

Программа

Программа – это

- алгоритм, записанный на каком-либо языке программирования
- набор команд для компьютера

Команда – это описание действий, которые должен выполнить компьютер.

- откуда взять исходные данные?
- что нужно с ними сделать?

Оператор – это команда языка программирования высокого уровня.

1970 – язык Паскаль (Н. Вирт)

Простейшая программа

название программы

```
program qq;  
begin { начало программы }  
end.  { конец программы }
```

комментарии в фигурных скобках
не обрабатываются



Что делает эта программа?

Вывод текста на экран

```
program qq;  
begin  
▶ write('2+');  
▶ writeln('2=?'); { на новую строку}  
▶ writeln('Ответ: 4');  
end.
```

Протокол:

2+

Ответ: 4

Задания

«4»: Вывести на экран текст "лесенкой"

Вася

пошел

гулять

«5»: Вывести на экран рисунок из букв

Ж

ЖЖЖ

ЖЖЖЖЖ

ЖЖЖЖЖЖЖ

НН НН

ZZZZZ

Переменные

Задача. Ввести с клавиатуры два числа и найти их сумму.

Протокол:

Введите два целых числа

компьютер

25 30

пользователь

$25+30=55$

компьютер считает сам!



1. Как ввести числа в память?
2. Где хранить введенные числа?
3. Как вычислить?
4. Как вывести результат?

Программа

```
program qq;  
begin  
  { ввести два числа }  
  { вычислить их сумму }  
  { вывести сумму на экран }  
end.
```

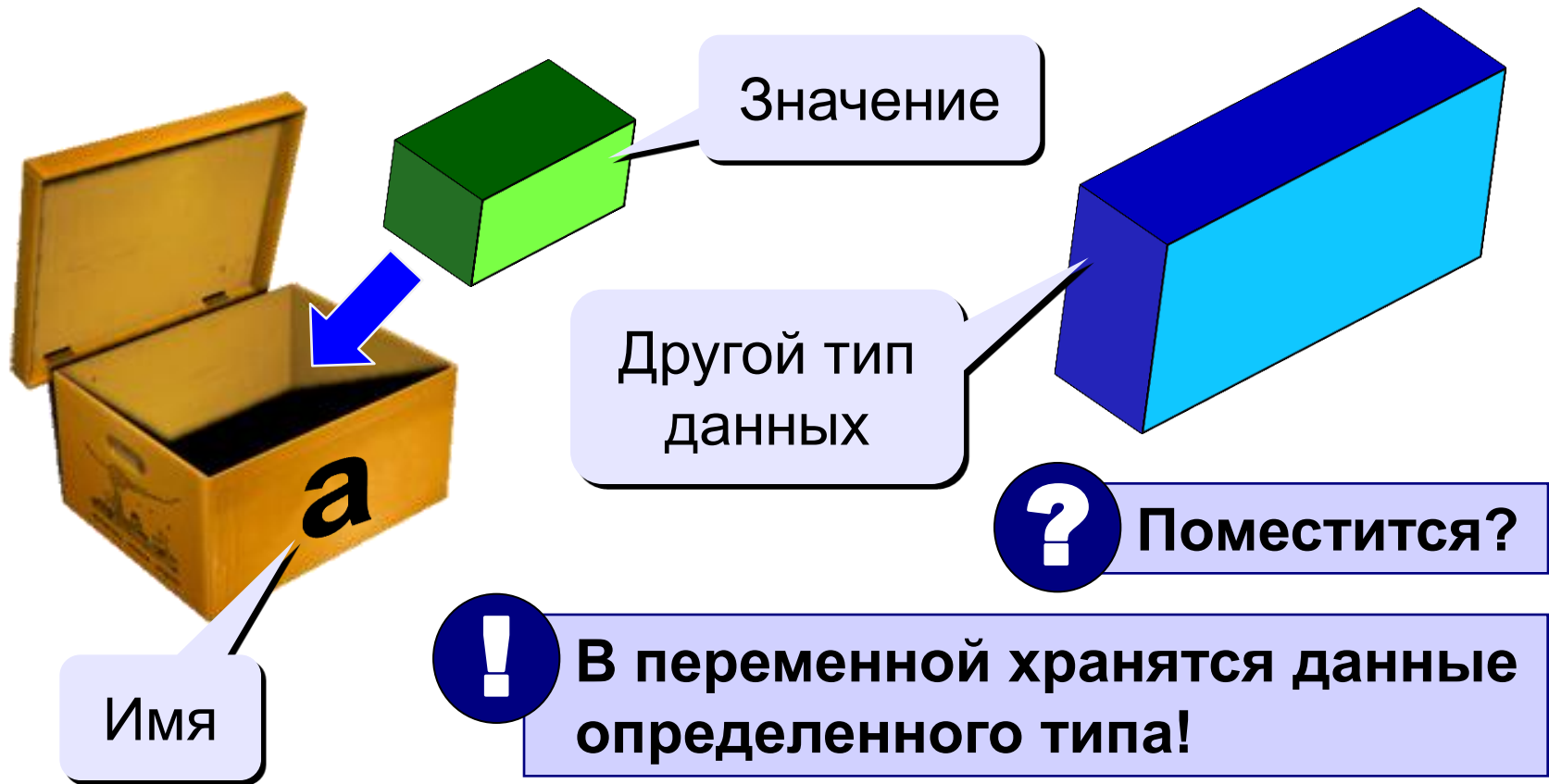
Псевдокод: алгоритм на русском языке с элементами Паскаля.



