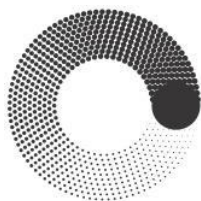


ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ



МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
ВЫСШАЯ ШКОЛА ПЕЧАТИ И МЕДИАИНДУСТРИИ

*Институт Принтмедиа и информационных технологий  
Кафедра Информатики и информационных технологий*

направление подготовки

09.03.02 «Информационные системы и технологии»

## ЛАБОРАТОРНАЯ РАБОТА № 6

Дисциплина: Введение в программирование.

Выполнил(а):

студент(ка) группы 191-726

Синельникова К.Т.

\_\_\_\_\_  
(Дата)

\_\_\_\_\_  
(Подпись)

Проверил: асс. Кононенко К.М.

\_\_\_\_\_  
(Дата)

\_\_\_\_\_  
(Подпись)

Замечания: \_\_\_\_\_

\_\_\_\_\_  
\_\_\_\_\_

Москва

2019

## Оглавление

<b>Теория.....</b>	<b>3</b>
<b>Задания .....</b>	<b>7</b>
<b>Блок-схемы .....</b>	<b>8</b>
<b>Код программы .....</b>	<b>11</b>
<b>Результат программы .....</b>	<b>14</b>

## Теория

### Арифметические операторы.

**Следующие операторы выполняют арифметические операции с числовыми типами:**

унарные — ++ (приращение), -- (уменьшение), + (плюс) и - (минус);

бинарные — \* (умножение), / (деление), % (остаток от деления), + (сложение) и - (вычитание).

Эти операторы поддерживают все целочисленные типы и типы с плавающей запятой.

### Оператор инкремента ++

Оператор инкремента ++ увеличивает операнд на 1. Операндом должна быть переменная, свойство или индексатор.

Оператор инкремента поддерживается в двух формах: постфиксный оператор инкремента (x++) и префиксный оператор инкремента (++x).

*Постфиксный оператор приращения*

Результатом x++ является значение x *перед* выполнением операции

*Префиксный оператор инкремента*

Результатом ++x является значение x *после* выполнения операции

### Оператор декремента --

Унарный оператор декремента -- уменьшает операнд на 1. Операндом должна быть переменная, свойство или индексатор.

Оператор декремента поддерживается в двух формах: постфиксный оператор декремента (x--) и префиксный оператор декремента (--x).

*Постфиксный оператор уменьшения*

Результатом x-- является значение x *перед* выполнением операции

### *Префиксный оператор декремента*

Результатом --x является значение x *после* выполнения операции

## **Операторы унарного плюса и минуса**

Унарный оператор + возвращает значение полученного операнда.

Унарный оператор - изменяет знак операнда на противоположный.

Унарный оператор - не поддерживает тип ulong.

## **Оператор умножения \***

Оператор умножения \* вычисляет произведение операндов.

Унарный оператор \* представляет собой оператор косвенного обращения к указателю.

## **Оператор деления /**

Оператор деления / делит левый операнд на правый.

### *Деление целых чисел*

Для операндов целочисленных типов результат оператора / является целочисленным типом, который равен частному двух операндов, округленному в сторону нуля. Чтобы получить частное двух операндов в виде числа с плавающей запятой, используйте тип float, double или decimal:

### *Деление чисел с плавающей запятой*

Для типов float, double и decimal результатом оператора / является частное двух операндов. Если один из операндов — это decimal, второй операнд не может быть ни float, ни double, так как ни float, ни double не преобразуются неявно в тип decimal. Необходимо явным образом преобразовать операнд float или double в тип decimal. Дополнительные сведения о неявных числовых преобразованиях см. в таблице неявных числовых преобразований.

## **Оператор остатка %**

Оператор остатка % вычисляет остаток от деления левого операнда на правый.

### *Целочисленный остаток*

Для целочисленных операндов результатом  $a \% b$  является значение, произведенное  $a - (a / b) * b$ . Знак ненулевого остатка такой же, как и у левого операнда. Используйте метод `Math.DivRem` для вычисления результатов как целочисленного деления, так и определения остатка.

