

# I. АЛГОРИТМЫ ЛИНЕЙНОЙ СТРУКТУРЫ

```
Console.WriteLine("Текст");  
Console.ReadLine(имя переменной);
```

Приступая к решению задач, следует вспомнить, что:

1. программы с линейной структурой являются простейшими и обычно используются для реализации **вычислений по формулам**;
2. в программах с линейной структурой **инструкции выполняются последовательно**, одна за другой;
3. алгоритм программы с линейной структурой может быть представлен следующим образом:



## 1 Программа, выводящая ПРИВЕТСТВИЕ.

Изменить программу, т. о. чтобы она также спрашивала Вашу ФАМИЛИЮ и соответственно выводила ее на экран.

```
using System;
namespace Imya1
{
    class Program
    {
        static void Main()
        {
            Console.Write("Введите имя: ");
            string imya = Console.ReadLine();

            Console.WriteLine("Добро пожаловать, " + imya + "!");
            Console.ReadLine();
        }
    }
}
```

## 2 Программа, вычисляющая СТОИМОСТЬ покупки (нескольких мониторов и принтеров). Изменить программу, т. о. чтобы она также спрашивала ПРОЦЕНТ СКИДКИ и соответственно выводила стоимость покупки со скидкой на экран.

```
using System;
namespace StoimostPokupok
{
    class Program
    {
        static void Main()
        {
            Console.Write("Укажите цену монитора: ");
            double cena1 = Convert.ToDouble(Console.ReadLine());

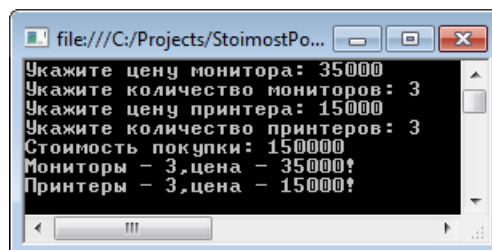
            Console.Write("Укажите количество мониторов: ");
            int kol1 = Convert.ToInt32(Console.ReadLine());

            Console.Write("Укажите цену принтера: ");
            double cena2 = Convert.ToDouble(Console.ReadLine());

            Console.Write("Укажите количество принтеров: ");
            int kol2 = Convert.ToInt32(Console.ReadLine());

            double rez = cena1 * kol1 + cena2 * kol2;

            Console.WriteLine("Стоимость покупки: {0}", rez);
            Console.WriteLine("Мониторы - {0}, цена - {1}!", kol1, cena1);
            Console.WriteLine("Принтеры - {0}, цена - {1}!", kol2, cena2);
            Console.ReadLine();
        }
    }
}
```



**3 Программа, вычисляющая ОБЪЕМ  $V$  цилиндра с радиусом основания  $R$  и высотой  $H$ . Формула для нахождения объема:  $V = \pi \cdot R^2 \cdot H$ .**

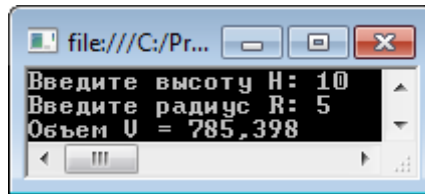
**Изменить программу, т. о. чтобы она так же вычисляла ПЛОЩАДЬ  $S$  боковой поверхности цилиндра с радиусом основания  $R$  и высотой  $H$ .**

**Формула нахождения площади:  $S = 2 \cdot \pi \cdot R \cdot H$ .**

```
using System;
namespace ObiemCilindra
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Введите высоту H: ");
            double H = Convert.ToDouble(Console.ReadLine());

            Console.WriteLine("Введите радиус R: ");
            double R = Convert.ToDouble(Console.ReadLine());

            double V = Math.PI * Math.Pow(R, 2) * H;
            Console.WriteLine("Объем V = {0:N3}", V);
            Console.ReadLine();
        }
    }
}
```



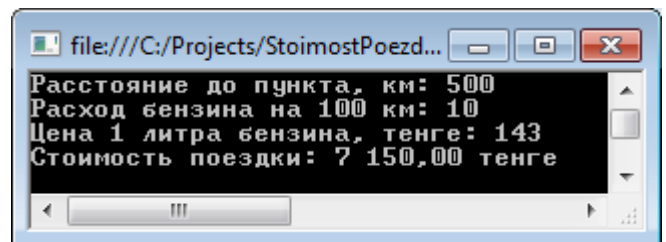
**4 Программа, вычисляющая стоимость поездки на автомобиле до определенного пункта в ПРЯМУЮ сторону.**

**Исходные данные:**

- расстояние до пункта, км;
- расход бензина на 100 км;
- цена одного литра бензина.

**Изменить программу, т. о. чтобы она также вычисляла стоимость поездки на автомобиле до определенного пункта В ПРЯМУЮ И ОБРАТНУЮ стороны.**

```
using System;
namespace StoimostPoezdki
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Расстояние до пункта, км: ");
            double rasst = Convert.ToDouble(Console.ReadLine());
            Console.WriteLine("Расход бензина на 100 км: ");
            double benz = Convert.ToDouble(Console.ReadLine());
            Console.WriteLine("Цена 1 литра бензина, тенге: ");
            double cena = Convert.ToDouble(Console.ReadLine());
            double rez = benz/100*rasst*cena;
            Console.WriteLine("Стоимость поездки: {0:N2} тенге", rez);
            Console.ReadLine();
        }
    }
}
```



**5 Программа, вычисляющая расстояния между ДВУМЯ точками плоскости  $A_1(x_1; y_1)$  и  $A_2(x_2; y_2)$ .**

$$A_1A_2 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

**Изменить программу, т. о. чтобы она также вычисляла расстояние ОТ НАЧАЛА КООРДИНАТ  $O(0;0)$  до точки  $A_1(x_1; y_1)$ . Таким образом, вышеуказанная формула примет вид:**

$$A_1O = \sqrt{x_1^2 + y_1^2}$$

**Примечание:**

**Введите координаты точки  $A_1$  (т.е. значения  $x_1$  и  $y_1$ ).**

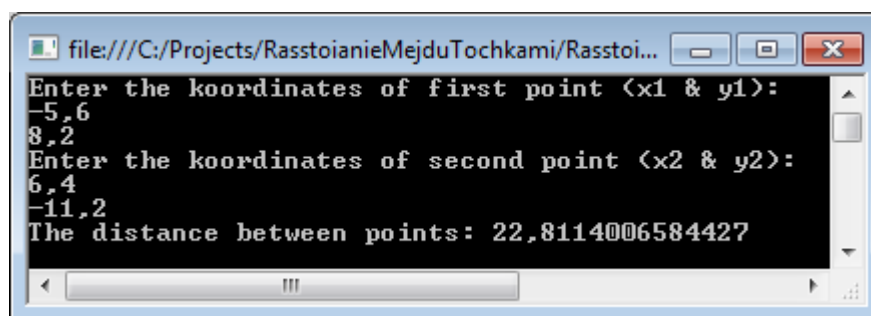
**Введите координаты точки  $A_2$  (т.е. значения  $x_2$  и  $y_2$ ).**

