



# ПРОГРАММИРОВАНИЕ РАЗВЕТВЛЯЮЩИХСЯ АЛГОРИТМОВ

## НАЧАЛА ПРОГРАММИРОВАНИЯ

9 класс



ИЗДАТЕЛЬСТВО

**БИНОМ**

# Ключевые слова

- **условный оператор**
- **сокращённая форма условного оператора**
- **составной оператор**
- **вложенные ветвления**



# Общий вид условного оператора

Полная форма условного оператора:

**if** <условие> **then** <оператор\_1> **else** <оператор\_2>

Сокращённая форма условного оператора:

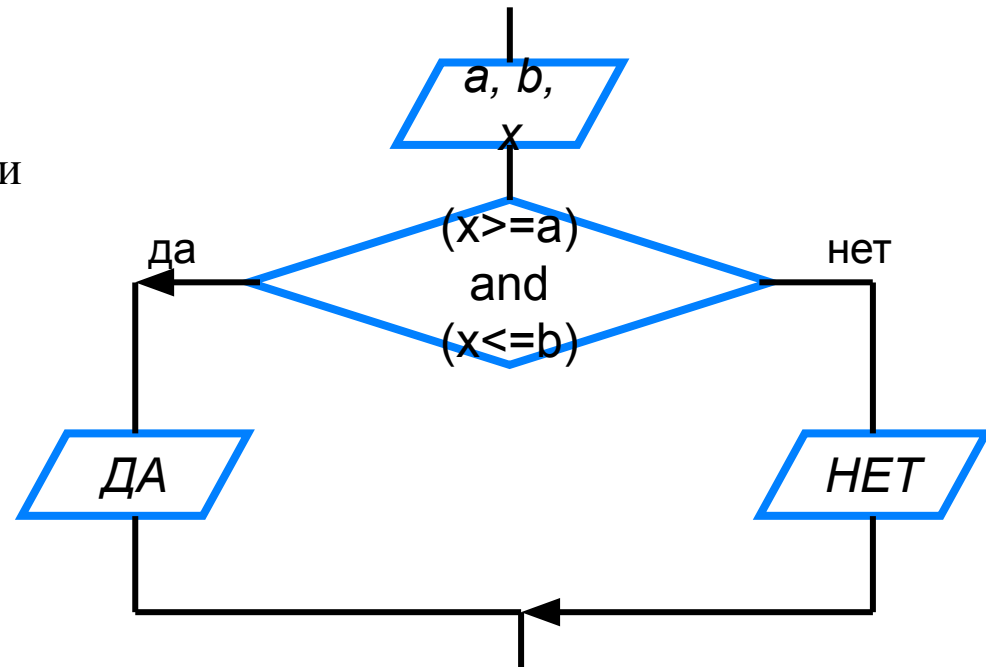
**if** <условие> **then** <оператор>

**!** Перед **else** знак «;» не ставится.



# Условный оператор

```
program n_9;  
  var x, a, b: real;  
begin  
  writeln ('Определение принадлежности  
           точки отрезку');  
  write ('Введите a, b>>');  
  readln (a, b);  
  write ('Введите x>>');  
  readln (x);  
  if (x>=a) and (x<=b) then  
    writeln ('Точка принадлежит отрезку')  
  else writeln ('Точка не принадлежит отрезку')  
end.
```



# Сокращённая форма условного оператора

```
program n_10;  
  var y, a, b, c: integer;
```

```
begin
```

```
  writeln ('Нахождение наибольшей из трёх величин');
```

```
  write ('Введите a, b, c>>');
```

```
  readln (a, b, c);
```

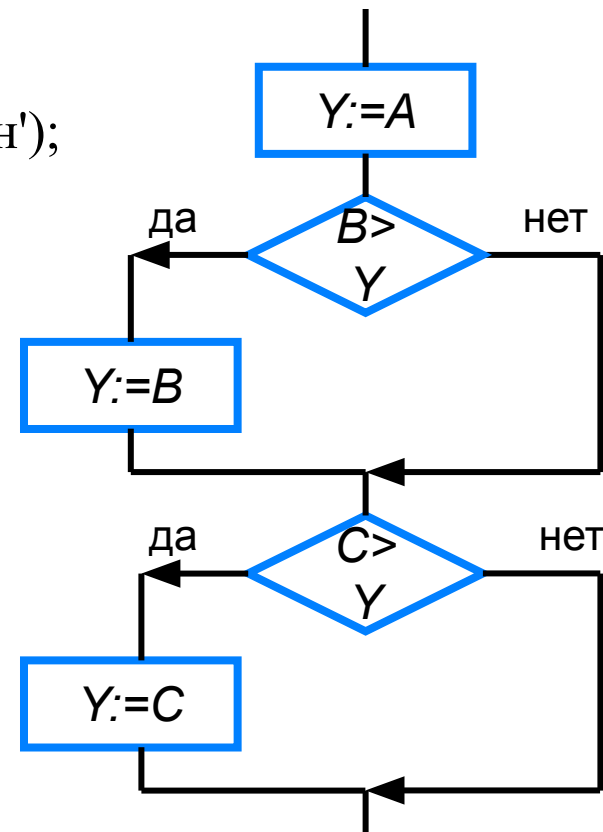
```
  y:=a;
```

```
  if (b>y) then y:=b;
```

```
  if (c>y) then y:=c;
```

```
  writeln ('y=', y)
```

```
end.
```

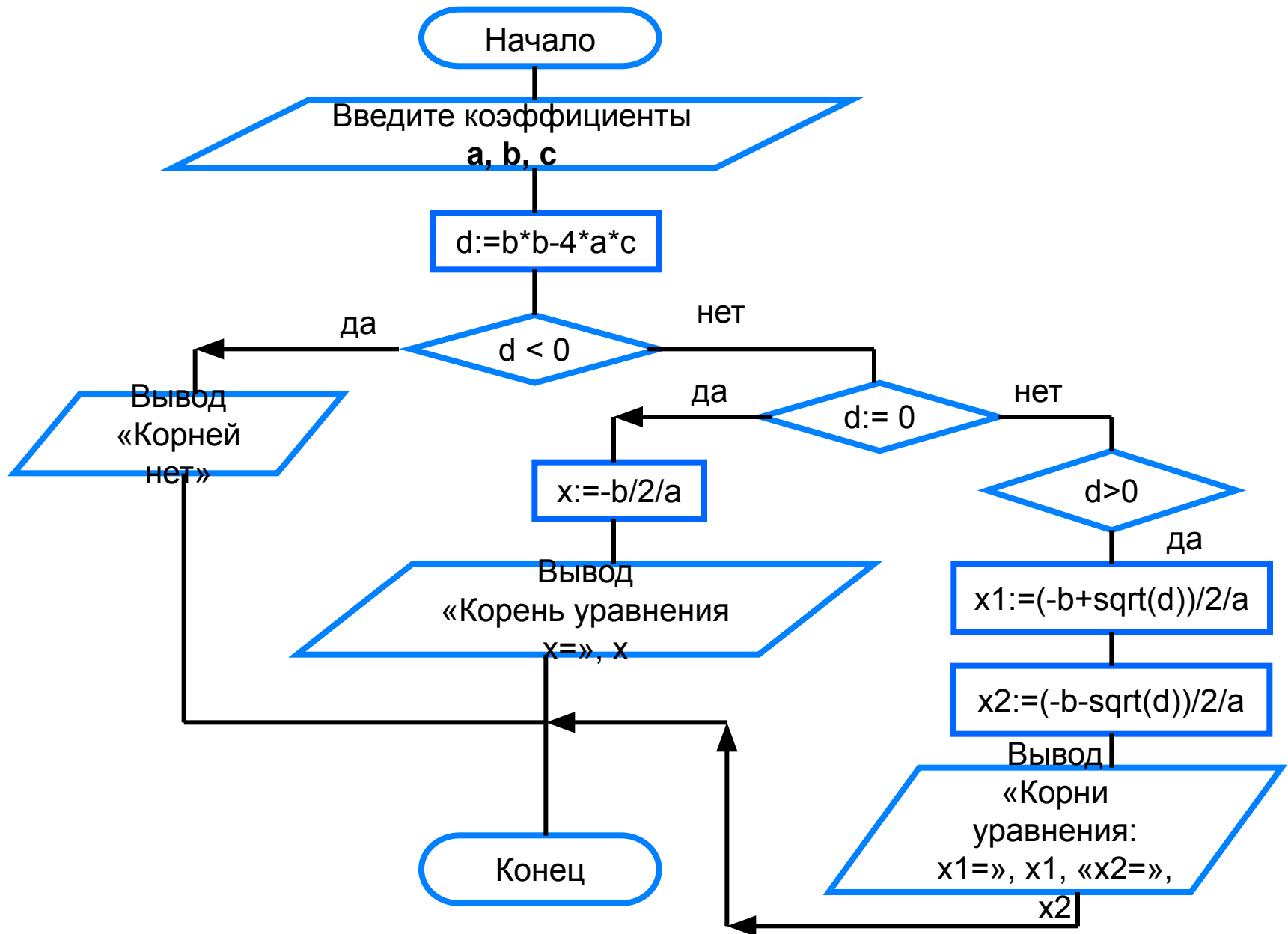


# Составной оператор

В условном операторе и после **then**, и после **else** можно использовать **только один оператор**.

Если в условном операторе после **then** или после **else** нужно выполнить **несколько операторов**, то используют **составной оператор** – конструкцию вида:  
**begin** <последовательность операторов> **end**

# Блок-схема решения КВУР



```
program n_11;  
  var a, b, c: real;  
  var d: real;  
  var x, x1, x2: real;
```

```
begin
```

```
  writeln ('Решение квадратного уравнения');
```

```
  write ('Введите коэффициенты a, b, c >>');
```

```
  readln (a, b, c);
```

```
  d:=b*b-4*a*c;
```

```
  if d<0 then writeln ('Корней нет');
```

```
  if d=0 then
```

```
    begin
```

```
      x:=-b/2/a;
```

```
      writeln ('Корень уравнения x=', x:9:3)
```

```
    end;
```

```
  if d>0 then
```

```
    begin
```

```
      x1:=(-b+sqrt(d))/2/a;
```

```
      x2:=(-b-sqrt(d))/2/a;
```

```
      writeln ('Корни уравнения:');
```

```
      writeln ('x1=', x1:9:3);
```

```
      writeln ('x2=', x2:9:3)
```

```
    end;
```

```
end.
```