### *Остаток с плавающей запятой*

Для операндов типа `float` и `double` результатом  $x \% y$  для конечных  $x$  и  $y$  будет значение  $z$ , так что:

знак  $z$ , если отлично от нуля, совпадает со знаком  $x$ ;

абсолютное значение  $z$  является значением, произведенным  $|x| - n * |y|$ ,

где  $n$  — это наибольшее возможное целое число, которое меньше или равно  $|x| / |y|$ , а  $|x|$  и  $|y|$  являются абсолютными значениями  $x$  и  $y$ , соответственно.

### *Примечание*

Этот метод вычисления остатка аналогичен тому, который использовался для целочисленных операндов, но отличается от IEEE 754. Если вам нужна операция остатка, которая соответствует IEEE 754, используйте метод **`Math.IEEERemainder`**.

Сведения о поведении оператора  $\%$  в случае неконечных операндов см. в разделе Оператор остатка спецификации языка C#.

Для операндов `decimal` оператор остатка  $\%$  эквивалентен оператору остатка типа `System.Decimal`.

## **Оператор сложения +**

Оператор сложения  $+$  вычисляет сумму своих операндов. Кроме того, оператор  $+$  можно использовать для объединения строк и делегатов.

## **Оператор вычитания -**

Оператор вычитания  $-$  вычитает правый операнд из левого. Кроме того, оператор  $-$  можно использовать для удаления делегатов.

## Приоритет и ассоциативность операторов

В следующем списке перечислены арифметические операторы в порядке убывания приоритета:

- Постфиксный инкремент  $x++$  и декремент  $x--$
- Префиксный инкремент  $++x$  и декремент  $--x$ , унарные операторы  $+$  и  $-$
- Мультипликативные операторы  $*$ ,  $/$ , и  $\%$
- Аддитивные операторы  $+$  и  $-$

## Задания

1. С начала суток прошло  $N$  секунд ( $N$  — целое). Найти количество секунд, прошедших с начала последней минуты.

2. Дни недели пронумерованы следующим образом: 0 — воскресенье, 1 — понедельник, 2 — вторник, . . . , 6 — суббота. Дано целое число  $K$ , лежащее в диапазоне 1–365. Определить номер дня недели для  $K$ -го дня года, если известно, что в этом году 1 января было понедельником.

3. Дни недели пронумерованы следующим образом: 1 — понедельник, 2 — вторник, . . . , 6 — суббота, 7 — воскресенье. Дано целое число  $K$ , лежащее в диапазоне 1–365, и целое число  $N$ , лежащее в диапазоне 1–7. Определить номер дня недели для  $K$ -го дня года, если известно, что в этом году 1 января было днем недели с номером  $N$ .

4. Даны целые положительные числа  $A$ ,  $B$ ,  $C$ . На прямоугольнике размера  $A \times B$  размещено максимально возможное количество квадратов со стороной  $C$  (без наложений). Найти количество квадратов, размещенных на прямоугольнике, а также площадь незанятой части прямоугольника.

5. Дан номер некоторого года (целое положительное число). Определить соответствующий ему номер столетия, учитывая, что, к примеру, началом 20 столетия был 1901 год.

