

Богатов Р.Н.

# Программирование на языке высокого уровня

---

Лекция 1.

Введение в программирование на C#.

Условный оператор. Типы данных. Цикл for.

Кафедра АСОИУ ОмГТУ, 2012

# Что такое программирование?

- Программировать умеют все! Даже моя жена Софья.
- Знаменитая программа Софьи, которую копируют чаще всего:

1. Смешать :
  - 1/3 ст. подсолнечного масла
  - 1/4 ст. воды
  - 1 ст. сахара
2. Добавить 2-3 размятых банана
3. По желанию добавить 1-2 взбитых яйца
4. Отдельно смешать сухими :
  - 2 ст. муки
  - 1 ч.л. соды
  - 1/2 ч.л. соли
  - 1/2 ч.л. корицы
  - ванилин
5. Все смешать и выпекать, пока не покоричневает

(Называется «Банановый хлеб»)

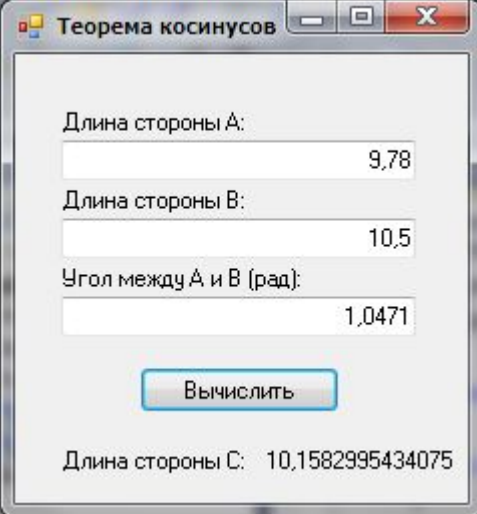
# К делу! Пример программы на C#

```
// исходные данные
double a, b, alpha;
a = 10.5;
b = 9.78;
alpha = Math.PI/3;
```

```
// исходные данные
double a, b, alpha;
a = Convert.ToDouble(textBox1.Text);
b = Convert.ToDouble(textBox2.Text);
alpha = Convert.ToDouble(textBox3.Text);

// вычисления
double c;
c = Math.Sqrt(a*a + b*b - 2*a*b*Math.Cos(alpha));

// вывод результата
label5.Text = Convert.ToString(c);
```



Теорема косинусов

Длина стороны А: 9,78

Длина стороны В: 10,5

Угол между А и В (рад): 1,0471

Вычислить

Длина стороны С: 10,1582995434075

# Условный оператор `if`

```
if (x < 0)
    сделать-то-то;
```

```
if (x < 0)
{
    сделать-одно;
    сделать-второе;
    сделать-что-то-ещё;
}
```

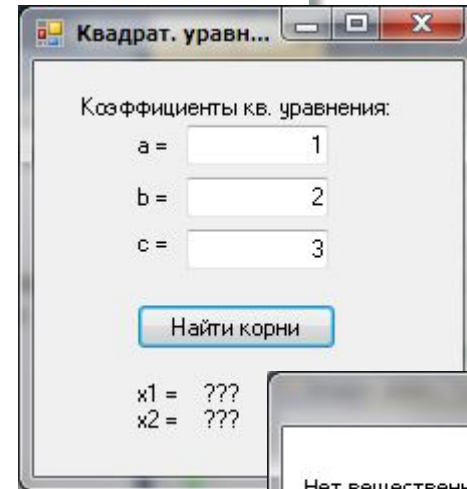
```
if (x < 0)
    сделать-то-то;
else
    сделать-другое;
```

```
if (x < 0)
{
    сделать-одно;
    сделать-второе;
}
else
{
    сделать-другое;
    и-ещё-кое-что;
}
```

# Решение квадратного уравнения

```
// исходные данные
double a, b, c;
a = Convert.ToDouble(textBox1.Text);
b = Convert.ToDouble(textBox2.Text);
c = Convert.ToDouble(textBox3.Text);

// вычисления
double D = b*b - 4*a*c;
if (D < 0)
{
    // вывод результата
    label6.Text = "???" ;
    label8.Text = "???" ;
    MessageBox.Show("Нет вещественных корней!" );
}
else
{
    double x1, x2;
    x1 = (-b - Math.Sqrt(D))/2/a;
    x2 = (-b + Math.Sqrt(D))/2/a;
    // вывод результата
    label6.Text = Convert.ToString(x1);
    label8.Text = Convert.ToString(x2);
}
}
```



Квадрат. уравн...

