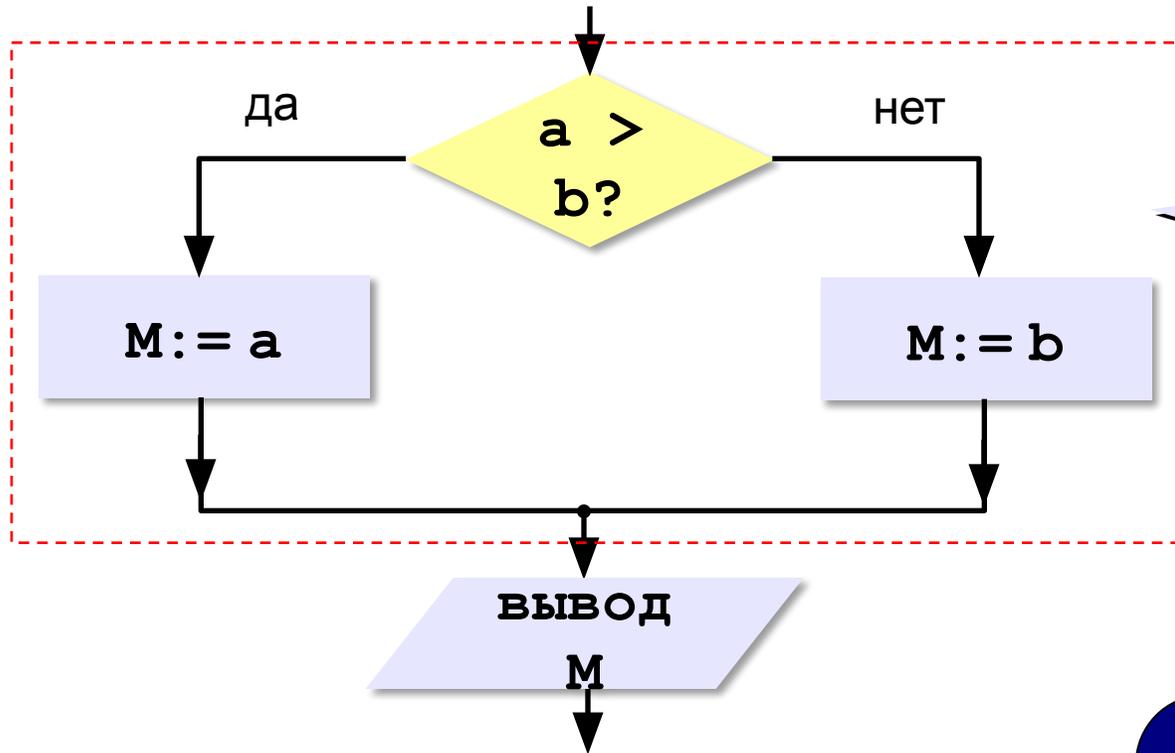


# УСЛОВНЫЙ ОПЕРАТОР

*Алгоритмы ветвления*

# Условный оператор

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

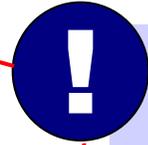


полная  
форма  
ветвления

? Если  $a = b$ ?

# Условный оператор: полная

```
if a > b then  
    M := a  
else  
    M := b;
```

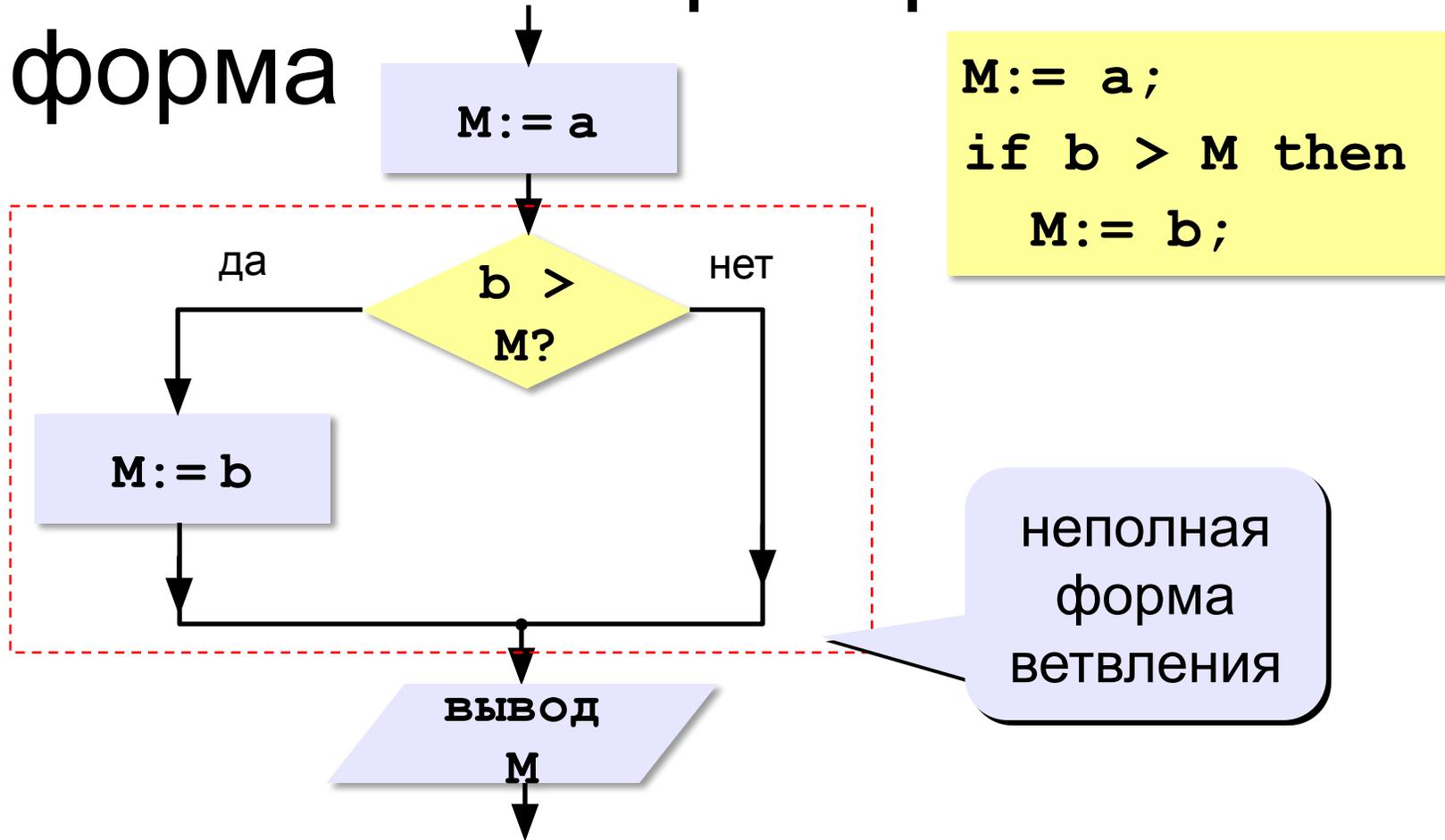


Перед `else` знак «;»  
**НЕ** ставится!

```
if a > b then begin  
    M := a;  
end  
else begin  
    M := b;  
end;
```

операторные  
скобки

# Условный оператор: неполная форма

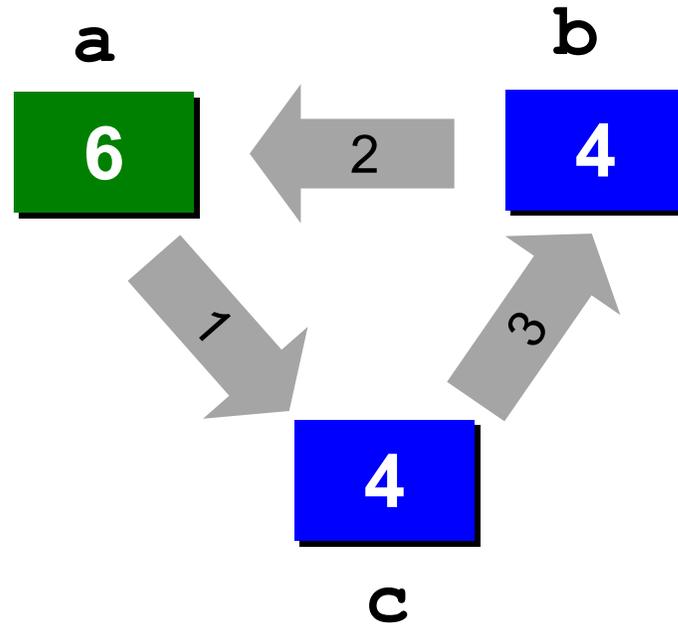


```
M := a;  
if b > M then  
  M := b;
```

# Условный оператор

```
if a < b then begin  
  c := a;  
  a := b;  
  b := c;  
end;
```

? Что делает?



? Можно ли обойтись без переменной **c**?

# Знаки отношений

>

<

больше, меньше

>=

больше или равно

<=

меньше или равно

=

равно

<>

не равно

# Вложенный условный оператор

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



Сколько вариантов?

```
if a > b then
  writeln('Андрей старше')
else
  if a = b then
    writeln('Одного возраста')
  else
    writeln('Борис старше');
```



Зачем нужен?

вложенный  
условный оператор

# Выделение структуры

```
( if a > b then write('A') else if a = b then  
write('=') else write('B');
```

```
if a > b then  
    write('A')  
else  
    if a = b then  
        write('=')  
    else write('B');
```

# Сложные условия

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

сложное условие

```
if (v >= 25) and (v <= 40) then  
    writeln('подходит')  
else  
    writeln('не подходит');
```

and

or

xor

not

исключающее  
«ИЛИ»

Приоритет :

- 1) **not**
- 2) **and**
- 3) **or, xor**
- 4) отношения (<, >, <=, >=, =, <>)



Почему скобки обязательны?

# Множественный выбор

```
if m = 1 then
    write('январь');
if m = 2 then
    write('февраль');
...
if m = 12 then
    write('декабрь');
```

```
case m of
    1: write('январь');
    2: write('февраль');
    ...
    12: write('декабрь')
else write('ошибка')
end;
```

# Использование списков и

Число дней в месяце:

Диагностика

```
case m of
  2: d:= 28;  { невисокосный год }
  1,3,5,7,8,10,12: d:= 31
  else d:= 30
end;
```

Социальный статус:

```
case v of
  0..6:  write(' дошкольник ');
  7..17: write(' школьник ')
  else   write(' взрослый ')
end;
```

# Множественный выбор

```
var c: char;  
...  
case c of  
  'a': begin  
    writeln('антилопа');  
    writeln('Анапа');  
  end;  
...  
  'я': begin  
    writeln('ягуар');  
    writeln('Якутск');  
  end  
else writeln('ошибка')  
end;
```

несколько  
операторов в  
блоке

# 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

# Задачи

«С»: Ввести последовательно возраст Антона, Бориса и Виктора. Определить, кто из них старше.

**Пример:**

Возраст Антона: 15

Возраст Бориса: 17

Возраст Виктора: 16

Ответ: Борис старше всех.

**Пример:**

Возраст Антона: 17

Возраст Бориса: 17

Возраст Виктора: 16

Ответ: Антон и Борис старше Виктора.

# Задачи

**«С»:** Напишите программу, которая получает возраст человека (целое число, не превышающее 120) и выводит этот возраст со словом «год», «года» или «лет». Например, «21 год», «22 года», «25 лет».

**Пример:**

Введите возраст: **18**

Вам 18 лет.

**Пример:**

Введите возраст: **21**

Вам 21 год.

**Пример:**

Введите возраст: **22**

Вам 22 года.

# Задачи

**«В»:** Напишите программу, которая получает номер месяца и выводит соответствующее ему время года или сообщение об ошибке.

## Пример:

Введите номер месяца :

5

Весна .

## Пример:

Введите номер месяца :

15

Неверный номер месяца .