**1). Write a C program to print all integers between starting and ending range divisible by 11.**

**CODE 1 :**

/\*Write a C program to print all integers between starting and ending range divisible by 11. \*/

#include <stdio.h>

int main()

{

int a, b;

printf("Enter the starting range : ");

scanf("%d", &a);

printf("Enter the ending range : ");

scanf("%d", &b);

printf("All integers between starting and ending range divisible by 11 is : \n");

for (int i = a; i < b; i++)

{

if (i % 11 == 0)

{

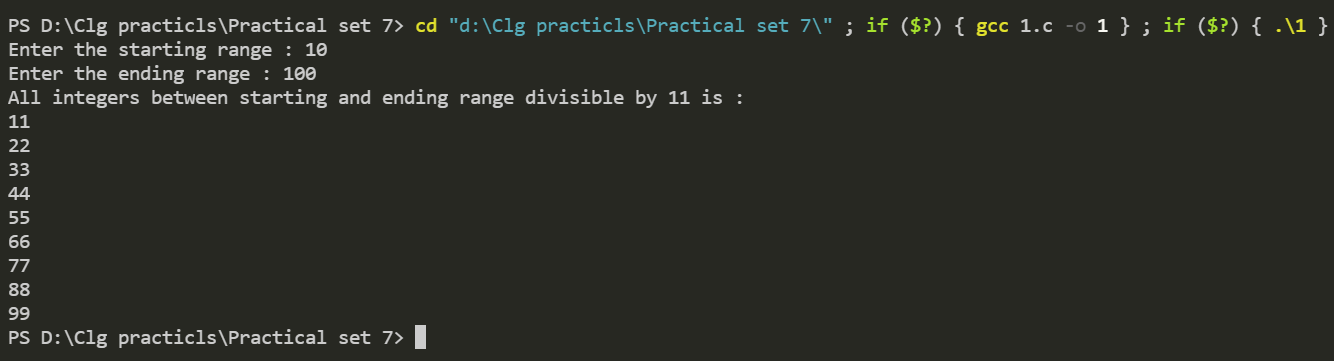
printf("%d\n", i);

}

}

return 0;

}

**OUTPUT 1 :**

**2). Write a C program to print a number and its square , cube for numbers.**

**CODE 2 :**

/\*Write a C program to print a number and its square , cube for numbers.\*/

#include <stdio.h>

int main()

{

int a;

printf("Enter the number : ");

scanf("%d", &a);

printf("The number is : %d\n", a);

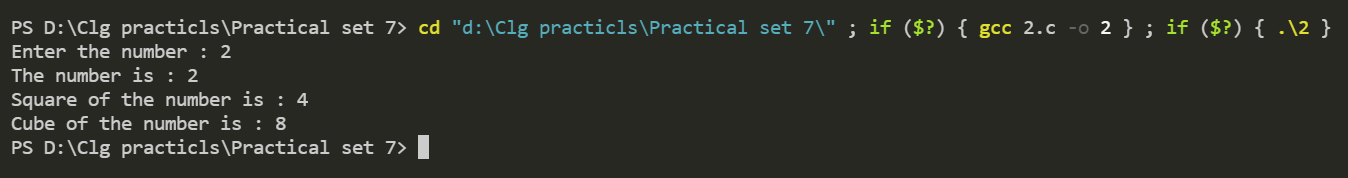
printf("Square of the number is : %d\n", a \* a);

printf("Cube of the number is : %d\n", a \* a \* a);

return 0;

}

**OUTPUT 2 :**

****

**3). Write a program to generate first n number of Fibonacci series.**

**CODE 3 :**

/\*Write a program to generate first n number of Fibonacci series\*/

#include <stdio.h>

int fibonacci(int n)

{

if (n == 1 || n == 2)

{

return n - 1;

}

else

{

return fibonacci(n - 1) + fibonacci(n - 2);

}

}

int main()

{

int n;

printf("Enter the number of terms of fibonacci series:");

scanf("%d", &n);

for (int i = 1; i <= n; i++)

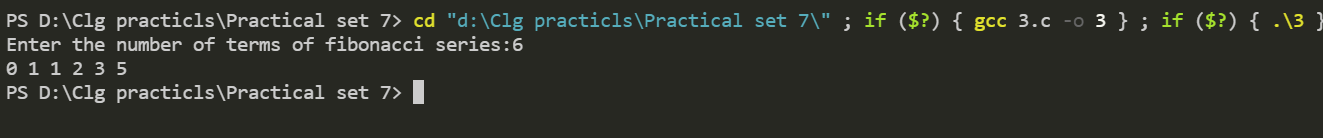
{

printf("%d ", fibonacci(i));

}

return 0;

