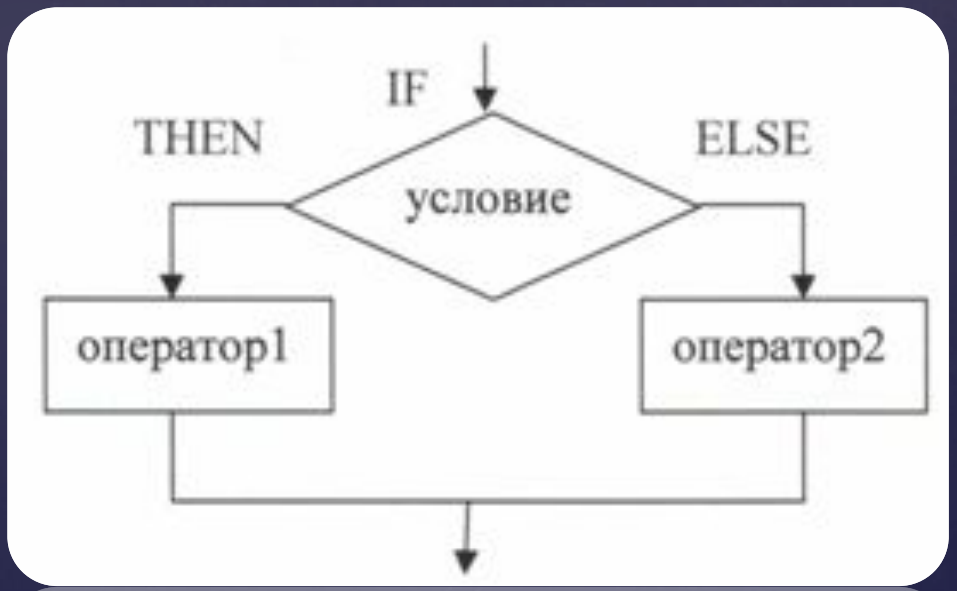
Условный оператор

{ Решение задач.

- Условный оператор реализует «ветвление», т.е. изменяет порядок выполнения операторов в зависимости от истинности или ложности некоторого условия.

IF <*Условие*> THEN <*Оператор1*> ELSE <*Оператор2*>;



Условный оператор.

Условный оператор.

- Условный оператор может иметь две формы (структуры) – полную или неполную. При неполной структуре *“ELSE”* не указывается.

IF ***<условие>*** **THEN** ***<оператор1>***

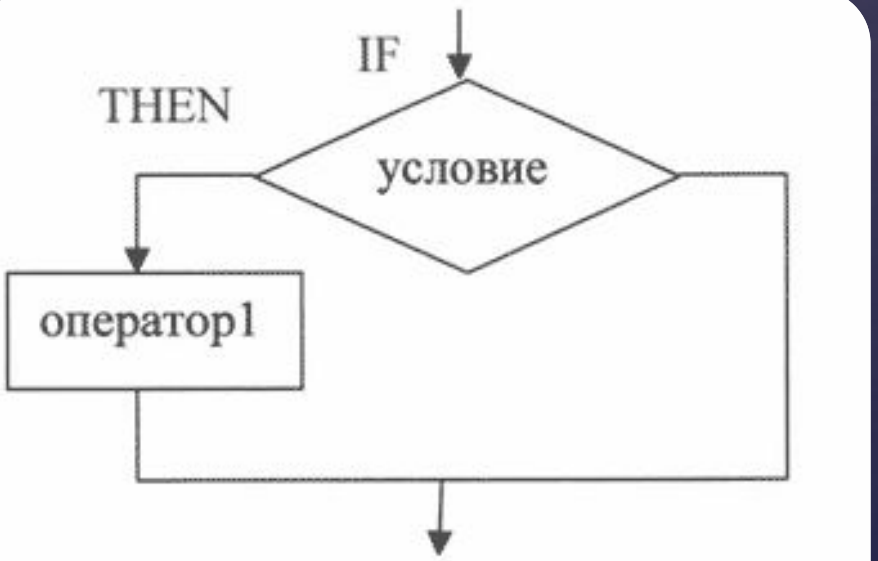


Рис. 2. *Неполная форма условного оператора*

- Условие – это логическое выражение, которое может быть записано в операторе явно или вычислено в программе. Для записи простых условий используются операции отношения:
- Операции отношения:

>	Больше
<	Меньше
=	Равно
>=	Больше либо равно
<=	Меньше либо равно
<>	Не равно

Условный оператор.

