

Программирование на языке Python

1. Введение

2. Ветвления

Программирование на языке Python

1. Введение

Алгоритм

Алгоритм – это четко определенный план действий для исполнителя.

Свойства алгоритма

- **дискретность**: состоит из отдельных шагов (команд)
- **понятность**: должен включать только команды, известные исполнителю (входящие в СКИ)
- **определенность**: при одинаковых исходных данных всегда выдает один и тот же результат
- **конечность**: заканчивается за конечное число шагов
- **массовость**: может применяться многократно при различных исходных данных
- **корректность**: дает верное решение при любых допустимых исходных данных

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?

Языки программирования

- **Машинно-ориентированные (низкого уровня)** - каждая команда соответствует одной команде процессора (ассемблер)
- **Языки высокого уровня** – приближены к естественному (английскому) языку, легче воспринимаются человеком, **не зависят от конкретного компьютера**
 - *для обучения*: Бейсик, Кумир, Паскаль, Python
 - *профессиональные*: Си, Python, Паскаль
 - *для задач робототехники и искусственного интеллекта*: Пролог, ЛИСП, C++, Python
 - *для Интернета*: HTML, CSS, JavaScript, Java, Python, PHP, ASP

Язык Python

- 1991** – разработан Гвидо ван Россумом
- объектно-ориентированный язык
 - успешно применяется для интернета



Из чего состоит программа?

Константа – постоянная величина, имеющая имя, в ПИТОНЕ нет констант.

Переменная – изменяющаяся величина, имеющая имя (ячейка памяти).

Процедура – вспомогательный алгоритм, описывающий некоторые действия (рисование окружности).

Функция – вспомогательный алгоритм для выполнения вычислений (вычисление квадратного корня, **sin**).

Имена программы, констант, переменных

Имена могут включать

- латинские буквы (A-Z)

заглавные и строчные буквы не различаются

- цифры

имя не может начинаться с цифры

- знак подчеркивания _

Имена **НЕ** могут включать

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

Какие имена правильные??

**AXby R&B 4Wheel Вася “PesBarbos”
TU154 [QuQu] _ABBA A+B**

Переменные

Язык Python чувствителен к регистру. Переменная Z и z – разные переменные. Python, в отличие от многих языков, не требует описания переменных.

Типы переменных:

- int { целая }
- float { вещественная }
- list { список, аналог массивов }
- str { символьная строка }
- bool { логическая }

Объявление переменных (выделение памяти):

```
int("88") результат 88
```

```
str(88) результат "88"
```

```
float(88) результат 88.00
```

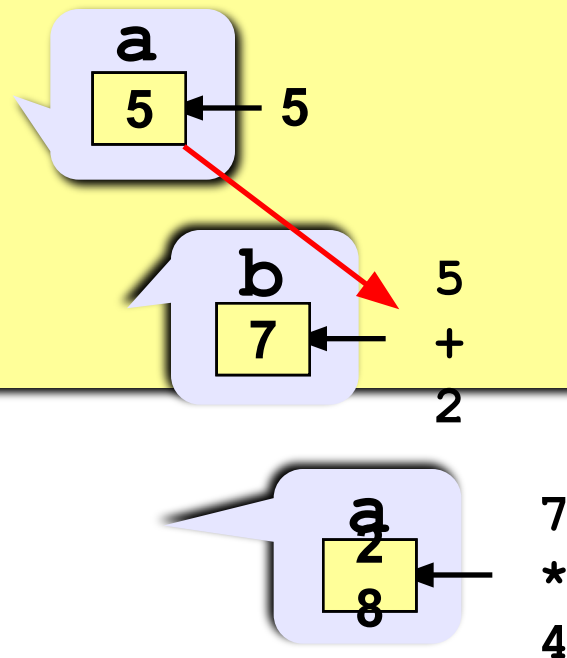
Как изменить значение переменной?

Оператор – это команда языка программирования высокого уровня.

Оператор присваивания служит для изменения значения переменной.

