



ПРОГРАММИРОВАНИЕ РАЗВЕТВЛЯЮЩИХСЯ АЛГОРИТМОВ

НАЧАЛА ПРОГРАММИРОВАНИЯ

9 класс

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:

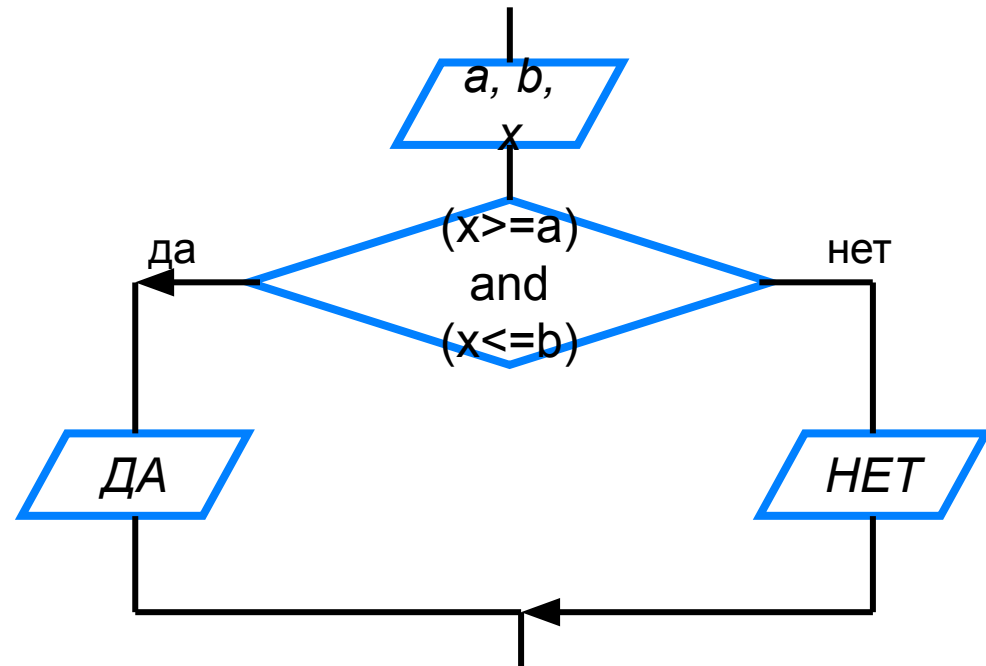
if <условие> **then** <оператор>

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



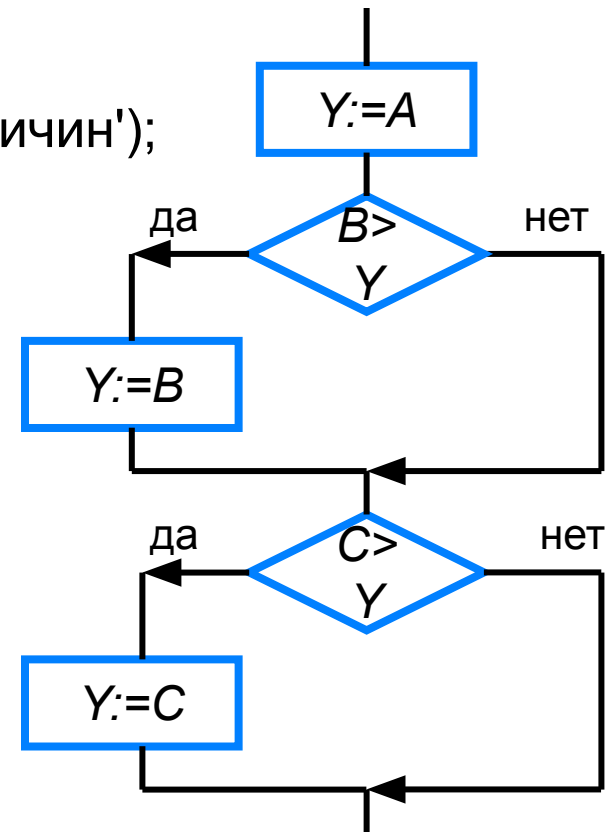
Условный оператор

```
program n_9;  
  var x, a, b: real;  
begin  
  readln (a, b);  
  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.
```

