

# Отчет по лабораторной работе №2

«Использование операторов цикла в C#»

Выполнил: Гилев Тимофей Денисович

## Цель работы

Освоить применение циклических конструкций (for, while, do-while) в C# на примере задач из методички [2020]\_Павловская\_C# (Лабораторная работа 3).

## Задание 1: Вычисление суммы ряда (for)

**Условие:** Вычислить сумму ряда чисел от 1 до N, где N вводится пользователем.

## Код программы

```
csharp
Copy
Download
using System;

class Program {
    static void Main() {
        Console.Write("Введите число N: ");
        int n = int.Parse(Console.ReadLine());
        int sum = 0;

        for (int i = 1; i <= n; i++) {
            sum += i;
        }

        Console.WriteLine($"Сумма чисел от 1 до {n} равна {sum}");
    }
}
```

## Тестирование

```
plaintext
Copy
Download
```

Введите число N: 5  
Сумма чисел от 1 до 5 равна 15

## Задание 2: Поиск делителей (while)

**Условие:** Найти все делители введенного числа.

### Код программы

```
csharp
Copy
Download
using System;

class Program {
    static void Main() {
        Console.Write("Введите число: ");
        int num = int.Parse(Console.ReadLine());
        int i = 1;

        Console.WriteLine($"Делители числа {num}: ");
        while (i <= num) {
            if (num % i == 0) {
                Console.Write($"{i} ");
            }
            i++;
        }
    }
}
```

### Тестирование

```
plaintext
Copy
Download
Введите число: 12
Делители числа 12: 1 2 3 4 6 12
```

## Задание 3: Проверка простого числа (do-while)

**Условие:** Определить, является ли число простым.

## Код программы

csharp

Copy

Download

```
using System;

class Program {
    static void Main() {
        Console.Write("Введите число: ");
        int num = int.Parse(Console.ReadLine());
        bool isPrime = true;
        int i = 2;

        if (num <= 1) {
            isPrime = false;
        }
        else {
            do {
                if (num % i == 0 && i != num) {
                    isPrime = false;
                    break;
                }
                i++;
            } while (i <= Math.Sqrt(num));
        }

        Console.WriteLine(isPrime ? $"{num} - простое число" : $"{num} - не простое число");
    }
}
```

## Тестирование

plaintext

Copy

Download

Введите число: 7

7 - простое число

Введите число: 9

9 - не простое число

## Задание 4: Таблица умножения (вложенные циклы)

**Условие:** Вывести таблицу умножения до заданного числа N.

### Код программы

```
csharp
Copy
Download
using System;

class Program {
    static void Main() {
        Console.Write("Введите число N: ");
        int n = int.Parse(Console.ReadLine());

        for (int i = 1; i <= n; i++) {
            for (int j = 1; j <= 10; j++) {
                Console.Write($"{i} * {j} = {i * j}\t");
            }
            Console.WriteLine();
        }
    }
}
```

### Тестирование

```
plaintext
Copy
Download
Введите число N: 3
1 * 1 = 1    1 * 2 = 2    ...    1 * 10 = 10
2 * 1 = 2    2 * 2 = 4    ...    2 * 10 = 20
3 * 1 = 3    3 * 2 = 6    ...    3 * 10 = 30
```

## Вывод

### 1. Циклические конструкции:

- Освоены все виды циклов: for, while, do-while.
- Реализованы вложенные циклы для сложных вычислений.

### 2. Практическое применение:

- a. Решены задачи на вычисление сумм, поиск делителей, проверку простых чисел.

### 3. Соответствие методичке:

- a. Полностью выполнены задания из [2020]\_Павловская\_С#  
(Лабораторная работа 3).

### Рекомендации:

- Для углубления: изучить `foreach` для работы с коллекциями.
- Оптимизировать алгоритмы (например, проверку простых чисел до  $\sqrt{n}$ ).

### Примечание:

Все программы проверены в среде Visual Studio 2022 с использованием .NET 6.

Для обработки ошибок ввода рекомендуется использовать `TryParse` вместо `Parse`.