}

**OUTPUT 3 :**

**4). Write a program to find out sum of first and last digit of a given number.**

**CODE 4 :**

/\*Write a program to find out sum of first and last digit of a given number. \*/

#include <stdio.h>

#include<conio.h>

int main()

{

int a, first, last, n;

printf("Enter the number : ");

scanf("%d", &a);

last = a % 10;

while (a >= 10)

{

a = a / 10;

}

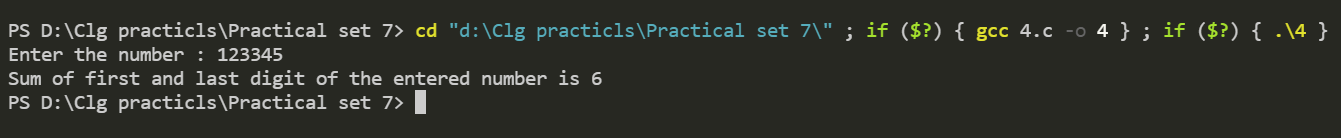
first = a;

printf("Sum of first and last digit of the entered number is %d", first + last);

return 0;

}

**OUTPUT 4 :**



**5). Write a C program to find the sum and average of different numbers which are accepted by user as many as user wants.**

**CODE 5 :**

/\*Write a C program to find the sum and average of different numbers which are accepted by the

user as many as the user wants.\*/

#include <stdio.h>

int main()

{

int sum = 0, N, i = 1;

printf("Enter how many numbers you want to take for calculation of sum and average: ");

scanf("%d", &N);

int Number[N];

for (i = 1; i <= N; i++)

{

printf("Enter the value of Number %d: ", i);

scanf("%d", &Number[N]);

sum += Number[N];

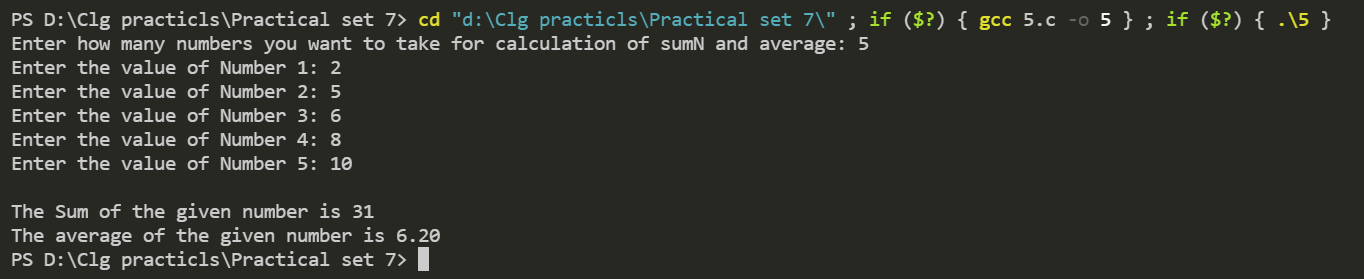
}

printf("\nThe Sum of the given number is %d", sum);

printf("\nThe average of the given number is %0.2f", (float)sum / N);

return 0;

}

**OUTPUT 5 :**

**6). Write a program to calculate average and total of 5 students for 3 subjects (use nested for loops).**

**CODE** 6 **:**

/\* 6.Write a program to calculate average and total of 5 students for 3

subjects (use nested for loops).\*/

#include <stdio.h>

int main()

{

int marks = 0;

for (int student = 1; student <= 5; student++)

{

printf("\n\*\*\*For student %d\*\*\*\n", student);

int sum = 0;

for (int subject = 1; subject <= 3; subject++)

{

printf("Enter the marks of subject %d : ", subject);

scanf("%d", &marks);

sum = sum + marks;

}

printf("\nThe total marks of student %d is %d", student, sum);

