C# PROGRAMS

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

namespace _03oct_C_
{
    internal class Program
    {
        static void Main(string[] args)
        {

            //Lab 1: Multiplication table of a given number
            //Assignment:
            //Write a C# program to print the multiplication table of a given number using the for
loop.

            //{
            //    Console.WriteLine("Enter the number for the multiplication table:");
            //    int number = Convert.ToInt32(Console.ReadLine());
            //    for (int i = 1; i <= 10; i++)
            //    {
            //        Console.WriteLine($"{number} x {i} = {number * i}");
            //    }
            //}



            //Lab: 2 Display a right - angled triangle
            //Assignment:
            //Write a C# program to display a right-angled triangle pattern using nested for loops.

            //{
            //    Console.WriteLine("Enter the number of rows for the triangle:");
            //    int rows = Convert.ToInt32(Console.ReadLine());
            //    for (int i = 1; i <= rows; i++)
            //    {
            //        for (int j = 1; j <= i; j++)
            //        {
            //            Console.Write("*");
            //        }
            //       Console.WriteLine();
            // }
            //}
```

//Lab 3: Sum of all even numbers between 1 and 100

//Write a C# program to calculate the sum of all even numbers between 1 and 100 using a while loop.

```
// {
//          int sum = 0;
//    int number = 2;
//          while (number <= 100)
//          {
//              sum += number;
//              number += 2;
//          }
//Console.WriteLine("The sum of all even numbers between 1 and 100 is: " + sum);
//}
```

//Lab 4: Multiplication table from 1 to 5

//Assignment:

//Write a C# program to print a multiplication table from 1 to 5 using nested while loops.

```
//{
//    int i = 1;
//    while (i <= 5)
//    {
//        int j = 1;
//        while (j <= 10)
//        {
//            Console.Write($"{i} x {j} = {i * j}\n");
//            j++;
//        }
//        Console.WriteLine();
//        i++;
//}
//}
```

```
        //Lab 5: Print Positive Number
        //Assignment:
        //Write a C# program to keep asking the user to enter a positive number and print it. The
program
        //should stop when the user enters a negative number.

        //{
        //    while (true)
        //    {
        //        Console.WriteLine("Enter a positive number :");
        //        int number = Convert.ToInt32(Console.ReadLine());
        //        if (number < 0)
        //        {
        //            Console.WriteLine("Negative number entered. Stopping the program.");
        //            break;
        //        }

        //        Console.WriteLine($"You entered: {number}");
        //    }
        //}




        // Lab 6: Create Menu Driven Calculator
        //Assignment:
        //Write a C# program to create a basic menu-driven calculator using nested do-while
loops. The
        //calculator should continue to ask the user for two numbers and an operation(+, -, *, /)
until the
        //user chooses to exit.
        //{
        //    char choice;

        //    do
        //    {
        //        Console.Write("Enter the first number: ");
        //        double num1 = Convert.ToDouble(Console.ReadLine());
        //        Console.Write("Enter the second number: ");
        //        double num2 = Convert.ToDouble(Console.ReadLine());
        //        Console.WriteLine("Choose an operation (+, -, *, /): ");
        //        char operation = Console.ReadKey().KeyChar;
        //        Console.WriteLine();
```

```
    //        switch (operation)
    //        {
    //            case '+':
    //                Console.WriteLine($"Result: {num1} + {num2} = {num1 + num2}");
    //                break;
    //            case '-':
    //                Console.WriteLine($"Result: {num1} - {num2} = {num1 - num2}");
    //                break;
    //            case '*':
    //                Console.WriteLine($"Result: {num1} * {num2} = {num1 * num2}");
    //                break;
    //            case '/':
    //                if (num2 != 0)
    //                {
    //                    Console.WriteLine($"Result: {num1} / {num2} = {num1 / num2}");
    //                }
    //                else
    //                {
    //                    Console.WriteLine("Error: Division by zero is not allowed.");
    //                }
    //                break;
    //            default:
    //                Console.WriteLine("Invalid operation! Please choose one of +, -, *, or /.");
    //                break;
    //        }
    //        Console.WriteLine("Do you want to perform another calculation? (y/n): ");
    //        choice = Console.ReadKey().KeyChar;
    //        Console.WriteLine();

    //    } while (choice == 'y' || choice == 'Y');

    //    Console.WriteLine("Thank you for using the calculator. Goodbye!");
    //}





    //  Lab 7: Print All Numbers From 1 to 100
    //Assignment:
    //Write a C# program to print all numbers from 1 to 100. Use the continue statement to
skip
    //numbers that are divisible by 3, and use the break statement to stop the loop if the
number
```

```
        //exceeds 50.

