

# Лабораторная работа №1

## «Переменные, арифметические операции, циклы и условия»

**Цель:** целью данной работы является получение опыта работы с переменными, арифметическими операциями, циклами и операторами выбора на языке высокого уровня C# в среде программирования Microsoft Visual Studio.

**Задание:** необходимо разработать 3 набора консольных программ, выполняющих следующие действия:

1. Пользователь вводит число, после чего в консоль выводится сообщение, является ли это число чётным или нечётным.

**Пример:**

Вход: 7	Вход: 8
Выход: “Нечётное”	Выход: “Чётное”

Для выполнения 1 задания вам понадобятся операторы:

1. **Console.ReadLine** – для считывания числа из консоли
2. **Console.WriteLine** – для вывода на экран сообщений
3. **%** – для проверки числа на чётность
4. **if** – для выбора варианта сообщения

2. Пользователь вводит число N. Далее вводит степень P, в которую необходимо возвести число, после чего в консоль выводится результат выполнения операции возведения в степень.

**Пример:**

Вход: 2, 8	Вход: 3, 2
Выход: 256	Выход: 9

Для выполнения 2 задания вам понадобятся операторы:

1. **Console.ReadLine** – для считывания чисел из консоли
2. **Console.WriteLine** – для вывода на экран сообщений
3. **\*** – для умножения чисел
4. **for** – цикл, в котором некая результирующая переменная будет умножаться на число N, P раз

3. Предложение пользователю выбрать операцию: +, -, \*, /. Далее пользователь может ввести первое число, затем пользователю предлагается ввести второе число (целое число), после чего в консоль выводится результат выбранной операции над введёнными числами.

## **Пример:**

Вход: ‘+’, 1, 2      Вход: ‘/’, 1, 2  
Вывод:  $1 + 2 = 3$     Вывод:  $1 / 2 = 0,50$

Для выполнения 3 задания вам понадобятся операторы:

1. **Console.ReadLine** – для считывания чисел из консоли
2. **Console.ReadKey** – для не буферизированного считывания символа из консоли
3. **Console.WriteLine** – для вывода на экран сообщений
4. **switch** – для определения действия, в зависимости от выбранной операции
5. \*, /, -, + – для выполнения арифметических операций

## Справочная информация:

Минимальная программа на языке C#, в среде разработке Visual Studio, выглядит следующим образом:

```
using System; //подключение библиотек
using System.Collections.Generic;

namespace TestApp //пространство имен приложения
{
    internal class Program //класс программы
    {
        static void Main(string[] args) //главная функция
        {
            Console.WriteLine("Hello World"); //вывод сообщения на экран
            Console.ReadKey(); //ожидание нажатия клавиши перед завершением работы
        }
    }
}
```

Объявление и инициализация переменных могут выглядеть следующим образом:

```
static void Main(string[] args) //главная функция
{
    int a; //создание целочисленной переменной
    a = 10; //присвоение значения переменной

    float pi = 3.14f; //создание вещественной переменной и присвоение ей значения
    char c = 'c'; //создание символьной переменной и присвоение ей значения
}
```

Арифметические операции:

```
int a = 1, b = 2; //создание и заполнение целочисленных переменных
float c1, c2, c3, c4, c5, c6; //создание и инициализация вещественной переменных

c1 = a + b; //сложение
c2 = a - b; //вычитание
c3 = a * b; //умножение
c4 = a / b; //целая часть от деления (оба операнда целочисленные)
c5 = (float)a / b; //деление (как минимум один операнд должен быть вещественного типа)
c6 = a % b; //остаток от деления

```

Условный оператор:

```
int a = 5, b = 10;
int max = 0;

if (a > b) //выбор большего из двух чисел
    max = a;
else
    max = b;

```

## Оператор множественного выбора:

```
int input;
input = Convert.ToInt32(Console.ReadLine()); //число, вводимое пользователем
switch (input) //сравнение значения input с доступными вариантами
{
    case 3: //вариант 1 - input == 3
        Console.WriteLine("Пользователь ввёл 3.");
        break;
    case 7: //вариант 2 - input == 7
        Console.WriteLine("Пользователь ввёл 7.");
        break;
    default: //вариант 3 - input == чему-то другому
        Console.WriteLine("Неправильный ввод.");
        break;
}
```

## Функции ввода-вывода:

```
int a;
a = Convert.ToInt32(Console.ReadLine()); //считывание целочисленного значения
float b;
b = Convert.ToSingle(Console.ReadLine()); //считывание вещественного значения

char c;
ConsoleKeyInfo key = Console.ReadKey(); //считывание кода нажатой клавиши
c = key.KeyChar; //получение символа по коду

Console.WriteLine("{0} {1} {2}", a, b, c); //вывод на экран значений переменных a, b и c
```

## Цикл FOR:

```
int sum = 0;
for (int i = 1; i <= 100; i++) //вычисление суммы чисел от 1 до 100
{
    //где: i имеет начальное значение 1 и изменяется на +1
    //до тех пор, пока не достигнет 100
    sum += i;
}
```

## Цикл WHILE:

```
//вычисление суммы чисел от 1 до 100
int i = 1, sum = 0;
while (i <= 100)
{
    sum = sum + i;
    i++; //увеличение i на 1
}
```

## Цикл DO – WHILE:

```
//цикл, который будет выполняться до тех пор
//пока пользователь не введёт 0
int num = 0;
do
{
    Console.WriteLine("Введите число: ");
    num = Convert.ToInt32(Console.ReadLine());
} while (num != 0);
```

В теле циклов могут использоваться операторы **break** и **continue**.

Оператор **break** обеспечивает немедленный выход из цикла.

Оператор **continue** вызывает прекращение текущей и начало следующей итерации.

## **Ссылки на описания используемых функций/операторов:**

1. **Console.ReadLine**,      **Console.WriteLine**,      **Console.ReadKey** –  
<https://devpractice.ru/c-sharp-lesson-3-console-io/#p33>
2. **if** – <https://devpractice.ru/c-sharp-lesson-6-if-and-for/>
3. **for** – <https://metanit.com/sharp/tutorial/2.6.php>
4. **switch** – <https://metanit.com/sharp/tutorial/3.45.php>
5. **\*, /, -, +, %** – <http://csharp.webdelphi.ru/arifmeticheskie-operatory-c/#i-3>