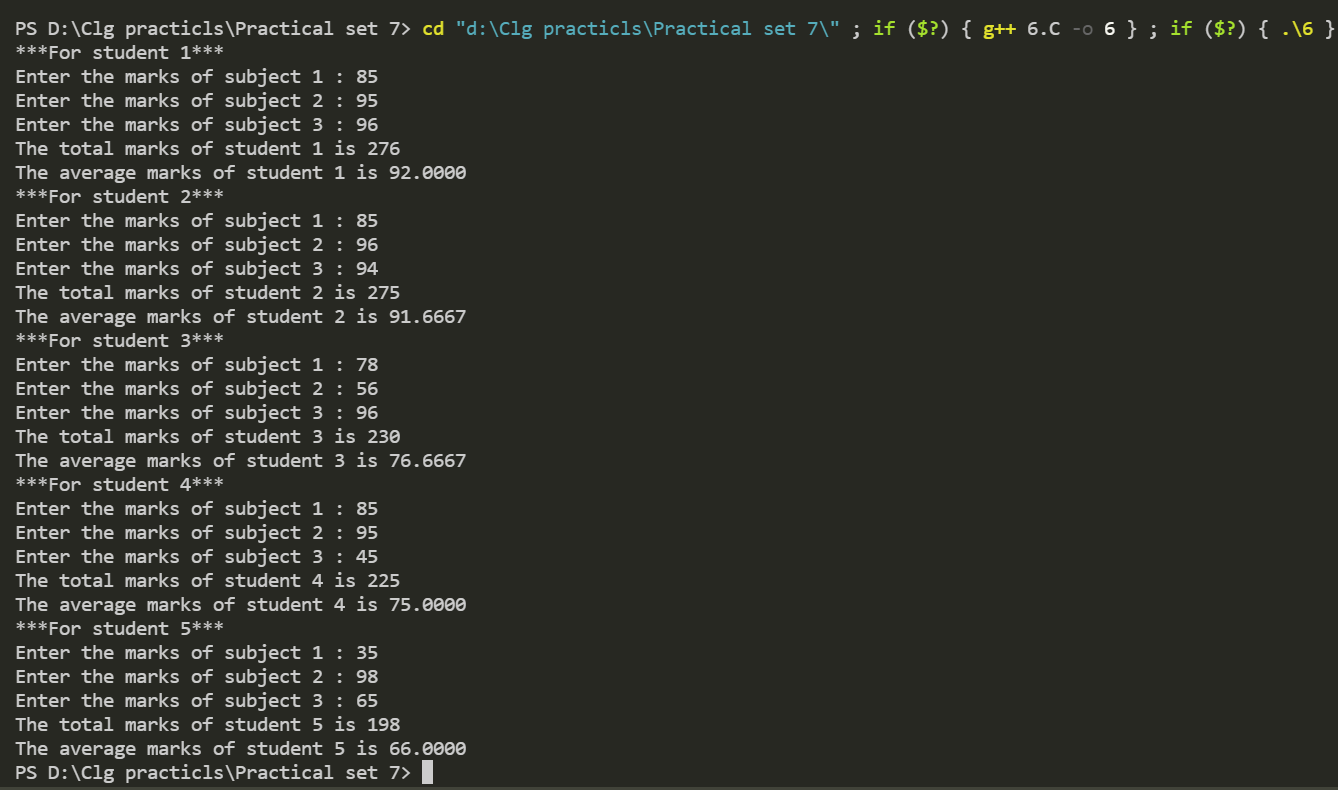
printf("\nThe average marks of student %d is %0.4f\n", student,

((float)sum) / 3);

}

return 0;

}

**OUTPUT 6 :**

**7). Write a C program to find 1+1/2!+1/3!+1/4!+...+1/n!.**

**CODE** 7 **:**

// 7.Write a C program to find 1+(1/2!)+(1/3!)+(1/4!)+.....+(1/n!).

#include <stdio.h>

int main()

{

int n, fact;

float sum = 0;

printf("\nEnter the value of n: ");

scanf("%d", &n);

for (int i = 1; i <= n; i++)

{

fact = 1;

for (int j = i; j > 0; j--)

{

fact = fact \* j;

}

sum = sum + (1.0 / fact);

}

if (n == 1 || n == 0)

printf("\n(1/1!) OR (1/0!) = 1.00");

else if (n == 2)

printf("\n(1/1!)+(1/2!) = %f", sum);

else if (n == 3)

printf("\n(1/1!)+(1/2!)+(1/3!) = %f", sum);

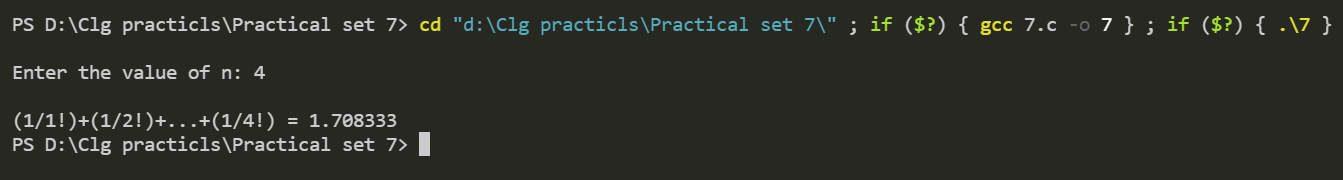
else

printf("\n(1/1!)+(1/2!)+...+(1/%d!) = %f", n, sum);

return 0;

}

**OUTPUT 7 :**



**8) Write a program to evaluate the series sum=1-x+x^2/2!- x^3/3!+x^4/4!......-x^9/9!**

**CODE 8 :**

/\*Write a program to evaluate the series sum=1-x+x^2/2!-x^3/3!+x^4/4!......-x^9/9\*/