**Для возведения в квадрат используйте методы класса `Math`:**

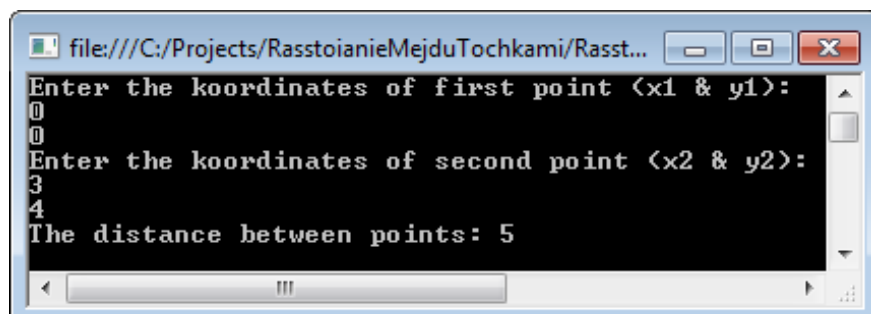
- 1. Для возведения в степень `Math.Pow(число, степень)`**
- 2. Для извлечения квадратного корня из числа `Math.Sqrt(число)`**

```
using System;
namespace RasstoianieMejduTochkami
{
    class Program
    {
        static void Main(string[] args)
        {
            double x1, y1, x2, y2, s;
            Console.WriteLine("Enter the koordinates of first point (x1 & y1):");
            x1 = double.Parse(Console.ReadLine());
            y1 = double.Parse(Console.ReadLine());
            Console.WriteLine("Enter the koordinates of second point (x2 & y2):");
            x2 = double.Parse(Console.ReadLine());
            y2 = double.Parse(Console.ReadLine());

            s = Math.Sqrt((x2-x1)*(x2-x1)+(y2-y1)*(y2-y1));
            Console.WriteLine("The distance between points: " + s);
            Console.ReadLine();
        }
    }
}
```



```
file:///C:/Projects/RasstoianieMejduTochkami/Rasstoi...
Enter the koordinates of first point (x1 & y1):
-5.6
8.2
Enter the koordinates of second point (x2 & y2):
6.4
-11.2
The distance between points: 22.8114006584427
```



```
file:///C:/Projects/RasstoianieMejduTochkami/Rasst...
Enter the koordinates of first point (x1 & y1):
0
0
Enter the koordinates of second point (x2 & y2):
3
4
The distance between points: 5
```

**6 Алгоритм, вычисляющий ПРОИЗВЕДЕНИЕ цифр целого ЧЕТЫРЕХЗНАЧНОГО числа. Изменить алгоритм и написать программу, т. о. чтобы также находилась СУММА цифр целого ШЕСТИЗНАЧНОГО числа.**

**Примечание:**

**Используйте операции: остатка от деления % и частного от деления /.**

**Разряд единиц:**

**1234 % 10 = 4**

**ed := n%10**

**Разряд сотен:**

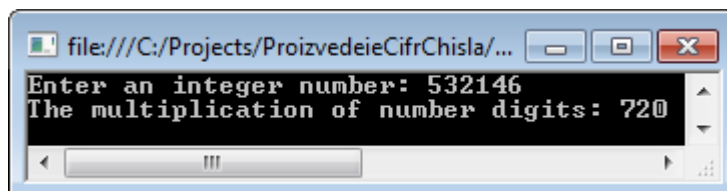
**1234 % 100 = 34**

**34 / 10=3**

**sot := n%100**

**sot := n%10**

```
using System;
namespace ProizvedeieCifrChisla
{
    class Program
    {
        static void Main(string[] args)
        {
            int m, n, s = 1;
            Console.WriteLine("Enter an integer number: ");
            n=int.Parse(Console.ReadLine());
            m = n;
            while (m != 0)
            {
                s = s * (m % 10); // s - произведение единиц
                m = m / 10;      // m - единицы числа
            }
            Console.WriteLine("The multiplication of number digits: " + s);
            Console.ReadLine();
        }
    }
}
```



## II. АЛГОРИТМЫ РАЗВЕТВЛЯЮЩЕЙСЯ СТРУКТУРЫ

IF (...) {...}

ЕСЛИ (...) ТО {...} ИНАЧЕ {...}

Приступая к решению задач, следует вспомнить, что:

1. конструкция **if** используется для выбора одного из двух направлений дальнейшего хода программы (последовательности инструкций, которые должны быть выполнены);

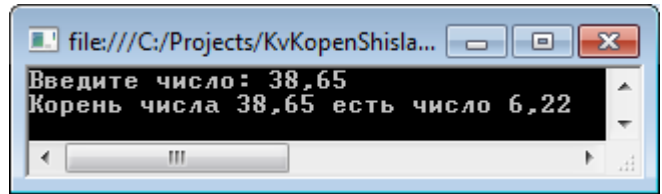


2. **выбор** последовательности инструкций **осуществляется** во время выполнения программы **в зависимости от выполнения условия**;
3. **условие** – это выражение логического типа, которое **может принимать одно из двух значений: true** (истина – условие выполняется) или **false** (ложь – условие не выполняется);
4. **при помощи** логических операций **&** (логическое И) и **|** (логическое ИЛИ) из простых условий **можно строить сложные условия**.

## 1 Программа, вычисляющая КОРЕНЬ числа.

Изменить программу, т.о. чтобы она также находила КВАДРАТ числа.

```
using System;
namespace KvKopenShisla
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Введите число: ");
            double chislo = Convert.ToDouble(Console.ReadLine());
            if (chislo < 0)
                Console.WriteLine("Корень не может быть найден!");
            else
                Console.WriteLine("Корень числа "+chislo+" есть число {0:N2} ",
                    Math.Sqrt(chislo));
            Console.ReadLine();
        }
    }
}
```



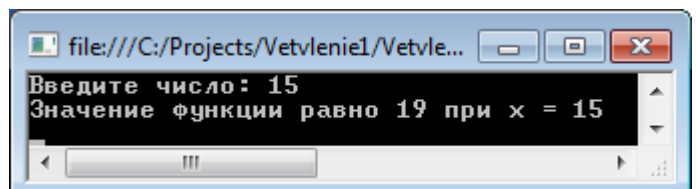
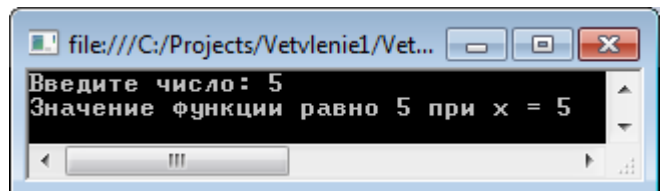
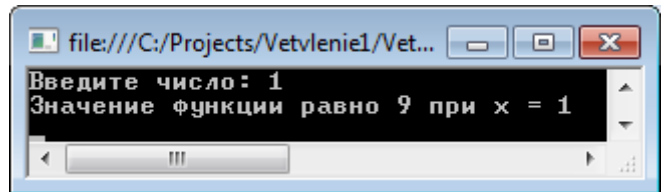
## 2 Программа, находящая ЗНАЧЕНИЕ функции:

$$y = \begin{cases} -2x + 11, & \text{если } x < 3; \\ 5, & \text{если } 3 \leq x \leq 8; \\ 2x - 11, & \text{если } x > 8; \end{cases}$$

Изменить программу, т.о. чтобы она находила ЗНАЧЕНИЕ следующей функции:

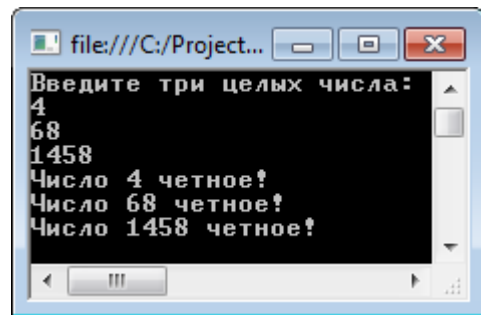
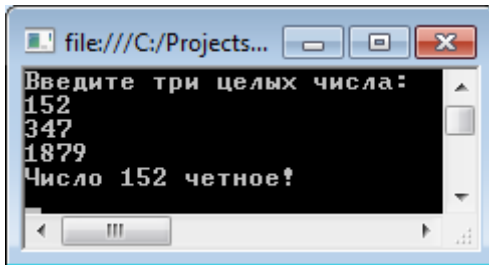
$$y = \begin{cases} -2, & \text{если } x < 5; \\ x, & \text{если } 5 \leq x \leq 11; \\ 2x, & \text{если } x > 11; \end{cases}$$

```
using System;
namespace Vetvlenie1
{
    class Program
    {
        static void Main()
        {
            Console.WriteLine("Введите число: ");
            int x = Convert.ToInt32(Console.ReadLine());
            int y;
            if (x < 3)
                y = -2 * x + 11;
            else
            {
                if (x <= 8)
                    y = 5;
                else
                    y = 2*x - 11;
            }
            Console.WriteLine("Значение функции равно "+y+" при x = "+x);
            Console.ReadLine();
        }
    }
}
```



