
21        if (index <= size / 2){
22            return;
23        }
24        reverse(str, index + 1, size);
25    }
26 }
```

```
PS C:\Users\nayna\Desktop\Programming\classes\Assignment C>
cd "C:\Users\nayna\Desktop\Programming\classes\Assignment
C"; if ($?) { gcc reverse-string-with-recursion.c -o rev
erse-string-with-recursion }; if ($?) { ./reverse-string-
with-recursion }
Enter a string to reverse: Priyanka
The string after reversing is: aknayirP
PS C:\Users\nayna\Desktop\Programming\classes\Assignment C>
```

Ans.

4> Write a program to find factorial using recursion.



```
factorial-with-recursion.c
1 #include <stdio.h>
2 long int multiplyNumbers(int n);
3 int main() {
4     int n;
5     printf("Enter a positive integer: ");
6     scanf("%d", &n);
7     printf("Factorial of %d = %ld", n, multiplyNumbers(n));
8     return 0;
9
10    long int multiplyNumbers(int n) {
11        if (n==1)
12            return n*multiplyNumbers(n-1);
13        else
14            return 1;
15    }
16 }
```

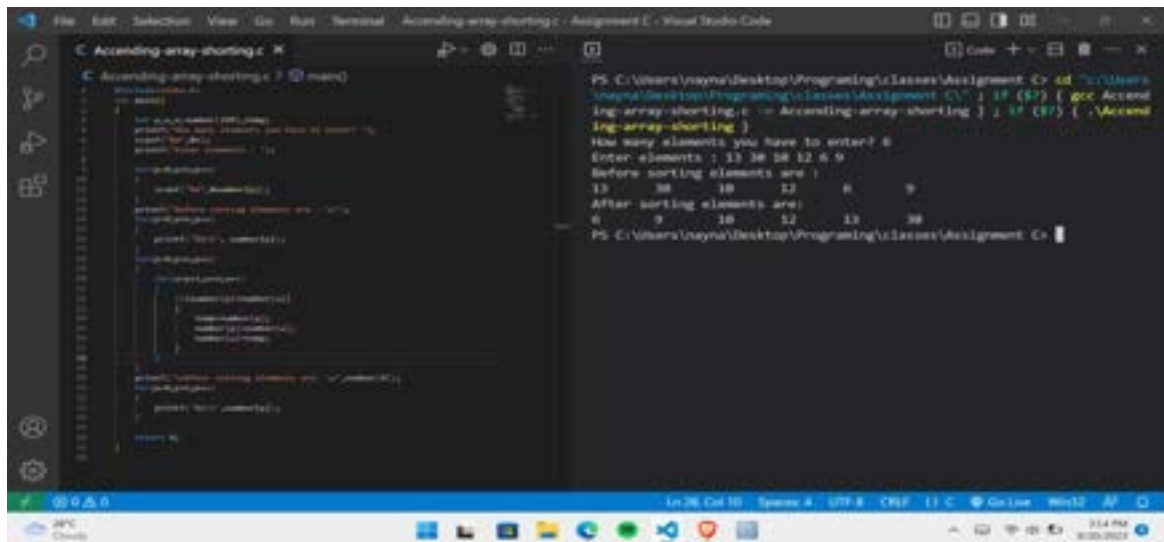
```
PS C:\Users\nayna\Desktop\Programming\classes\Assignment
C> cd "C:\Users\nayna\Desktop\Programming\classes\Assigne
ment C"; if ($?) { gcc factorial-with-recursion.c -o fa
ctorial-with-recursion }; if ($?) { ./factorial-with-re
cursion }
Enter a positive integer: 6
Factorial of 6 = 720
PS C:\Users\nayna\Desktop\Programming\classes\Assignment
C>
```

Ans.

5> Write a program to take two Array input from user and sort them in ascending or descending order as per user's choice.

Ans.

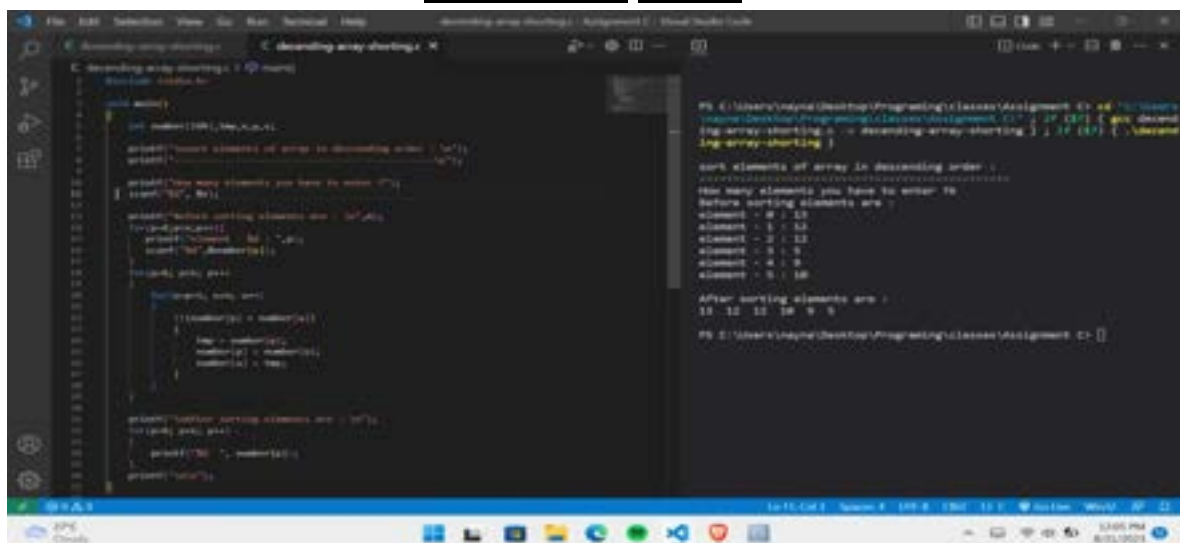
Acceding Order



The screenshot shows a Visual Studio Code editor with a C++ program titled "Ascending-array-sorting.c". The program prompts the user to enter the number of elements and then the elements themselves. It then sorts the array in ascending order and displays the result. The terminal output shows the following sequence of events:

```
PS C:\Users\usayna\Desktop\Programming\classes\Assignment C> cd "C:\Users\usayna\Desktop\Programming\classes\Assignment C" & gcc Ascending-array-sorting.c -o Ascending-array-sorting & .\Ascending-array-sorting
How many elements you have to enter? 6
Enter elements : 13 38 18 12 6 9
Before sorting elements are :
13 38 18 12 6 9
After sorting elements are:
6 9 12 18 38 13
PS C:\Users\usayna\Desktop\Programming\classes\Assignment C>
```

Descending Order

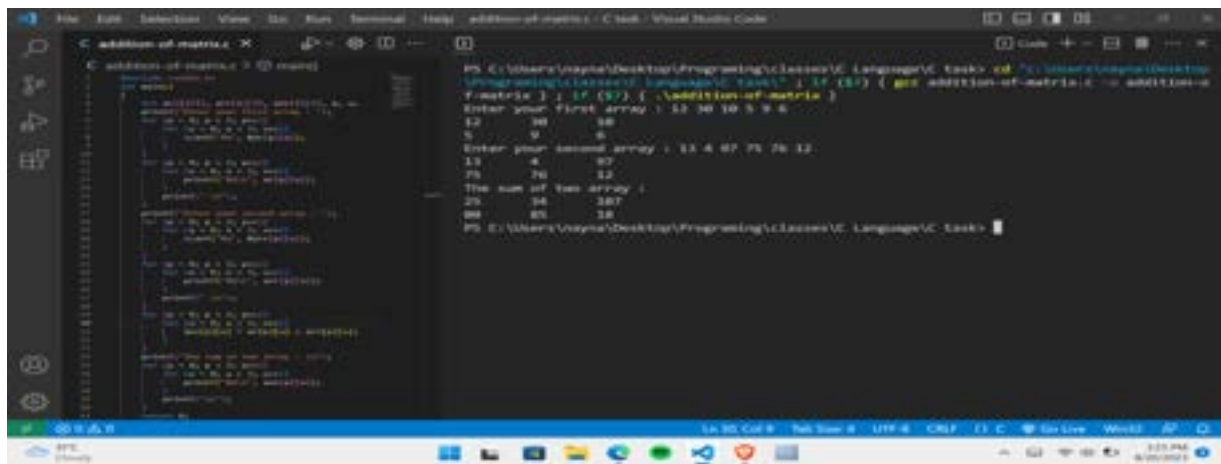


The screenshot shows a Visual Studio Code editor with a C++ program titled "Descending-array-sorting.c". The program prompts the user to enter the number of elements and then the elements themselves. It then sorts the array in descending order and displays the result. The terminal output shows the following sequence of events:

```
PS C:\Users\usayna\Desktop\Programming\classes\Assignment C> cd "C:\Users\usayna\Desktop\Programming\classes\Assignment C" & gcc Descending-array-sorting.c -o Descending-array-sorting & .\Descending-array-sorting
sort elements of array in descending order :
How many elements you have to enter 10
Before sorting elements are :
element - 0 : 13
element - 1 : 38
element - 2 : 12
element - 3 : 6
element - 4 : 9
element - 5 : 18
After sorting elements are :
18 12 13 38 9 6
PS C:\Users\usayna\Desktop\Programming\classes\Assignment C>
```

6> Write a program to make Addition, Subtraction and Multiplication of two matrix using 2-D Array.

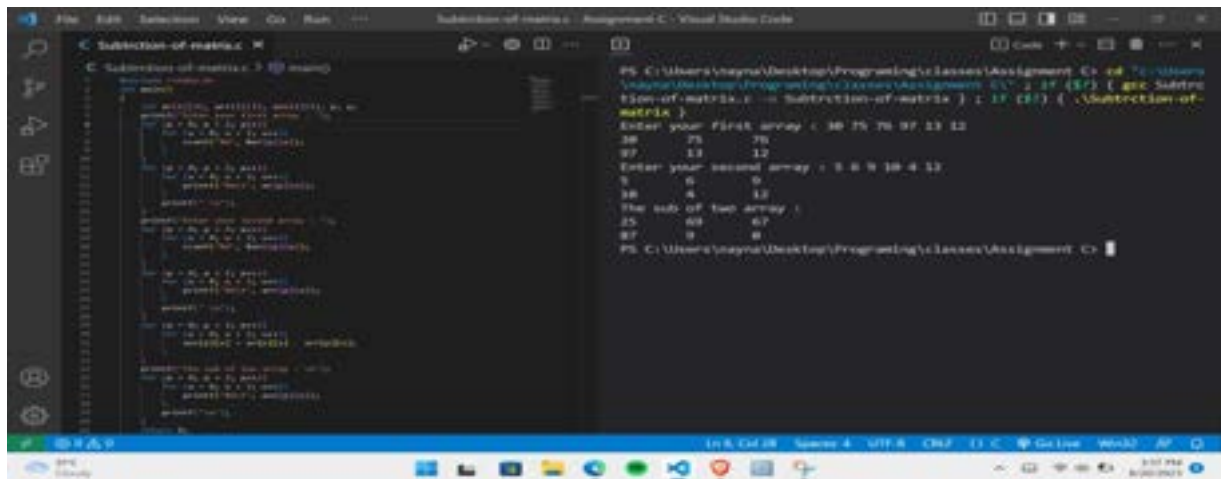
Ans. Addition of two matrix using 2-D Array



The screenshot shows a C++ program in Visual Studio Code titled "addition-of-matrix.c". The code defines two 3x3 matrices, A and B, and a result matrix C. It uses nested loops to read values for matrix A, then matrix B, and finally to calculate the sum of corresponding elements in A and B, storing the result in C. The program then prints the resulting matrix C. The terminal output shows the execution of the program with the following input and output:

```
PS C:\Users\Vaanya\Desktop\Programming\Classes\VC_Language\VC_tasks> cd "C:\Users\Vaanya\Desktop\Programming\Classes\VC_Language\VC_tasks" & gcc addition-of-matrix.c -o addition-of-matrix.exe & .\addition-of-matrix.exe
Enter your first array : 12 34 56 78 90 12 34 56 78
12 34 56
78 90 12
Enter your second array : 11 4 87 75 76 12 11 4 87
11 4 87
75 76 12
The sum of two array :
23 38 143
53 166 90
```

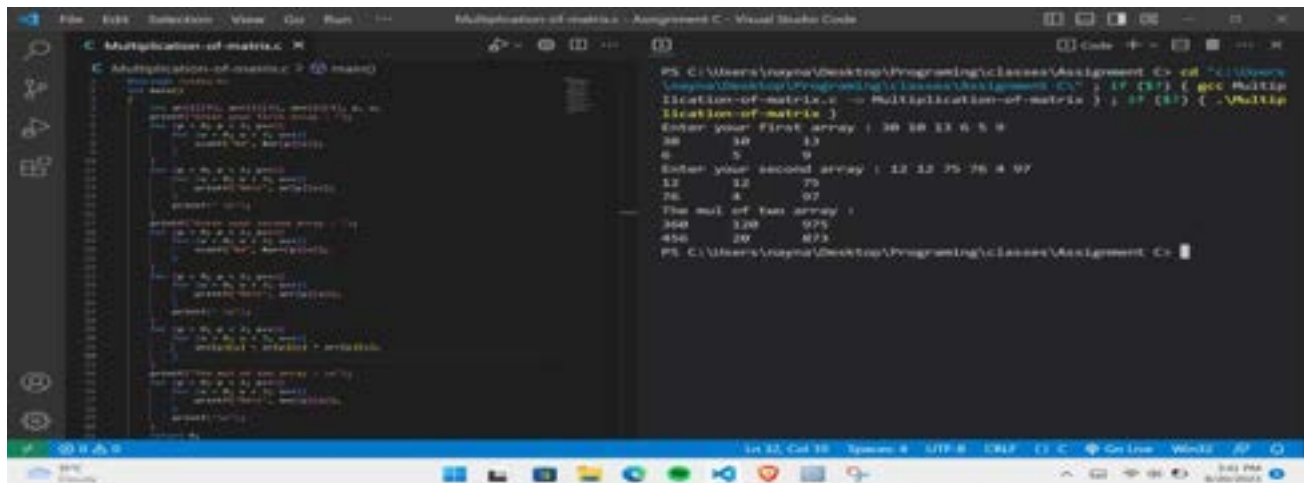
Subtraction of two matrix using 2-D Array



The screenshot shows a C++ program in Visual Studio Code titled "Subtraction-of-matrix.c". The code defines two 3x3 matrices, A and B, and a result matrix C. It uses nested loops to read values for matrix A, then matrix B, and finally to calculate the difference of corresponding elements in A and B, storing the result in C. The program then prints the resulting matrix C. The terminal output shows the execution of the program with the following input and output:

```
PS C:\Users\Vaanya\Desktop\Programming\Classes\Assignment_C> cd "C:\Users\Vaanya\Desktop\Programming\Classes\Assignment_C" & gcc Subtraction-of-matrix.c -o Subtraction-of-matrix.exe & .\Subtraction-of-matrix.exe
Enter your first array : 38 75 76 97 13 12 38 75 76
38 75 76
97 13 12
Enter your second array : 5 6 9 10 4 12 5 6 9
5 6 9
10 4 12
The sub of two array :
33 69 67
87 9 0
```

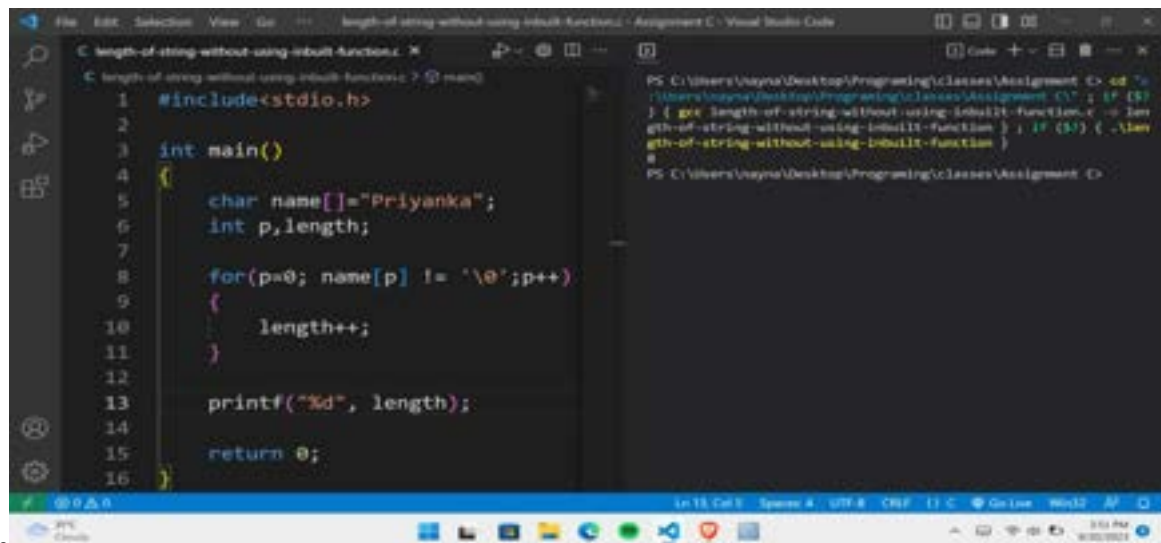
Multiplication of two matrix using 2-D Array



The screenshot shows a Visual Studio Code editor with a C program titled "Multiplication of matrix.c". The program uses nested loops to calculate the product of two 3x3 matrices. The terminal output shows the execution of the program, where the user enters the elements of two matrices, and the program prints the resulting 3x3 product matrix.

```
PS C:\Users\Vaanya\Desktop\Programming\classes\Assignment C> cd "C:\Users\Vaanya\Desktop\Programming\classes\Assignment C" & if ($?) { gcc Multiplication-of-matrix.c -o Multiplication-of-matrix } & if ($?) { .\Multiplication-of-matrix }
Enter your first array : 30 30 33 6 5 9
30 30 33
6 5 9
Enter your second array : 12 12 75 76 4 97
12 12 75
76 4 97
The mul of two array :
360 320 575
456 20 873
PS C:\Users\Vaanya\Desktop\Programming\classes\Assignment C>
```

7> Write a program to Find out length of string without using inbuilt function.



The screenshot shows a Visual Studio Code editor with a C program titled "length-of-string-without-using-inbuilt-function.c". The program defines a main function that takes a string "Priyanka" and iterates through it to count the number of characters, excluding the null terminator. The result is printed using printf.

```
PS C:\Users\Vaanya\Desktop\Programming\classes\Assignment C> cd "C:\Users\Vaanya\Desktop\Programming\classes\Assignment C" & if ($?) { gcc length-of-string-without-using-inbuilt-function.c -o length-of-string-without-using-inbuilt-function } & if ($?) { .\length-of-string-without-using-inbuilt-function }
5
PS C:\Users\Vaanya\Desktop\Programming\classes\Assignment C>
```

Ans.

8> Write a program to reverse a string and check if the string is palindrome or not.