#include <stdio.h>

#include <math.h>

int main()

{

float x, sum = 1, fact = 1;

printf("Enter the value of x: ");

scanf("%f", &x);

for (int i = 1; i <= 9; i++)

{

fact = 1;

for (int j = i; j >= 1; j--)

{

fact = fact \* j;

}

sum += pow(-x, i) / fact;

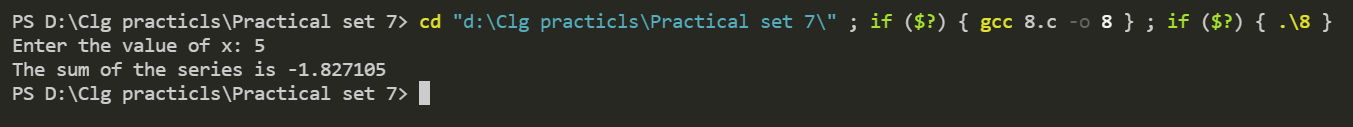
}

printf("The sum of the series is %f", sum);

return 0;

}

**OUTPUT 8 :**

****

**9) Write a program to display even numbers up to n . use break and continue.**

**CODE** 9 **:**

// 9.Write a program to display even numbers up to n.(use break and continue)

#include <stdio.h>

int main()

{

int n, i = 0;

printf("Enter the value of n for the display even numbers up to n: ");

scanf("%d", &n);

if (n >= 0)

{

printf("\nThe even numbers: \n");

}

while (i != n + 1)

{

if (n <= 0)

{

printf("Please Enter valid number.");

break;

}

if (i % 2 == 0)

{

printf("%d ", i);

i++;

continue;

}

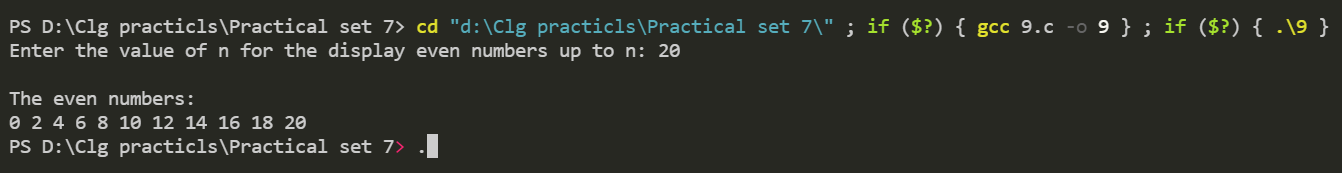
i++;

}

return 0;

}

**OUTPUT** 9 **:**

****

**10) Write a program to print all Armstrong numbers in a given range.**

**CODE 10 :**

// 10.Write a program to print all Armstrong numbers in a given range.

#include <stdio.h>

void armstrong(int start, int end)

{

int temp, sum, r;

for (int i = start; i <= end; i++)

{

temp = i;

sum = 0;

while (temp != 0)

{

r = temp % 10;

temp = temp / 10;

sum = sum + (r \* r \* r);

}

if (sum == i)

{

printf("%d\n", i);

}

}

}

int main()

{

int start, end;

label:

printf("\nInput starting number of range: ");

scanf("%d", &start);

printf("Input ending number of range: ");

scanf("%d", &end);

if (start > end)

{

printf("\nPlease!!Enter valid range.");

goto label;

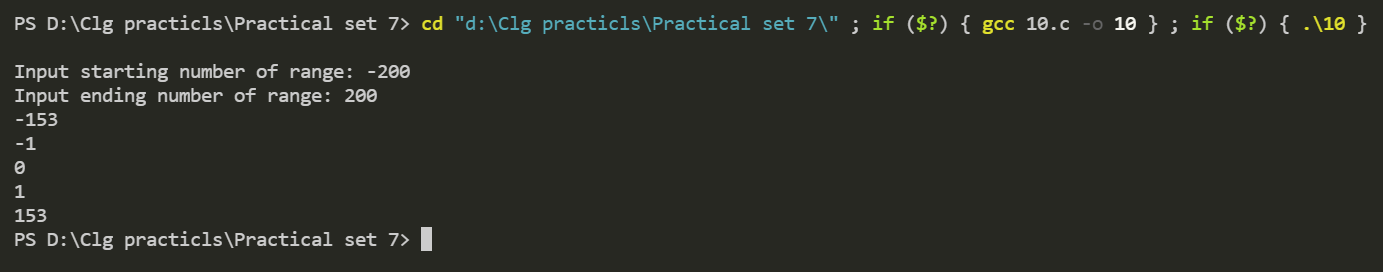
}

armstrong(start, end);

return 0;

}

**OUTPUT** 10 **:**

****