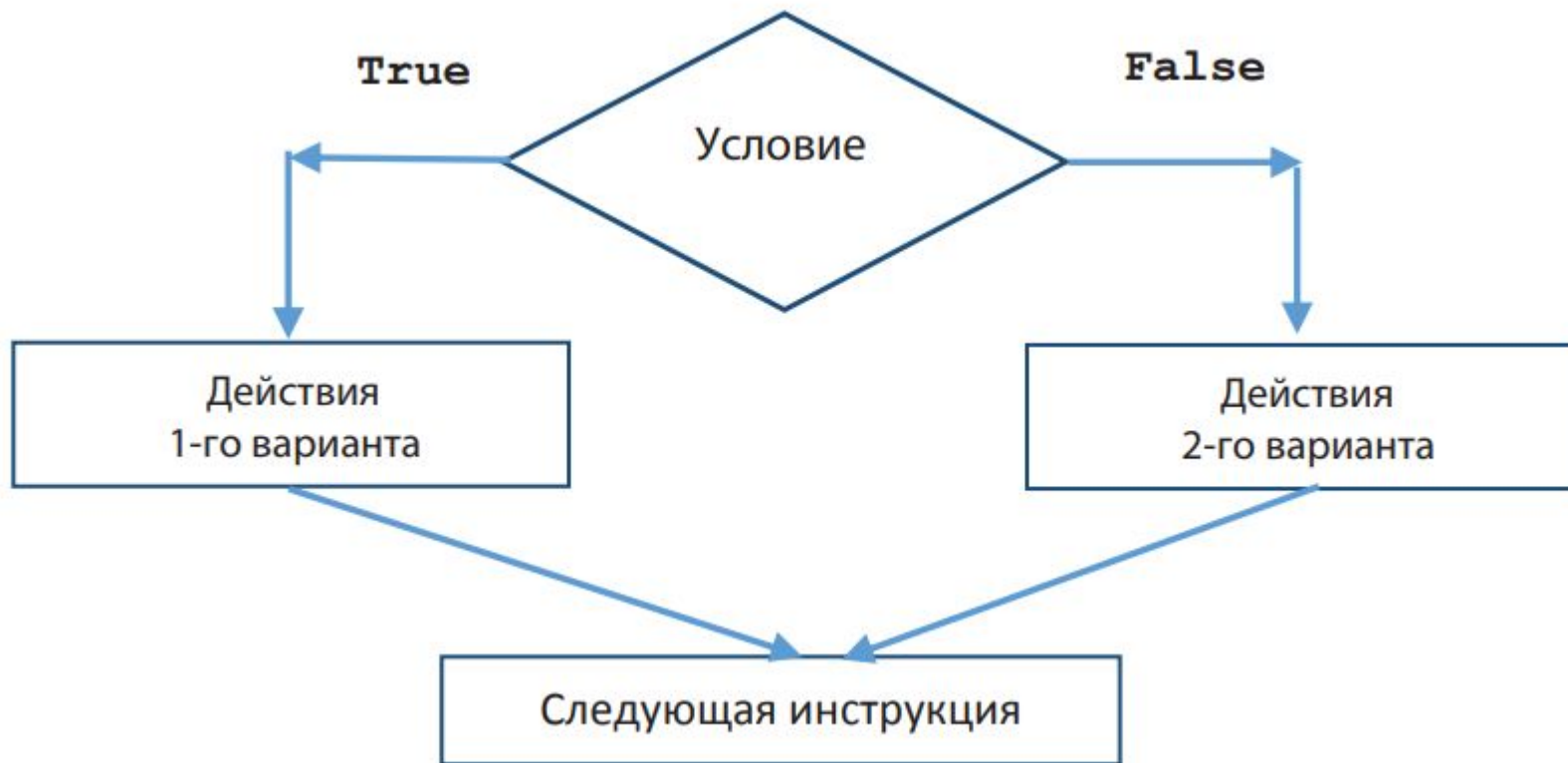


Условный оператор

Задача: **изменить порядок действий** в зависимости от выполнения некоторого условия.



Структура полного условного оператора

if <условие>:

 <Действия 1-го варианта (1-я серия инструкций)>

else:

 <Действия 2-го варианта (2-я серия инструкций)>

В самом простом случае – это два арифметических выражения между которыми записан знак операции сравнения.

В языке Python есть 6 операций сравнения:

	Знак операции	Означает
1	<	Меньше
2	<=	Меньше либо равно
3	>	Больше
4	>=	Больше либо равно
5	==	Равно
6	!=	Не равно

Сложные условия

Задача: набор сотрудников в возрасте **25-40 лет**
(включительно).