```

1 #include <string.h>
2 #include <stdio.h>
3
4 int main()
5 {
6     char name[] = "Naman";
7     printf("%s\n", strcmp(name, "Naman"));
8
9     char name[] = { "Naman" };
10    int i = 0;
11    int n = strlen(name) - 1;
12
13    while (i < n) {
14        if (name[i++] != name[n--]) {
15            printf("No is not a palindrome\n", name);
16            return 0;
17        }
18    }
19    printf("No is a palindrome\n", name);
20    return 0;
21 }

```

The screenshot shows a C++ IDE with a file named 'string_palindrome.c'. The code defines a function `isPalindrome` that takes a string `str` and returns `1` if it is a palindrome and `0` otherwise. The `main` function tests this with the string "Naman". The output in the console shows that "Naman" is not a palindrome, which is correct.

Ans.

9> Write a program of employee structure that provides the following information - Print and Display Employee-number, Employee-name, Address and Age.

The screenshot shows a Visual Studio Code editor with two panes. The left pane displays the source code for a C program named 'structure-employee.c'. The code includes a structure declaration for 'employee' with fields 'emp_name', 'emp_address', 'emp_age', and 'emp_number'. It also includes a 'main' function that declares a variable 's1' of type 'employee', initializes it with values for Priyanka, and prints the details.

```

1  #include <stdio.h>
2
3  /*structure declaration*/
4  struct employee
5  {
6      char emp_name[30], emp_address[30];
7      int emp_age, emp_number;
8  };
9
10 int main()
11 {
12     struct employee s1 = {"Priyanka", "Raj",
13                           29, 134567890};
14     printf("Employee name : %s \nEmployee
15 address : %s \nEmployee age : %d
16 \nEmployee number : %d", s1.emp_name, s1.
17 emp_address, s1.emp_age, s1.emp_number);
18
19     return 0;
20 }

```

The right pane shows the terminal output of the program, which prints the employee details for Priyanka.

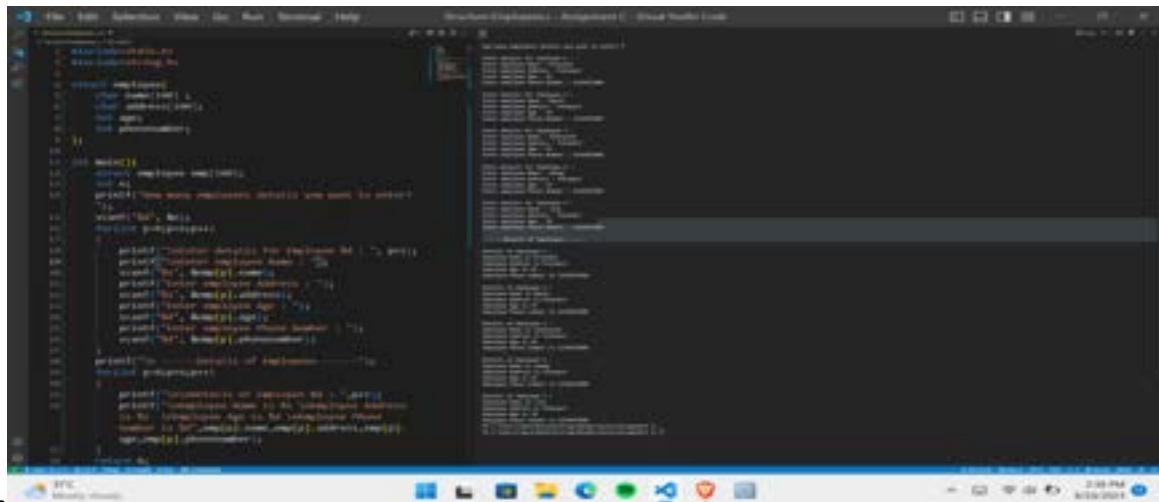
```

PS C:\Users\rayna\Desktop\Programming\classes\Assignment C> cd C:\Users\rayna\Desktop\Programming\classes\Assignment C> gcc structure-employee.c -o structure-employee
Employee name : Priyanka
Employee address : Raj
Employee age : 29
Employee number : 134567890
PS C:\Users\rayna\Desktop\Programming\classes\Assignment C>

```

Ans.

10> Write a program of structure for five employees that provides the following information -print and display Employee number, Employee name, Address and Age.



