

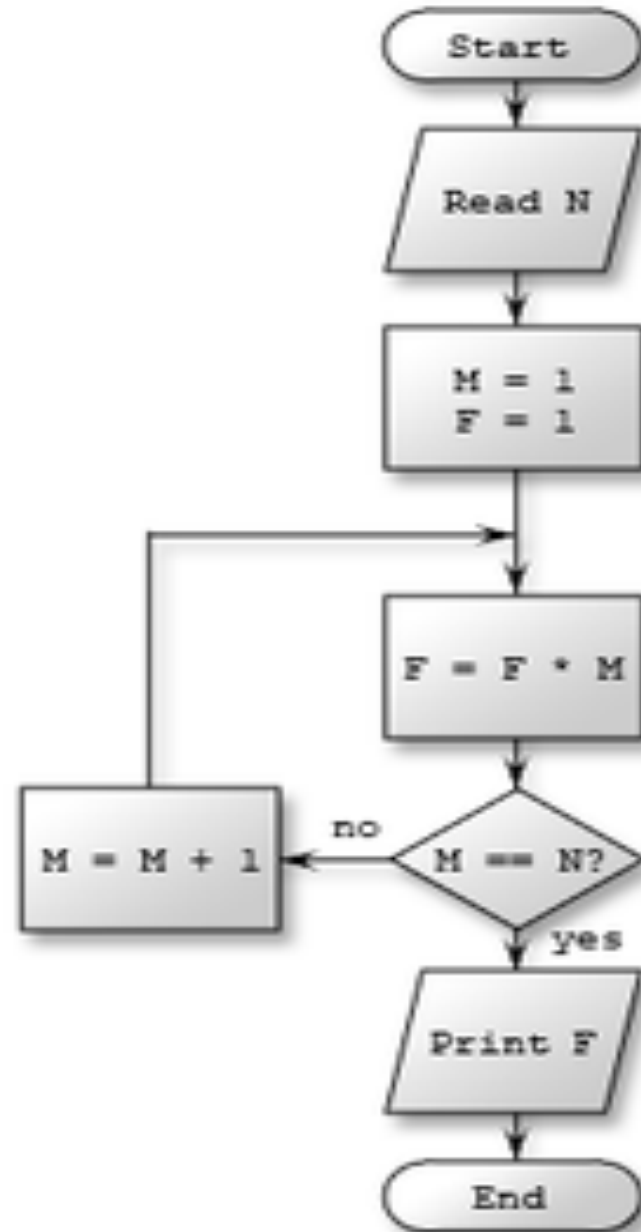
□ Что такое алгоритмы?

□ Какие виды алгоритмов вы знаете?

□ Для чего необходимы блок-схемы?

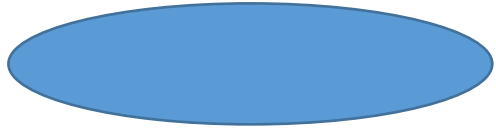


Блок схема?

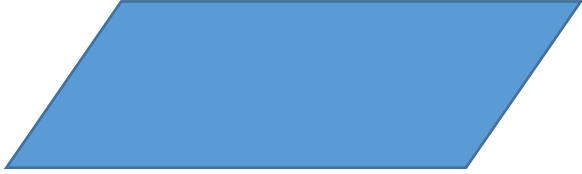


**Блок - схема** — графическое представление алгоритма.





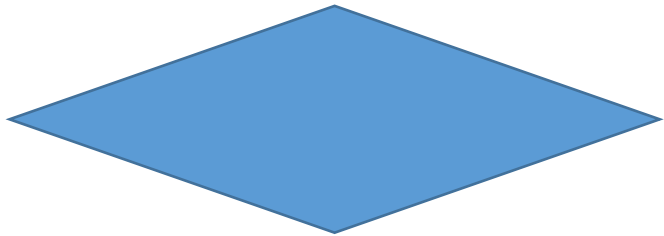
Блок начала и конца алгоритма



Блок ввода и вывода  
информации



Блок  
действия



Блок выбора или  
ветвления

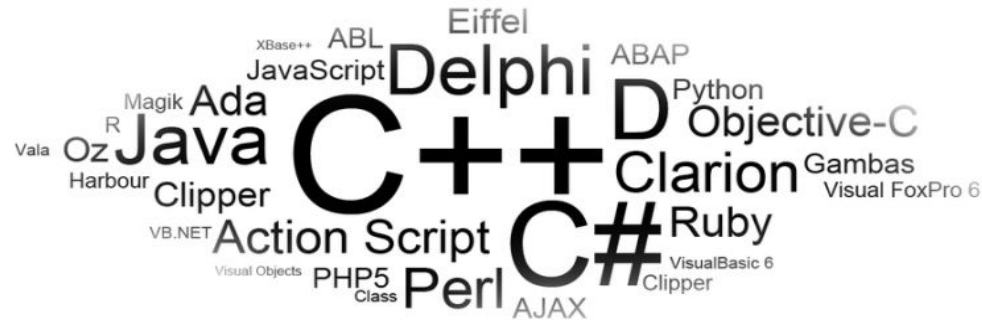
# ВИДЫ АЛГОРИТМОВ





- Линейный
- Разветвляющийся
- Циклический

# Введение в объектно-ориентированный язык программирования





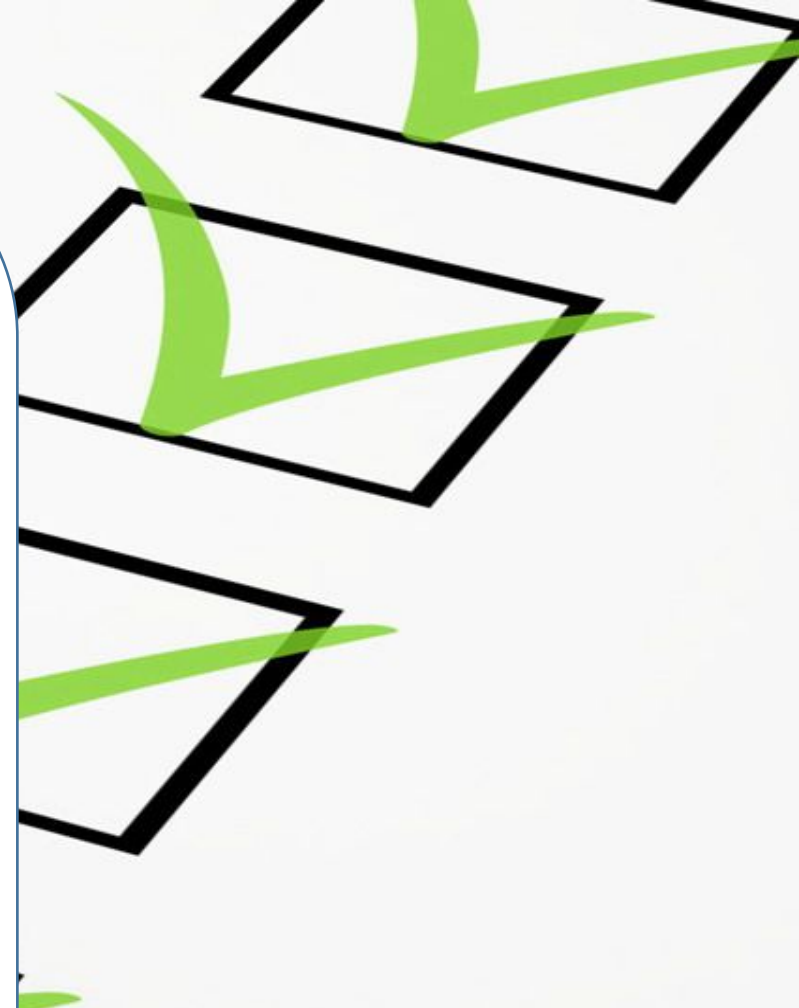
## Цель урока:

9.4.3.2 использовать объектно-ориентированный язык программирования для решения простых задач

9.2.4.1 использовать различные типы данных в ООП

## Критерии успеха:

- ✓ разрабатывает алгоритм решения предоставленной задачи как минимум двумя способами
- ✓ сравнивает эффективность разработанных алгоритмов и выбирает оптимальный вариант
- ✓ переводит один из разработанных алгоритмов в код языка программирования



# Vocabulary

Russian	English
Объект	
Блок схема	
Ориентированный	
Решение	
Задача	
Эффективность	
Оптимальный	

**Объектно-ориентированное программирование** (в дальнейшем ООП) — вид программирования, в которой основными концепциями являются понятия **объектов и классов**.

**Объект** — это сущность, экземпляр класса, которой можно посылать сообщения, и которая может на них реагировать, используя свои данные.

**Класс** — это набор объектов или методов, обладающих одинаковыми свойствами и поведением

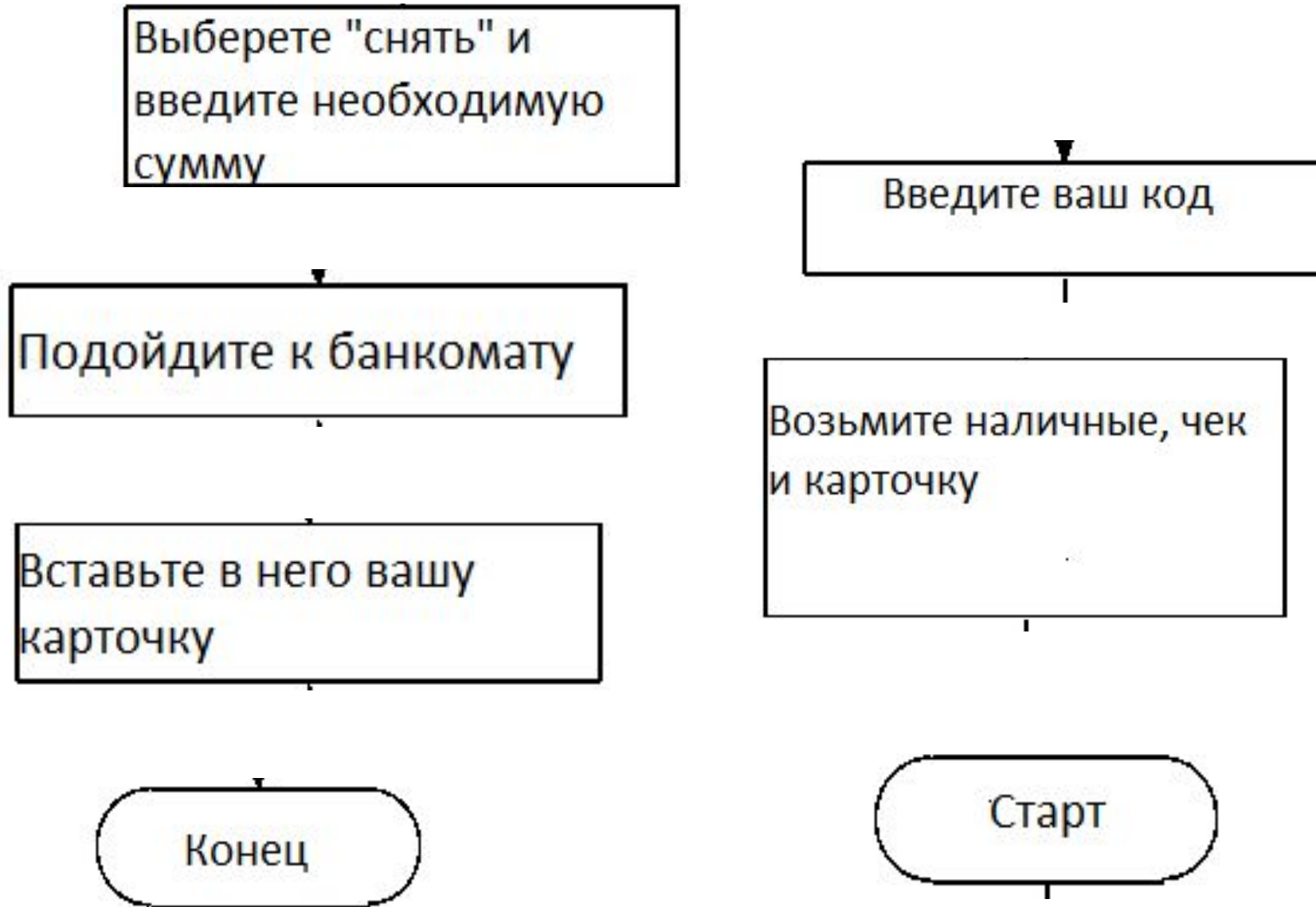
**Свойства** – это характеристики объекта.