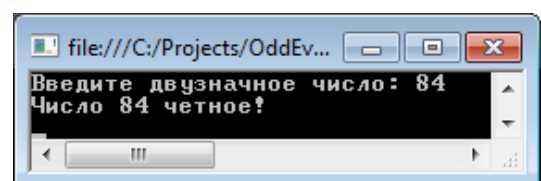
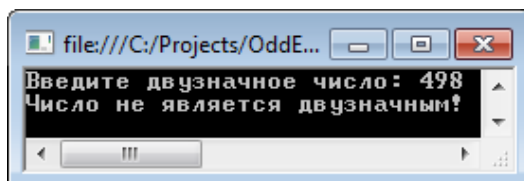
### 3 Программа, определяющая среди ТРЕХ целых чисел ЧЕТНОЕ. Изменить программу, т.о. чтобы она определяла среди ДВУХ целых чисел НЕЧЕТНОЕ.

```
using System;
namespace ChetNechetChisla
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Введите три целых числа:");
            int ch1 = Convert.ToInt32(Console.ReadLine());
            int ch2 = Convert.ToInt32(Console.ReadLine());
            int ch3 = Convert.ToInt32(Console.ReadLine());
            if (ch1 % 2 == 0)
                Console.WriteLine("Число " + ch1 + " четное!");
            if (ch2 % 2 == 0)
                Console.WriteLine("Число " + ch2 + " четное!");
            if (ch3 % 2 == 0)
                Console.WriteLine("Число " + ch3 + " четное!");
            Console.ReadLine();
        }
    }
}
```



### 4 Программа, определяющая является ли целое число НЕЧЕТНЫМ ДВУЗНАЧНЫМ числом. Изменить программу, т.о. чтобы она определяла является ли целое число ЧЕТНЫМ ОДНОЗНАЧНЫМ числом.

```
using System;
namespace OddEven2digitNumber
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Введите двузначное число: ");
            int ch = Convert.ToInt32(Console.ReadLine());
            // -10 < [-9...9] < 10
            if (((ch < 10) & (ch > -10)) | (ch < -99) | (ch > 99))
                Console.WriteLine("Число не является двузначным!");
            else
            {
                if ((ch % 2) == 0)
                    Console.WriteLine("Число " + ch + " четное!");
                else
                    Console.WriteLine("Число " + ch + " нечетное!");
            }
            Console.ReadLine();
        }
    }
}
```





**5 Программа, определяющая максимальное число из ТРЕХ заданных.  
Изменить программу, т.о. чтобы она определяла максимальное число из  
ПЯТИ заданных.**

```
using System;
namespace Max3chisel
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Введите три целых числа (a,b,c):");
            int a = Convert.ToInt32(Console.ReadLine());
            int b = Convert.ToInt32(Console.ReadLine());
            int c = Convert.ToInt32(Console.ReadLine());
            int max;

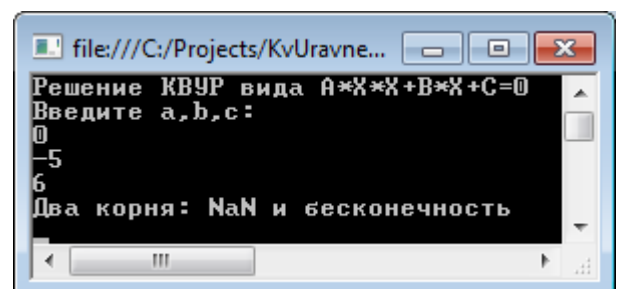
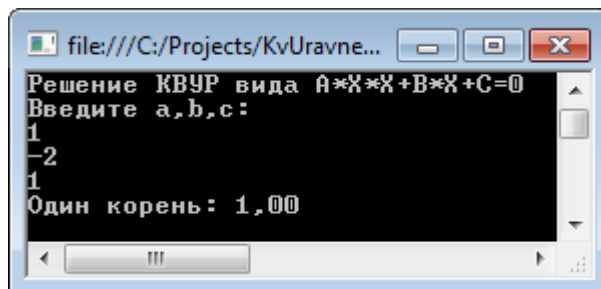
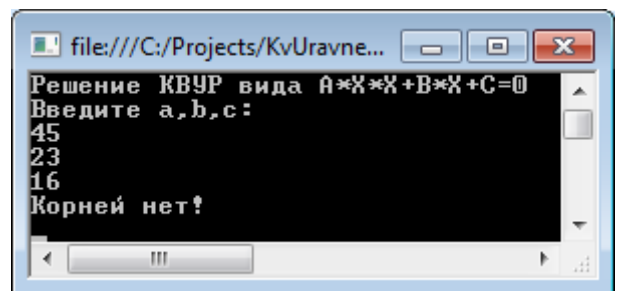
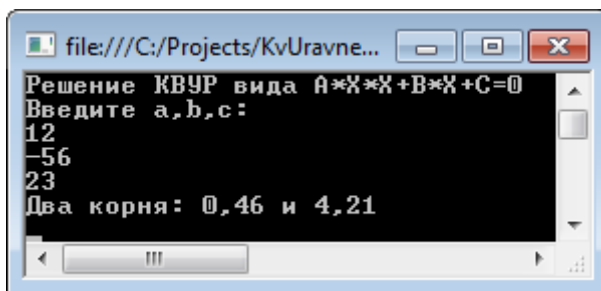
            if (a > b) max = a;
            else      max = b;

            if (max > c)      ;
            else      max = c; // максимальное число уже найдено
            Console.WriteLine("Максимальное число " + max);
            Console.ReadLine();
        }
    }
}
```

## 6 Программа, решающая КВАДРАТНОЕ уравнение.

Изменить программу, т.о. чтобы она выводила сообщение об ошибке, если пользователь вводит значение коэффициента А равное НУЛЮ.

```
using System;
namespace KvUraveniel
{
    class Program
    {
        static void Main(string[] args)
        {
            double a,b,c,d,x1,x2;
            Console.WriteLine( "Решение КВУР вида А*Х*Х+В*Х+С=0");
            Console.WriteLine("Введите a,b,c: ");
            a = Convert.ToDouble(Console.ReadLine());
            b = Convert.ToDouble(Console.ReadLine());
            c = Convert.ToDouble(Console.ReadLine());
            d = Math.Pow(b,2)-(4*a*c);
            if (d < 0)
                Console.WriteLine("Корней нет! ");
            else
            {
                if (d == 0)
                {
                    x1 = -b / (2 * a);
                    Console.WriteLine("Один корень: {0:N2}",x1);
                }
                else
                {
                    x1 = (-b-Math.Sqrt(d))/(2*a);
                    x2 = (-b+Math.Sqrt(d))/(2*a);
                    Console.WriteLine("Два корня: {0:N2} и {1:N2}",x1,x2);
                }
            }
            Console.ReadLine();
        }
    }
}
```



### III. СОСТАВЛЕНИЕ ПРОГРАММ, НА ОСНОВЕ ОПЕРАТОРА ВАРИАНТА SWITCH ( ... ) { CASE ... ; } ВЫБОР ... ИЗ ...

Оператора варианта `switch()` используется в тех случаях, когда в зависимости от значений какой-либо переменной необходимо выполнить те или иные операторы:

```
switch (управляющая переменная)
{
    case набор значений_1 : оператор_1;
    case набор значений_2 : оператор_2;
    ...
    case набор значений_n : оператор_n;
    default : /* блок default может отсутствовать */
              альтернативный оператор;
}
```

#### Принцип работы оператора

Если управляющая переменная принимает значение из набора значений\_1, то выполняется оператор\_1.