```
#include <iostream>
using namespace std;

struct Employee {
    int empno;
    string empname;
    string address;
    int age;
};

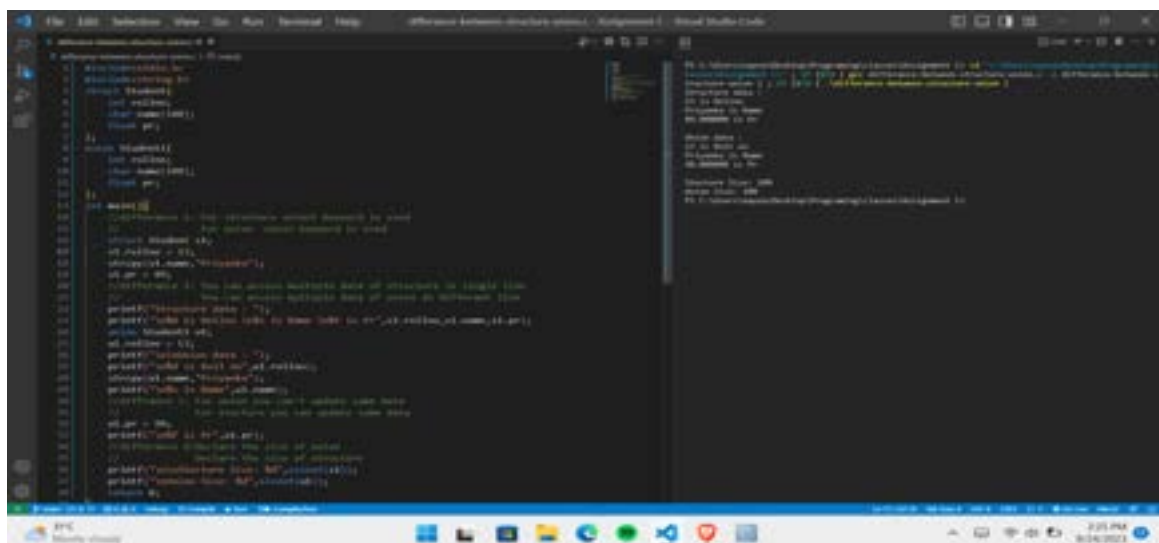
int main() {
    Employee emp[5];
    for (int i = 0; i < 5; i++) {
        cout << "Enter Employee " << i << " details: ";
        emp[i].empno = 1000 + i;
        emp[i].empname = "Employee " << i << " ";
        emp[i].address = "Address of Employee " << i << " ";
        emp[i].age = 20 + i;
    }

    cout << "\n\nDetails of Employees:-\n\n";
    for (int i = 0; i < 5; i++) {
        cout << "Employee no. is: " << emp[i].empno << "\n";
        cout << "Employee name is: " << emp[i].empname << "\n";
        cout << "Employee Age is: " << emp[i].age << "\n";
        cout << "Employee Address is: " << emp[i].address << "\n";
    }

    return 0;
}
```

Ans.

11> Write a program to show the difference between Structure and Union.



```
#include <iostream>
using namespace std;

struct Student {
    int rollno;
    string name;
    int age;
};

union Student {
    int rollno;
    string name;
    int age;
};

int main() {
    cout << "Difference 1:- The structure Student takes 12 bytes of memory.\n";
    cout << "The union Student takes 4 bytes of memory.\n";

    Student s;
    cout << "Enter rollno, name, age: ";
    s.rollno = 101;
    s.name = "Rishi";
    s.age = 20;

    cout << "\n\nDetails of Student:-\n\n";
    cout << "rollno: " << s.rollno << "\n";
    cout << "name: " << s.name << "\n";
    cout << "age: " << s.age << "\n";

    return 0;
}
```

Ans.