Пример:

```
a=2  
b=a+2  
a=b*4  
print(a)
```



Оператор присваивания



<имя переменной> = <выражение>

Арифметическое выражение может включать

- имена переменных
- знаки арифметических операций:

+ - * / // % **

умножение

деление

деление
нацело

остаток от
деления

Возведение
в степень

- вызовы функции
- круглые скобки ()

Какие операторы неправильные?

a := 5

10 = x;

y = 7,8

b = 2.5

x = 2*(a + y)

a = b + x

имя переменной должно
быть слева от знака **=**

целая и дробная часть
отделяются **точкой**

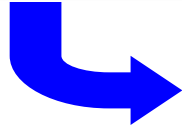
[illegible]

Порядок выполнения операций

- вычисление выражений в скобках
- умножение, деление, //, % слева направо
- сложение и вычитание слева направо

2 3 5 4 1 7 8 6

$$Z = (5 * a * c + 3 * (c - d)) / a * (b - c) / b$$

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


$$z = \frac{5ac + 3(c - d)}{ab} (b - c)$$

2 6 3 4 7 5 1 12 8 11 10

$$X = (a^2 * a + 5 * c * c - d * (a + b)) / ((c + d) * (d - 2 * a))$$

Оператор ввода

```
a=input( )      { ввод значения  
                  переменной a }
```

```
print('Введите число')  
a = int(input())  
c = a + 2  
print ( 'c=', c )
```

Протокол:

компьютер

Введите число

25

c=27

пользователь

Оператор вывода

```
print ( a )      { вывод значения  
                  переменной a }
```

```
print ( ' Hello! ' ) { вывод текста }
```

```
print ( 'Otvet: ', c ) { вывод  
                        текста и значения переменной c }
```

```
print ( a+b ) { вывод суммы чисел }
```


Сложение двух чисел

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

Простейшее решение:

```
a=int(input())  
b=int(input())  
c := a + b  
print ( c )
```

Сложение чисел

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?

Сумма: псевдокод

ввести два числа

вычислить их сумму

вывести сумму на экран

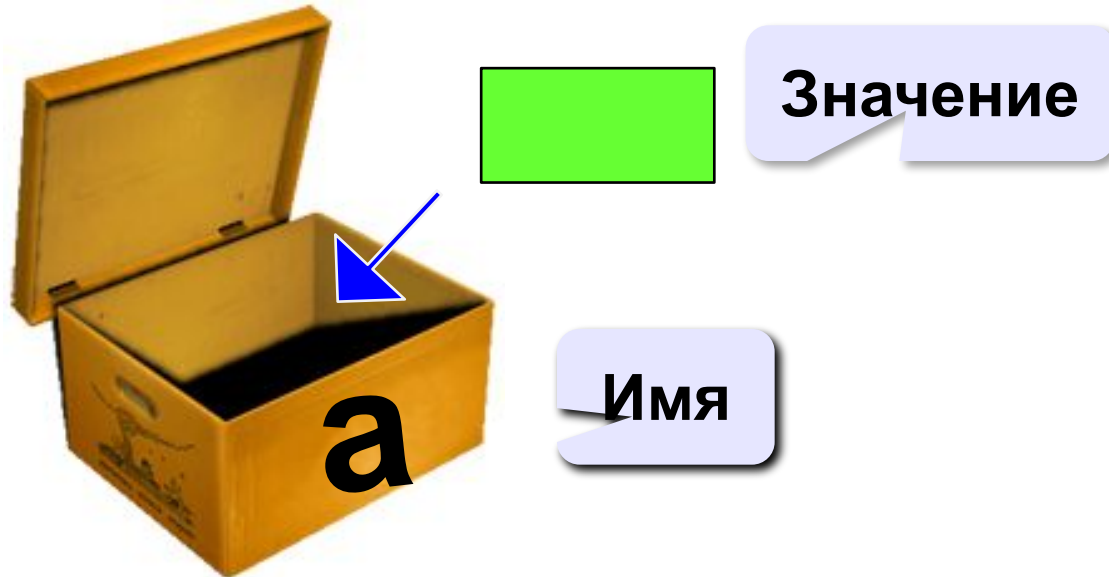
Псевдокод – алгоритм на русском языке с элементами языка программирования.



Компьютер не может исполнить псевдокод!

Переменные

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



Имена переменных

МОЖНО использовать

- латинские буквы (A-Z, a-z)

заглавные и строчные буквы **различаются**

- русские буквы (**не рекомендуется!**)
- цифры

имя не может начинаться с цифры

- знак подчеркивания _

НЕЛЬЗЯ использовать

• ~~еёёбки~~

• ~~знаки +, =, !, ? и др.~~

Какие имена правильные?

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

Типы переменных

```
a = 4  
print ( type(a) )  
<class 'int'>
```

целое число
(*integer*)

```
a = 4.5  
print ( type(a) )  
<class 'float'>
```

вещественное
число

```
a = "Вася"  
print ( type(a) )  
<class 'str'>
```

символьная строка

```
a = True  
print ( type(a) )  
<class 'bool'>
```

логическая

Зачем нужен тип переменной?

Тип определяет:

- область допустимых значений**
- допустимые операции**
- объём памяти**
- формат хранения данных**

Как записать значение в переменную?

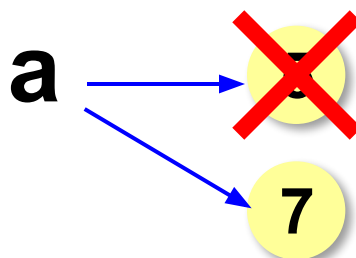
оператор
присваивани
я



При записи нового значения
старое удаляется из памяти!

a = 5

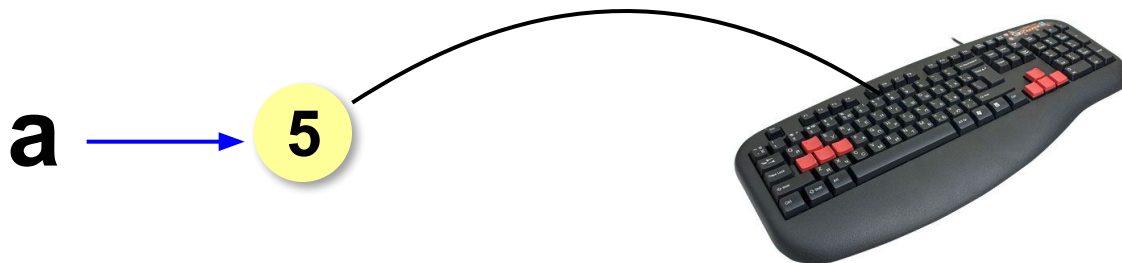
a = 7



Оператор – это команда языка
программирования (инструкция).

Оператор присваивания – это команда для
записи нового значения переменной.

Ввод значения с клавиатуры



1. Программа ждет, пока пользователь введет значение и нажмет *Enter*.
2. Введенное значение записывается в переменную **a** (связывается с именем **a**)

Ввод значения с клавиатуры

```
a = input ()
```

ввести строку с клавиатуры и связать с переменной a

```
b = input ()
```

```
c = a + b
```

```
print ( c )
```

Протокол:

21

33

2133



Почему?



Результат функции input – строка

символов!

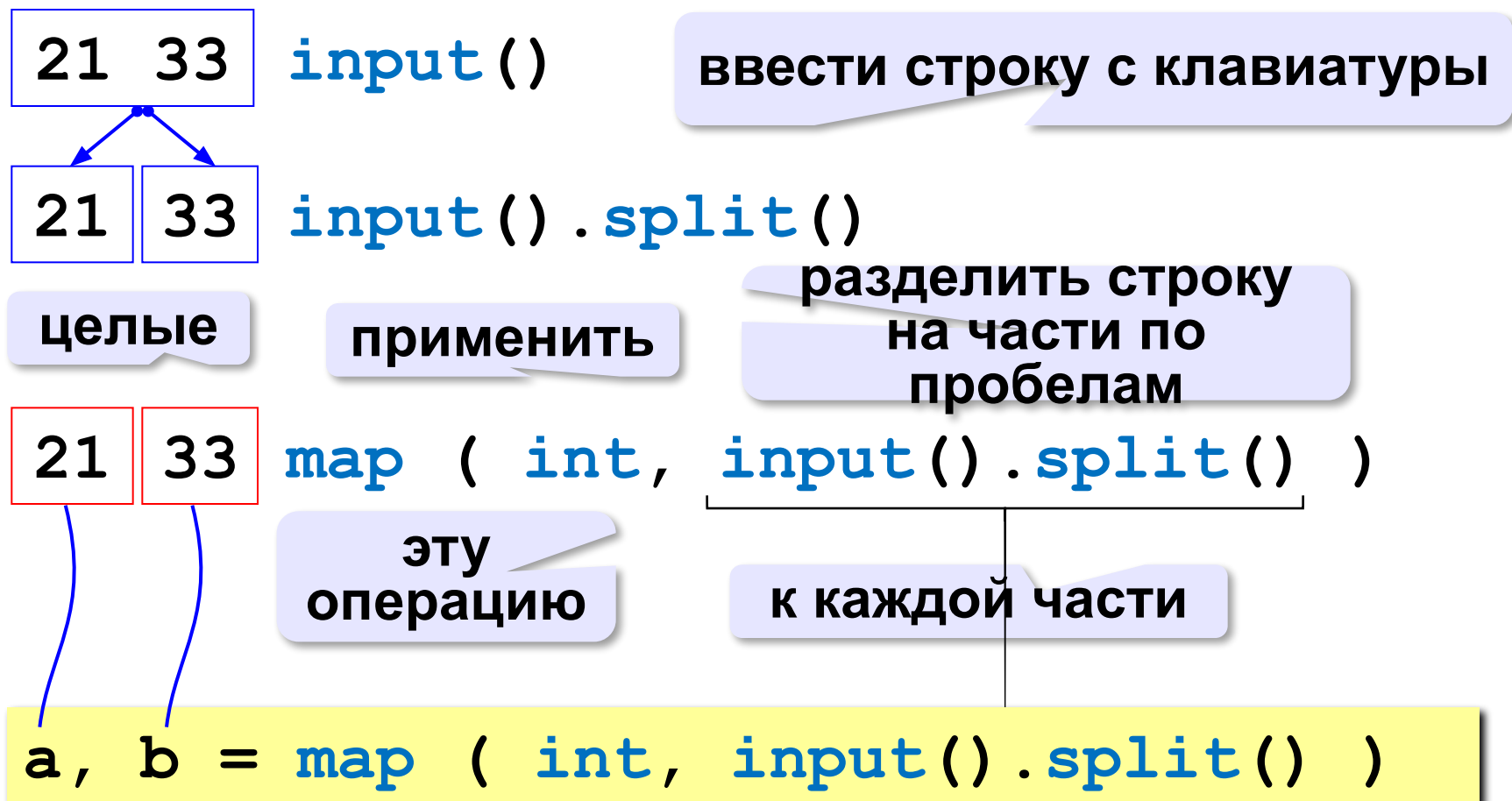
преобразовать
в целое число

```
a = int ( input () )
```

```
b = int ( input () )
```