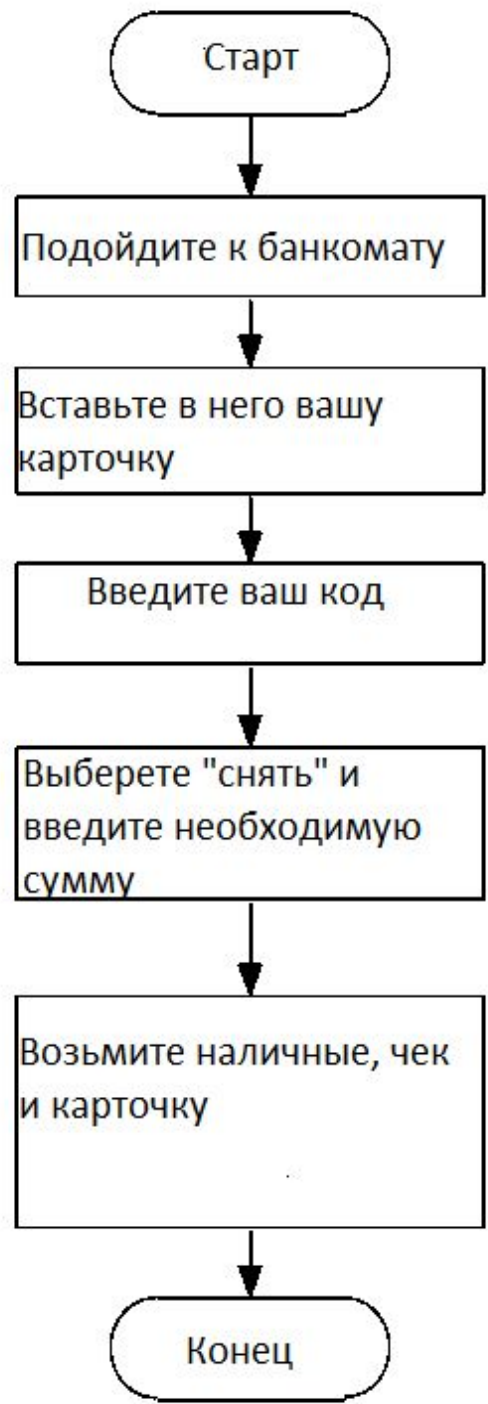
**События** – это то, что происходит в реальном мире и определяет поведение объекта.

**Методы** – это способы реакции объекта на события.

## Работа в группе (2 мин)

**Задание: Соберите блок-схему по порядку**





**Составьте самостоятельно в  
тетрадах блок-схему перехода  
дороги (3 мин)**

# Один из вариантов блок-схемы перехода дороги





**Обсудите в паре устно.  
Как можно сопоставить части  
алгоритма по приготовлению  
торта:**

**Рецепт**

**ВВОД**

**Ингредиент  
ы**

**пошаговая процедура  
или список  
инструкций по  
выполнению работы**

**Духовка и  
посуда**

**ВЫХОД**

**Торт**

**аппаратное  
обеспечение**

**Укажите соответствие частей (2  
минуты)**

**Ответ**

**Рецепт**

пошаговая процедура  
или список  
инструкций по  
выполнению работы

**Ингредиенты**

**ВВОД**

**Духовка и  
посуда**

аппаратное  
обеспечение

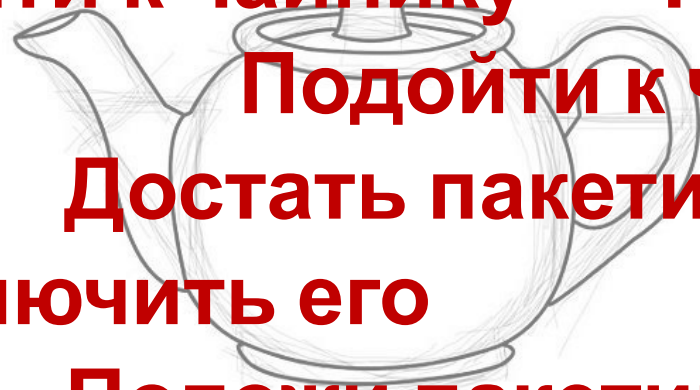
**Торт**

**ВЫХОД**

## Работа в группе (3 минуты)

Задание: Соберите алгоритм «Чай» по порядку

Подойти к чайнику      Подойти к буфету



Подойти к чайнику

Достать пакетики с чаем

Включить его

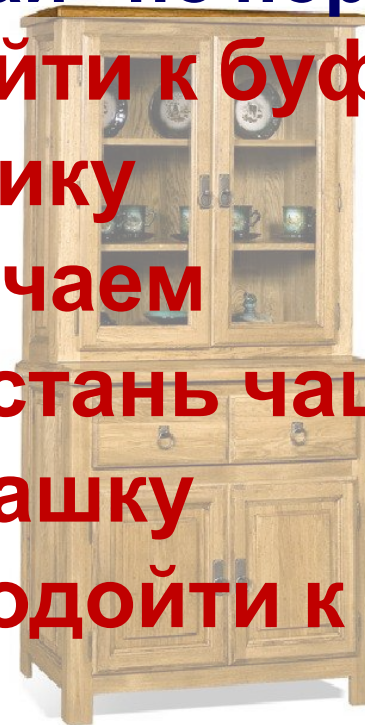
Достань чашку

Положи пакетик в чашку

Налей воды в чашку

Подойти к

буфету



© ВИЛЕЙСКАЯ МЕБЕЛЬНАЯ ФАБРИКА

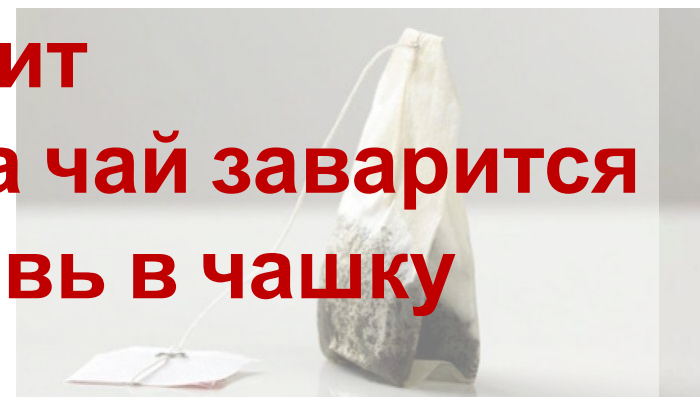
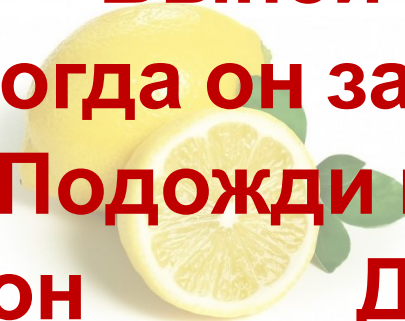
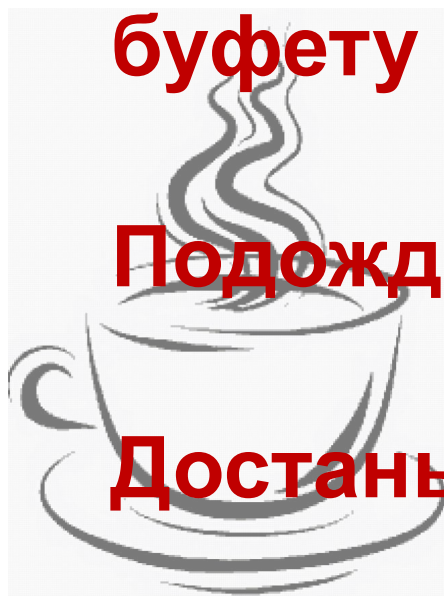
Выпей чай