        //{
        //    for (int i = 1; i <= 100; i++)
        //    {
        //        if (i % 3 == 0)
        //        {
        //            continue;
        //        }
        //        if (i > 50)
        //        {
        //            break;
        //        }
        //        Console.WriteLine(i);
        //    }
```

```
        //  Lab 8: Print All Numbers From 1 to 100
        //Assignment:
        //Write a C# program to print all numbers from 1 to 100. Use the continue statement to
skip
        //numbers that are divisible by 3, and use the break statement to stop the loop if the
number
        //exceeds 50.

        //{
        //    for (int i = 1; i <= 100; i++)
        //    {
        //        if (i % 3 == 0)
        //        {
        //            continue;
        //        }
        //        if (i > 50)
        //        {
        //            break;
        //        }
        //        Console.WriteLine(i);
        //    }
```

```
//Lab 9: Factorial Calculation
//Assignment:
//Write a C# program to calculate the factorial of a given number using a while loop.

//{
//    Console.Write("Enter a number to calculate its factorial: ");
//    int number = Convert.ToInt32(Console.ReadLine());
//    int factorial = 1;
//    int i = number;
//    while (i > 0)
//    {
//        factorial *= i;
//        i--;
//    }
//    Console.WriteLine("The factorial of " + number + " is: " + factorial);
//}
```

```
//Lab 10: Number Pyramid
//Assignment:
//Write a C# program to print a number pyramid using a nested while loop. Example for 5 rows:
//    1
//   2 2
//  3 3 3
// 4 4 4 4
//5 5 5 5 5

//    {
//       Console.Write("Enter the number of rows for the pyramid: ");
//    int rows = Convert.ToInt32(Console.ReadLine());
//    int i = 1;
//    while (i <= rows)
//    {
//        int j = 1;
//        while (j <= i)
//        {
//            Console.Write(i + " ");
//            j++;
```

```
        //       }

        //Console.WriteLine();
        //       i++;
        //   }




        //Lab 11: Guess the Number Game
        //Assignment:
        //Write a C# program that generates a random number between 1 and 100. The user
has to guess
        //   the number, and the program should give hints("too high" or "too low") until the user
guesses
        //    correctly.Use a nested do-while loop to allow the user to play multiple rounds.

        //{
        //    Random random = new Random();
        //    char playAgain;

        //    do
        //    {
        //        int randomNumber = random.Next(1, 101);
        //        int guess;
        //        bool isGuessedCorrectly = false;
        //        Console.WriteLine("I have generated a random number between 1 and 100.");
        //        do
        //        {
        //            Console.Write("Enter your guess: ");
        //            guess = Convert.ToInt32(Console.ReadLine());

        //            if (guess < randomNumber)
        //            {
        //                Console.WriteLine("Too low! Try again.");
        //            }
        //            else if (guess > randomNumber)
        //            {
        //                Console.WriteLine("Too high! Try again.");
        //            }
        //            else
        //            {
        //                Console.WriteLine("Congratulations! You've guessed the number correctly.");
//                isGuessedCorrectly = true;
//            }
//        } while (!isGuessedCorrectly);
//        Console.Write("Do you want to play again? (y/n): ");
//        playAgain = Console.ReadKey().KeyChar;
//        Console.WriteLine();
//    } while (playAgain == 'y' || playAgain == 'Y');
//    Console.WriteLine("Thank you for playing! Goodbye!");
```

```
//Lab 12: Simple Calculator
//Assignment:
//Write a C# program to create a simple calculator using a switch-case statement. The program
//should handle addition, subtraction, multiplication, and division.

//{
//    Console.WriteLine("Simple Calculator");
//    Console.WriteLine("-----------------");
//    Console.Write("Enter the first number: ");
//    double num1 = Convert.ToDouble(Console.ReadLine());
//    Console.Write("Enter the second number: ");
//    double num2 = Convert.ToDouble(Console.ReadLine());
//    Console.WriteLine("Select an operation: ");
//    Console.WriteLine("1. Addition (+)");
//    Console.WriteLine("2. Subtraction (-)");
//    Console.WriteLine("3. Multiplication (*)");
//    Console.WriteLine("4. Division (/)");
//    Console.Write("Enter your choice (1/2/3/4): ");
//    int choice = Convert.ToInt32(Console.ReadLine());
//    double result;
//    switch (choice)
//    {
//        case 1: // Addition
//            result = num1 + num2;
//            Console.WriteLine($"Result: {num1} + {num2} = {result}");
//            break;
//        case 2: // Subtraction
//            result = num1 - num2;
//            Console.WriteLine($"Result: {num1} - {num2} = {result}");
//            break;
//        case 3: // Multiplication
//            result = num1 * num2;
//            Console.WriteLine($"Result: {num1} * {num2} = {result}");
//            break;
//        case 4: // Division
//                // Check for division by zero
//            if (num2 != 0)
//            {
//                result = num1 / num2;
//                Console.WriteLine($"Result: {num1} / {num2} = {result}");
//            }
//            else
//            {
//                Console.WriteLine("Error: Division by zero is not allowed.");
//            }
//            break;
//        default:
//            Console.WriteLine("Invalid choice! Please select a valid operation.");
//            break;
//}
```