## Блок-схемы

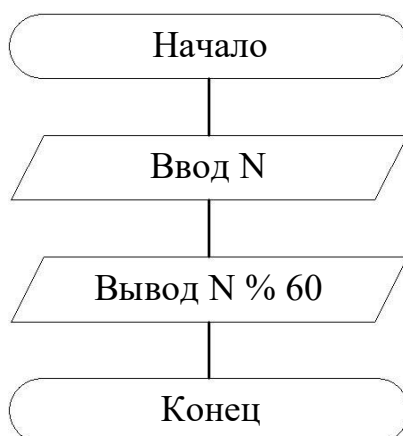


Рисунок 1 — Блок-схема к заданию 1

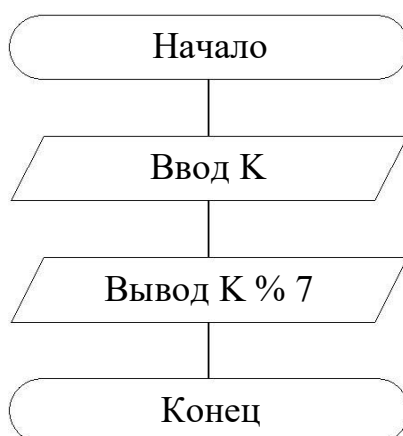


Рисунок 2 — Блок-схема к заданию 2



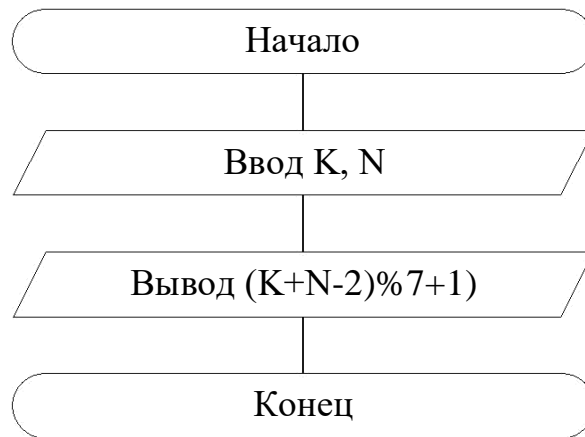


Рисунок 3 — Блок-схема к заданию 3

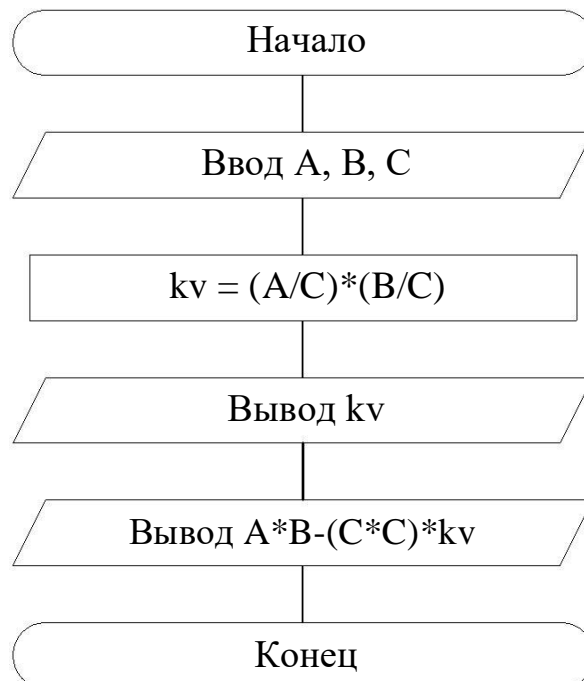


Рисунок 4 — Блок-схема к заданию 4

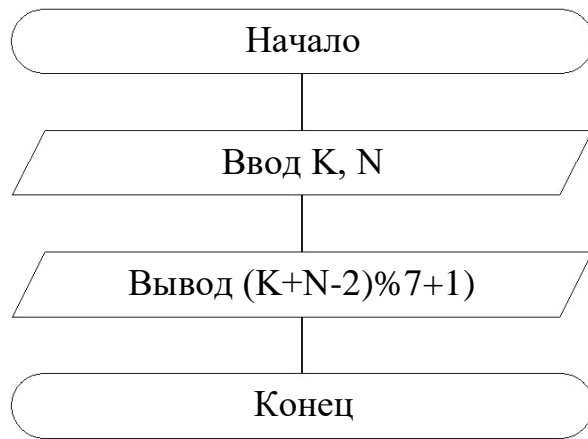


Рисунок 5 — Блок-схема к заданию 5

## Код программы

### Листинг 1 — Задание 1 (Количество секунд, прошедших с начала последней минуты)

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace Код_Лабораторной_6__1_5__шарп
7. {
8.     class Program
9.     {
10.         static void Main(string[] args)
11.         {
12.             int N;
13.             Console.Write("Введите количество секунд с начала суток: ");
14.             N = int.Parse(Console.ReadLine());
15.             Console.WriteLine("Количество секунд прошедших с последней минуты: "
+ N % 60);
16.             Console.ReadKey();
17.         }
18.     }
19. }
```

### Листинг 2 — Задание 2 (Определение номера дня недели)

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace Код_Лабораторной_6__1_5__шарп
7. {
8.     class Program
9.     {
10.         static void Main(string[] args)
11.         {
12.             int K;
13.             Console.Write("Введите число, день недели которого вы хотите найти:
");
14.             K = int.Parse(Console.ReadLine());
15.             Console.WriteLine("День недели для " + K + " дня: " + K % 7);
16.             Console.WriteLine("0 – воскресенье, 1 – понедельник, 2 – вторник, .
. . , 6 – суббота.");
17.             Console.ReadKey();
18.         }
19.     }
20. }
```