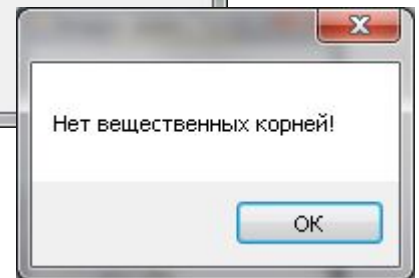
Коэффициенты кв. уравнения:

a =

b =

c =

x1 = ???  
x2 = ???



# Встроенные типы данных C#

Тип	Диапазон значений	Размер (байт)
sbyte	-128 ... 127	1
byte	0 ... 255	1
short	-32 768 ... 32 767	2
ushort	0 ... 65 535	2
int	-2 147 483 648 ... 2 147 483 647	4
uint	0 ... 4 294 967 295	4
long	-9 223 372 036 854 775 808 ... 9 223 372 036 854 775 807	8
ulong	0 ... 18 446 744 073 709 551 615	8
float	$\pm 1,5 \cdot 10^{-45} \dots \pm 3,4 \cdot 10^{+38}$ , точность – 7 разрядов	4
double	$\pm 5,0 \cdot 10^{-324} \dots \pm 1,7 \cdot 10^{+308}$ , точность – 16 разрядов	8
decimal	$\pm 1 \cdot 10^{-28} \dots \pm 7,9 \cdot 10^{+28}$ , точность – 28 разрядов	16
bool	true или false	1
char	0 ... 65 535 (код символа)	2

# Цикл с заданным числом повторений

```
// какой из циклов будет работать дольше?
```

```
// какой из циклов будет работать дольше?
```

```
// n - требуемое количество точек
```

```
for (int i = 1; i <= n; i++)
```

```
{
```

```
    textBox1.Text = textBox1.Text + "Итерация " + i + "\r\n";
```

```
}
```

```
for
```

```
{
```

```
    что-то-там;
```

```
    и ещё что-то;
```

```
    и ещё;
```

```
}
```

```
    что-то-там;
```

```
ещё;
```

```
    что-то-там;
```

```
ещё что-то;
```

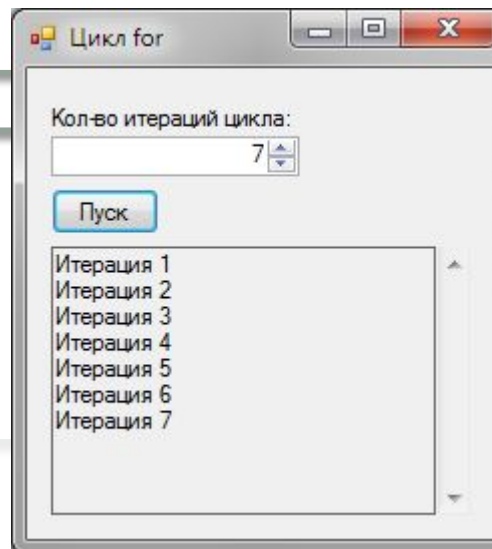
```
ещё;
```

```
    что-то-там;
```

```
ещё что-то;
```