Подождать, когда он закипит

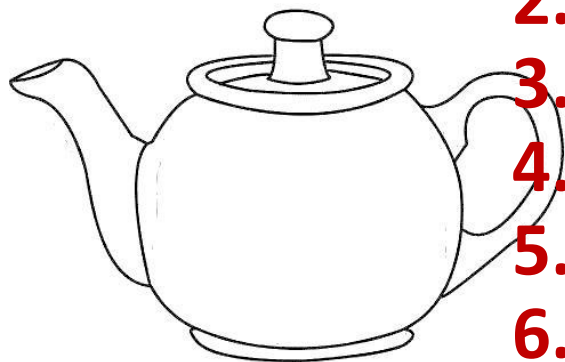
Подожди пока чай заварится

Достань лимон

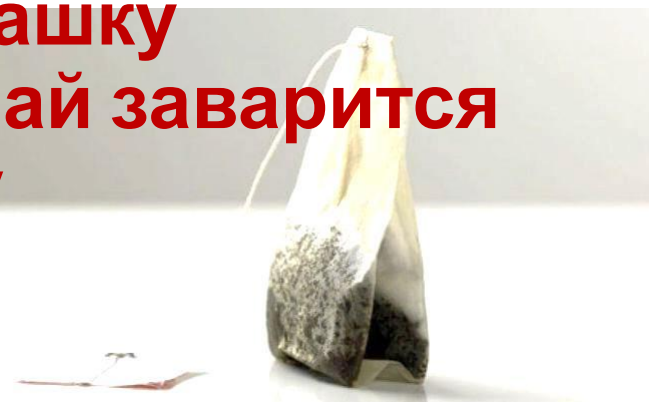
Добавь в чашку



## Один из вариантов сбора алгоритма «Чай»:



1. Подойти к буфету
2. Достать пакетики с чаем
3. Подойти к чайнику
4. Включить его
5. Подождать, когда он закипит
6. Подойти к буфету
7. Достань чашку
8. Достань лимон
9. Подойти к чайнику
10. Положи пакетик в чашку
11. Налей воды в чашку
12. Подожди пока чай заварится
13. Добавь в чашку
14. Выпей чай.



**Задание (2 мин). Напишите два способа математического алгоритма нахождения периметра прямоугольника, где**  
 **$P$ -периметр,  $L$  –длина,  $B$  - ширина:**

**Решение:**

$$P = ???$$

**Или**

$$P = ???$$



**Задание. Напишите два  
способа математического  
алгоритма нахождения  
периметра прямоугольника,  
где  
Р-периметр, L –длина, В -  
ширина:**

$$P = L+L+B+B = 2L + 2B$$

**Или (Or)**

$$P = 2(L+B)$$

Какой способ эффективней?

$$P = L+L+V+V = 2L + 2V$$

Или (Or)

$$P = 2(L+V)$$

# Какие можно сделать выводы?

1. У одной задачи может быть \_\_\_\_\_ алгоритмических решений.
2. Некоторые решения могут быть \_\_\_\_\_, чем другие.
3. В чем причина того, что одно решение может быть лучше (например, более эффективно), а другое может быть хуже?



**Какие требования можно выставить к любому алгоритму?**

# Требования к любому алгоритму:

- **Максимальная скорость выполнения алгоритма.**
- **Минимальный объем памяти**
- **Проста и понятность, что позволяет легче отлаживать программу.**

**Алгоритм должен использовать только те команды, которые понятны исполнителю. При этом про любой алгоритм можно сказать, что:**

- алгоритм получает на вход данные (в виде - цифры или слова);
- алгоритм обрабатывает полученные данные по шагам, вычисляя на каждом шаге промежуточные данные. Этот процесс может быть конечным и бесконечным.
- Если процесс работы алгоритма конечен, то результат работы алгоритма – это данные, вычисленные на последнем шаге.
- если процесс работы алгоритма бесконечен (т. е. , зациклился), то результат его работы - не определён.

