**От Виктора**

**Задача 1**: Напишите программу, которая бесконечно запрашивает целые числа с консоли. Программа завершается при вводе символа ‘q’ или при вводе числа, сумма цифр которого чётная.

using System;

namespace PrimeNumbersProgram

{

class Program

{

static void Main(string[] args)

{

// Создаем цикл, который будет повторяться бесконечно (используем while с условием true, чтобы создать бесконечный цикл.)

while (true)

{

// Запрашиваем у пользователя целое число или символ 'q': выводим сообщение с помощью метода Write класса Console и считываем введенное значение с помощью метода ReadLine.

Console.Write("Введите целое число или \"q\", чтобы выйти из игры: ");

string input = Console.ReadLine()!;

// Проверяем, является ли введенный символ 'q': сравниваем введенное значение (приведенное к нижнему регистру с помощью метода ToLower) с символом 'q'

if (!string.IsNullOrEmpty(input) && input.Trim().ToLower() == "q")

{

// Если значения совпадают, выводим сообщение о завершении программы и выходим из цикла с помощью ключевого слова break.

Console.WriteLine("Выход из программы...");

break;

}

// Проверяем, удалось ли преобразовать введенное значение в целое число: используем метод TryParse класса int, Если преобразование удалось, сохраняем результат в переменную number.

if (int.TryParse(input, out int number))

{

// Если удалось, вызываем метод IsEvenDigitSum для проверки суммы цифр числа на четность: передаем значение переменной number в метод IsEvenDigitSum и проверяем, является ли сумма цифр числа четной.

if (IsEvenDigitSum(number!))

{

// Если сумма цифр четная, выводим сообщение о завершении программы и выходим из цикла: выводим сообщение с помощью метода WriteLine класса Console и выходим из цикла с помощью ключевого слова break.

Console.WriteLine("Сумма цифр четная. Выход из программы...");

break;

}

}

else

{

// Если не удалось преобразовать введенное значение в целое число, выводим сообщение об ошибке с помощью метода WriteLine класса Console.

Console.WriteLine("Ошибка: недопустимый ввод.");

}

}

}

static bool IsEvenDigitSum(int number)

{

// Объявляем переменную для хранения суммы цифр числа: создаем переменную sum, которая будет хранить сумму цифр числа.

int sum = 0;

// Используем метод Math.Abs для получения абсолютного значения числа: вызываем метод Abs класса Math, чтобы получить абсолютное значение переменной number (чтобы работать только с положительными числами).

number = Math.Abs(number);

// Используем цикл while для вычисления суммы цифр числа:цикл while, который будет выполняться, пока значение переменной number больше 0.

while (number > 0)

{

// На каждой итерации добавляем к переменной sum последнюю цифру числа (остаток от деления на 10) и удаляем ее из числа (делением на 10).

sum += number % 10;

number /= 10;

}

// Проверяем, является ли значение переменной sum (сумма цифр) четным, и возвращаем соответствующее булево значение.

return sum % 2 == 0;

}

}

}

От Константина

// 1. Задайте одномерный массив, заполненный случайными

// числами. Определите количество простых чисел в этом

// массиве.

using System;

class Program

{

static void Main()

{

//создаем объект Random для генерации случайных чисел

Random random = new Random();

//создаем массив из 10 случайных чисел

int[] numbers = new int[10];

for(int i = 0; i < numbers.Length; i++)

{

numbers[i] = random.Next(1, 100); //генерация случайного числа от 1 до 100

Console.Write($"[{numbers[i]}] ");

}

//определение кол-во простых чисел в массиве

int count = 0;

foreach (var number in numbers)

{

if (IsPrime(number))

{

Console.WriteLine($"{number} ");

count++;

}

}

//вывод результата

Console.WriteLine("Количество простых чисел в массиве: "+ count);

}

//метод для определения является ли число простым

static bool IsPrime(int number)

{

if (number < 2) return false;

for (int i = 2; i < number; i++)

{

if (number % i == 0)

{

return false;

}

}

return true;

}

}