- Если по условию задачи после THEN или ELSE необходимо выполнить больше одного оператора, то тогда используют операторные скобки BEGIN…END.

- **IF** *<условие>* **THEN**
  BEGIN
  <оператор 1>;
  <оператор 2>;
  …
  <оператор n>;
  END
  ELSE
  BEGIN
  <оператор 1>;
  <оператор 2>;
  …
  <оператор n>;
  END;

Условный оператор.

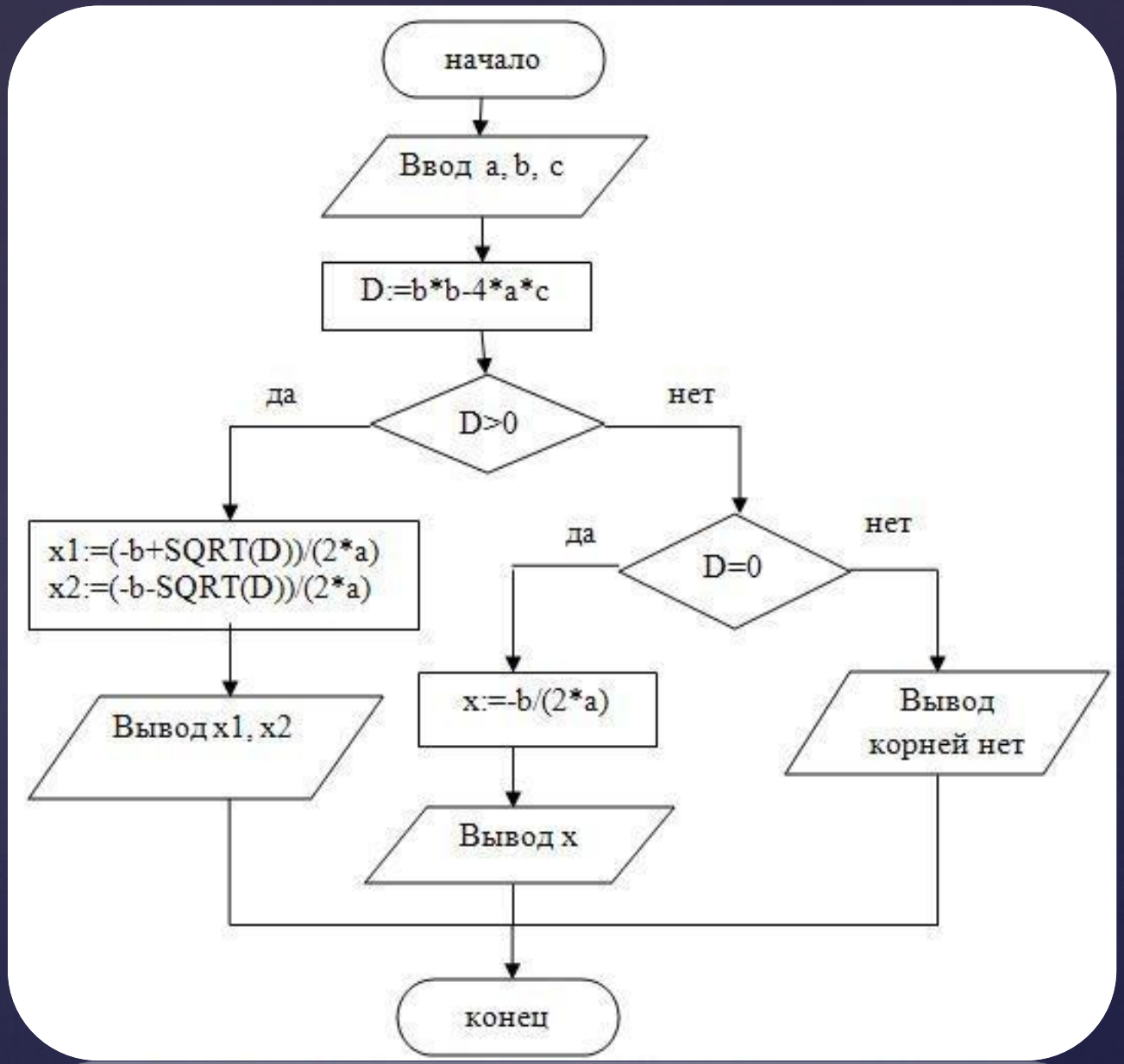
```
PROGRAM  zadacha1;
VAR A,B:INTEGER; //описываем переменные A и B целыми числами
BEGIN
WRITE('A='); //Вводим с клавиатуры числа A и B
READLN(A);
WRITE('B=');
READLN(B);
WRITE ('Большее число : ');
IF A>B THEN WRITELN (A) //Если A>B , то выводим на экран A, иначе выводим B
ELSE WRITELN (B);
END.
```

