

## PROGRAM1

### FIBONACCI SERIES

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Fibonacci_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //Printing fibonacci series

            int firstNumber = 0, secondNumber = 1, currentNumber = 0, input;

            Console.Write("\nEnter number of values in the series to print : ");
            input = Convert.ToInt32(Console.ReadLine());

            if (input == 1)
                Console.Write("\nSeries : " + firstNumber);
            else
            {
                Console.Write("\nSeries : " + firstNumber + "\t" + secondNumber);

                for (int i = 1; i <= input - 2; i++)
                {
                    currentNumber = firstNumber + secondNumber;
                    Console.Write("\t" + currentNumber);
                    firstNumber = secondNumber;
                    secondNumber = currentNumber;
                }

                Console.ReadLine();
            }
        }
    }
}
```

#### OUTPUT

Enter number of values in the series to print : 8

Series : 0      1      1      2      3      5      8      13

## PROGRAM2

### ARMSTRONG NUMBER

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Armstrong_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //Checking for armstrong number

            int input, copy, remainder = 0, sum = 0;

            //takig input
            Console.Write("\nEnter the 3 digit number to check : ");
            input = Convert.ToInt32(Console.ReadLine());
            copy = input;

            //evaluating sum of digits of given number
            while(copy != 0)
            {
                remainder = copy % 10;
                sum = sum + (remainder * remainder * remainder);
                copy = copy / 10;
            }

            //checking if the sum of digits is same as the given input
            if (sum == input)
                Console.Write("\n{0} is an armstrong number", input);
            else
                Console.Write("\n{0} is not an armstrong number", input);

            Console.ReadLine();
        }
    }
}
```

#### OUTPUT

```
Enter the 3 digit number to check : 407

407 is an armstrong number
```

### PROGRAM3

#### ARMSTRONG NUMBER USING FUNCTION

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Armstrong2_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //checking for armstrong number using function

            int input;

            Console.Write("\nEnter the 3 digit number to be checked : ");
            input = Convert.ToInt32(Console.ReadLine());

            if (CheckArmstrong(input))
                Console.Write("\n{0} is an armstrong number", input);
            else
                Console.Write("\n{0} is not an armstrong number", input);

            Console.ReadLine();
        }

        static bool CheckArmstrong(int argument)
        {
            int remainder = 0, sum = 0, copy;

            copy = argument;
            while (copy != 0)
            {
                remainder = copy % 10;
                sum = sum + remainder * remainder * remainder;
                copy = copy / 10;
            }

            return (sum == argument) ? (true) : (false);
        }
    }
}
```

#### OUTPUT

```
Enter the 3 digit number to be checked : 371
```

```
371 is an armstrong number
```

## PROGRAM4

### ARMSTRING NUMBERS IN A RANGE

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Armstrong3_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //checking for armstrong numbers in a given range

            int startingNumber, endingNumber;

            //taking inputs
            Console.Write("\nEnter the starting 3 digit number : ");
            startingNumber = Convert.ToInt32(Console.ReadLine());
            Console.Write("\nEnter the ending 3 digit number : ");
            endingNumber = Convert.ToInt32(Console.ReadLine());

            //calling the method
            CheckArmstrong(startingNumber, endingNumber);

            Console.ReadLine();
        }

        static void CheckArmstrong(int startingNumber, int endingNumber)
        {
            //looping through the range
            for (int n = startingNumber; n < endingNumber; n++)
            {
                int copy, remainder = 0, sum = 0;

                //checking if the number is armstrong
                copy = n;
                while (copy != 0)
                {
                    remainder = copy % 10;
                    sum += remainder * remainder * remainder;
                    copy /= 10;
                }

                //printing armstrong numbers in the given range
                if (sum == n)
                    Console.Write("\n{0}", n);
            }
        }
    }
}
```

#### OUTPUT

Enter the starting 3 digit number : 100

Enter the ending 3 digit number : 999

153

370

371

407

## PROGRAM5

### SUM OF DIGITS OF A GIVEN NUMBER

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace SumOfDigits_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //finding sum of digits of a given number

            int input,sum = 0;

            //taking the input from user
            Console.Write("\nEnter the number : ");
            input = Convert.ToInt32(Console.ReadLine());

            //calling the method
            sum = FindSum(input);

            //printing the sum
            Console.Write("\nSum of digits of {0} is {1} ",input,sum);