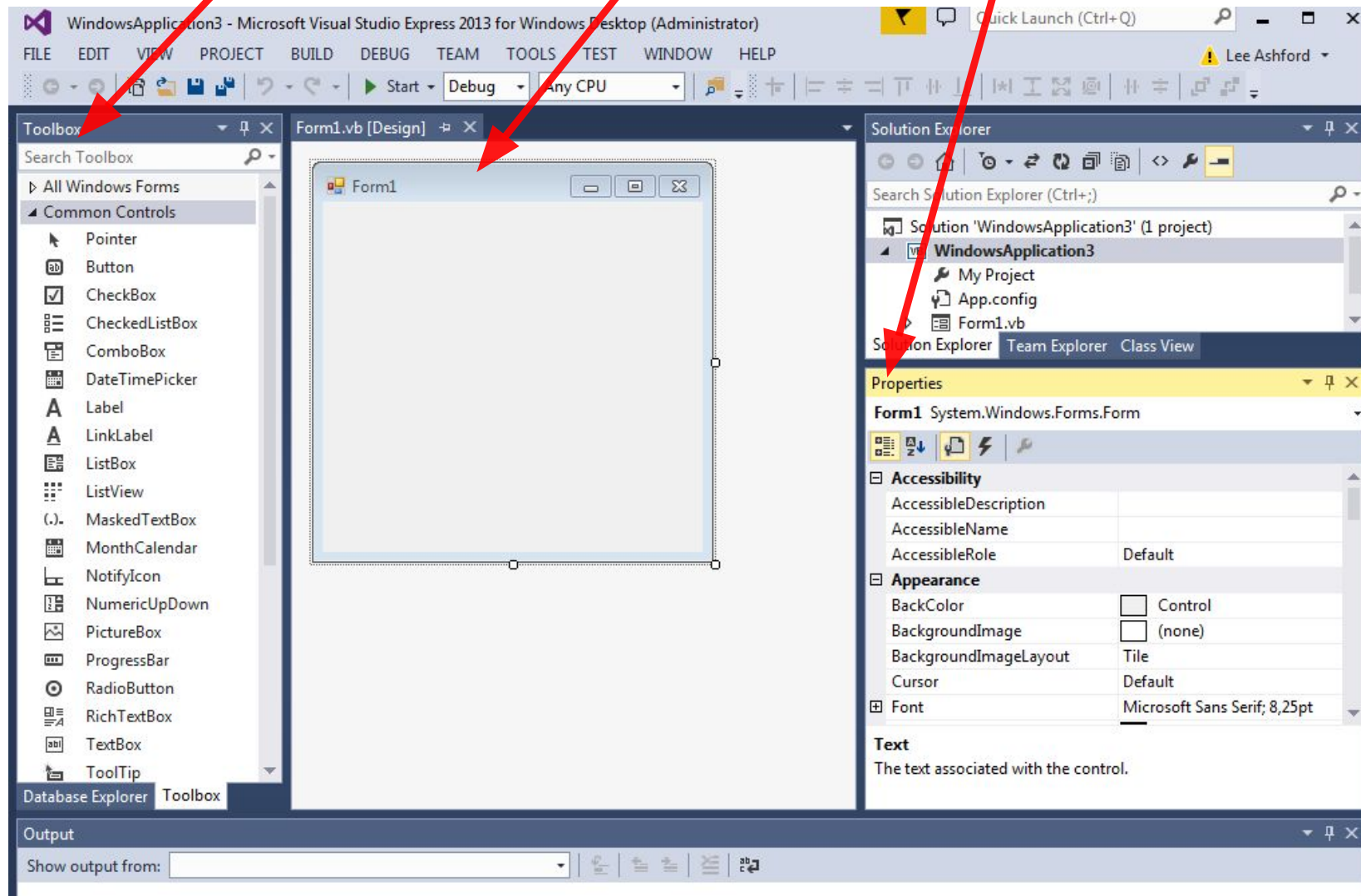
# Интегрированная среда разработки



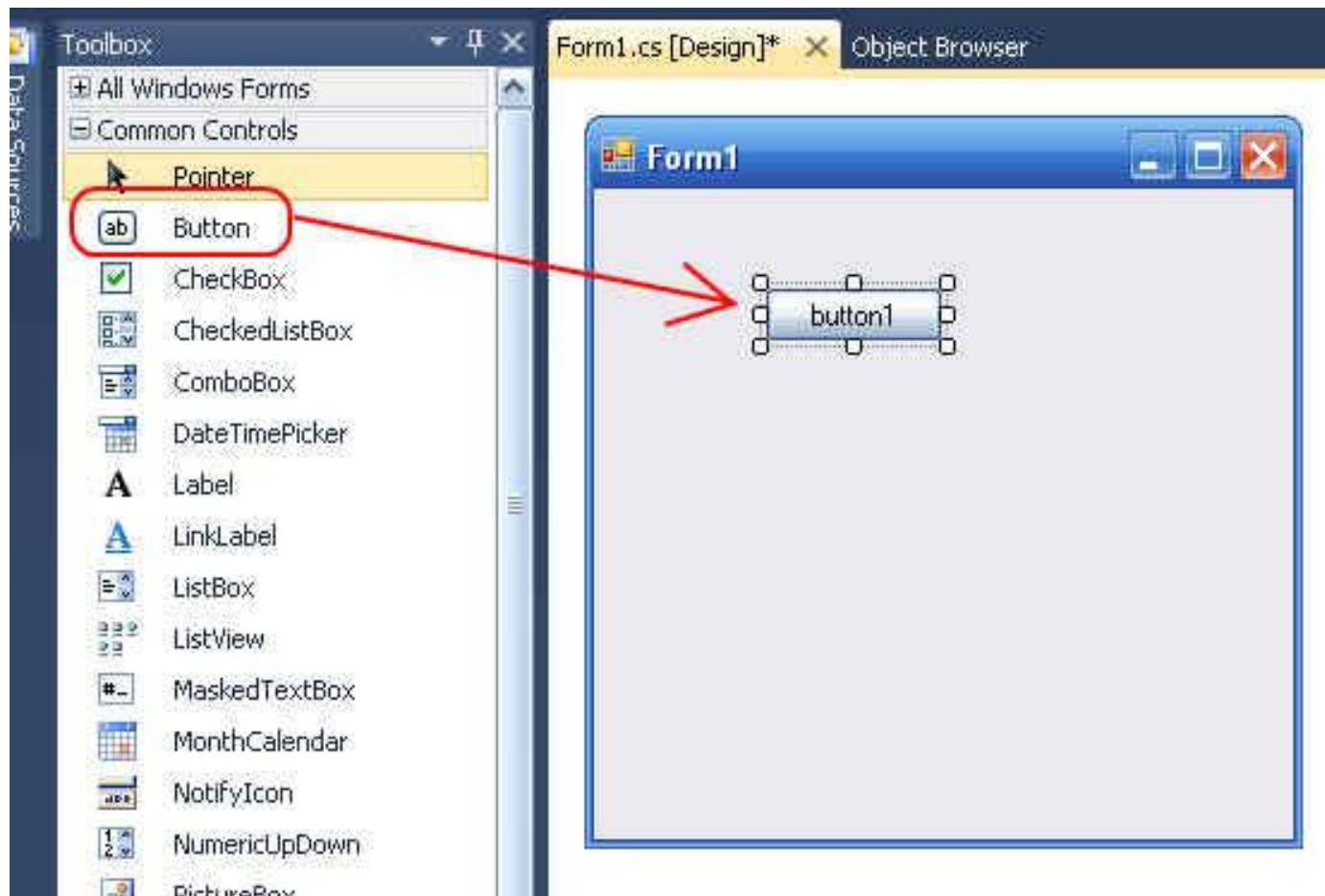
Visual Basic

Панель инструментов  
Форма

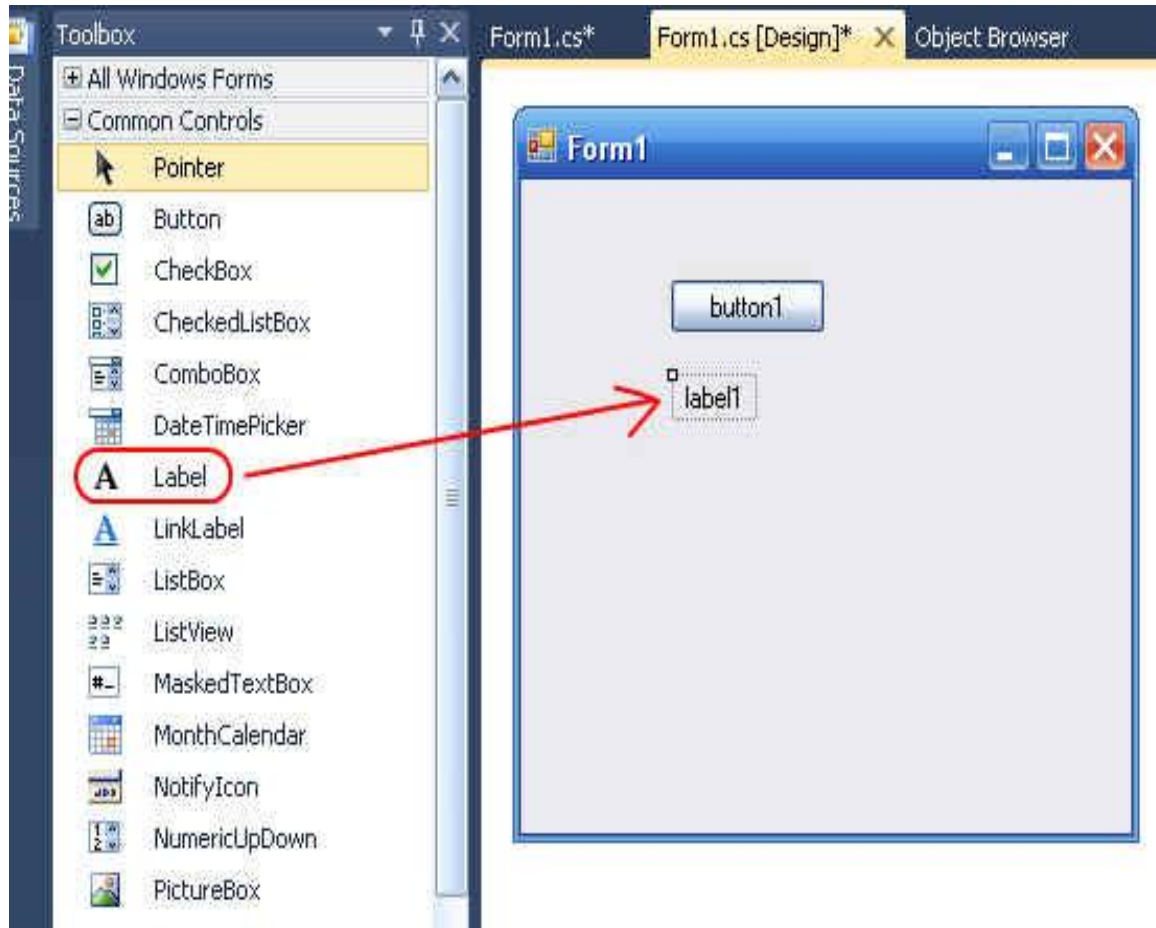
Свойства  
компонента



# «Button» («кнопка»)



# «Label» («надпись»)



- После нажатия кнопки, у нас откроется окно редактирования

кода:

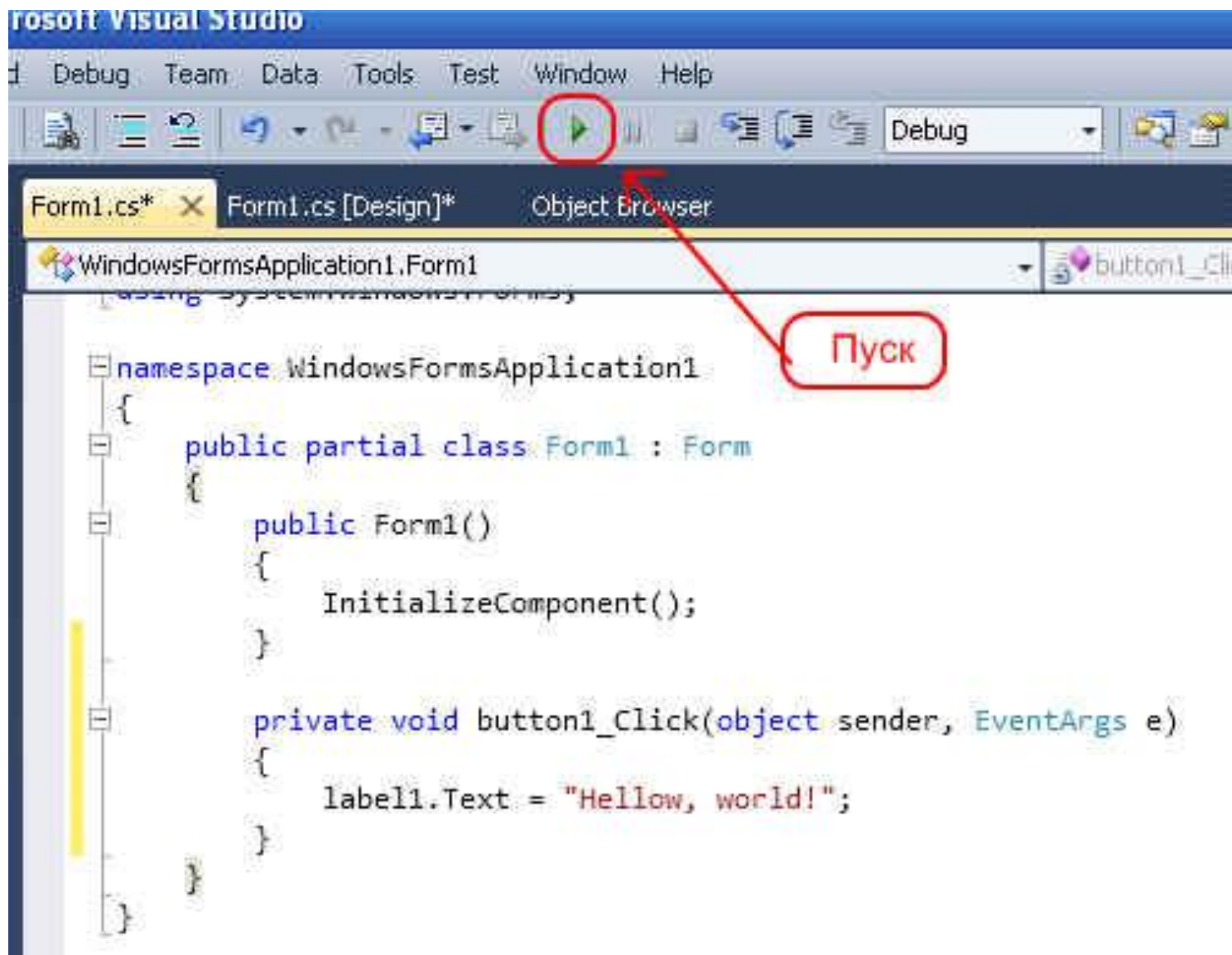


```
Form1.cs* X Form1.cs [Design]* Object Browser
WindowsFormsApplication1.Form1 button1_Click
namespace WindowsFormsApplication1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            |
        }
    }
}
```



- введем вот такой код и запустим программу:



Наша программа скомпилируется и запустится:



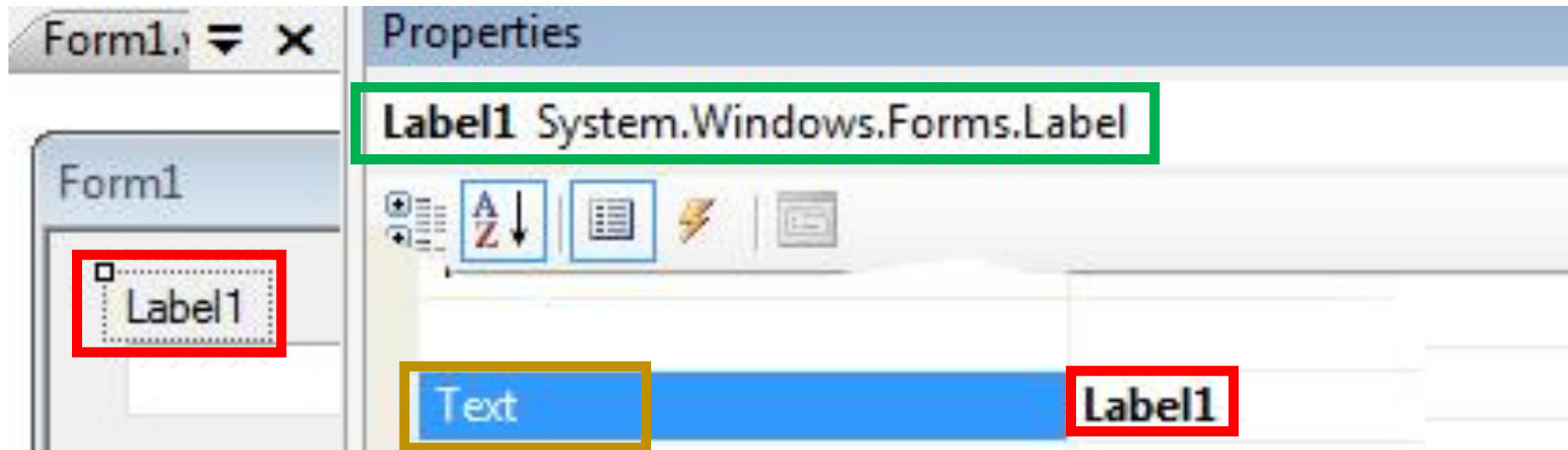
Нажмем на Button1 и увидим надпись "Hello, world!":



# Каждый объект Visual Basic имеет свойство и значение свойства

Объект . Свойство = Значение

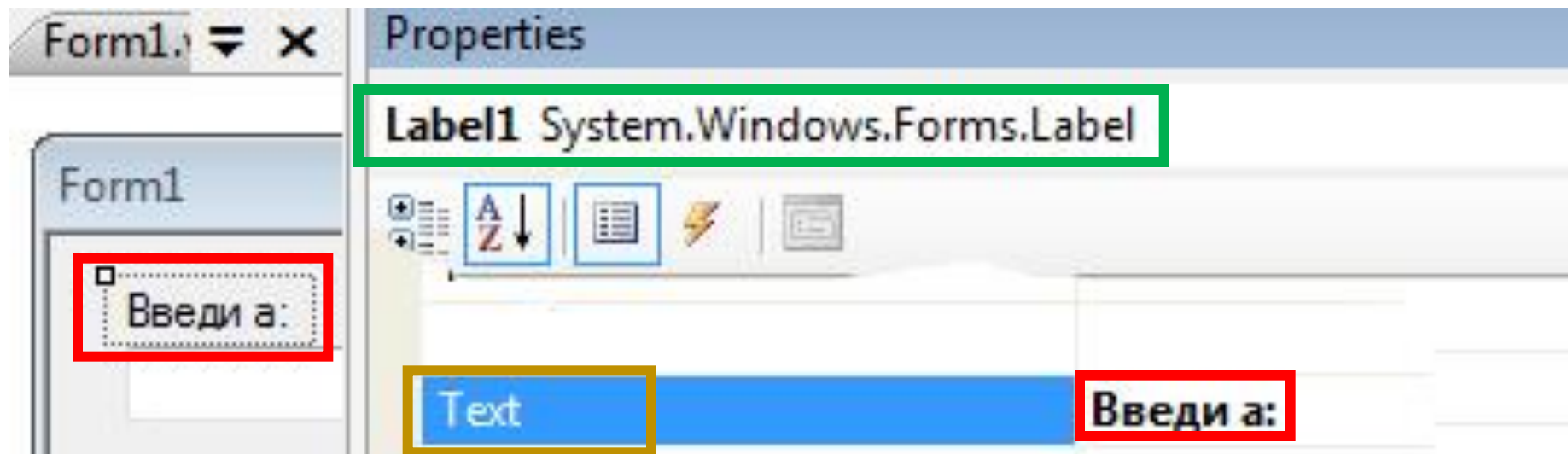
Label1 . Text = "Label1"



# Изменяем значение свойства объекта Label1

Объект . Свойство = Значение

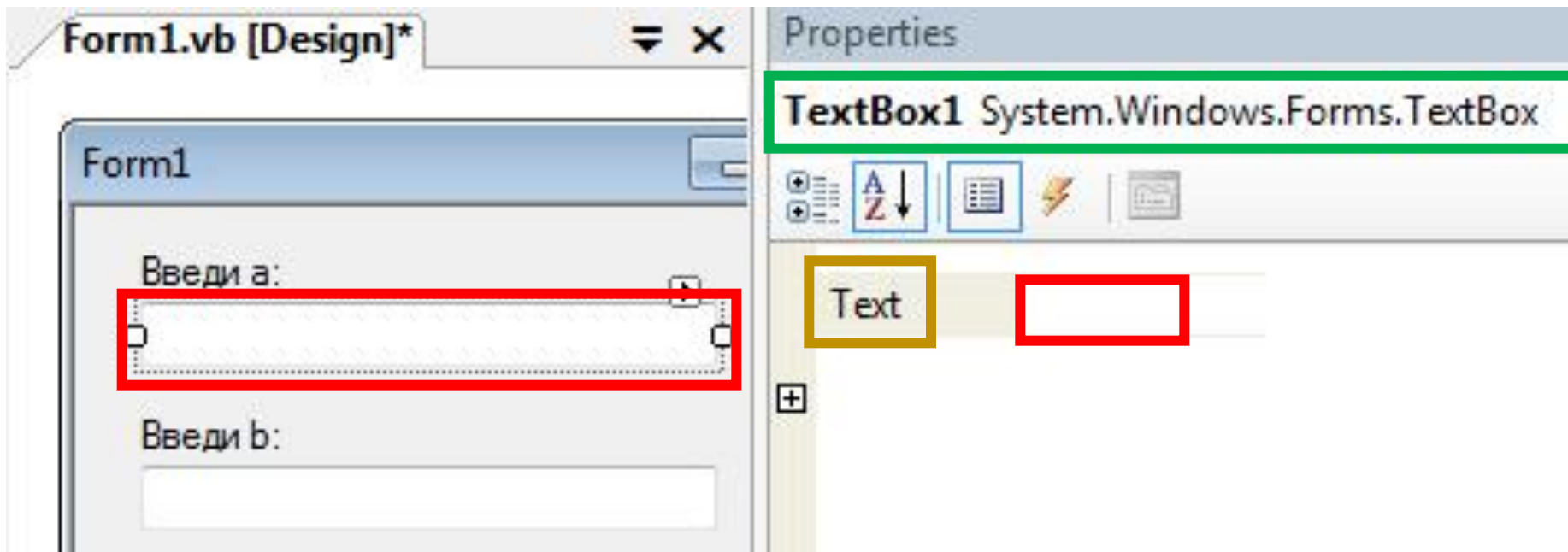
Label1 . Text = "Введи а:"