```
//Lab 13: Sum of Digits
//Assignment:
//       Write a C# program that accepts a number from the user and calculates the sum of its digits using
//a do-while loop.
//{
//    Console.Write("Enter a number: ");
//    int number = Convert.ToInt32(Console.ReadLine());
//    int sum = 0;
//    int originalNumber = number;
//    if (number < 0)
//    {
//        number = Math.Abs(number);
//    }
//    do
//    {
//        int digit = number % 10;
//        sum += digit;
//        number /= 10;
```

```
    //    } while (number > 0);
    //    Console.WriteLine($"The sum of the digits of {originalNumber} is: {sum}");




    //Lab 14: Finding the Largest Number
    //Assignment:
    //Write a C# program that accepts 10 numbers from the user and finds the largest
number using for loop.

    //{
    //    int largest = int.MinValue;
    //    int number;

    //    Console.WriteLine("Enter 10 numbers:");
    //    for (int i = 1; i <= 10; i++)
    //    {
    //        Console.Write($"Number {i}: ");
    //        number = Convert.ToInt32(Console.ReadLine());
    //        if (number > largest)
    //        {
    //            largest = number;
    //        }
    //    }
    //    Console.WriteLine($"The largest number entered is: {largest}");




    //Lab 15: Sum of Squares
    //Assignment:
    //Write a C# program to calculate the sum of squares of all numbers from 1 to a
user-provided
    //number using a while loop.

    //{
    //    Console.Write("Enter a positive integer: ");
    //    int n = Convert.ToInt32(Console.ReadLine());
    //int sum = 0;
    //int i = 1;
    //while (i <= n)
    //{
    //    sum += i* i;
    //i++;
    //}
    //    Console.WriteLine($"The sum of squares from 1 to {n} is: {sum}");
```

```
    //Lab 16: Countdown Timer
    //Assignment:
    // Write a C# program to simulate a countdown timer from 10 to 1 using a do-while loop,
    //displaying each second.

    //{
    //     int countdown = 10;

    //     do
    //     {
    //         Console.WriteLine(countdown);
    //         countdown--;
    //         Thread.Sleep(1000);
    //     } while (countdown > 0);

    //     Console.WriteLine("Countdown finished!");
    //   }
```

```
    //17. Finding the First Multiple of 5
    //Assignment:
    //Write a C# program to find and print the first multiple of 5 in a given list of numbers.
Use the
    //break statement to exit the loop once a multiple of 5 is found.


    // {
    //     int[] numbers = { 8, 3, 12, 20, 7, 14 };
    //     for (int i = 0; i < numbers.Length; i++)
    //     {
    //         if (numbers[i] % 5 == 0)
    //       {
    //           Console.WriteLine($"The first multiple of 5 is: {numbers[i]}");
    //           break;
    //       }
    //    }
    //}
```

```
    //18.Print Non-Multiples of 3
    //Assignment:
    // Write a C# program to print numbers from 1 to 20, but skip numbers that are divisible
by 3 using
    //the continue statement.


    //{
    //   for (int i = 1; i <= 20; i++)
    //   {
    //      if (i % 3 == 0)
    //      {
    //         continue;
    //      }

    //      Console.WriteLine(i);
    //   }
    //}
```

```
    //Lab 19: Checking for Palindrome
    //Assignment:
    //Write a C# program to check if a given number is a palindrome using a while loop.