сложное условие

```
if v >= 25 and v <= 40 :  
    print ("подходит")  
else:  
    print ("не подходит")
```

and «И»

or «ИЛИ»

not «НЕ»

Приоритет :

- 1) отношения (<, >, <=, >=, ==, !=)
- 2) **not** («НЕ»)
- 3) **and** («И»)
- 4) **or** («ИЛИ»)

Пример. Даны два вещественных числа a и b . Если первое больше второго, то увеличить каждое число в 2 раза, иначе – уменьшить в два раза.

Соответствующая программа:

```
a = float(input('a = '))
b = float(input('b = '))
if a > b:
    a = a * 2
    b = b * 2
else:
    a = a/2
    b = b/2
print('a =', a)
print('b =', b)
```

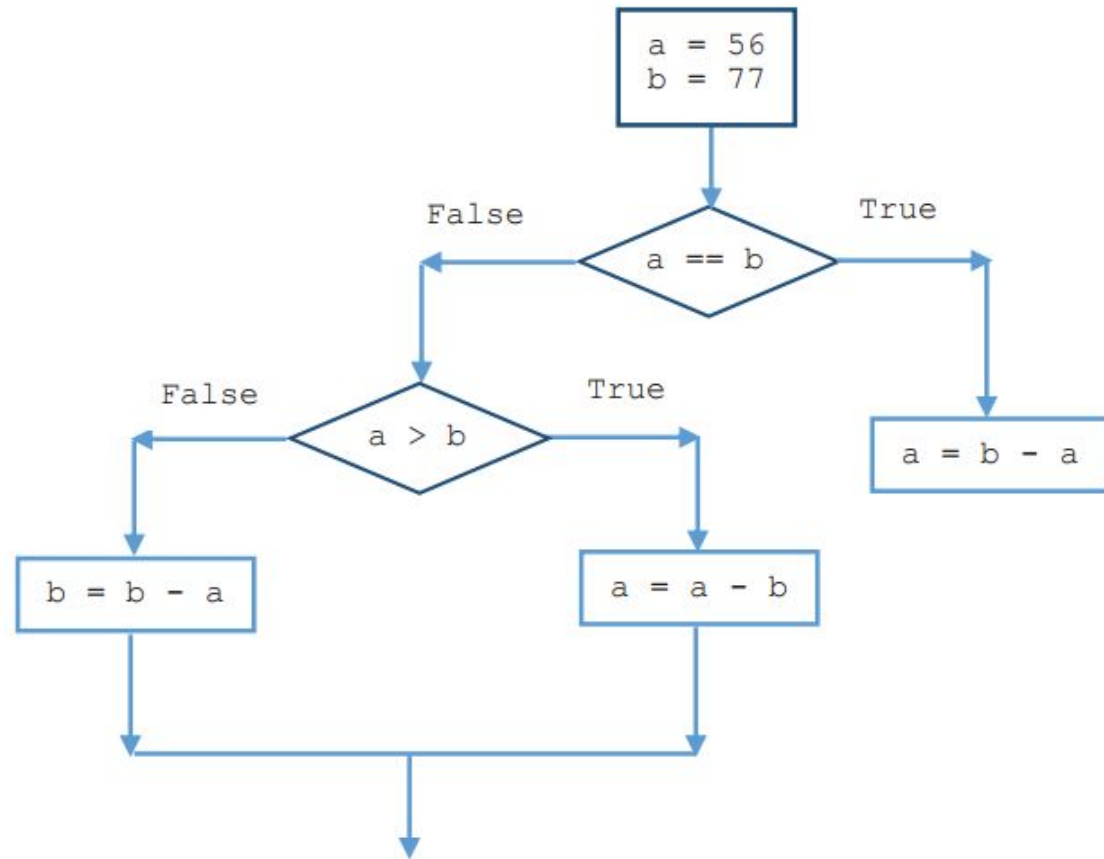


Таблица истинности

<i>A</i>	<i>B</i>	Not <i>A</i>	<i>A</i> And <i>B</i>	<i>A</i> Or <i>B</i>	<i>A</i> Xor <i>B</i>
T	T	F	T	T	F
T	F	F	F	T	T
F	F	T	F	F	F
F	T	T	F	T	T

Свойства условий:

1. Сокращение длинных условий

- использовать обратный слэш («\»):

if $v < 400$ and $v \neq 2$ and $v \neq 3$ and $v \neq 12$ and $v \neq 13$ and $v \neq 22$ and $v \neq 23$: ...

- взять все условие в скобки (перенос внутри скобок разрешен):

if ($v < 400$ and $v \neq 2$ and $v \neq 3$ and $v \neq 12$ and $v \neq 13$ and $v \neq 22$ and $v \neq 23$): ...