### Листинг 3 — Задание 3 (Определение номера дня недели)

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace Код_Лабораторной_6__1_5__шарп
7. {
8.     class Program
9.     {
10.         static void Main(string[] args)
11.         {
12.             int K, N;
13.             Console.WriteLine("Дни недели пронумерованы следующим образом: 1 –
                понедельник, 2 – вторник, . . . , 6 – суббота, 7 – воскресенье.");
14.             Console.Write("Введите день недели с которого начался год: ");
15.             N = int.Parse(Console.ReadLine());
16.             Console.Write("Введите день: ");
17.             K = int.Parse(Console.ReadLine());
18.             Console.WriteLine("День недели для " + K + " дня, начавшегося с " +
                N + " дня недели: " + ((K + N - 2) % 7 + 1));
19.             Console.ReadKey();
20.         }
21.     }
22. }
```

### Листинг 4 — Задание 4 (Нахождение количества квадратов на прямоугольнике и площади незанятой части прямоугольника)

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace Код_Лабораторной_6__1_5__шарп
7. {
8.     class Program
9.     {
10.         static void Main(string[] args)
11.         {
12.             int A, B, C, kv;
13.             Console.Write("Введите A: ");
14.             A = int.Parse(Console.ReadLine());
15.             Console.Write("Введите B: ");
16.             B = int.Parse(Console.ReadLine());
17.             Console.Write("Введите C: ");
18.             C = int.Parse(Console.ReadLine());
19.             kv = (A / C) * (B / C);
20.             Console.WriteLine("Количество квадратов, размещенных на
                прямоугольнике со стороной C: " + kv);
21.             Console.WriteLine("Площадь незанятой части прямоугольника: " + (A *
                B - (C * C) * kv));
22.             Console.ReadKey();
23.         }
24.     }
25. }
```

## Листинг 5 — Задание 5 (Определение номера столетия)

```
1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace Код_Лабораторной_6__1_5__шарп
7. {
8.     class Program
9.     {
10.         static void Main(string[] args)
11.         {
12.             int A;
13.             Console.Write("Введите номер года: ");
14.             A = int.Parse(Console.ReadLine());
15.             Console.WriteLine("Номер столетия: " + (A = (A - 1) / 100 + 1 ));
16.             Console.ReadKey();
17.         }
18.     }
19. }
```

## Результат программы

```
C:\Лабы С#\Лаб.раб. 6\Код Лабораторной 6 (1-5) sharp\bin\Debug\K
Введите количество секунд с начала суток: 385
Количество секунд прошедших с последней минуты: 25
```

Рисунок 6 — Результат выполнения программы 1

```
C:\Лабы С#\Лаб.раб. 6\Код Лабораторной 6 (1-5) sharp\bin\Debug\Код Лабораторной 6 (1-5)
Введите число, день недели которого вы хотите найти: 7
День недели для 7 дня: 0
0 - воскресенье, 1 - понедельник, 2 - вторник, . . . , 6 - суббота.
```

Рисунок 7 — Результат выполнения программы 2

```
C:\Лабы С#\Лаб.раб. 6\Код Лабораторной 6 (1-5) sharp\bin\Debug\Код Лабораторной 6 (1-5) sharp.exe
Дни недели пронумерованы следующим образом: 1 - понедельник, 2 - вторник, . . . , 6 - суббота, 7 - воскресенье.
Введите день недели с которого начался год: 1
Введите день: 14
День недели для 14 дня, начавшегося с 1 дня недели: 7
```

Рисунок 8 — Результат выполнения программы 3

```
C:\Лабы С#\Лаб.раб. 6\Код Лабораторной 6 (1-5) sharp\bin\Debug\Код Лабораторной 6 (1-5) u
Введите A: 3
Введите B: 4
Введите C: 2
Количество квадратов, размещенных на прямоугольнике со стороной C: 2
Площадь незанятой части прямоугольника: 4
```

Рисунок 9 — Результат выполнения программы 4

```
C:\Лабы С#\Лаб.раб. 6\Код Лабораторной 6 (1-5) sharp
Введите номер года: 1743
Номер столетия: 18
```

Рисунок 10 — Результат выполнения программы 5