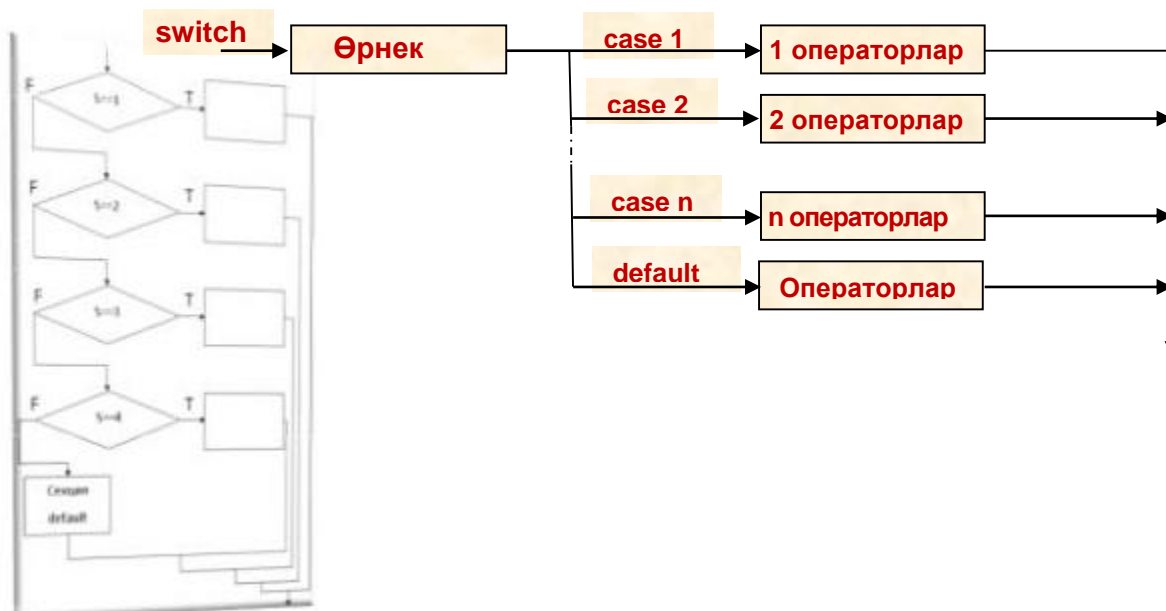
Если управляющая переменная принимает значение из набора значений\_2, то выполняется оператор\_2.

Если управляющая переменная принимает значение из набора значений\_n, то выполняется оператор\_n.

Если управляющая переменная не принимает ни одно значение из имеющихся наборов, то выполняется альтернативный\_оператор.

Приступая к решению задач, следует помнить, что:

- оператор `switch()` используется для выбора одного из нескольких направлений дальнейшего хода программы;
- выбор операторов осуществляется во время выполнения программы в зависимости от значения управляющей переменной;
- в качестве управляющей переменной можно использовать переменную целого (`int`) или символьного (`char`) типа.

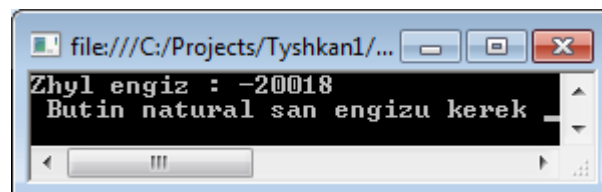
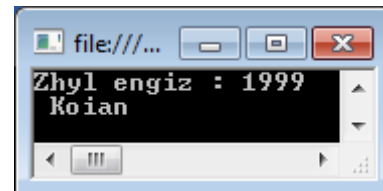
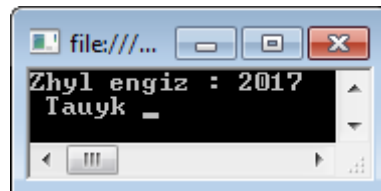


# 1 Программа выводит наименование животного, соответствующее введенному номеру года по ВОСТОЧНОМУ КАЛЕНДАРЮ.

Изменить программу, т.о. чтобы она выводила сообщение об ошибке с указанием типа введенной информации, если пользователь вводит отрицательное или дробное значение, или строку.

```
using System;
namespace Tyshkan1
{
    class Program
    {
        static void Main(string[] args)
        {
            int gyl;
            Console.WriteLine("Zhyl engiz : ");
            gyl = int.Parse(Console.ReadLine());
            switch (gyl % 12)
            {
                case 0: Console.WriteLine(" Meshin "); break;
                case 1: Console.WriteLine(" Tauyk "); break;
                case 2: Console.WriteLine(" It "); break;
                case 3: Console.WriteLine(" Dongyz "); break;
                case 4: Console.WriteLine(" Tushkan "); break;
                case 5: Console.WriteLine(" Syir "); break;
                case 6: Console.WriteLine(" Barys "); break;
                case 7: Console.WriteLine(" Koian "); break;
                case 8: Console.WriteLine(" Ulu "); break;
                case 9: Console.WriteLine(" Zhylan "); break;
                case 10: Console.WriteLine(" Zhylky "); break;
                case 11: Console.WriteLine(" Koy "); break;

                default: Console.WriteLine("Butin natural san engizu kerek"); break;
            } Console.ReadLine();
        }
    }
}
```



### 3 Программа, вычисляющая стоимость междугороднего телефонного разговора. Исходными данными являются код города и продолжительность разговора.

Город	Код	Цена минуты, тенге
Астана	717	25,00
Актобе	713	33,20

Введите код города

> 713

Введите продолжительность разговора, мин.

> 3

**Город: Астана**

**Цена минуты: 25.00 тенге**

**Стоимость разговора: 75 тенге**

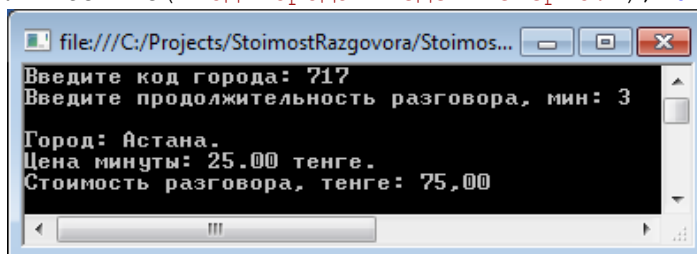
### Изменить программу, увеличив количество поддерживаемых городов.

```
using System;
namespace StoimostRazgovora
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Введите код города: ");
            int kod = Convert.ToInt32(Console.ReadLine());
            int min;
            switch (kod)
            {
                case 713:
                    Console.WriteLine("Введите продолжительность разговора, мин: ");
                    min = Convert.ToInt32(Console.ReadLine());
                    Console.WriteLine();
                    Console.WriteLine("Город: Актобе.");
                    Console.WriteLine("Цена минуты: 33.20 тенге.");
                    Console.WriteLine("Стоимость разговора, тенге: {0:N2}", min*33.2);
                    break;

                case 728:
                    Console.WriteLine("Введите продолжительность разговора, мин: ");
                    min = Convert.ToInt32(Console.ReadLine());
                    Console.WriteLine();
                    Console.WriteLine("Город: Тараз.");
                    Console.WriteLine("Цена минуты: 12.00 тенге.");
                    Console.WriteLine("Стоимость разговора, тенге: {0:N2}", min*12.0);
                    break;

                case 717:
                    Console.WriteLine("Введите продолжительность разговора, мин: ");
                    min = Convert.ToInt32(Console.ReadLine());
                    Console.WriteLine();
                    Console.WriteLine("Город: Астана.");
                    Console.WriteLine("Цена минуты: 25.00 тенге.");
                    Console.WriteLine("Стоимость разговора, тенге: {0:N2}", min*25.0);
                    break;

                default: Console.WriteLine("Код города введен неверно!");
                    break;
            }
            Console.ReadLine();
        }
    }
}
```



## 6 Программа, которая вычисляет дату следующего дня.

### Примечание:

Месяц содержит 31 день: 1, 3, 5, 7, 8, 10, 12.

30 дней: 4, 6, 9, 11.

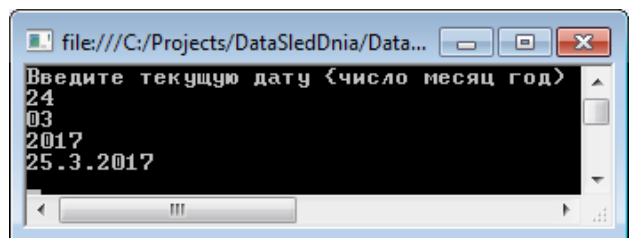
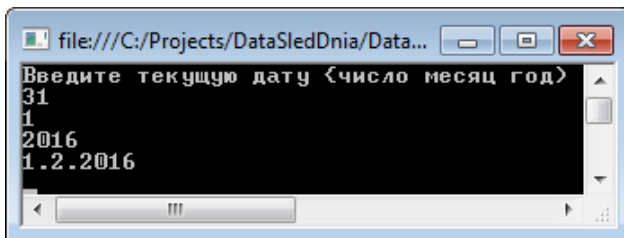
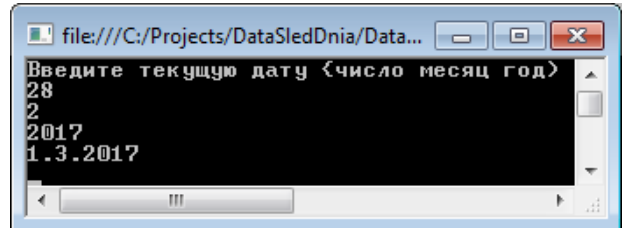
29/28 дней: 2 (високосный, т.е. остаток от деления на 4 равен нулю/обычный).