# Объект «TextBox1» Visual Basic имеет свойство и значение свойства

Объект . Свойство = Значение

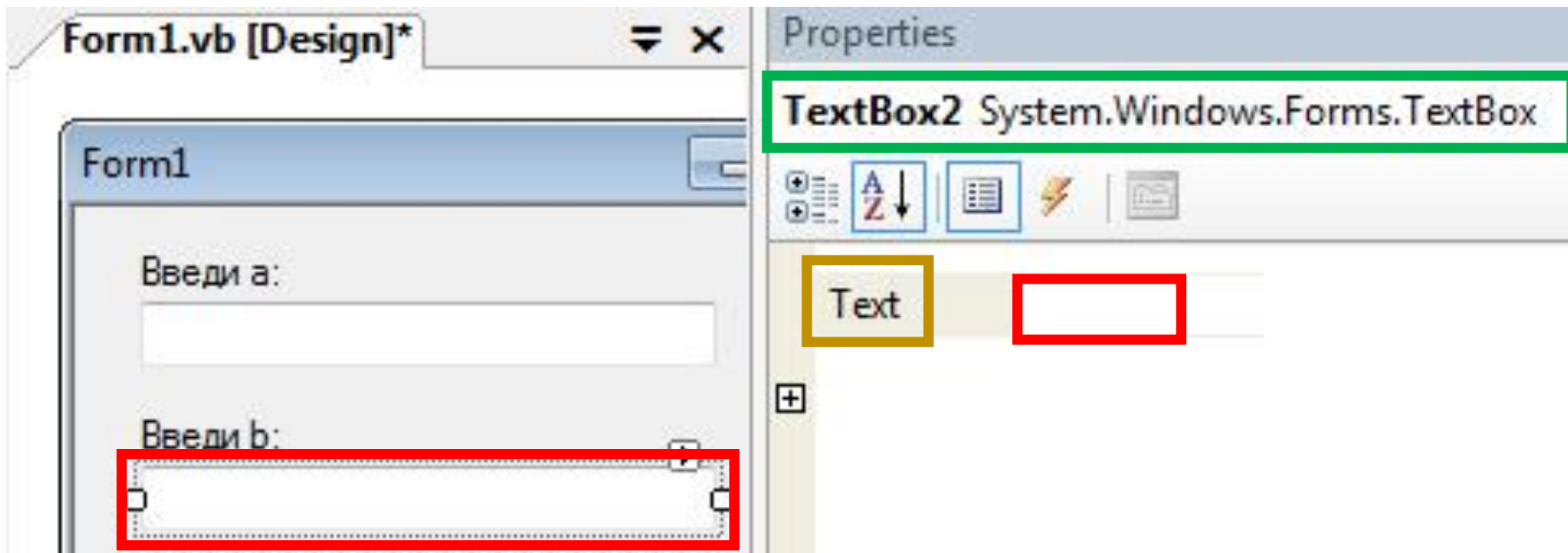
TextBox1 . Text = ""



# Объект «TextBox2» Visual Basic имеет свойство и значение свойства

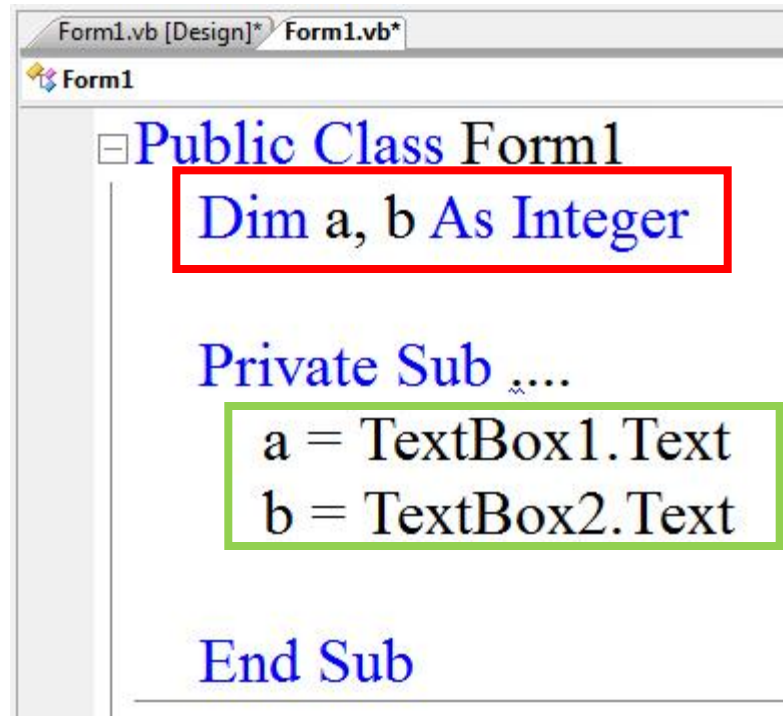
Объект . Свойство = Значение

TextBox2 . Text = ""



**Объявление 2 переменных **a** и **b**  
как целочисленные типа Integer.**

**Присваивание 2 объектам  
«TextBox1» и «TextBox2»  
2 переменных **a** и **b**.**



```
Form1.vb [Design]* Form1.vb*
Form1
Public Class Form1
    Dim a, b As Integer

    Private Sub ...
        a = TextBox1.Text
        b = TextBox2.Text
    End Sub
```

# Сравнение между Pascal и Visual Basic

Label1

напоминает нас об операторе  
вывода подсказки

**WRITE('...');**

напоминает нас  
об операторе

**READLN(...);**

ввода с  
клавиатуры



# Сравнение между Pascal и Visual Basic

Введи а:

это объект «**TextBox1**»

где:

```
Dim a, b As Integer
```

```
a = TextBox1.Text
```

```
WRITE('Введи а:');
```

```
READLN(a);
```

# Сравнение между Pascal и Visual Basic

Введи b:

это объект «**TextBox2**»

где:  
Dim a, b As Integer  
b = TextBox2.Text

```
WRITE('Введи b:');  
READLN(b);
```

# Достигли ли мы цели урока?

## Цель урока:

Разрабатывать различные алгоритмы решения одной и той же задачи



# Рефлексия

Не удалось  
разобраться  
в теме

Остались  
вопросы

Тема  
раскрыта,  
все  
понятно

# Домашнее задание

- У меня в жизни есть  
3 принципа
- Наследование,  
инкапсуляция  
и полиморфизм?