Условный оператор.

- В качестве оператора в команде ветвления может быть другой условный оператор. В этом случае получаем вложенные ветвления. Рассмотрим на примере.

- **Задача. Составить программу для решения квадратного уравнения $ax^2 + bx + c = 0$.**

Решение задач.



Решение задач.

```
PROGRAM  zadacha2;
VAR a,b,c,D,x,x1,x2:REAL;
BEGIN
      WRITE('a='); READLN(a);
      WRITE('b='); READLN(b);
      WRITE('c='); READLN(c);
      D:= b*b - 4*a*c;
      WRITE ('Корни уравнения: ');
      IF D>0 THEN
            BEGIN
                  x1:=(-b+SQRT(D))/(2*a);
                  x2:=(-b-SQRT(D))/(2*a);
                  WRITELN ('x1= ',x1:5:2,'x2= ',x2:5:2);
            END
            ELSE
                  IF D=0 THEN
                        BEGIN
                              x:= -b/(2*a);
                              WRITELN ('x= ',x:5:2);
                        END
                        ELSE
                              WRITELN ('Корней нет');
END.
```

Решение задач.

- Условие может быть записано и сложным логическим выражением. В сложных логических выражениях используются логические операции:

  AND (И) **OR** (ИЛИ) **NOT** (НЕ).

Отношения, связываемые логическими операциями, заключаются в скобки.

Например, требуется определить, есть ли среди чисел a, b, c хотя бы одно равное нулю. Для этого можно составить следующий условный оператор:

- IF (a=0) **OR** (b=0) **OR** (c=0) THEN WRITE(‘YES’) ELSE WRITE(‘NO’);

Условный оператор.

- Также в сложном условии могут быть использованы и мультипликативные операции:

MOD – остаток от деления,

DIV – целая часть от деления.

- Например, требуется определить, является ли число а – нечетным.

- **IF a MOD 2<>0 THEN WRITELN(‘YES’) ELSE WRITELN(‘NO’);**

Условный оператор.

- **Задача.** Написать программу, проверяющую, принадлежит ли число интервалу (0;5].

- **Решение.** Обозначим за x число, вводимое с клавиатуры пользователем. Принадлежность числа x заданному интервалу определяется следующим условием: 0<x<=5.

- PROGRAM zadacha3;
  VAR x:REAL; {описываем переменную x}
  BEGIN
  WRITE('x='); {Вводим с клавиатуры x}
  READLN(x);
  {Проверяем принадлежность x заданному интервалу}
  IF (x>0) and (x<=5) THEN WRITELN ('YES') ELSE WRITELN ("NO');
  END.

Решение задач.

Домашнее задание

<u>Составить и записать в тетради программу к задаче</u>

- Дано целое число. Определить оканчивается ли оно цифрой 7.