Введите текущую дату (число месяц год)

> 31 12 2016

**1.1.2017**

```
using System;
namespace DataSledDnia
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Введите текущую дату {число месяц год} ");
            int d = Convert.ToInt32(Console.ReadLine());
            int m = Convert.ToInt32(Console.ReadLine());
            int y = Convert.ToInt32(Console.ReadLine());
            bool b = false;
            // разместить проверку вводимых пользователем данных
            switch (m)
            {
                case 1: case 3: case 5: case 7: case 8: case 10: case 12:
                    { if (d == 31) b = true; break; }
                case 4: case 6: case 9: case 11:
                    { if (d == 30) b = true; break; }
                case 2:
                    if (d == 28)
                    { if (y % 4 != 0) b = true; }
                    break;
            }
            if (b)
            {
                d = 1; m += 1;
                if (m > 12)
                {
                    m = 1; y += 1;
                }
            }
            else d += 1;
            Console.WriteLine(d + "." + m + "." + y);
            Console.ReadLine();
        }
    }
}
```



Дополнить программу проверкой вводимых пользователем данных:

Год должен принадлежать отрезку от 2000 до 2999.

Месяц не должен превышать 12.

```
если (год < 2000) или (год > 2999) или (месяц > 12) то  
вывод('Ошибка!');  
return;
```

В обычном году не должно быть более 28 дней, а в високосном более 29.

```
если (год % 4 != 0) то  
начало  
    если день > 28 то  
        начало  
            вывод('Ошибка!');  
            return;  
        конец;  
    конец  
иначе  
начало  
    если день > 29 то  
        начало  
            вывод('Ошибка!');  
            return;  
        конец;  
    конец;
```

#### 4. АЛГОРИТМЫ ЦИКЛИЧЕСКОЙ СТРУКТУРЫ: ЦИКЛ С ПРЕДУСЛОВИЕМ

**ПОКА (...) ДЕЛАТЬ {...}**  
**WHILE (...) {...}**

Приступая к решению задач, следует вспомнить, что:

Обращение к оператору **while** означает **пока (...) делать ...** и выглядит так:

```
while (условие) {  
    оператор 1;  
    оператор 2;  
    ...  
    оператор n;  
}
```

Операторы, внутри блока { } будут выполняться циклически, пока логическое условие **ИСТИННО**.

Если условие **ложно**, то выполняются операторы, следующие за телом цикла.

Т.о. с помощью циклов типа **пока** можно составить любые повторяющиеся фрагменты алгоритмов.

Цикл **пока** обычно используют, когда **число повторений заранее неизвестно** (например, цикл до достижения требуемой точности результата или цикл до первого отрицательного элемента массива и т.п.).

**Замечание:**

- ✓ Чтобы цикл мог завершиться, содержимое его тела должно обязательно влиять на условие цикла.

1. Определить наибольший общий делитель (НОД) двумя описанными ниже способами для чисел:

1 вариант:	2 вариант:	3 вариант:	4 вариант:
А) 130 и 75;	А) 135 и 80;	А) 140 и 85;	А) 145 и 90;
Б) 23 и 7;	Б) 22 и 6;	Б) 21 и 5;	Б) 20 и 4;
В) 45 и 18;	В) 50 и 23;	В) 55 и 28;	В) 60 и 33;
Г) 64 и 10.	Г) 63 и 9.	Г) 62 и 8.	Г) 61 и 7.

Способ 1 (алгоритм Евклида):

Пусть даны два числа: 264 и 192.

1. Берется большее число и делится на меньшее.  
 $264 : 192$  берем по 1, остаток равен 72.
2. Затем меньшее число делится на первый остаток.  
 $192 : 72$  берем по 2, остаток равен 48.
3. Затем первый остаток делится на второй.  
 $72 : 48$  берем по 1, остаток равен 24.
4. Второй на третий и так далее до тех пор, пока не получится в остатке нуль.  
 $48 : 24$  берем по 1, остаток равен 0.
5. Последний делитель будет НОД данных чисел.



## Способ 2:

264	192
72	120
72	48
24	48
24	24

Даны два числа: 264 и 192.

Если они равны, то НОД найден, иначе определить большее из чисел.

Заменить большее число разностью большего и меньшего, в нашем случае:

264 заменить на разность  $264 - 192$ .

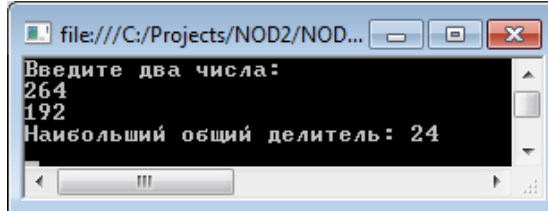
Повторять действия пока числа не будут равны.

НОД равен 24.

## Способ 1:

```
using System;
namespace NOD1 // алгоритм Евклида
{
    class Program
    {
        static void Main(string[] args)
        {
            int n1, n2, nod, ostatek;
            Console.WriteLine("Введите два числа: ");
            n1 = Convert.ToInt32(Console.ReadLine());
            n2 = Convert.ToInt32(Console.ReadLine());

            while (n1 % n2 != 0)
            {
                ostatek = n1 % n2;
                n1 = n2;
                n2 = ostatek;
            }
            nod = n2;
            Console.WriteLine("Наибольший общий делитель: "+nod);
            Console.ReadLine();
        }
    }
}
```

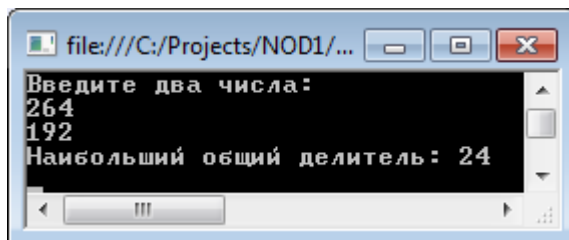


## Способ 2:

```
using System;
namespace NOD2 // Другой алгоритм
{
    class Program
    {
        static void Main(string[] args)
        {
            int n1, n2;
            Console.WriteLine("Введите два числа: ");
            n1 = Convert.ToInt32(Console.ReadLine());
            n2 = Convert.ToInt32(Console.ReadLine());

            while (n1 != n2)
            {
                if (n1 >= n2) n1 = n1 - n2;
                else n2 = n2 - n1;
            }

            Console.WriteLine("Наибольший общий делитель: "+n1);
            Console.ReadLine();
        }
    }
}
```



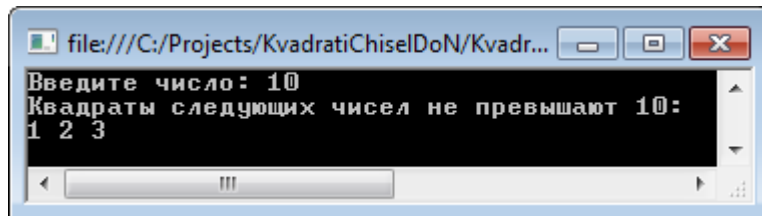
### 3. Программа для указанного числа n:

а) выводящая для натурального числа (1, 2, 3, ...), квадрат которых не превышает n.

```
using System;
namespace ChisloKvadratBolsheN // Квадрат сл. чисел не больше n
{
    class Program
    {
        static void Main()
        {
            int n, i;

            Console.Write("Введите число: ");
            n = Convert.ToInt32(Console.ReadLine());

            i = 1;
            Console.WriteLine("Квадраты следующих чисел не превышают {0}: ",n);
            while (Math.Pow(i, 2) <= n)
            {
                Console.Write(i+" ");
                i++;
            }
            Console.ReadLine();
        }
    }
}
```



