LAB TASK 8

Name : Mubasher Shahzad
Roll No: 22P-9002
Subject : Programming Fundamentals

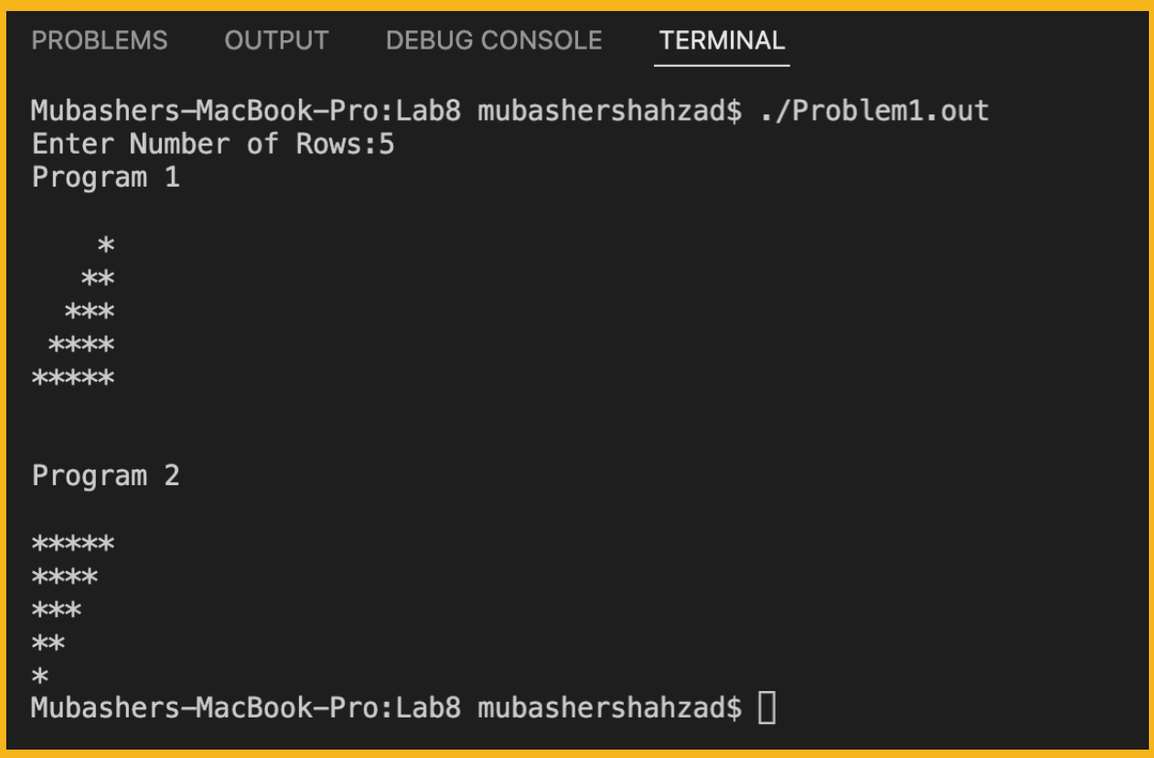
Problem 1:

Write a C program that prints the following patterns separately one below the other.

Code:

```
// Write a C program that prints the following patterns separately one below the other.
#include <stdio.h>
#include <math.h>
int main()
{
    int i, j, r;

    printf("Enter Number of Rows:");
    scanf("%d", &r);

    printf("Program 1\n\n");
    for (i = 1; i <= r; i++)
    {

        for (j = i; j < r; j++)
        {
            printf(" ");
        }
        for (j = 1; j <= i; j++)
        {
            printf("*");
        }

        printf("\n");
    }

    printf("\n\nProgram 2\n\n");
    for (i = 1; i <= r; i++)
    {

        for (j = i; j <= r; j++)
        {
            printf("*");
        }

        printf("\n");
    }

    return 0;
}
```

Output:



Problem 2:

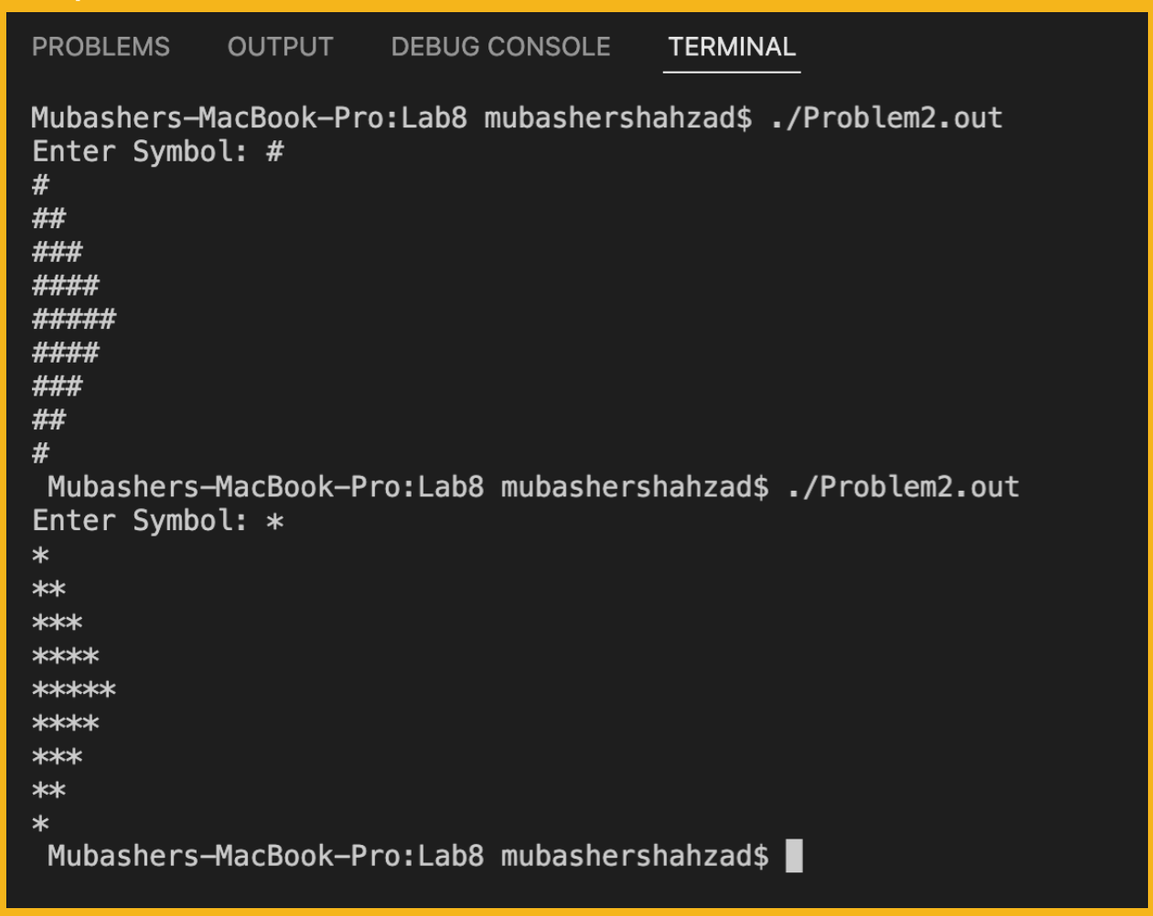
Write a C function to construct the following pattern, using a for loop. User should enter a symbol e.g *,^,@,- etc, and your function should have a pattern like that of that symbol.

Code:

```
#include <stdio.h>
#include <math.h>
char Diamond(char sym);
int main()
{
    char sym;
    printf("Enter Symbol: ");
    scanf(" %c", &sym);
    char c = Diamond(sym);
    printf(" %c", c);
    return 0;
}
char Diamond(char sym)
{
    int col = 1;
    for (int i = 1; i < 10; i++)
    {

        for (int j = 1; j <= col; j++)
        {
            printf("%c", sym);
        }
        if (i < 5)
        {
            col++;
```

```
        }
        else
        {
            col--;
        }

        printf("\n");
    }
}
```

Output:



Problem 3:

Write a Program that takes ‘n’ and ‘x’ from the user and computes the following series.

Code:

```
#include <stdio.h>

void main()
{
    float x, sum, no_row;
    int i, n;
    printf("Input the value of x :");
    scanf("%f", &x);
    printf("Input number of terms : ");
```

```
    scanf("%d", &n);
    sum = 1;
    no_row = 1;
    for (i = 1; i < n; i++)
    {
        no_row = no_row * x / (float)i;
        sum = sum + no_row;
    }
    printf("\nThe sum  is : %f\n", sum);
}
```

Output:

```
PROBLEMS     OUTPUT     DEBUG CONSOLE     TERMINAL

Mubashers-MacBook-Pro:Lab8 mubashershahzad$ ./Problem3.out
Input the value of x :5
Input number of terms : 3

The sum  is : 18.500000
Mubashers-MacBook-Pro:Lab8 mubashershahzad$
```

Problem 4:

A “Perfect” number is a positive whole number that is the sum of its proper divisors (including 1 and
excluding the number itself). For example, the proper divisors of 6 are 1, 2, 3 and 1 + 2 + 3 =
6. So, 6 is a perfect number. Similarly, 28 is also a perfect number.
Write a program that displays the first 4 perfect numbers.

Code:

```
#include <stdio.h>
void main()
{
    int n = 1, rem, s, c = 0;
    for (int k = 1; k > 0; k++)
    {
        s = 0;
        for (int i = 1; i <= n / 2; i++)
        {
            rem = n % i;
            if (rem == 0)
            {
                s = s + i;
            }
        }
        if (s == n)
        {
            c++;
            printf(" %d is a Perfect Divisible or Perfect Number\n", n);
            if (c == 4)
```

```
        {
            break;
        }
    }
    n++;
  }
}
```

Output: