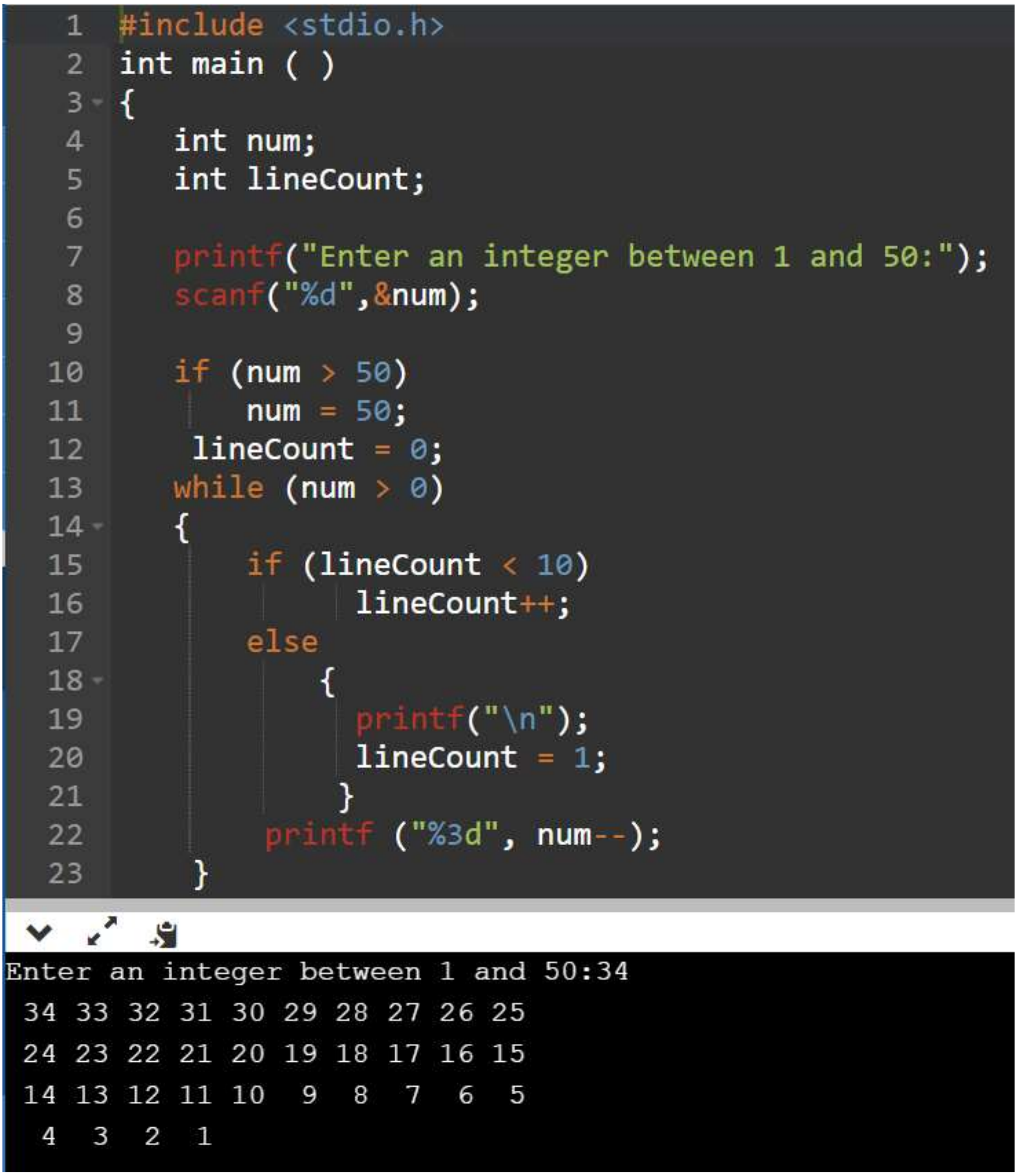
Birla Institute of Technology & Science, Pilani, Hyderabad Campus

First Semester 2020-2021

Computer Programming (CS F111) Lab 7

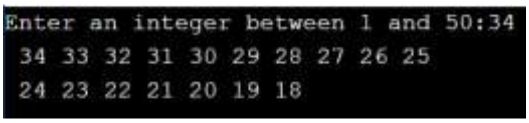
Prog 1: Write a program using while loop that prints 10 numbers per line. Refer to the code on gdbonline and its' output below:



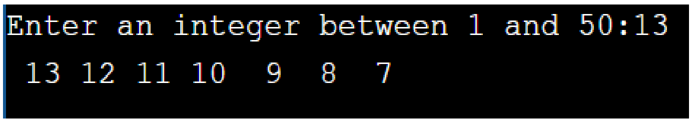
```
#include <stdio.h>
int main ( )
{
    int num;
    int lineCount;

    printf("Enter an integer between 1 and 50:");
    scanf("%d",&num);

    if (num > 50)
        num = 50;
     lineCount = 0;
    while (num > 0)
    {
        if (lineCount < 10)
              lineCount++;
        else
            {
              printf("\n");
              lineCount = 1;
            }
         printf ("%3d", num--);
     }
```

```
Enter an integer between 1 and 50:34
 34 33 32 31 30 29 28 27 26 25
 24 23 22 21 20 19 18 17 16 15
 14 13 12 11 10  9  8  7  6  5
  4  3  2  1
```

Task 1: Print only the output up to **one more than the half** of the number given as input. Your modified code should give the below output: (Sample output):

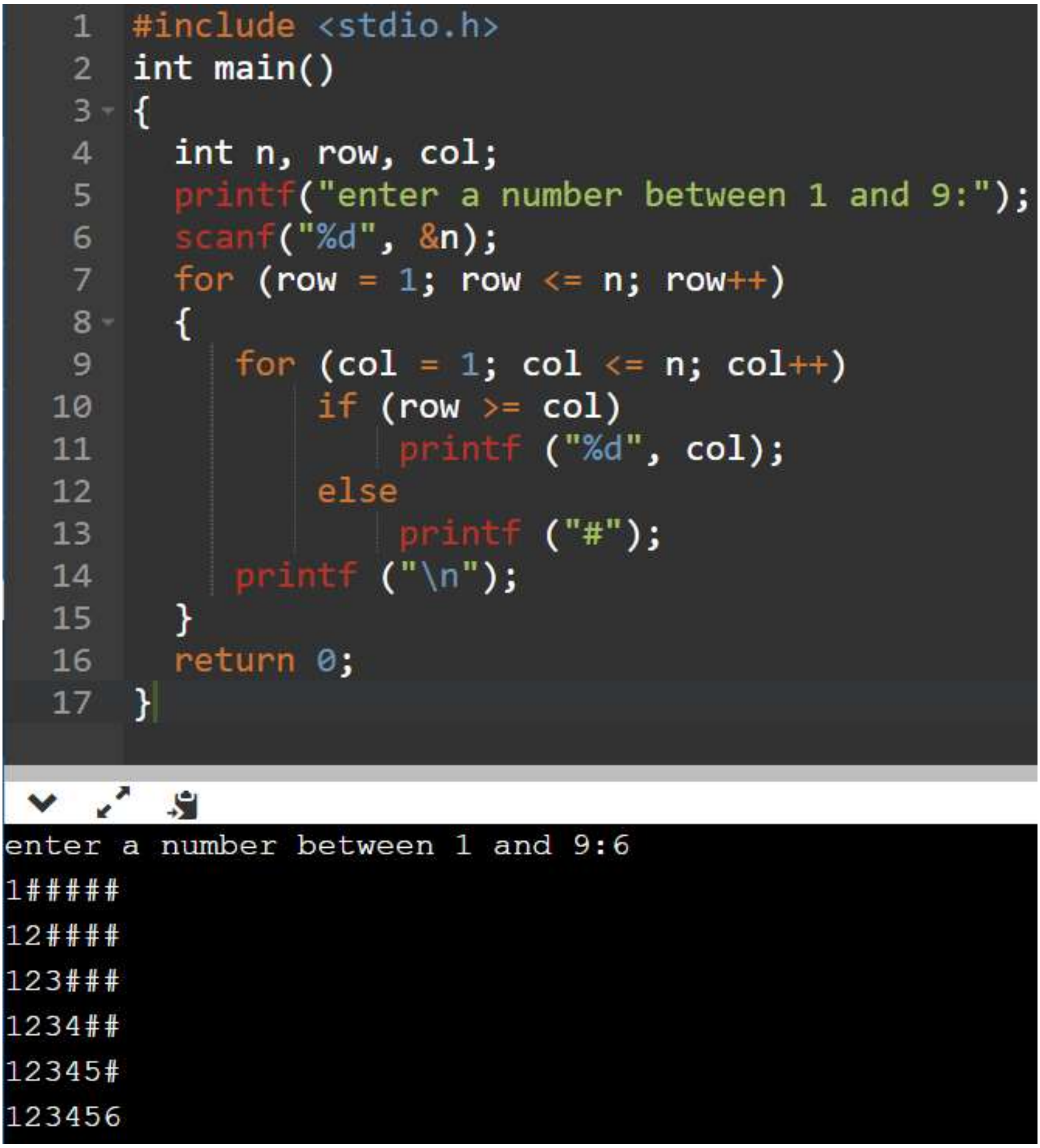


```
Enter an integer between 1 and 50:34
 34 33 32 31 30 29 28 27 26 25
 24 23 22 21 20 19 18
```



```
Enter an integer between 1 and 50:13
 13 12 11 10  9  8  7
```

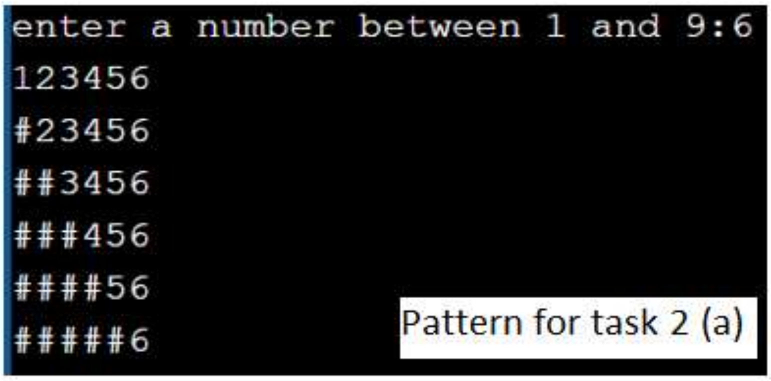
Prog 2: Write a program using for loop that generates a pattern of numbers and #s as shown in the output.



```c
#include <stdio.h>
int main()
{
   int n, row, col;
   printf("enter a number between 1 and 9:");
   scanf("%d", &n);
   for (row = 1; row <= n; row++)
   {
      for (col = 1; col <= n; col++)
          if (row >= col)
              printf ("%d", col);
          else
              printf ("#");
      printf ("\n");
   }
   return 0;
}
```

```
enter a number between 1 and 9:6
1#####
12####
123###
1234##
12345#
123456
```

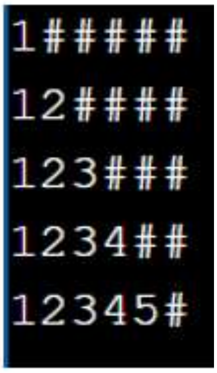
Task 2: a) Modify the above program to generate the pattern shown in first figure below. b) Modify the above code to generate the pattern shown in second figure.



```
enter a number between 1 and 9:6
123456
#23456
##3456
###456
####56
#####6
```

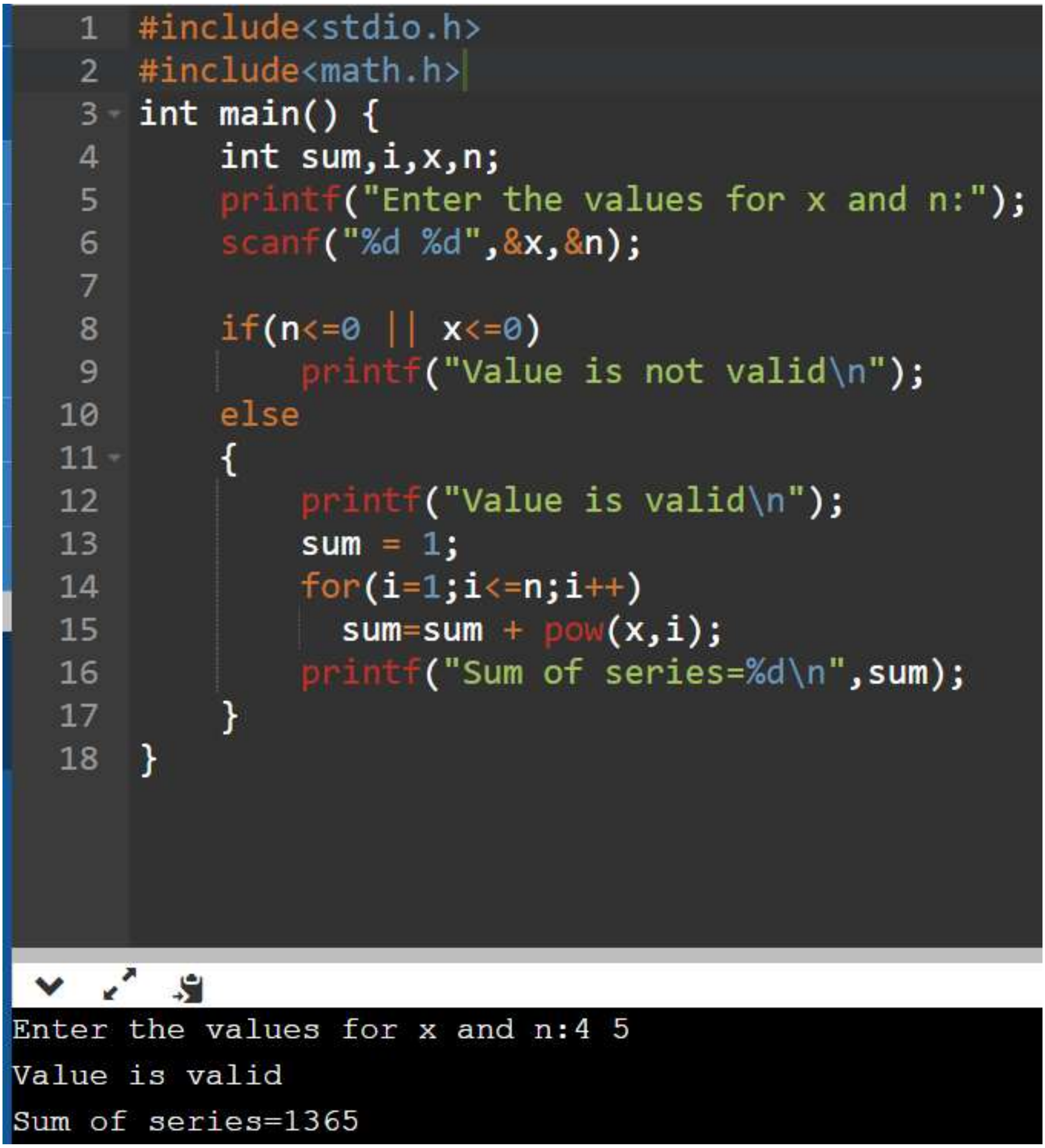
Pattern for task 2 (a)