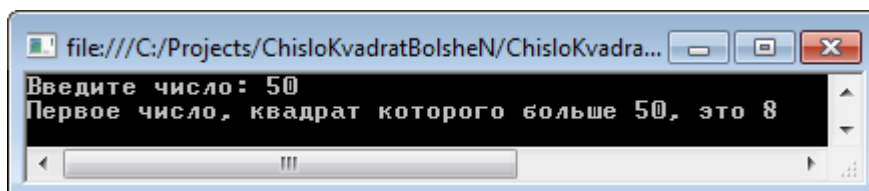
б) выводящая первое натуральное число, квадрат которого больше n.

```
using System;
namespace ChisloKvadratBolsheN // Первое число, квадрат кот-го больше n
{
    class Program
    {
        static void Main()
        {
            int n, i;

            Console.Write("Введите число: ");
            n = Convert.ToInt32(Console.ReadLine());

            i = 1;
            Console.WriteLine("Первое число, квадрат которого больше "+n);
            while (Math.Pow(i, 2) <= n)
            {
                i++;
            }
            Console.Write(i+" ");

            Console.ReadLine();
        }
    }
}
```



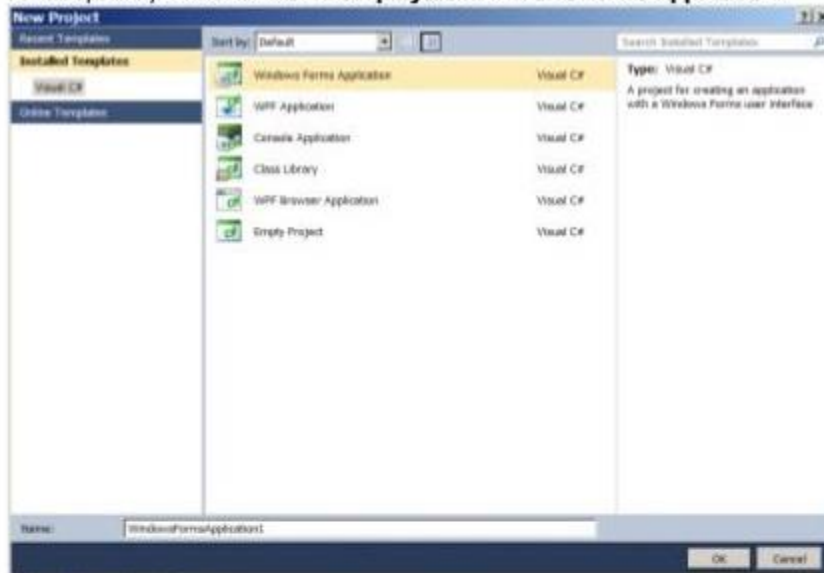
2 Написать программу, в которой ПК загадывает случайное число от 1 до 100. Пользователь должен угадать это число. Пользователю предоставляется возможность ввода попыток «до победного». При работе программы выводятся сообщения «Больше» и «Меньше», при попадании на задуманное число ПК сообщает об этом. Одновременно подсчитывается затраченное число шагов на отгадывание задуманного числа.

3 Написать программу определяющую, сколько цифр содержит натуральное число N.

# Программирование на языке C# с помощью Windows-форм

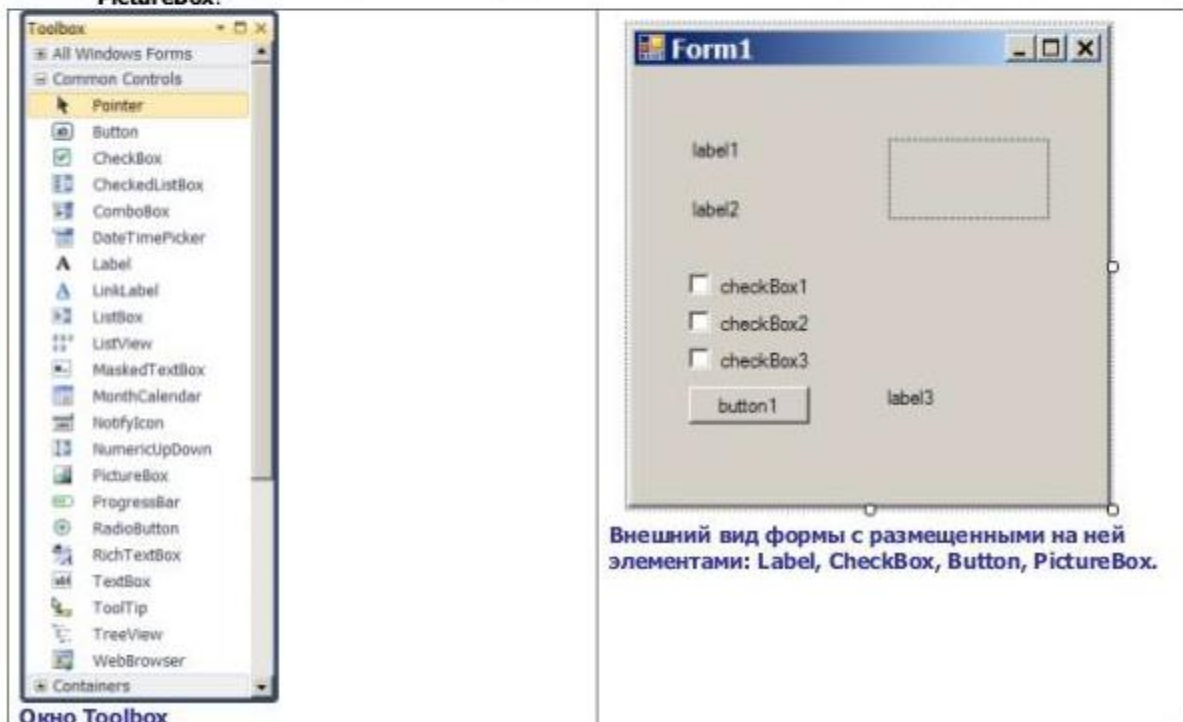
## Порядок работы:

1. Запустить среду программирования **Microsoft Visual C# 2010 Express**.
2. Выбрать пункт меню **File – New project – Windows Forms Application**.



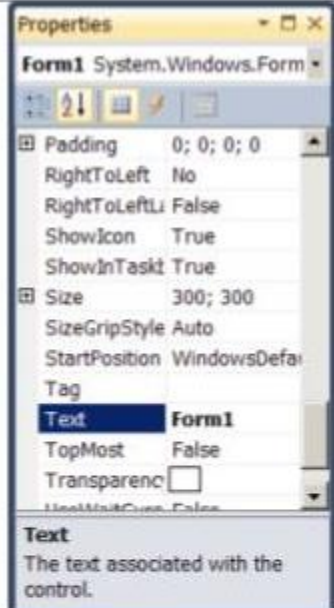
Окно **New Project**.

3. Используя окно **Toolbox** разместить на форме основные элементы: **Label**, **CheckBox**, **Button**, **PictureBox**.



Внешний вид формы с размещенными на ней элементами: **Label**, **CheckBox**, **Button**, **PictureBox**.

4. Используя окно **Properties** выполнить настройку формы и ее компонентов.



**Properties**

Form1 System.Windows.Form

Padding 0; 0; 0; 0

RightToLeft No

RightToLeftLayout False

ShowIcon True

ShowInTaskbar True

Size 300; 300

SizeGripStyle Auto

StartPosition WindowsDefault

Tag

**Text** Form1

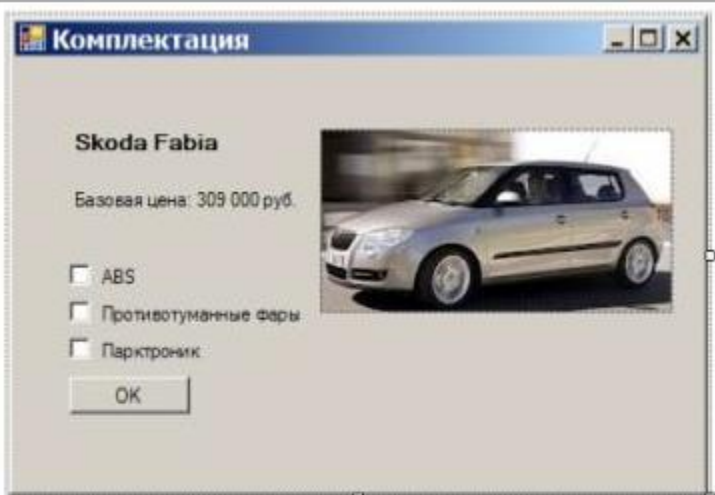
TopMost False

Transparent

UseWaitCursor False

**Text**

The text associated with the control.