Ввод двух значений в одной строке

```
a, b = map ( int, input().split() )
```



Ввод с подсказкой

```
a = input ( "Введите число: " )
```

Введите число: 26

подсказка



Что не
так?

```
a = int( input("Введите число: ") )
```

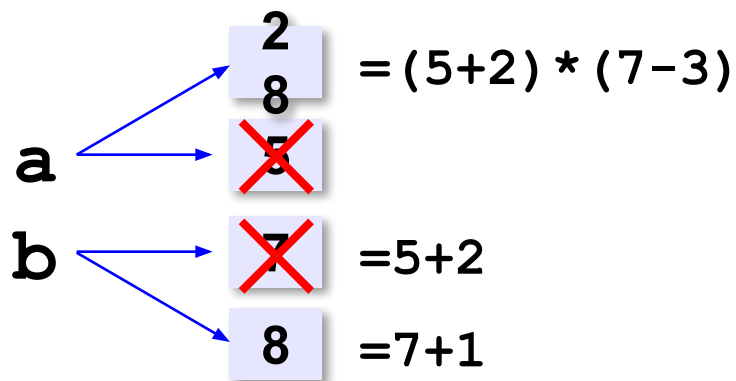
Изменение значений переменной

$a = 5$

$b = a + 2$

$a = (a + 2) * (b - 3)$

$b = b + 1$



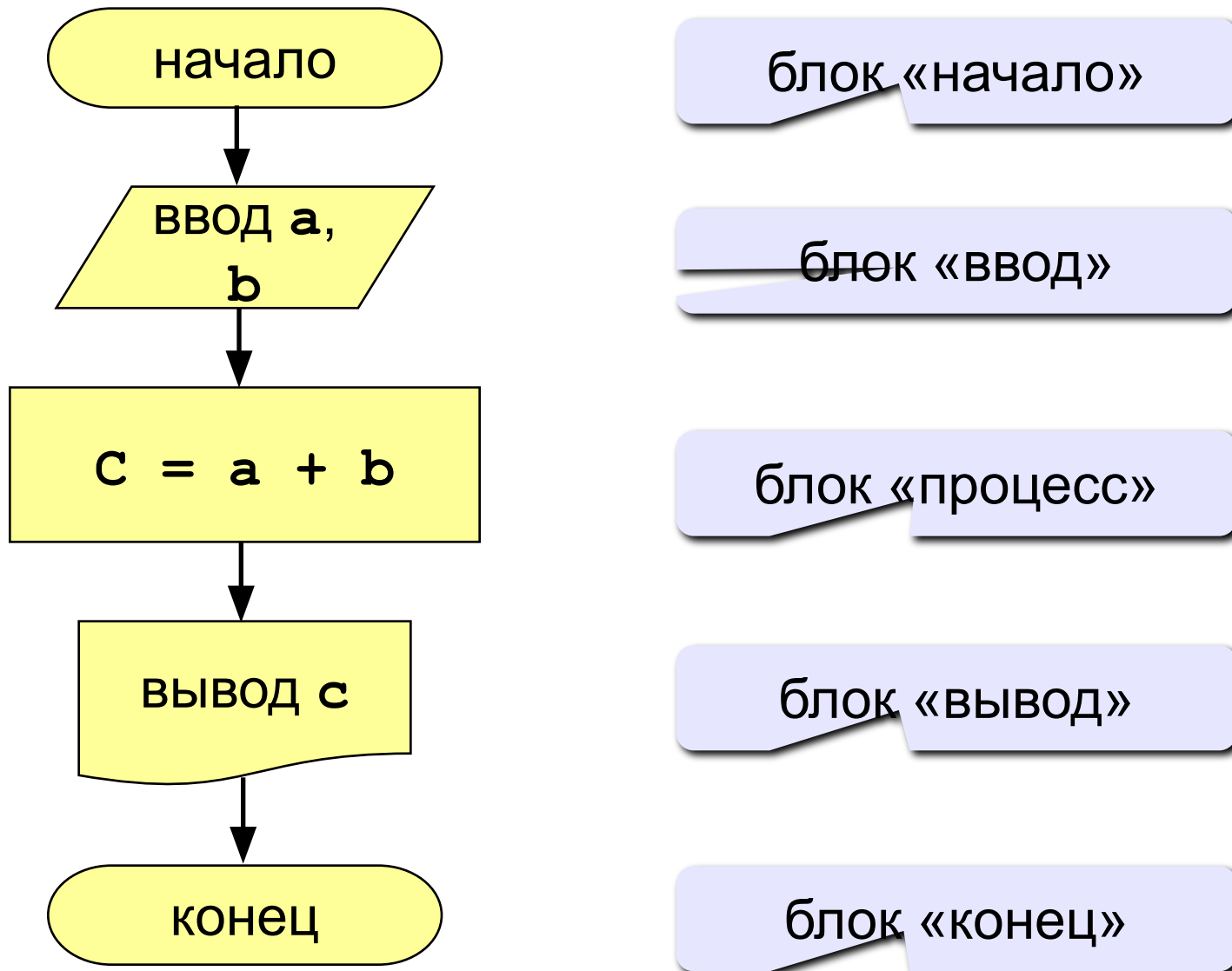
Форматирование вывода

```
import math  
x=math.pi  
print ("%%.50f" % (x))
```