Компьютер не может исполнить псевдокод!

Переменные

Переменная – это величина, имеющая имя, тип и значение. Значение переменной можно изменять во время работы программы.



Имена переменных

В именах **МОЖНО** использовать

- латинские буквы (A-Z)

заглавные и строчные буквы не различаются

- цифры

имя не может начинаться с цифры

- знак подчеркивания _

В именах **НЕЛЬЗЯ** использовать

- русские буквы
- пробелы
- скобки, знаки +, =, !, ? и др.

Какие имена правильные??

AXby R&B 4Wheel Вася "PesBarbos"
TU154 [QuQu] _ABBA A+B

Переменные

Типы переменных:

- integer { целая }
- real { вещественная }
- и другие...

Объявление переменных:

Выделение
места в памяти

variable – переменная

тип – целые

```
var a, b, c: integer;
```

СПИСОК ИМЕН
переменных

Как записать значение в переменную?

Оператор
присваивания

```
a := 5;
```

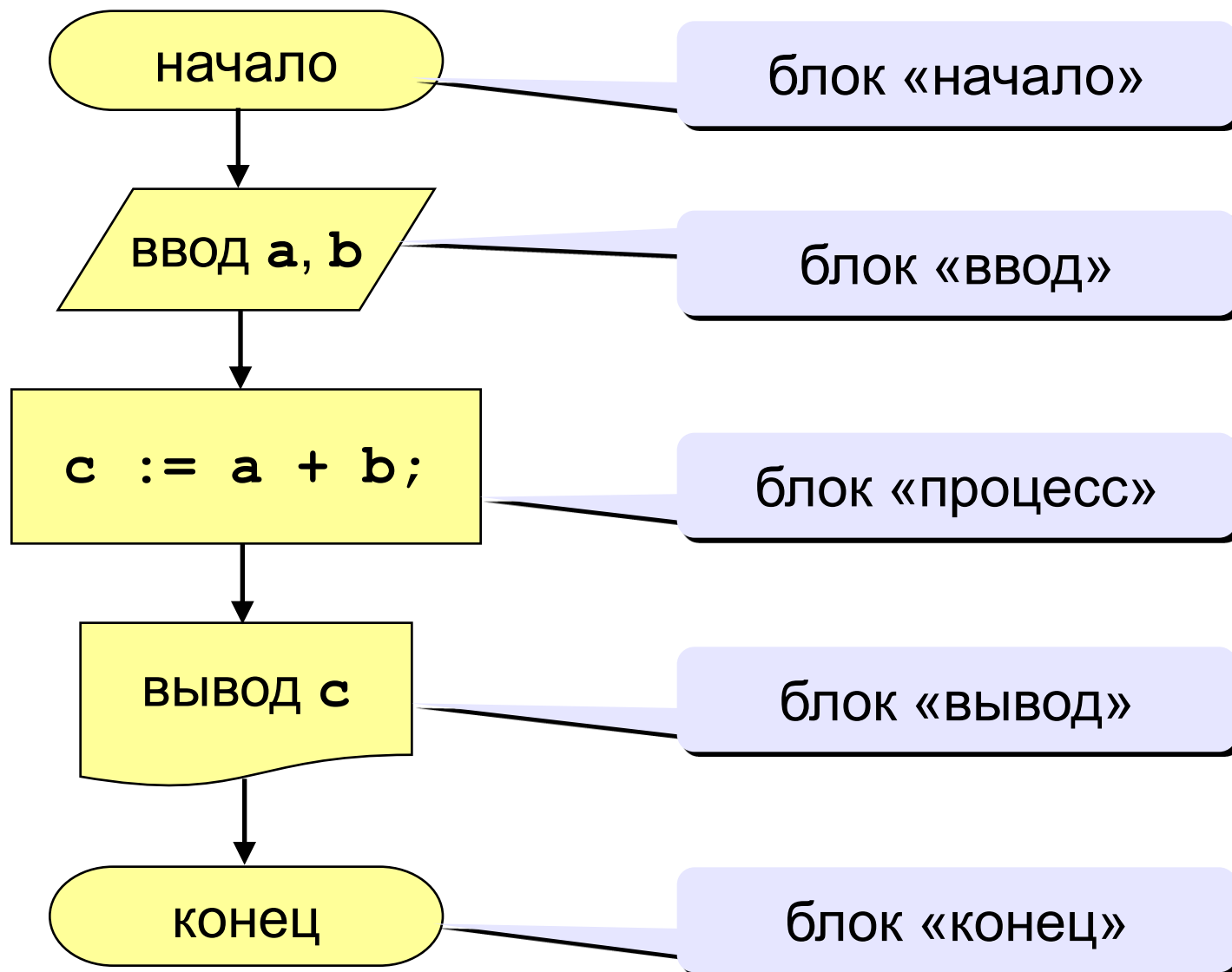


При записи нового
значения старое
стирается!

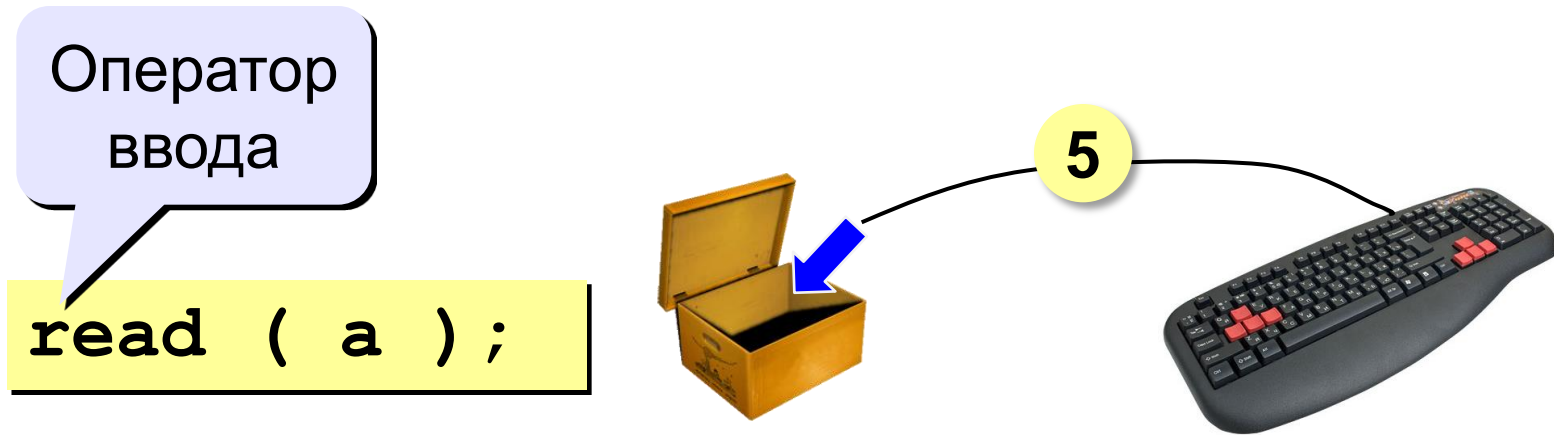
Оператор – это команда языка программирования (инструкция).