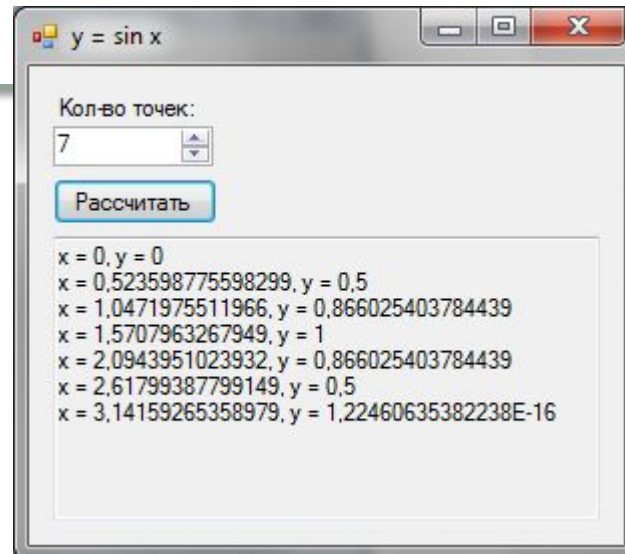
```
ещё;
```

} N раз



# Задачи на циклы: табуляция функции

```
double x, y;  
  
// n - требуемое количество точек  
for (int i = 0; i < n; i++)  
{  
    x = Math.PI * i / (n-1);  
    y = Math.Sin(x);  
    textBox1.Text = textBox1.Text + "x = " + x + ", y = " + y + "\r\n";  
}
```





# Задачи на циклы: поиск минимума/максимума

```
double x, y;
double ymax = double.NegativeInfinity, xbest = double.NaN;

// n - требуемое количество точек
for (int i = 0; i < n; i++)
{
    x = -0.5 + 2.0*i/(n-1); // всегда в интервале [-0.5; 1.5]
    y = x*x*x - x*x - x;
    textBox1.Text = textBox1.Text + "x = " + x + ", y = " + y + "\r\n";
    if (y > ymax)
    {
        ymax = y;
        xbest = x;
    }
}

label3.Text = Convert.ToString(ymax);
label5.Text = Convert.ToString(xbest);
```

```
x = 0,8333333333333333, y = -0,949074074074074
x = 1, y = -1
x = 1,166666666666667, y = -0,939814814814815
x = 1,333333333333333, y = -0,740740740740741
x = 1,5, y = -0,375
```

ymax = 0,185185185185185

y =  $x^3 - x^2 - x$

Кол-во точек:

```
x = -0,166666666666667, y = 0,134259259259259
x = 0, y = 0
x = 0,166666666666667, y = -0,189814814814815
x = 0,333333333333333, y = -0,407407407407407
x = 0,5, y = -0,625
x = 0,666666666666667, y = -0,814814814814815
x = 0,833333333333333, y = -0,949074074074074
x = 1, y = -1
x = 1,166666666666667, y = -0,939814814814815
x = 1,333333333333333, y = -0,740740740740741
x = 1,5, y = -0,375
```

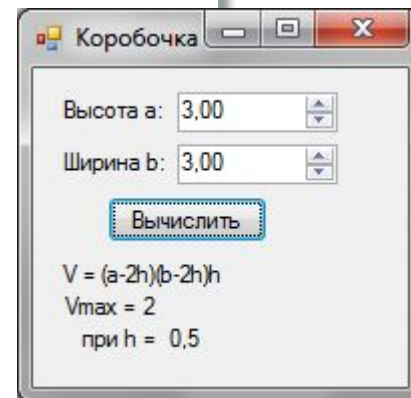
ymax = 0,185185185185185 при x = -0,333333333333333

# Задача про коробочку. Численное решение

У листа бумаги  $a \times b$  отрезают по углам четыре квадрата со стороной  $h$ . Полученную крестообразную выкройку сворачивают в коробочку (параллелепипед).

При каком  $h$  коробочка будет иметь максимальный объём?

```
double dh = a/2/9999;  
double V, h, Vmax = 0, hbest = 0;  
for (int i = 0; i < 10000; i++)  
{  
    h = dh * i;  
    V = (a-2*h) * (b-2*h) * h;  
    if (V > Vmax)  
    {  
        Vmax = V;  
        hbest = h;  
    }  
}  
  
label3.Text = Convert.ToString(Vmax);  
label5.Text = Convert.ToString(hbest);
```



## Домашнее задание

Вычислить сумму ряда  $1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!}$  для заданного  $n$  и сравнить полученный результат с числом Эйлера.

# Бонус

**Самостоятельно написать на C# какую-нибудь программу и выслать исполнимый модуль на адрес [bogatovrn@asoiu.com](mailto:bogatovrn@asoiu.com)**

Требования:

- программа должна отличаться от задач, разбираемых на лекциях или практических занятиях и заданных на дом
- программа должна сообщать ФИО и группу автора