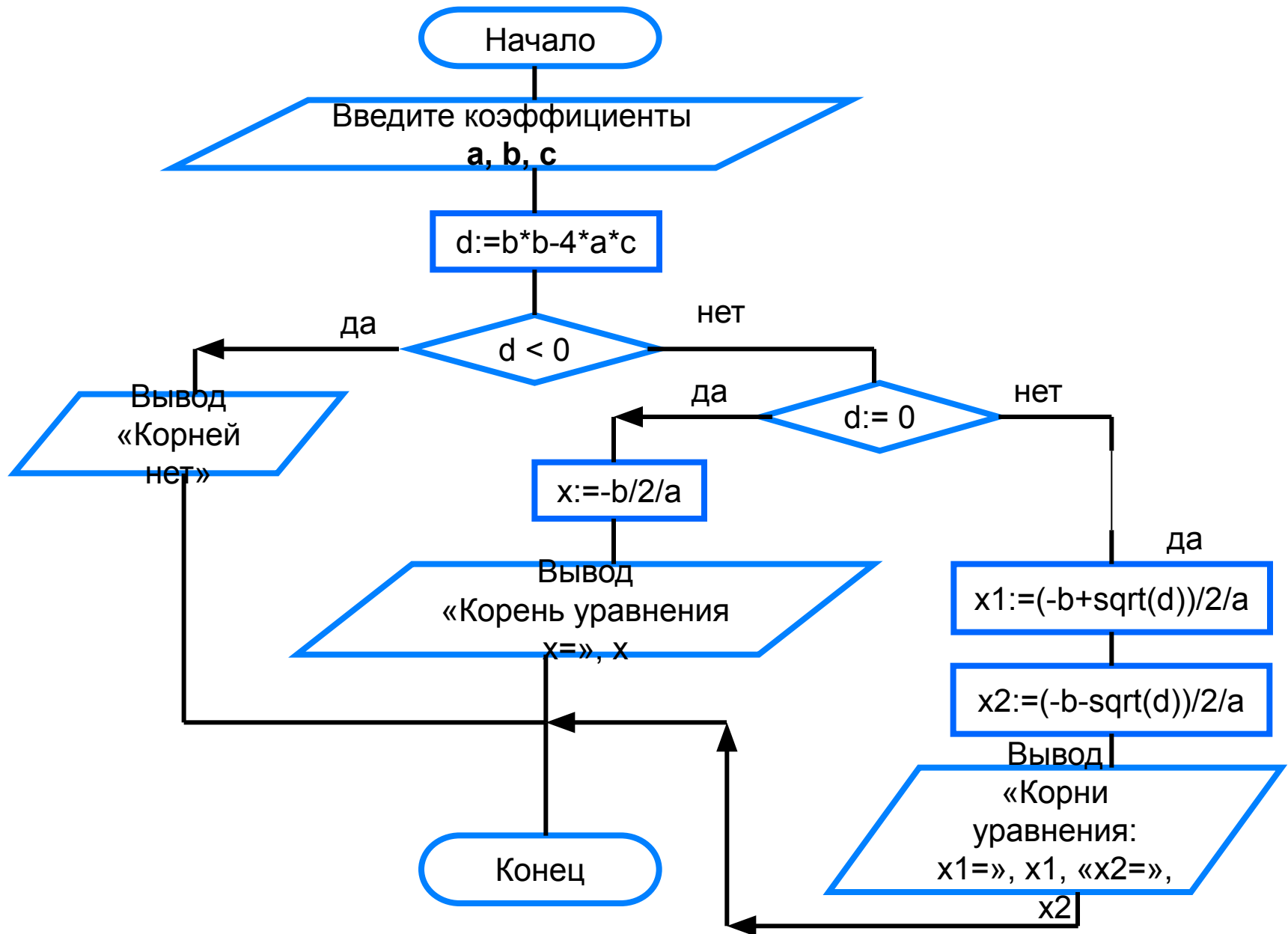


Составной оператор

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)
```

```
    end;
```

```
  if d>0 then
```

```
    begin
```

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

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

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

```
      writeln ('x1=', x1);
```

```
      writeln ('x2=', x2)
```

```
    end;
```

```
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

Является ли условным оператором последовательность символов?

а) if $x < y$ then $x := 0$ else read (y)

б) if $x \geq y$ then $x := 0$; $y := 0$ else write (z)

в) if $x < y < z$ then $a := a + 1$

Вопросы и задания

Используя составной оператор, упростите следующий фрагмент программы:

```
if a>b then c:=1;
```

```
if a>b then d:=2;
```

```
if a<=b then c:=3;
```

```
if a<=b then d:=4
```

Вопросы и задания

Домашнее задание

Дано трёхзначное число.

Напишите программу, которая определяет:

есть ли среди цифр заданного целого трёхзначного числа одинаковые;

Пример входных данных	Пример выходных данных
123	Нет
121	Да
222	Да

Вопросы и задания

Дано трёхзначное число.
Напишите программу, которая определяет:

является ли число «перевёртышем», т. е. числом, десятичная запись которого читается одинаково слева направо и справа налево.

Пример входных данных	Пример выходных данных
123	Нет
121	Перевёртыш
222	Перевёртыш

Вопросы и задания

Домашнее задание

Даны две точки в плоской прямоугольной системе координат. Напишите программу, определяющую, которая из точек находится ближе к началу координат.

Пример входных данных	Пример выходных данных
Координаты 1-й точки >> 1, 2 Координаты 2-й точки >> 3, 4	Первая точка ближе

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*