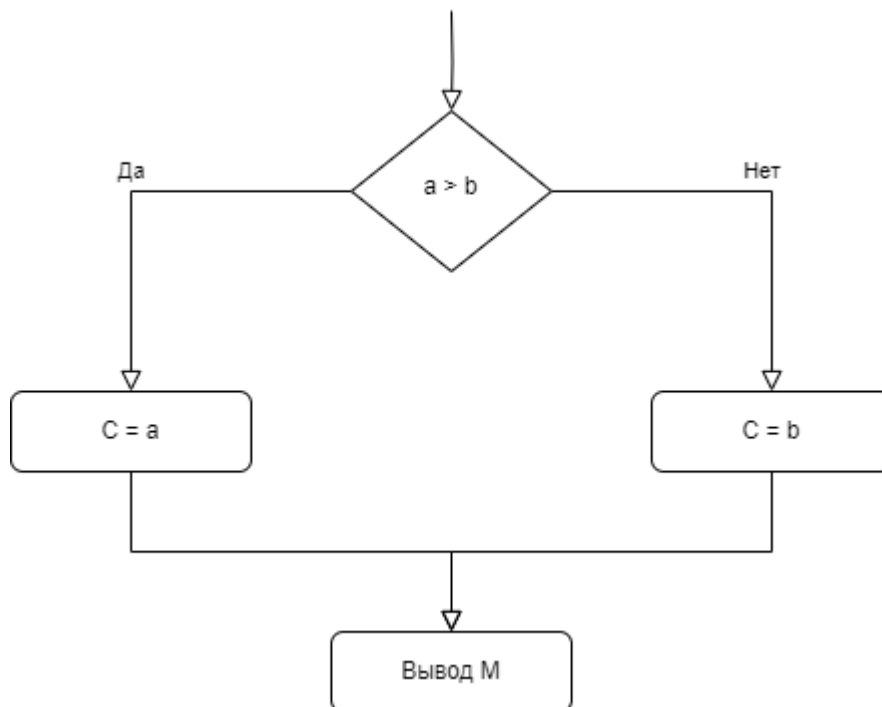


Ветвление



Для этого используется условный оператор if. Задача условного оператора изменить порядок действий в зависимости от выполнения некоторого условия.

Полная запись условного оператора:

```
In [14]: a = 10; b = 20
if a > b: # Если a больше b, то
    c = a # Присваиваем c значение a
else: # Иначе,
    c = b # Присваиваем c значение b
print(c) # Вывод c
```

20

Неполная форма:

```
In [15]: a = 10; b = 20
c = b
if a > b: # Если a больше b, то
    c = a # Присваиваем c значение a
#Иначе ничего не делаем
print(c) # Вывод c
```

20

И каскадная проверка условий:

```
In [18]: a = 7
if a < 2:
    print("Меньше двух")
elif a < 3:
```

```

    print("Меньше трёх")
elif a<4:
    print("Меньше четырёх")
elif a<5:
    print("Меньше пяти")
elif a<6:
    print("Меньше шести")
elif a<7:
    print("Менеше семи")
elif a<8:
    print("Меньше восьми")

```

Меньше восьми

Вложенные условные операторы:

```

In [17]: a = 11; b = 22
        c = 1
        if a > b:
            if a%10==0:
                c = a
            else:
                c = -a
        else:
            if a%10==0:
                c = b
            else:
                c = -b
        print(c)

```

-22

elif ____

==

else: if ____:

С каждым уровнем вложенности добавляется 1 табуляционный отступ от начала строки. Отступ ставиться клавишей Tab на клавиатуре.

Логические операции:

>, < - больше, меньше

>= - больше или равно

<= - меньше или равно

== - равно

!= - не равно

```

In [19]: print(10<20)

```

True

```

In [20]: print(10>20)

```

False

```
In [24]: print(10>=10)
```

True

```
In [25]: print(10<=10)
```

True

```
In [27]: print(10>=8)
```

True

```
In [28]: print(20<=10)
```

False

```
In [21]: print(10==20)
```

False

```
In [23]: print(10!=20)
```

True

Сложные условия

Условия могут объединяться в сложные условия, которая состоят из нескольких условий.

Для этого используются логические операторы `not` («НЕ»), `and` («И») и `or` («ИЛИ»)

```
In [29]: a=10;b=15
if a>3 and a%5==0:
    print("а больше трёх И без остатка делится на пять ")
```

а больше трёх И без остатка делится на пять

Приоритет логических операций:

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