

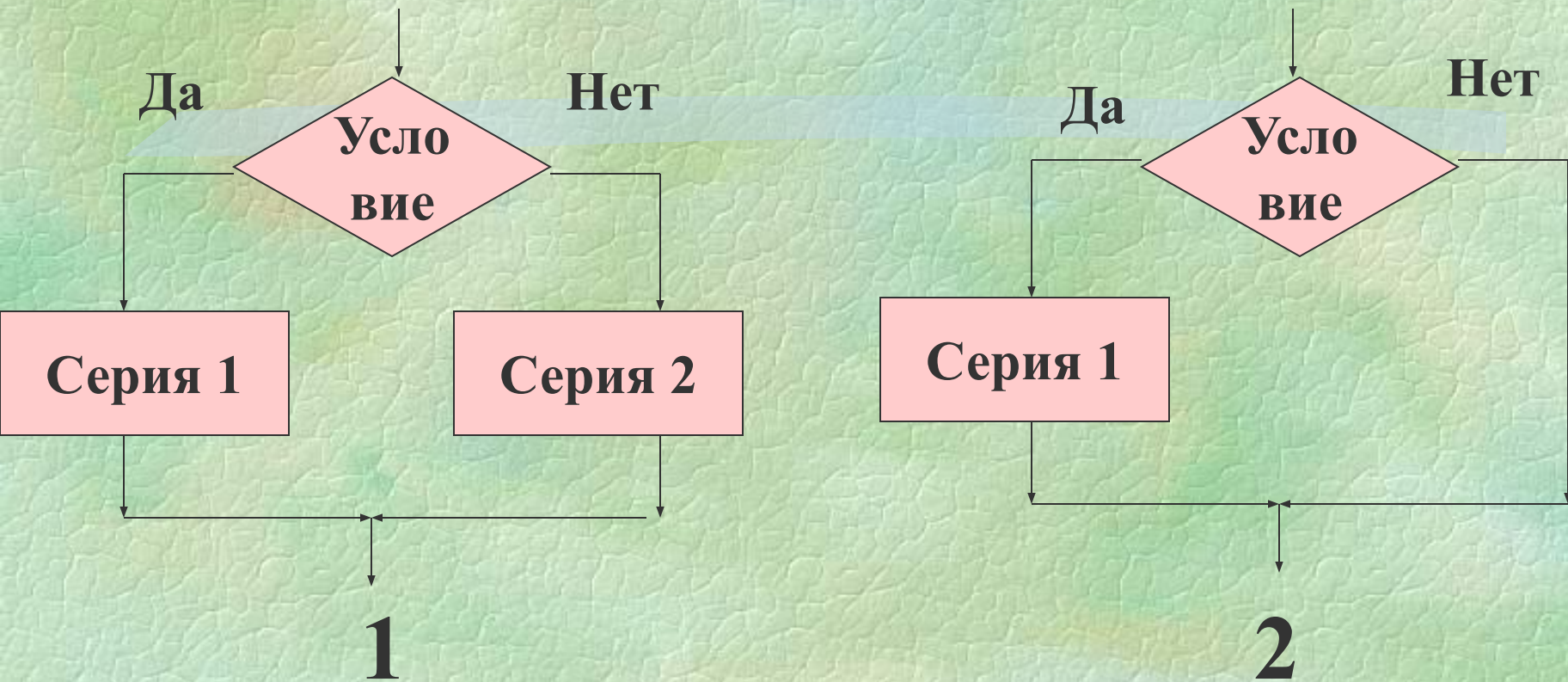
Операторы ветвления

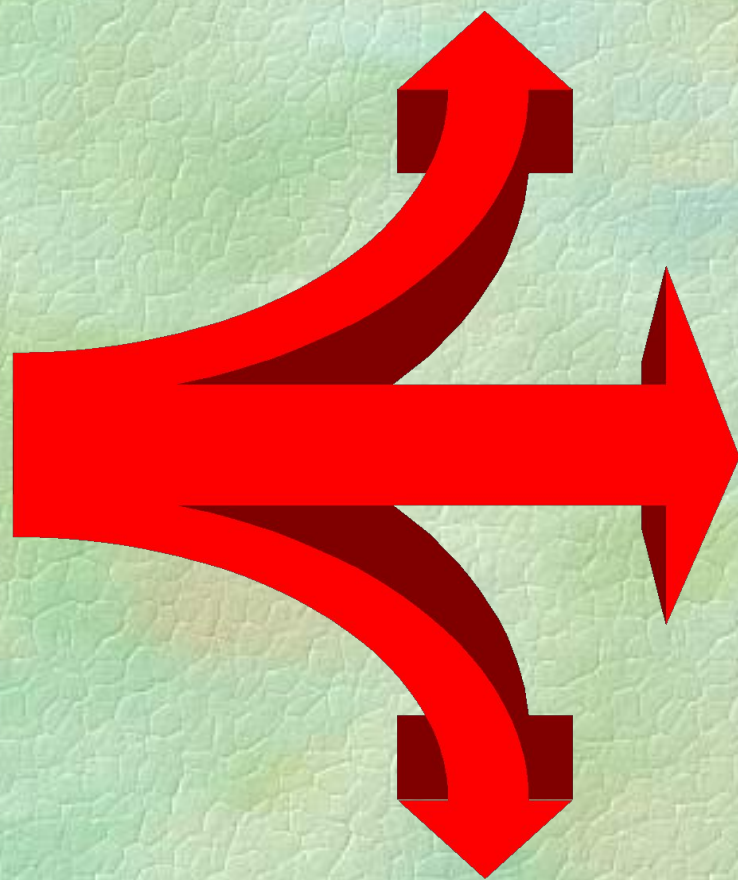
Prezented.Ru

Ветвление - такая форма организации действий, при которой в зависимости от выполнения некоторого условия совершается