# Вложенные ветвления

Возможна следующая конструкция:

```
if <условие1> then
```

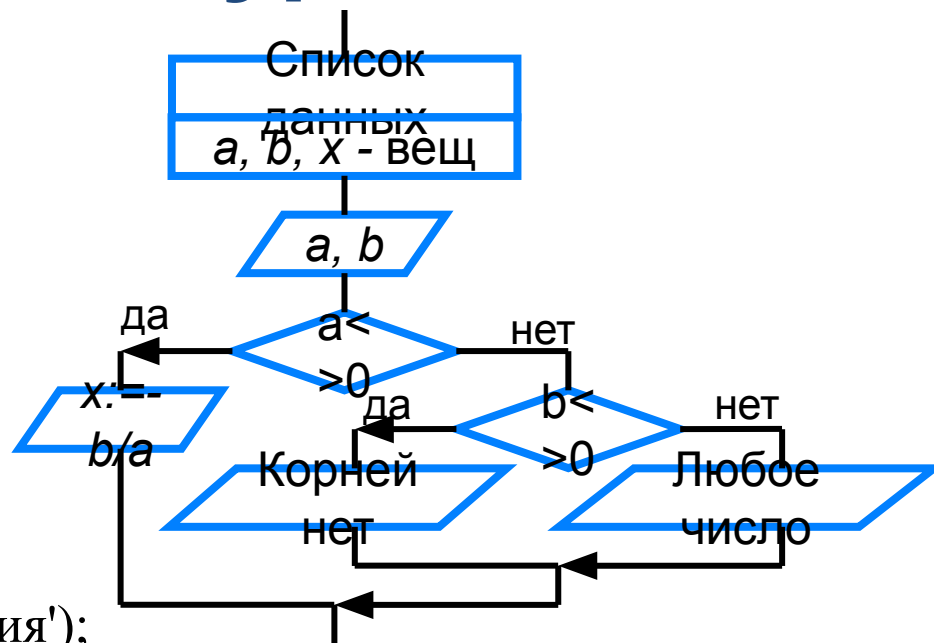
```
    if <условие2> then <оператор1>
```

```
        else <оператор2>
```

```
    else <оператор3>
```

**!** **else** всегда относится к ближайшему оператору **if**

# Решение линейного уравнения



```
program n_12;  
  var a, b, x: real;  
begin
```

```
  writeln ('Решение линейного уравнения');  
  write ('Введите коэффициенты a , b>>');  
  readln (a, b);
```

```
  if a <> 0 then // неравно
```

```
    begin
```

```
      x := -b/a;
```

```
      writeln ('Корень уравнения x=', x:9:3)
```

```
    end
```

```
  else if b <> 0 then writeln ('Корней нет') // неравно
```

```
    else writeln ('x – любое число');
```

```
end.
```

# Самое главное

При записи на языке Паскаль разветвляющихся алгоритмов используют условный оператор:

***if*** <условие> ***then*** <оператор\_1> ***else*** <оператор\_2>

Для записи неполных ветвлений используется сокращённый условный оператор:

***if*** <условие> ***then*** <оператор>

Если при некотором условии требуется выполнить определённую последовательность операторов, то их объединяют в один составной оператор, имеющий вид:

***begin*** <последовательность операторов> ***end.***



# Вопросы и задания

Напишите программу, которая для заданной пары  $(x, y)$  вычисляет значение функции  $f(x, y)$  по формуле:

$$f(x, y) = \begin{cases} 2x + 3y, & \text{если } x \leq 5 \text{ и } y \leq 3 \\ 4x - 2y, & \text{если } x > 5 \text{ и } y > 3 \\ x^2 + y^2, & \text{иначе} \end{cases}$$

где  $x$  и  $y$  – координаты точки. Пример входных данных:  $x=1, y=2$ . Пример выходных данных:  $f(1, 2) = 8$ .

Пример входных данных

Пример выходных данных

Ответ пользователя проверяется и комментируется.

Координаты 1-го поля  $x_1, y_1$

Координаты 2-го поля  $x_2, y_2$

Координаты 3-го поля  $x_3, y_3$

Координаты 4-го поля  $x_4, y_4$

Координаты 5-го поля  $x_5, y_5$

Координаты 6-го поля  $x_6, y_6$

Координаты 7-го поля  $x_7, y_7$

Координаты 8-го поля  $x_8, y_8$

Координаты 9-го поля  $x_9, y_9$

Координаты 10-го поля  $x_{10}, y_{10}$

Примеры входных данных и соответствующих им выходных данных разработайте самостоятельно.

# Опорный конспект

Условный оператор

Полная форма

*if* <условие> *then* <оператор\_1> *else* <оператор\_2>

Неполная форма

*if* <условие> *then* <оператор>

Составной оператор

*begin* <последовательность операторов> *end*