ВСЕГО
СИМВОЛОВ

в дробной
части

Блок-схема линейного алгоритма



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

$$4+5+7=16$$

$$4*5*7=140$$

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

Стандартные функции

`abs(x)` — модуль числа

`int(x)` — преобразование к целому числу

`round(x)` — округление

`import math`

подключить
математический
модуль

`sqrt(x)` — квадратный корень

`sin(x)` — синус угла, заданного **в радианах**

`cos(x)` — косинус угла, заданного **в радианах**

`exp(x)` — экспонента e^x

`ln(x)` — натуральный логарифм

`floor(x)` — округление «вниз»

`x = math.floor(1.6) # 1`

`x = math.ceil(1.6) # 2`

«вверх»

`x = math.floor(-1.6) #-2`

`x = math.ceil(-1.6) #-1`

Генератор случайных чисел

```
import random
```

англ. *random* – случайный

Генератор на $[0,1)$:

```
X = random() ; # псевдослучайное число  
Y = random()   # это уже другое число!
```

Целые числа на отрезке $[a,b]$:

```
X = randint(a, b) # псевдослучайное число  
Y = randint(a, b) # это уже другое число!
```

Программирование на языке Python

2. Ветвления

Разветвляющиеся алгоритмы

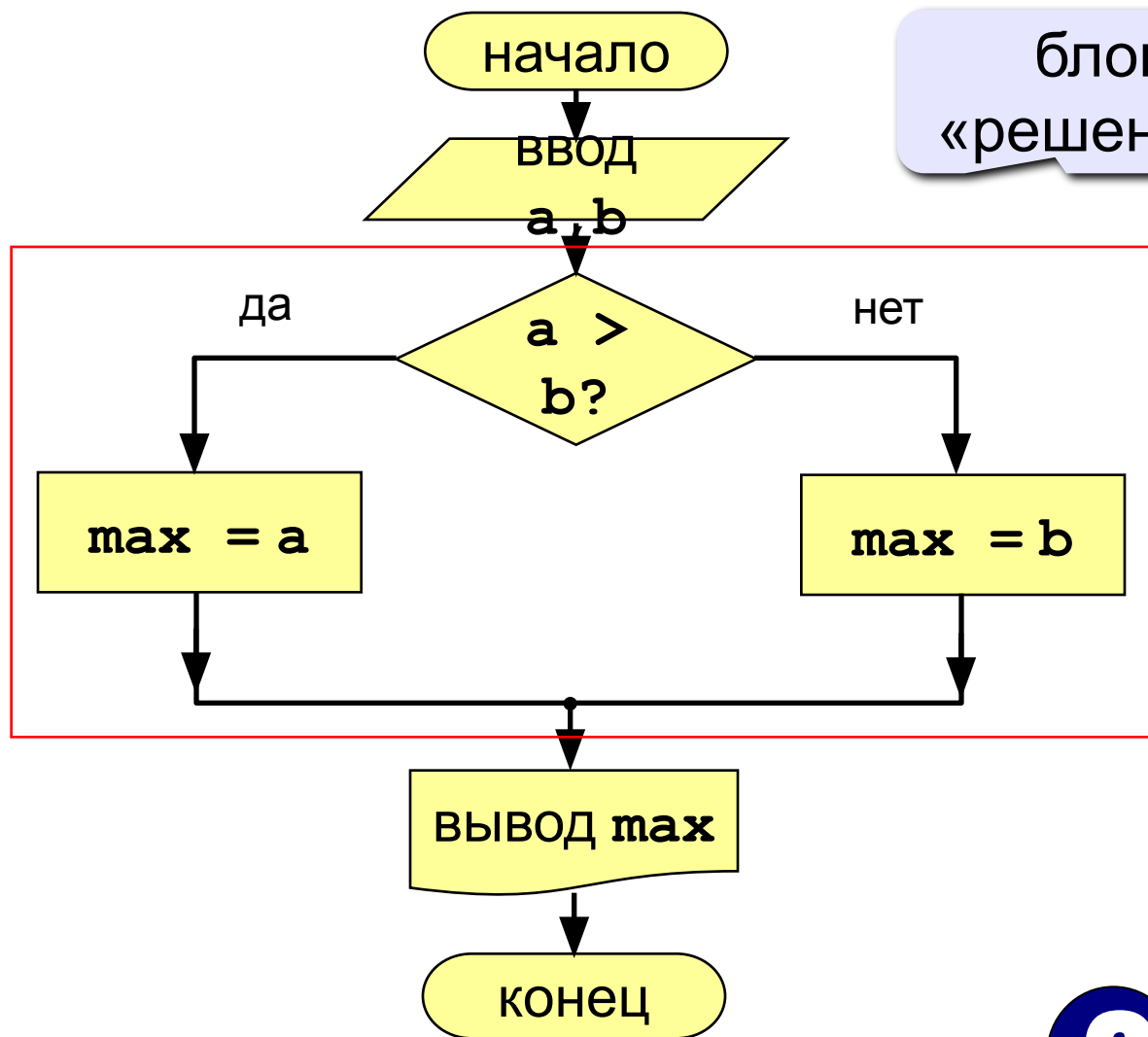
Задача. Ввести два целых числа и вывести на экран наибольшее из них.