            Console.ReadLine();
        }

        //method is calculating sum of digits of the given number
        static int FindSum(int input)
        {
            int remainder = 0, sum = 0, copy;

            copy = input;
            while(copy != 0)
            {
                remainder = copy % 10;
                sum += remainder;
                copy /= 10;
            }

            return sum;
        }
    }
}
```

#### OUTPUT

```
Enter the number : 7865

Sum of digits of 7865 is 26
```

## PROGRAM6

### REVERSE OF GIVEN NUMBER

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Reverse_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //printing reverse of given number
            int input, reverse;

            //taking the input
            Console.Write("\nEnter the number : ");
            input = Convert.ToInt32(Console.ReadLine());

            //calling the method
            GetReverse(input);

            Console.ReadLine();
        }

        //calculating reverse of give number
        static void GetReverse(int input)
        {
            int copy, remainder = 0;

            Console.Write("\nReverse of {0} is : ", input);
            copy = input;
            while(copy != 0)
            {
                remainder = copy % 10;
                Console.Write(remainder);
                copy /= 10;
            }
        }
    }
}
```

#### OUTPUT

Enter the number : 6785

Reverse of 6785 is : 5876

## PROGRAM7

### PALINDROME NUMEBR

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Palindrome_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //checking if given input is palindrome

            int input;

            //taking user input
            Console.Write("\nEnter the number : ");
            input = Convert.ToInt32(Console.ReadLine());

            //calling method and printing result
            if(CheckPalindrome(input))
                Console.Write("\n{0} is palindrome",input);
            else
                Console.Write("\n{0} is not palindrome",input);

            Console.ReadLine();
        }

        //method to check if given input is palindrome
        static bool CheckPalindrome(int input)
        {
            int copy, remainder = 1, reverse = 0;

            copy = input;
            while(copy > 0)
            {
                remainder = copy % 10;
                reverse = reverse * 10 + remainder;
                copy = copy / 10;
            }

            return (input == reverse) ? true : false;
        }
    }
}
```

#### OUTPUT

```
Enter the number : 6886

6886 is palindrome
```



## PROGRAM8

### SWAP NUMBERS USING THIRD VARIABLE

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace SwapNumbers_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //swapping numbers using 3rd variable

            int firstNumber, secondNumber, intermediate;

            Console.Write("\nEnter first number : ");
            firstNumber = Convert.ToInt32(Console.ReadLine());
            Console.Write("\nEnter second number : ");
            secondNumber = Convert.ToInt32(Console.ReadLine());

            Console.Write("\nValues before swapping : {0} {1}", firstNumber, secondNumber);

            intermediate = firstNumber;
            firstNumber = secondNumber;
            secondNumber = intermediate;

            Console.Write("\nValues after swapping : {0} {1}", firstNumber, secondNumber);

            Console.ReadLine();
        }
    }
}
```

#### OUTPUT

```
Enter first number : 67

Enter second number : 87

Values before swapping : 67 87
Values after swapping : 87 67
```

## PROGRAM9

### SWAP NUMBERS WITHOUT USING THIRD VARIABLE

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace SwapNumbers2_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //swapping 2 numbers without using third variable

            int firstNumber, secondNumber;

            //taking inputs
            Console.Write("\nEnter first number : ");
            firstNumber = Convert.ToInt32(Console.ReadLine());
            Console.Write("\nEnter second number : ");
            secondNumber = Convert.ToInt32(Console.ReadLine());

            Console.Write("\nValues before swapping : {0} {1}", firstNumber, secondNumber);

            //swapping numbers
            firstNumber = firstNumber + secondNumber;
            secondNumber = firstNumber - secondNumber;
            firstNumber = firstNumber - secondNumber;

            //printing outputs
            Console.Write("\nValues after swapping : {0} {1}", firstNumber, secondNumber);

            Console.ReadLine();
        }
    }
}
```

#### OUTPUT

```
Enter first number : 7
Enter second number : 56
Values before swapping : 7 56
Values after swapping : 56 7
```

## PROGRAM10

### PRINT STARS IN RIGHT ANGLED TRIANGLE PATTERN

#### CODE

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Pattern_new
{
    internal class Program
    {
        static void Main(string[] args)
        {
            //printing star pattern using for loop

            int rows;

            //taking input from user
            Console.Write("\nEnter number of rows to print : ");
            rows = Convert.ToInt32(Console.ReadLine());

            //printing pattern
            for(int i = 1; i <= rows; i++)
            {
                Console.Write("\n");
                for(int j = 1; j <= i; j++)
                    Console.Write("* ");
            }

            Console.ReadLine();
        }
    }
}
```

#### OUTPUT

Enter number of rows to print : 9

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