Pattern for task 2 (b) ➪



```
1#####
12####
123###
1234##
12345#
```

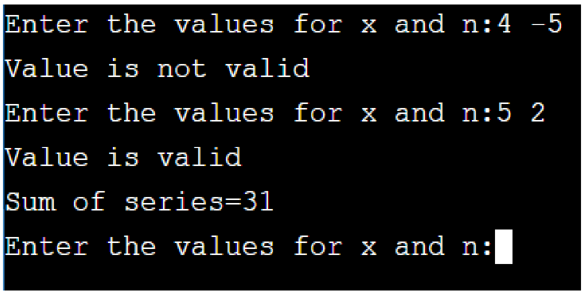
Prog 3: Write a C program to read in two numbers, x and n, and then compute the sum of this geometric progression: $1+x+x^2+x^3+\ldots+x^n$. Below run is for x = 4, n = 5.



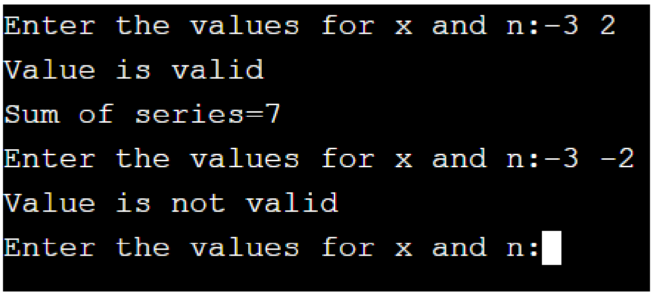
```c
#include<stdio.h>
#include<math.h>
int main() {
	int sum,i,x,n;
	printf("Enter the values for x and n:");
	scanf("%d %d",&x,&n);

	if(n<=0 || x<=0)
		printf("Value is not valid\n");
	else
	{
		printf("Value is valid\n");
		sum = 1;
		for(i=1;i<=n;i++)
		  sum=sum + pow(x,i);
		printf("Sum of series=%d\n",sum);
	}
}
```

```
Enter the values for x and n:4 5
Value is valid
Sum of series=1365
```

Task 3: Modify the above program such that for negative exponents your program prints an error message "Value is not valid" and then goes back and reads the next pair of numbers (x and n) without computing the sum. Below are some sample outputs.

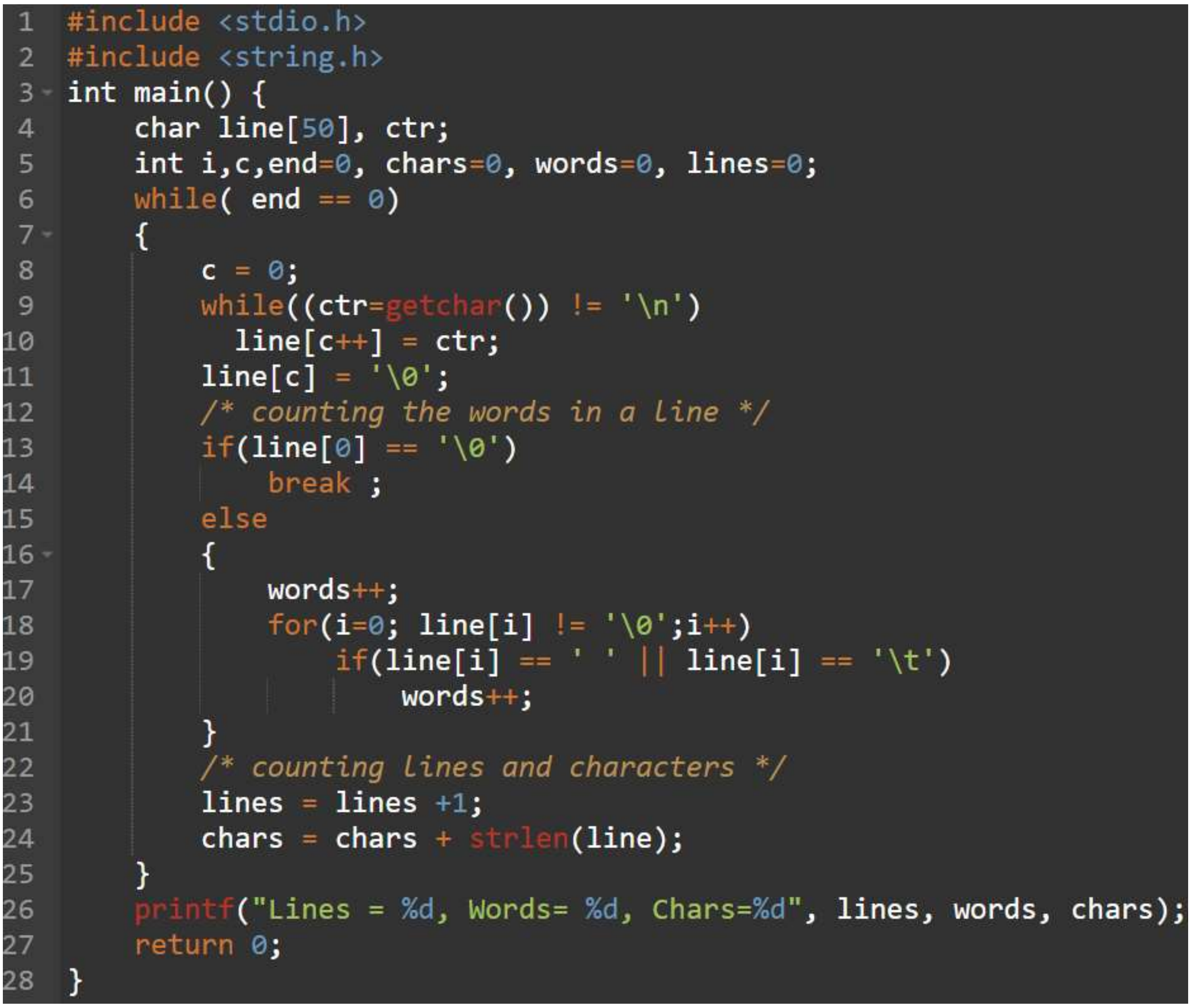


```
Enter the values for x and n:4 -5
Value is not valid
Enter the values for x and n:5 2
Value is valid
Sum of series=31
Enter the values for x and n:
```



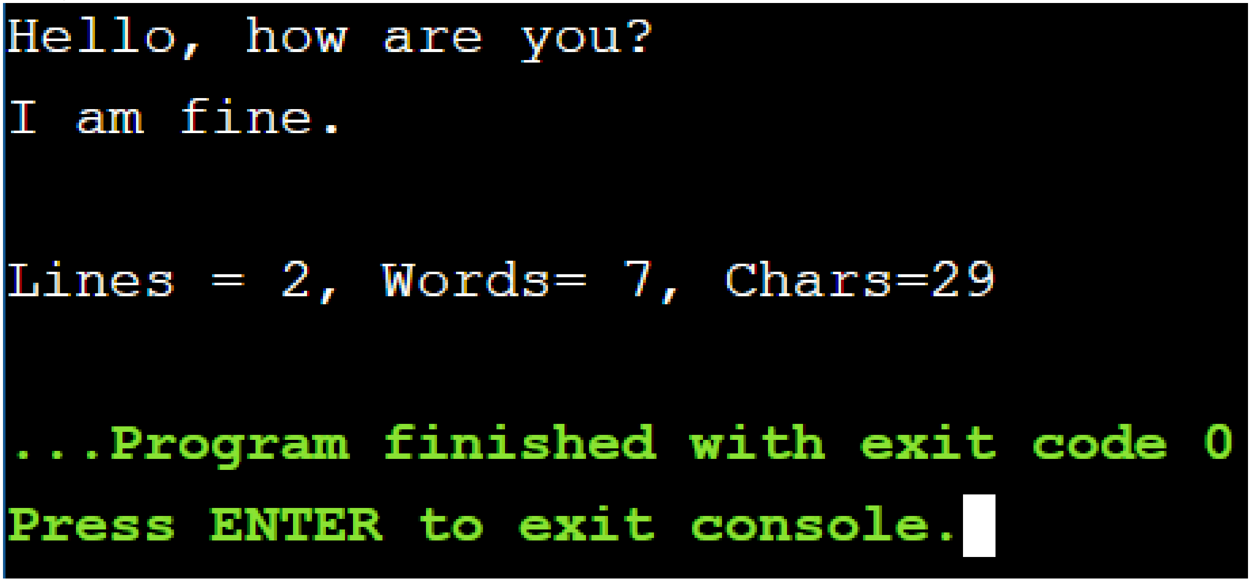
```
Enter the values for x and n:-3 2
Value is valid
Sum of series=7
Enter the values for x and n:-3 -2
Value is not valid
Enter the values for x and n:
```

Prog 4: Write a C program to count the lines, words and characters in a given text.



```c
#include <stdio.h>
#include <string.h>
int main() {
    char line[50], ctr;
    int i,c,end=0, chars=0, words=0, lines=0;
    while( end == 0)
    {
        c = 0;
        while((ctr=getchar()) != '\n')
          line[c++] = ctr;
        line[c] = '\0';
        /* counting the words in a line */
        if(line[0] == '\0')
            break ;
        else
        {
            words++;
            for(i=0; line[i] != '\0';i++)
                if(line[i] == ' ' || line[i] == '\t')
                    words++;
        }
        /* counting lines and characters */
        lines = lines +1;
        chars = chars + strlen(line);
    }
    printf("Lines = %d, Words= %d, Chars=%d", lines, words, chars);
    return 0;
}
```

Output:

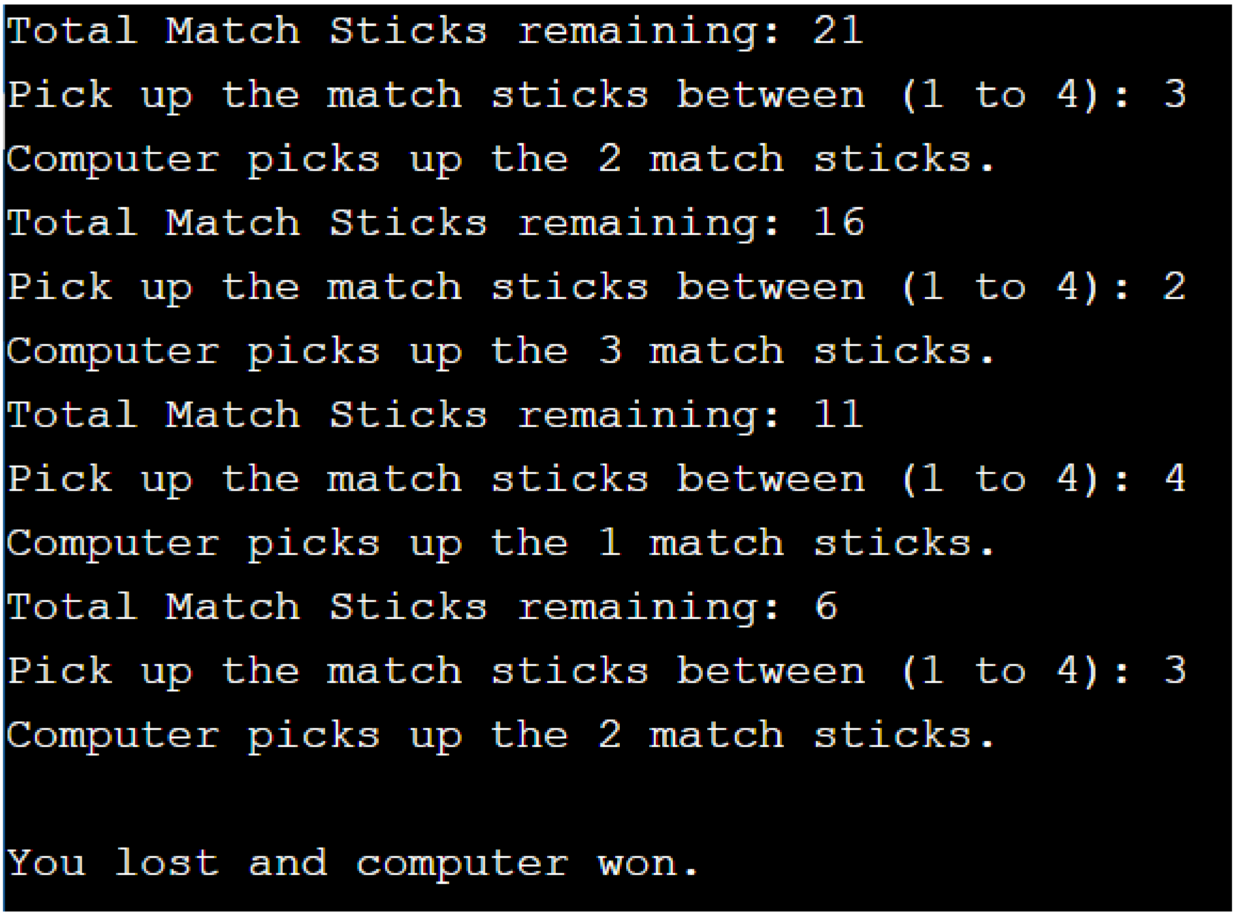


```
Hello, how are you?
I am fine.


Lines = 2, Words= 7, Chars=29


...Program finished with exit code 0
Press ENTER to exit console.
```

Task 4: Modify the above program so that you can additionally print how many number of special characters are there in the input. For ex, in the above example run, it is 3 (, ? .).

Prog 5: Write a program for a matchstick game being played between the computer and a user. Your program should ensure that the computer always wins alaways. Rules for the game are as follows: - There are 21 matchsticks, - The computer asks the user to pick 1, 2, 3, or 4 matchsticks, - After user picks, the customer does its picking. And this pattern repeats… - Whoever is forced to pick up the last matchstick loses the game. Output is as below:

```
#include<stdio.h>
int main() {
   int match_sticks = 21, user_choice, computer_choice;
   while(match_sticks>=1)
   {
      printf("Total Match Sticks remaining: %d\n", match_sticks);
      printf("Pick up the match sticks between (1 to 4): "); scanf("%d", &user_choice);
      if(user_choice > 4) {
         printf("Invalid Entry: Game ends...");  break;
      }
      computer_choice = 5 - user_choice;
      printf("Computer picks up the %d match sticks.\n", computer_choice);
      match_sticks = match_sticks-user_choice-computer_choice;
      if(match_sticks==1) {
         printf("\nYou lost and computer won.");   break;
      }
   }
   return(0);
}
```



```
Total Match Sticks remaining: 21
Pick up the match sticks between (1 to 4): 3
Computer picks up the 2 match sticks.
Total Match Sticks remaining: 16
Pick up the match sticks between (1 to 4): 2
Computer picks up the 3 match sticks.
Total Match Sticks remaining: 11
Pick up the match sticks between (1 to 4): 4
Computer picks up the 1 match sticks.
Total Match Sticks remaining: 6
Pick up the match sticks between (1 to 4): 3
Computer picks up the 2 match sticks.

You lost and computer won.
```

Submission instructions: Submit as usual all the programs and their tasks.