**Комплектация**

**Skoda Fabia**

Базовая цена: 309 000 руб.

ABS

Противотуманные фары

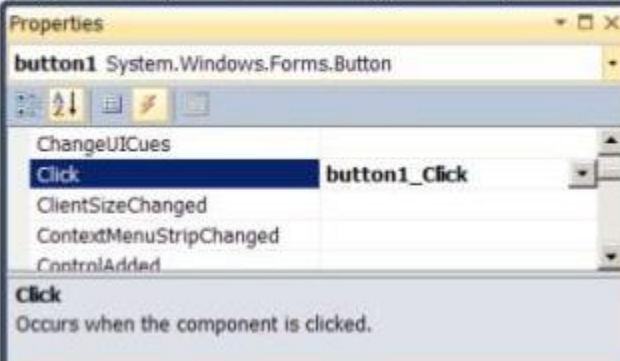
Парктроник

OK

**Внешний вид формы с настроенными на ней элементами: Label, CheckBox, Button, PictureBox.**

**Окно Properties**

5. Добавить обработчик события двойного щелчка на кнопке ОК.



**Properties**

button1 System.Windows.Forms.Button

ChangeUICues

**Click** button1\_Click

ClientSizeChanged

ContextMenuStripChanged

ControlAdded

**Click**

Occurs when the component is clicked.

6. Написать код **обработчика события двойного щелчка на кнопке OK.**

```
private void button1_Click(object sender, EventArgs e)
{
    double sum;
    double skidka;
    double cena;

    sum = 309000;
    if (checkBox1.Checked)
    {
        //abs
        sum += 8390;
    }

    if (checkBox2.Checked)
    {
        //fan
        sum += 5990;
    }

    if (checkBox3.Checked)
    {
        //parktronic
        sum += 7590;
    }

    cena = sum;

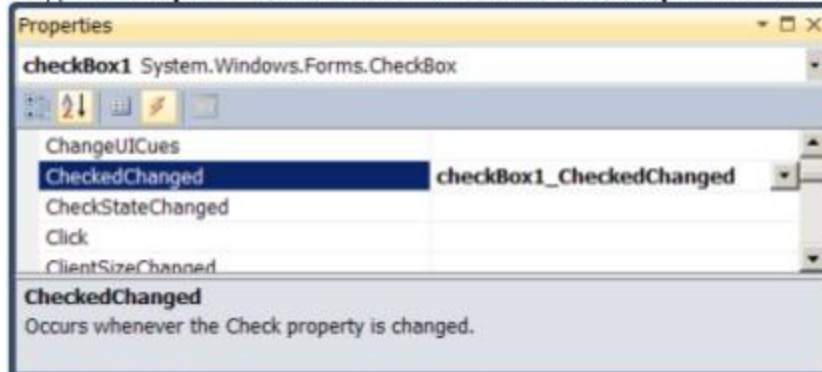
    string st;

    st = "Цена в выбранной комплектации: " + sum.ToString("C");

    if ((checkBox1.Checked) && (checkBox2.Checked) && (checkBox3.Checked))
    {
        //skidka
        skidka = sum * 0.01;
        cena = cena - skidka;
        st += "\nСкидка 1%: " + skidka.ToString("C") + "\nИтого: " + cena.ToString("C");
    }

    label3.Text = st;
}
```

7. Добавить **обработчик события изменения состояния переключателя.**



Окно выбора события для переключателя.

8. Написать код **обработчика события изменения состояния переключателя.**

```
private void checkBox1_CheckedChanged(object sender, EventArgs e)
{
    label3.Text = "";
}
```

9. Запустить приложение кнопкой **Start Debugging** . При необходимости исправить ошибки и запустить приложение повторно.

10. **Закреть** приложение и **сохранить** его на диске в своей папке.

# Язык программирования C#

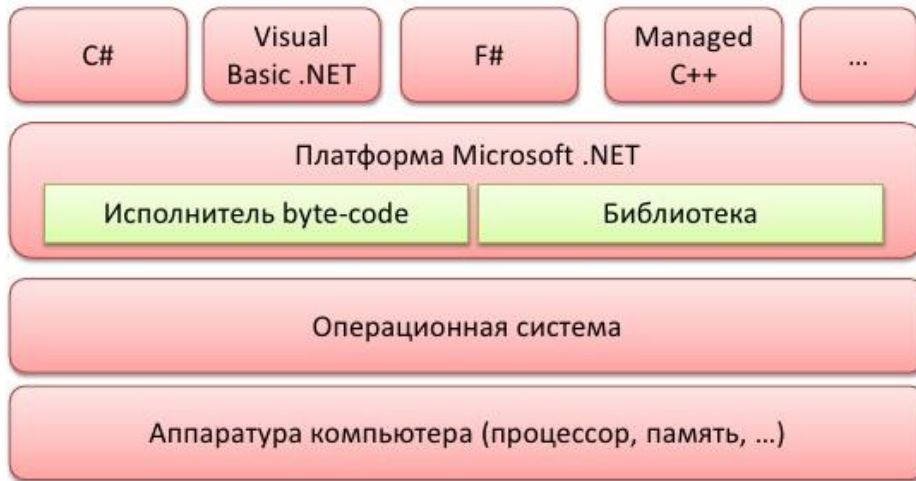
Дмитрий Сошников

[dmitryso@microsoft.com](mailto:dmitryso@microsoft.com)

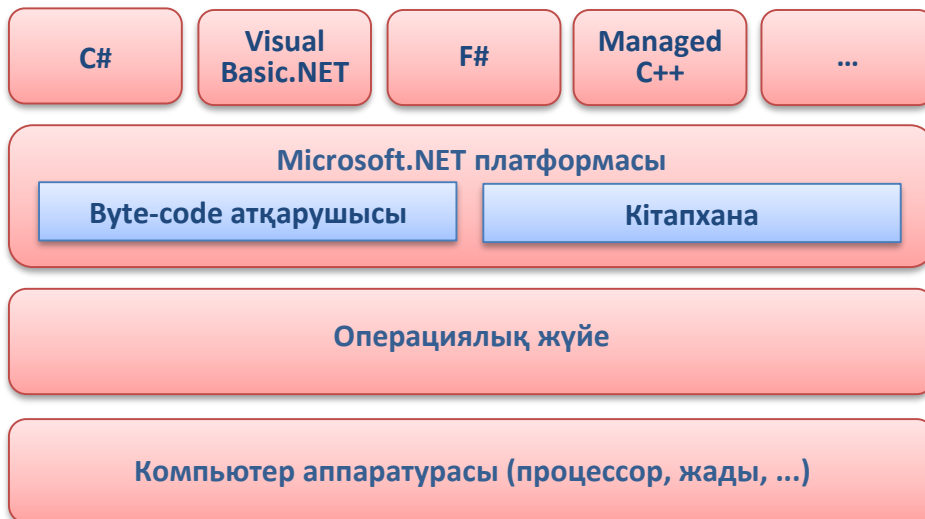




# Платформа .NET



# .NET платформасы



# Переменные и типы данных

- `int` – целый тип
- `float` – вещественный тип
- `double` – вещественный с двойной точностью
- `bool` – логический (значения `true/false`)
- Массивы, ссылки, делегаты, ...
- Объектный
  - `MyObject`, `List<bool>`, ...