2. Разрешены двойные неравенства,

if $A < B < C$: ... означает то же самое, что и if $A < B$ and $B < C$:

3. **Условие - логические функции**, то есть функции, возвращающие результат логического типа

```
n = int(input('Введите целое число '))
```

```
if Chet(n):
```

```
    print('Это число четное')
```

```
else:
```

```
    print('Это число нечетное')
```

где Chet() – функция, возвращающая результат True, если ее параметр (значение, указанное в скобках) является четным числом, и False – в противном случае;

4. **оператор in** (оператор проверки принадлежности), который проверяет, принадлежит ли некоторый объект (число, символ, переменная и т. п.) набору значений (списку, строке, диапазону чисел и т. п.):

```
a = 3
```

```
if a in range(10):
```

```
    sim = input('Введите символ ')
```

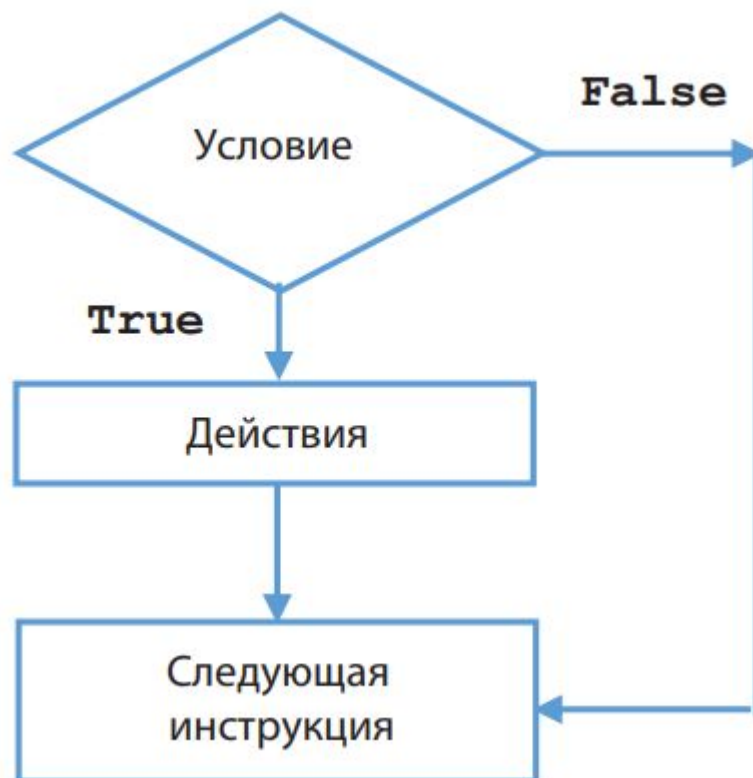
```
    s = input('Введите строку символов ')
```

```
if sim in s:
```

```
    Zvet = 'Зеленый'
```

```
if Zvet in Raduga:
```

Неполная форма



```
M = a
if b > a:
    M = b
```

Решение в стиле Python:

```
M = max(a, b)
```

```
M = a if a > b else b
```



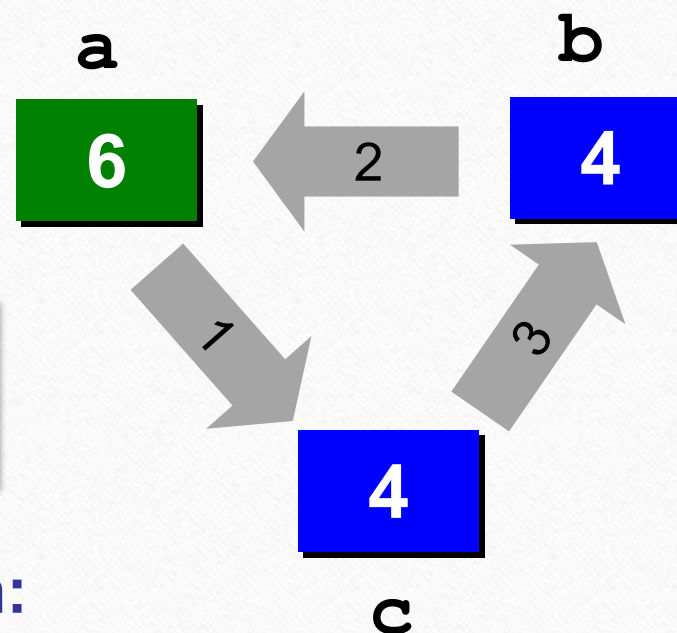
```
if a > b:
```

```
    c = a
```

```
    a = b
```

```
    b = c
```

? Что делает?



? Можно ли обойтись без переменной c?

Решение в стиле Python:

```
a, b = b, a
```

Вложенные условные операторы

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

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

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%.