Оператор присваивания – это команда для записи нового значения в переменную.

Блок-схема линейного алгоритма



Как ввести значение с клавиатуры



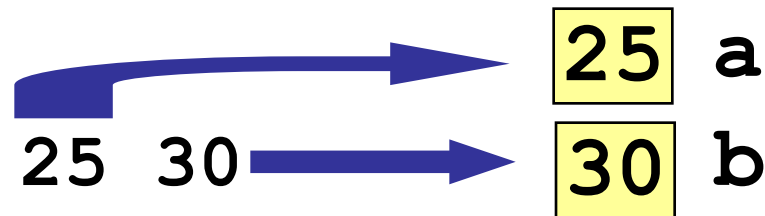
1. Программа ждет, пока пользователь введет значение и нажмет *Enter*.
2. Введенное значение записывается в переменную *a*.

Ввод значений двух переменных

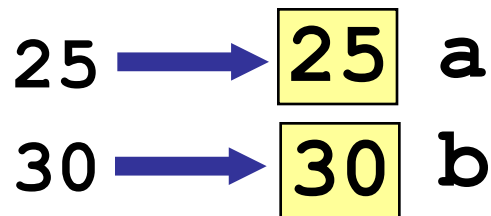
```
read ( a , b ) ;
```

Ввод значений двух переменных (через пробел или *Enter*).

через пробел:



через *Enter*.



Оператор вывода

`write(a);` { вывод значения
переменной a }

`writeln(a);` { вывод значения
переменной a и переход
на новую строку }

`writeln('Привет!');` { вывод текста }

`writeln('Ответ: ', c);`

{ вывод текста и значения переменной c }

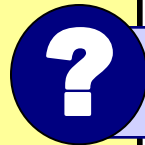
`writeln (a, '+', b, '=', c);`

Сложение двух чисел

Задача. Ввести два целых числа и вывести на экран их сумму.

Простейшее решение:

```
program qq;  
var a, b, c: integer;  
begin  
    read ( a, b );  
    c := a + b;  
    writeln ( c );  
end.
```



Что плохо?

Полное решение

```
program qq;  
var a, b, c: integer;  
begin  
    writeln('Введите два целых числа');  
    read ( a, b );  
    c := a + b;  
    writeln ( a, '+', b, '=', c );  
end.
```

Протокол:

компьютер

Введите два целых числа

25 30

пользователь

25+30=55

Как изменить значение переменной?

Пример:

```
program qq;
```

```
var a, b: integer;
```

```
begin
```

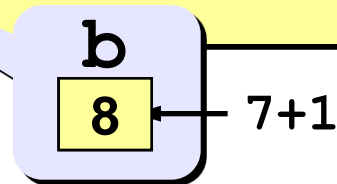
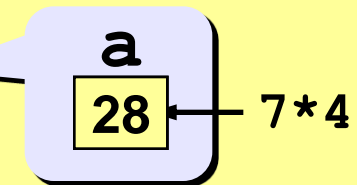
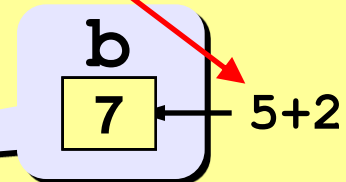
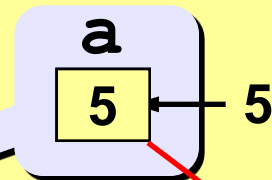
```
  a := 5;
```

```
  b := a + 2;
```

```
  a := (a + 2) * (b - 3);
```

```
  b := b + 1;
```

```
end.
```