    //   {
    //    Console.Write("Enter a number: ");
    //    int originalNumber = int.Parse(Console.ReadLine());
    //int reversedNumber = 0;
    //int number = originalNumber;
```

```
//     while (number > 0)
//       {
//           int digit = number % 10;
//reversedNumber = reversedNumber* 10 + digit;
//             number /= 10;
//        }

//    if (originalNumber == reversedNumber)

//       Console.WriteLine($"{originalNumber} is a palindrome.");

//       else
//              Console.WriteLine($"{originalNumber} is not a palindrome.");
//    }
```

```
//20.Lab 20: Average of Positive Numbers
//Assignment:
//Write a C# program to calculate the average of positive numbers entered by the user.
The
//program should stop when the user enters a negative number using a do-while loop

//{
//    double sum = 0;
//    int count = 0;
//    double number;

//    do
//    {
//       Console.Write("Enter a positive number (negative to stop): ");
//       number = double.Parse(Console.ReadLine());

//       if (number >= 0)
//       {
//           sum += number;
//           count++;
//       }

//    } while (number >= 0);
//    if (count > 0)
//    {
//        double average = sum / count;
//        Console.WriteLine($"The average of the positive numbers is: {average:F2}");
//    }
//    else
//    {
//        Console.WriteLine("No positive numbers were entered.");
//    }
//}
```

```
//21. Reverse a Number
//Assignment:
//Write a C# program to reverse a given number using a do-while loop.

//{
//    Console.Write("Enter a number to reverse: ");
//    int originalNumber = int.Parse(Console.ReadLine());
//    int reversedNumber = 0;

//    do
//    {
//        int digit = originalNumber % 10;
//        reversedNumber = reversedNumber * 10 + digit;
//        originalNumber /= 10;
//    } while (originalNumber > 0);

//    Console.WriteLine($"The reversed number is: {reversedNumber}");
//}
```

```
//22.: Menu-Driven String Operations
//Assignment:
//Write a C# program that presents a menu to the user for various string operations:
//1. Reverse the string
//2. Convert to uppercase
//3. Convert to lowercase
```

```
//4. Find the length of the string
//Use a switch-case statement to implement this.

//{
//    Console.Write("Enter a string: ");
//    string input = Console.ReadLine();

//    Console.WriteLine("\nSelect an operation:");
//    Console.WriteLine("1. Reverse the string");
//    Console.WriteLine("2. Convert to uppercase");
//    Console.WriteLine("3. Convert to lowercase");
//    Console.WriteLine("4. Find the length of the string");

//    Console.Write("Enter your choice (1-4): ");
//    int choice = int.Parse(Console.ReadLine());

//    switch (choice)
//    {
//        case 1:
//            char[] charArray = input.ToCharArray();
//            Array.Reverse(charArray);
//            string reversedString = new string(charArray);
//            Console.WriteLine($"Reversed string: {reversedString}");
//            break;
//        case 2:
//            string upperString = input.ToUpper();
//            Console.WriteLine($"Uppercase string: {upperString}");
//            break;
//        case 3:
//            string lowerString = input.ToLower();
//            Console.WriteLine($"Lowercase string: {lowerString}");
//            break;
//        case 4:
//            int length = input.Length;
//            Console.WriteLine($"Length of the string: {length}");
//            break;
//        default:
//            Console.WriteLine("Invalid choice. Please select a number between 1 and 4.");
//            break;
//    }
//}
```

//23.Skip Multiples of 4

//Assignment:

//Write a C# program to print numbers from 1 to 50 but skip numbers that are divisible by 4 using the continue statement.

```
//{
//    for (int i = 1; i <= 50; i++)
//    {
//        if (i % 4 == 0)
//        {
//            continue;
//        }

//        Console.WriteLine(i);
//    }
//}
```

//24.Fibonacci Sequence

//Assignment:

// Write a C# program to print the first 10 numbers in the Fibonacci sequence using a for loop.

```
//{
//    int n = 10; // Number of Fibonacci numbers to print
//    int firstNumber = 0, secondNumber = 1;

//    Console.WriteLine("Fibonacci Sequence:");

//    for (int i = 0; i < n; i++)
//    {
//        Console.WriteLine(firstNumber); // Print the current Fibonacci number

//        // Calculate the next Fibonacci number
//        int nextNumber = firstNumber + secondNumber;
//        firstNumber = secondNumber; // Update firstNumber
//        secondNumber = nextNumber; // Update secondNumber
//    }
```

```
//25.: Find All Armstrong Numbers
//Assignment:
//Write a C# program to find all Armstrong numbers between 1 and 500 using a while loop. (An
//Armstrong number is a number that is equal to the sum of the cubes of its digits).