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



блок
«решение»

полная
форма
ветвления



Если $a = b$?

Вариант 1. Программа

```
print('input 2 number')  
a=int(input())  
b=int(input())  
if a > b :  
    max = a  
  
else :  
    max = b  
  
print ('Max number', max)
```

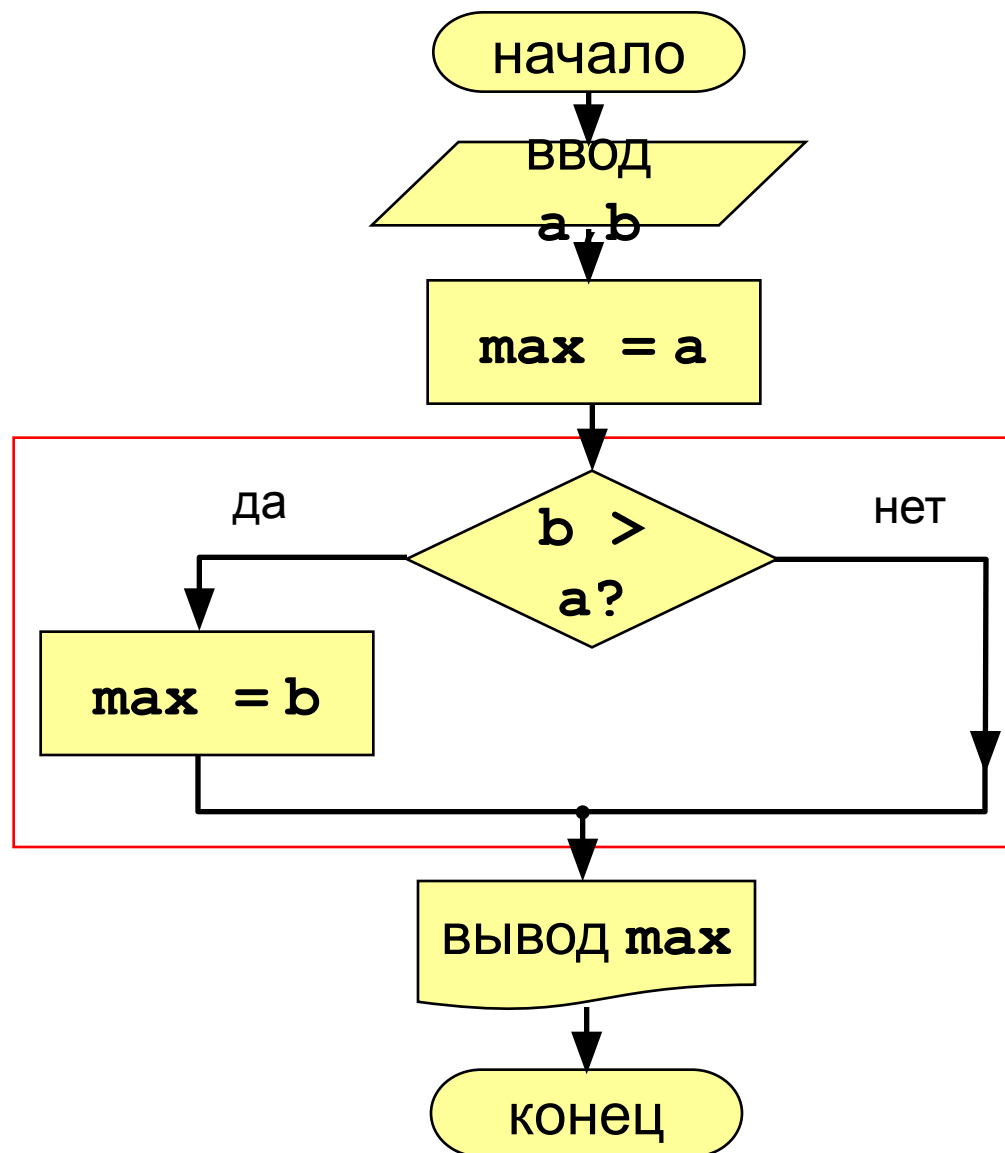
Условный оператор

```
if <условие> :  
    {что делать, если условие верно}  
  
else :  
    {что делать, если условие неверно}
```

Особенности:

- вторая часть (***else*** ...) может отсутствовать (неполная форма)

Вариант 2. Блок-схема



неполная
форма
ветвления

Вариант 2. Программа

```
print('input 2 number')
```

```
a=input()
```

```
b=input()
```

```
max = a
```

```
if b > a :
```

```
    max = b
```

```
print ('Max number', max)
```

неполная
форма
условного
оператора

Задания

«4»: Ввести три числа и найти наибольшее из них.

Пример:

Введите три числа:

4 15 9

Наибольшее число 15

«5»: Ввести пять чисел и найти наибольшее из них.

Пример:

Введите пять чисел:

4 15 9 56 4

Наибольшее число 56

Знаки отношений

>	<	больше, меньше