Арифметические операции

+ сложение **-** вычитание

***** умножение **/** деление

div деление нацело (остаток отбрасывается)

mod остаток от деления

```
var a, b: integer;  
begin  
    a := 7*3 - 4;  
    a := a * 5;  
    b := a div 10;  
    a := a mod 10;  
end.
```

Порядок выполнения операций

- 1) вычисление выражений в скобках
- 2) умножение, деление, **div**, **mod** слева направо
- 3) сложение и вычитание слева направо

1 2 4 3 6 5

```
z := (5*a+c) / (a*b) * (b-c) ;
```

$$x = \frac{5c^2 - d(a+b)}{(c+d)(d-2a)}$$

$$z = \frac{5a+c}{ab} (b-c)$$

2 3 5 4 1 10 6 9 8 7

```
x := (5*c*c - d*(a+b)) / ((c+d) * (d-2*a)) ;
```


Какие операторы неправильные?

```
program qq;  
var a, b: integer;  
    x, y: real;  
begin  
    a := 5;  
    10 := x;  
    y := 7,8;  
    b := 2.5;  
    x := 2*(a + y);  
    a := b + x;  
end.
```

имя переменной должно
быть слева от знака :=

целая и дробная часть
отделяются **точкой**

нельзя записывать
вещественное значение в
целую переменную

Вывод целых чисел

```
program qq;  
var a, b: integer;  
begin  
  a := 15;  
  b := 45;  
  writeln ( a, b );  
  writeln ( a:4, b:4 );  
end.
```

1545

15 45

СИМВОЛОВ
на ЧИСЛО

Вывод вещественных чисел

```

program qq;
var x: real;
begin
  x := 12.345678;
  writeln ( x );
  writeln ( x:10 );
  writeln ( x:7:2 );
end.

```

ВСЕГО
СИМВОЛОВ

1,234568·10¹

1.234568000E+01

1.235E+01

12.35

ВСЕГО
СИМВОЛОВ

в дробной
части

Задания

«4»: Ввести три числа, найти их сумму и произведение.

Пример:

Введите три числа:

4 5 7

$$4+5+7=16$$

$$4*5*7=140$$

«5»: Ввести три числа, найти их сумму, произведение и среднее арифметическое.

Пример:

Введите три числа:

4 5 7

$$4+5+7=16$$

$$4*5*7=140$$

$$(4+5+7) / 3 = 5.33$$

Разветвляющиеся алгоритмы

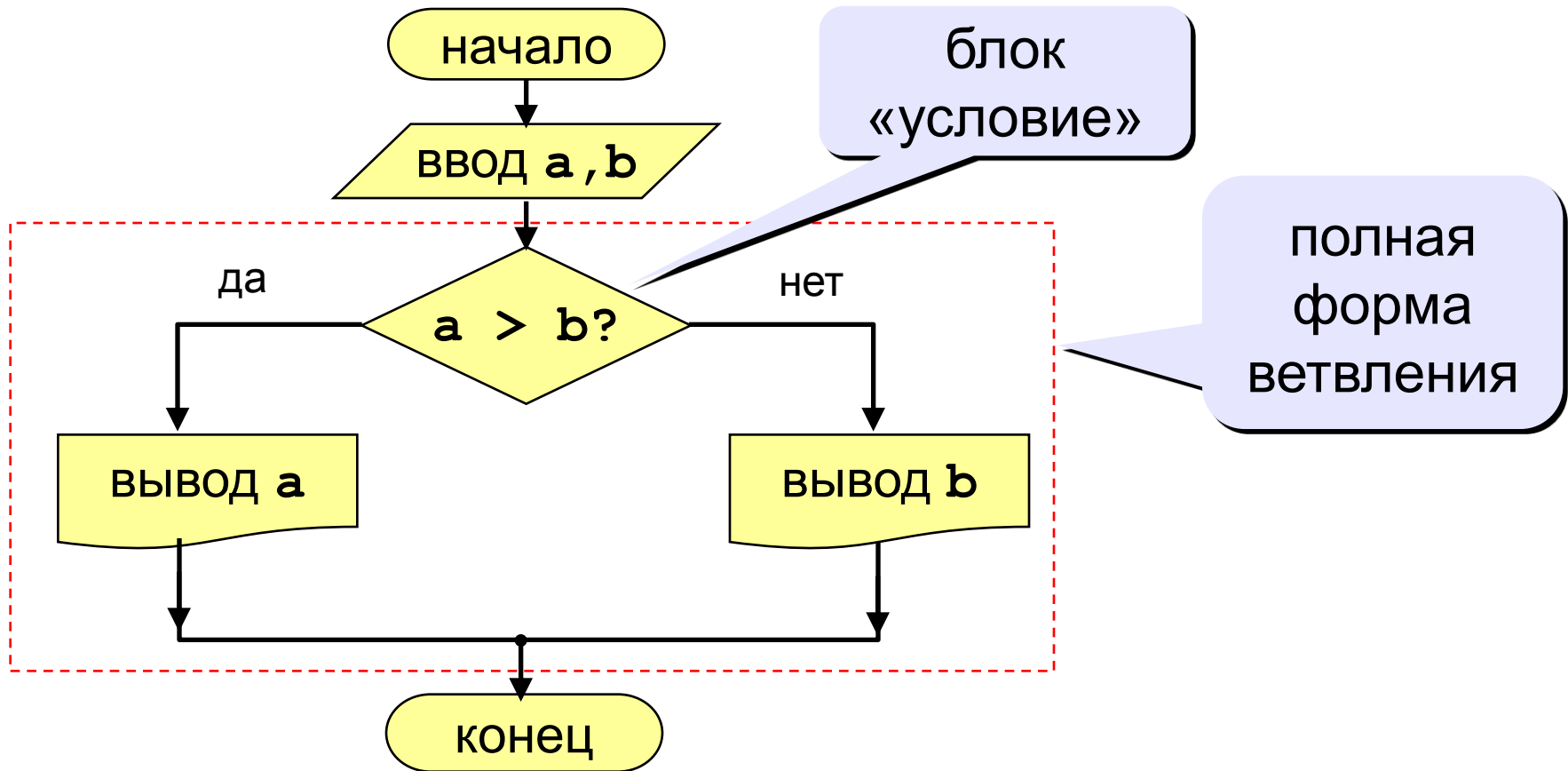
Задача. Ввести два целых числа и вывести на экран наибольшее из них.

Идея решения: надо вывести на экран первое число, если оно больше второго, или второе, если оно больше первого.

Особенность: действия исполнителя зависят от некоторых условий (*если ... иначе ...*).

Алгоритмы, в которых последовательность шагов зависит от выполнения некоторых условий, называются **разветвляющимися.**

Вариант 1. Блок-схема



Если $a = b$?

Вариант 1. Программа

```
program qq;  
var a, b: integer;  
begin  
    writeln('Введите два целых числа');  
    read ( a, b );  
    if a > b then  
        write(a)  
    else  
        write(b);  
end.
```