//{
//    int number = 1;
//    Console.WriteLine("Armstrong numbers between 1 and 500:");

//    while (number <= 500)
//    {
//        int sum = 0;
//        int temp = number;
//        while (temp > 0)
//        {
//            int digit = temp % 10;
//            sum += digit * digit * digit;
//            temp /= 10;
//        }

//        if (sum == number)
//        {
//            Console.WriteLine(number);
//        }

//        number++;
//    }
//}
```

```
//26.Lab 26: Menu-Driven Number System Conversion
//Assignment:
// Write a C# program to create a menu-driven system for number conversions:
//1. Binary to Decimal
//2. Decimal to Binary
//3. Decimal to Hexadecimal
```

```
    //Use a switch-case statement to implement this.
    //{
    //    while (true)
    //    {
    //        Console.WriteLine("\nMenu:");
    //        Console.WriteLine("1. Binary to Decimal");
    //        Console.WriteLine("2. Decimal to Binary");
    //        Console.WriteLine("3. Decimal to Hexadecimal");
    //        Console.WriteLine("4. Exit");
    //        Console.Write("Select an option (1-4): ");

    //        int choice = int.Parse(Console.ReadLine());

    //        switch (choice)
    //        {
    //            case 1:
    //                Console.Write("Enter a binary number: ");
    //                string binaryInput = Console.ReadLine();
    //                int decimalValue = Convert.ToInt32(binaryInput, 2);
    //                Console.WriteLine($"Decimal value: {decimalValue}");
    //                break;

    //            case 2:
    //                Console.Write("Enter a decimal number: ");
    //                int decimalInput = int.Parse(Console.ReadLine());
    //                string binaryValue = Convert.ToString(decimalInput, 2);
    //                Console.WriteLine($"Binary value: {binaryValue}");
    //                break;

    //            case 3:
    //                Console.Write("Enter a decimal number: ");
    //                int decimalInputHex = int.Parse(Console.ReadLine());
    //                string hexadecimalValue = Convert.ToString(decimalInputHex, 16).ToUpper();
    //                Console.WriteLine($"Hexadecimal value: {hexadecimalValue}");
    //                break;

    //            case 4:
    //                Console.WriteLine("Exiting the program.");
    //                return;

    //            default:
    //                Console.WriteLine("Invalid choice. Please select a number between 1 and
4.");
    //                break;
```

```
        //        }
        //    }
        //}





        //27.Reverse a String
        //Assignment:
        //Write a C# program to reverse a given string using a for loop.
        //{
        //    Console.Write("Enter a string to reverse: ");
        //    string input = Console.ReadLine();
        //    string reversedString = "";
        //    for (int i = input.Length - 1; i >= 0; i--)
        //    {
        //        reversedString += input[i];
        //    }

        //    Console.WriteLine($"Reversed string: {reversedString}");
        //}





        //28.Count Vowels and Consonants in a String
        //Assignment:
        //Write a C# program to count the number of vowels and consonants in a given string
using a for loop.
        //{
        //    Console.Write("Enter a string: ");
        //    string input = Console.ReadLine();

        //    int vowelCount = 0;
        //    int consonantCount = 0;
        //    string vowels = "aeiouAEIOU";
        //    for (int i = 0; i < input.Length; i++)
        //    {
        //        char currentChar = input[i];
        //        if (char.IsLetter(currentChar))
        //        {
        //            if (vowels.Contains(currentChar))
        //          {
        //              vowelCount++;
        //          }
        //          else
        //          {
        //              consonantCount++;
        //          }
        //      }
        //  }

        //  Console.WriteLine($"Number of vowels: {vowelCount}");
        //  Console.WriteLine($"Number of consonants: {consonantCount}");
        //}
```

```
//29.Find Maximum and Minimum
//Assignment:
//Write a C# program to find the maximum and minimum of 5 numbers entered by the user using a for loop.
//{
//  int[] numbers = new int[5];
//  int max, min;
//  Console.WriteLine("Enter 5 numbers:");

//  for (int i = 0; i < 5; i++)
//  {
//      Console.Write($"Number {i + 1}: ");
//      numbers[i] = int.Parse(Console.ReadLine());
//  }
//  max = min = numbers[0];
//  for (int i = 1; i < 5; i++)
//  {
//      if (numbers[i] > max)
//      {
//          max = numbers[i];
//      }

//      if (numbers[i] < min)
//      {
//          min = numbers[i];
//      }
```

```
            //    }

            //    Console.WriteLine($"Maximum number: {max}");
            //    Console.WriteLine($"Minimum number: {min}");
            //}





            //30.Count Digits of a Number
            //Assignment:
            //Write a C# program to count the number of digits in a given number using a while loop.
            //{
            //    Console.Write("Enter a number: ");
            //    int number = Math.Abs(int.Parse(Console.ReadLine()));
            //    int count = 0;
            //    if (number == 0)
            //    {
            //        count = 1;
            //    }
            //    else
            //    {
            //        while (number > 0)
            //        {
            //            number /= 10;
            //            count++;
            //        }
            //    }

            //    Console.WriteLine($"Number of digits: {count}");
            //}
        }
    }
}
```