>=		больше или равно
<=		меньше или равно
==		равн о
!=		не равно

Вложенные условные операторы

Задача: в переменных *a* и *b* записаны возрасты
Андрея и Бориса. Кто из них старше?



Сколько
вариантов?

```
if a > b:  
    print("Андрей старше")  
else:  
    if a == b:  
        print("Одного возраста")  
    else:  
        print("Борис старше")
```



Зачем
нужен?

вложенный
условный
оператор

Каскадное ветвление

```
if a > b:  
    print("Андрей старше")  
elif a == b:  
    print("Одного возраста")  
else:  
    print("Борис старше")
```



`elif = else if`

Каскадное ветвление

```
cost = 1500
if cost < 1000:
    print ( "Скидок нет." )
elif cost < 2000:
    print ( "Скидка 2%." )
elif cost < 5000:
    print ( "Скидка 5%." )
else:
    print ( "Скидка 10%." )
```

первое сработавшее
условие



Что
выведет?

Скидка 2%.

Задачи

«А»: Ввести три целых числа, найти максимальное из них.

Пример:

Введите три целых числа:

1 5 4

Максимальное число 5

«В»: Ввести пять целых чисел, найти максимальное из них.

Пример:

Введите пять целых чисел:

1 5 4 3 2

Максимальное число 5