полная форма
условного
оператора

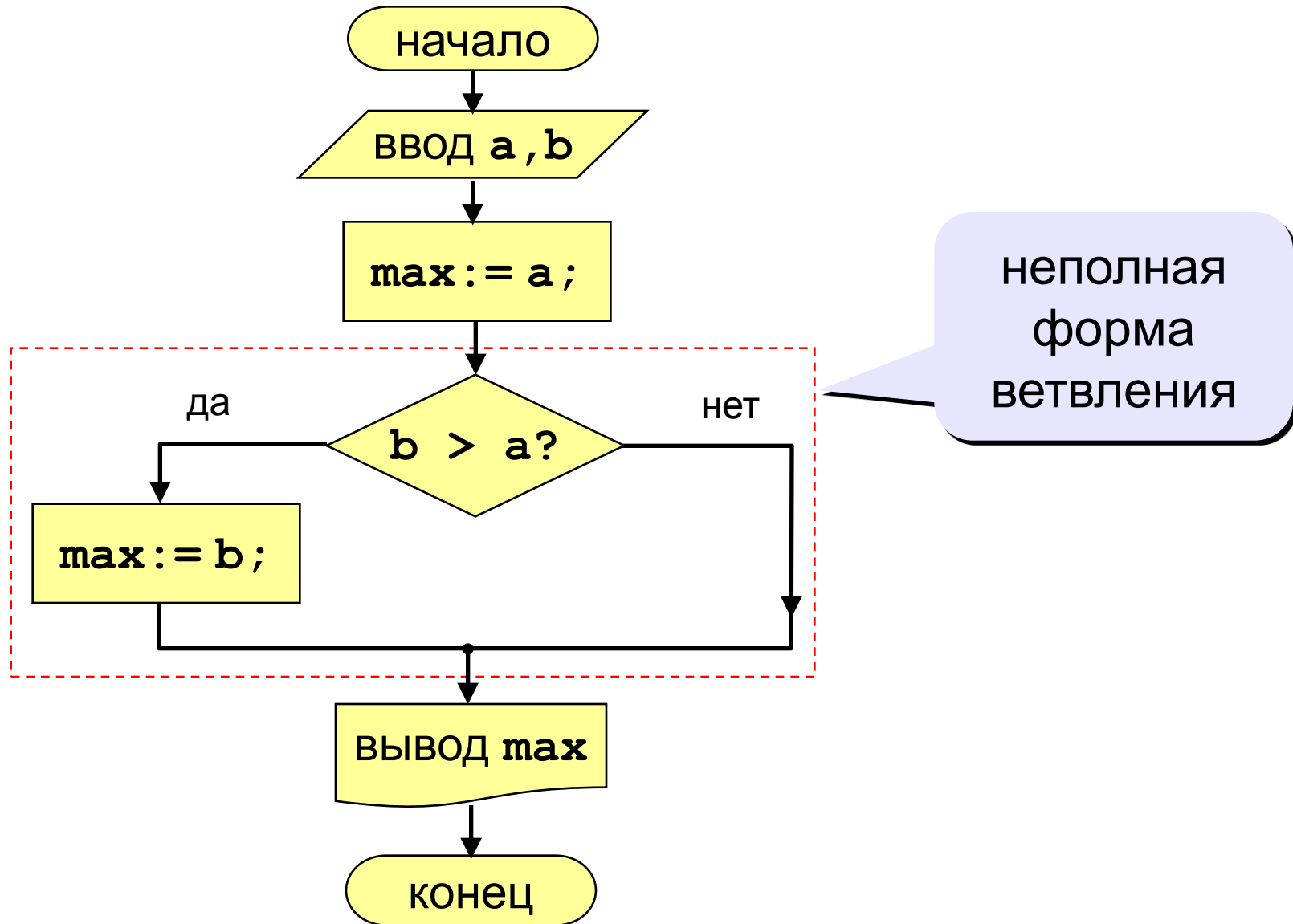
Условный оператор

```
if <условие> then оператор1
    {что делать, если условие верно}
else оператор2;
{что делать, если условие неверно}
```

Особенности:

- перед **else** **НЕ** ставится точка с запятой
- вторая часть (**else** ...) может отсутствовать (неполная форма)
- если после **then** или **else** требуется более одного оператора, то ставятся операторные скобки **begin** и **end** (*составной оператор*)

Вариант 2. Блок-схема



Вариант 2. Программа

```
program qq;  
var a, b, max: integer;  
begin  
  writeln('Введите два целых числа');  
  read ( a, b );  
  max := a;  
  if b > a then  
    max := b;  
  writeln ('Наибольшее число ', max);  
end.
```

неполная
форма
условного
оператора

Вариант 2Б. Программа

```
program qq;  
var a, b, max: integer;  
begin  
    writeln('Введите два целых числа');  
    read ( a, b );  
    max := b;  
    if a > b then  
        max := a;  
    writeln ('Наибольшее число ', max);  
end.
```