# Оператор присваивания

```
using System;

namespace MyApplication
{
    class Program
    {
        static void Main(string[] args)
        {
            float a = float.Parse(args[0]);
            float b = float.Parse(args[1]);
            float c = float.Parse(args[2]);
            var d = b * b - 4 * a * c;
            var x1 = -b + Math.Sqrt(d) / 2 / a;
            var x2 = -b - Math.Sqrt(d) / 2 / a;
            Console.WriteLine("x1={0}, x2={1}", x1, x2);
        }
    }
}
```

# Операторы передачи управления

- Условный оператор if-then-else
- Циклы
  - С предусловием while ... do
  - С постусловием do ... while
  - Со счётчиком for

# Условный оператор

```
using System;

namespace MyApplication
{
    class Program
    {
        static void Main(string[] args)
        {
            float a = float.Parse(args[0]);
            float b = float.Parse(args[1]);
            float c = float.Parse(args[2]);
            var d = b * b - 4 * a * c;
            if (d >= 0)
            {
                var x1 = -b + Math.Sqrt(d) / 2 / a;
                var x2 = -b - Math.Sqrt(d) / 2 / a;
                Console.WriteLine("x1={0}, x2={1}", x1, x2);
            }
            else Console.WriteLine("No solutions");
        }
    }
}
```

# Цикл со счётчиком, функции

```
using System;

namespace ConsoleApplication2
{
    class Program
    {
        static int fact(int x)
        {
            int f = 1;
            for (int i = 1; i <= x; i++)
            {
                f = f * i;
            }
            return f;
        }

        static void Main(string[] args)
        {
            for (var i = 1; i < 10; i++)
            {
                Console.WriteLine("{0}!= {1}", i, fact(i));
            }
            Console.ReadKey();
        }
    }
}
```

# Классы, объекты

```
class Person
{
    public Person(string n) { Name = n; }
    public Person(string n, Person f, Person m)
    {
        Name = n; Father = f; Mother = m;
    }
    public Person Father, Mother;
    public string Name { get; set; }
    public void Introduce()
    {
        Console.WriteLine("I am {0}, my father is {1}, mother is {2}",
            Name, Father.Name, Mother.Name);
    }
}
```

```
var V = new Person("Vasya", new Person("Petya"), new Person("Masha"));
V.Introduce();
```

# Полезные классы в библиотеке

- Списки List<...>
- Словари Dictionary<TKey,TValue>
- Работа с XML, сетевыми подключениями, сжатие данных, криптография, графика, ...

```
var L = new List<Person>();  
  
L.Add(V);  
L.Add(new Person("Jack",V,new Person("Katya")));  
  
L.Sort(new Comparison<Person>((p1, p2) => string.Compare(p1.Name, p2.Name)));  
  
foreach (var x in L) Console.WriteLine(x.Name);  
  
L.ForEach(P => P.Introduce());
```

# C# - не только для консольного программирования!

- Windows Forms
- Windows Presentation Foundation
  
- ASP.NET Web Forms
- ASP.NET MVC
  
- Silverlight
  
- XNA Game Studio (Windows, Xbox, Windows Phone 7, Zune)
  
- Windows Phone 7 (Silverlight, XNA)
- Xbox (XNA)



# Дополнительные ресурсы для изучения

- Центр начинающего разработчика:  
<http://msdn.microsoft.com/ru-ru/beginner/default.aspx>
  - Книга «С# для начинающих» – скачать PDF или читать онлайн <http://msdn.microsoft.com/ru-ru/beginner/ee344863.aspx>
- Интернет-университет Intuit.ru
  - Основы программирования на С#: <http://www.intuit.ru/department/pl/csharp/>
  - Введение в программирование на С# 2.0: <http://www.intuit.ru/department/pl/csharp20/>
- MSDN:
  - Центр Visual C#: <http://msdn.microsoft.com/ru-ru/vcsharp/default.aspx>
  - Интерактивный учебник по С#

## КОНТРОЛЬНАЯ РАБОТА

Ф.И.О. \_\_\_\_\_

1. Что понимается под программами **линейной структуры**?

---

---

2. Запишите полную конструкцию оператора разветвляющейся структуры **если**.

---

---

3. Запишите конструкцию оператора **выбор**.

---

---

4. Дайте определение **цикла**.

---

---

5. Какие три вида **циклов** вы знаете?

---

---

6. Написать программу, которая требует ввести **два числа**, а затем сообщает которое из них делится на **три** без остатка, а которое с остатком.

<hr/> <hr/> <hr/> <hr/> <hr/>	<hr/> <hr/> <hr/> <hr/> <hr/>
-------------------------------	-------------------------------

7. Написать программу, которая требует ввести **фамилию, имя, отчество и дату рождения**, а затем выводит эти данные **5 раз** на экран.

<hr/> <hr/> <hr/> <hr/> <hr/>	<hr/> <hr/> <hr/> <hr/> <hr/>
-------------------------------	-------------------------------

8. Написать программу, вычисляющую значения функции  $y = x^2 + 1$  на отрезке **[1; 100]** с шагом равным **3**.

<hr/> <hr/> <hr/> <hr/> <hr/>	<hr/> <hr/> <hr/> <hr/> <hr/>
-------------------------------	-------------------------------

4 Программа, которая после введенного с клавиатуры числа (в диапазоне от 1 до 999), обозначающего денежную единицу, дописывает слово "рубль" в правильной форме. Например, 12 рублей, 21 рубль и т. д.

Примечание:

0	-лей	10	-лей	20	-лей	30	-лей
1	-ль			21	-ль	31	-ль
2	-ля			22	-ля	32	-ля
3	-ля			23	-ля	33	-ля
4	-ля			24	-ля	34	-ля
5	-лей	15	-лей	25	-лей	35	-лей
6	-лей	16	-лей	26	-лей	36	-лей
7	-лей	17	-лей	27	-лей	37	-лей
8	-лей	18	-лей	28	-лей	38	-лей
9	-лей	19	-лей	29	-лей	39	-лей

100	-лей	110	-лей	120	-лей	130	-лей
101	-ль			121	-ль	131	-ль
102	-ля			122	-ля	132	-ля
103	-ля			123	-ля	133	-ля
104	-ля			124	-ля	134	-ля
105	-лей	115	-лей	125	-лей	135	-лей
106	-лей	116	-лей	126	-лей	136	-лей
107	-лей	117	-лей	127	-лей	137	-лей
108	-лей	118	-лей	128	-лей	138	-лей
109	-лей	119	-лей	129	-лей	139	-лей

Пример работы оператора % (остатка от деления):

0 mod 100 = 0	10 mod 100 = 10	100 mod 100 = 0	110 mod 100 = 10
1 mod 100 = 1	11 mod 100 = 11	101 mod 100 = 1	111 mod 100 = 11
2 mod 100 = 2	12 mod 100 = 12	102 mod 100 = 2	112 mod 100 = 12

```
using System;

class Class
{
    public static void Main()
    {
        Console.WriteLine("Введите число рублей от 1 до 999: ");
        int rub = Convert.ToInt32(Console.ReadLine());

        if ((rub % 100 > 10) & (rub % 100 < 15))
            Console.WriteLine(rub + " рублей.");
        else
        {
            int ostatok = rub % 10;
            switch (ostatok)
            {
                case 1:
                    Console.WriteLine(rub + " рубль.");
                    break;

                case 2:
                case 3:
                case 4:
                    Console.WriteLine(rub + " рубля.");
                    break;

                case 5:
                case 6:
                case 7:
                case 8:
                case 9:
                case 0:
                    Console.WriteLine(rub + " рублей.");
                    break;
            }
        }

        Console.ReadLine();
    }
}
```

5 Написать программу, которая после введенного с клавиатуры числа (в диапазоне от 1 до 99), обозначающего денежную единицу, дописывает слово "копейка" в правильной форме. Например, 5 копеек, 41 копейка и т. д.

**Примечание:**

0	-еек			20	-еек	30	-еек
1	-ка			21	-ка	31	-ка
2	-ки			22	-ки	32	-ки
3	-ки			23	-ки	33	-ки
4	-ки			24	-ки	34	-ки
5	-еек			25	-еек	35	-еек
6	-еек			26	-еек	36	-еек
7	-еек			27	-еек	37	-еек
8	-еек			28	-еек	38	-еек
9	-еек			29	-еек	39	-еек

7 Программа, определяющая, какие из ТРЕХ чисел ПРИНАДЛЕЖАТ отрезку [a; b].  
Изменить программу, т.о. чтобы она определяла, какие из ДВУХ чисел НЕ ПРИНАДЛЕЖАТ отрезку [a; b].

**Примечание:**

Введите два числа (a и b).

Проверьте правильность ввода отрезка (если  $a \geq b$ , то отрезок задан неверно).

Введите три целых числа.

Проверьте, принадлежит ли отрезку первое, затем второе и третье введенное число заданному отрезку.