одна или другая последовательность действий

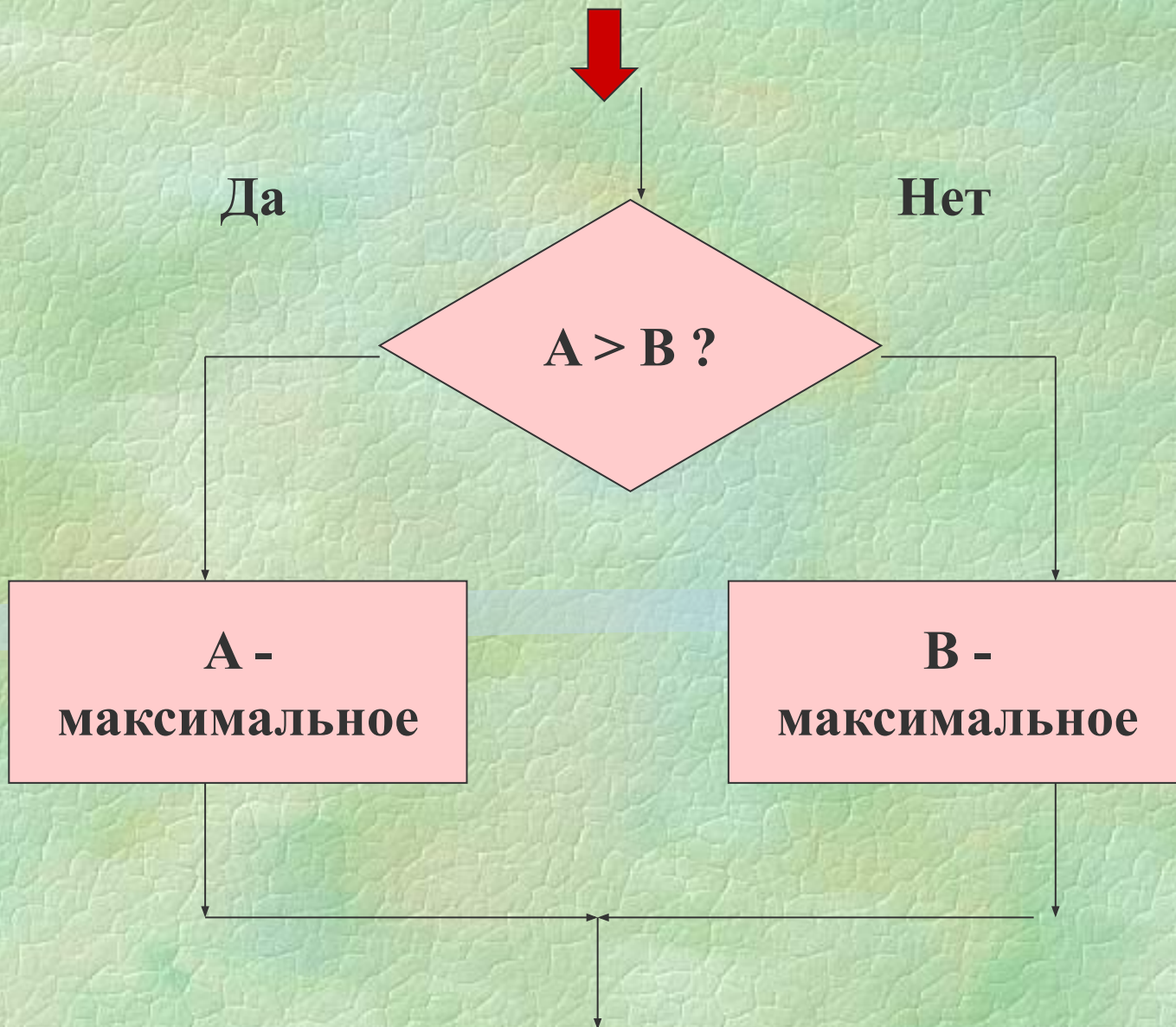
КОМАНДА ВЕТВЛЕНИЯ ИМЕЕТ ПОЛНУЮ (1)

ИЛИ СОКРАЩЕННУЮ ФОРМУ(2)





**Ветвление в
полном
варианте**



Оператор ветвления в полном варианте

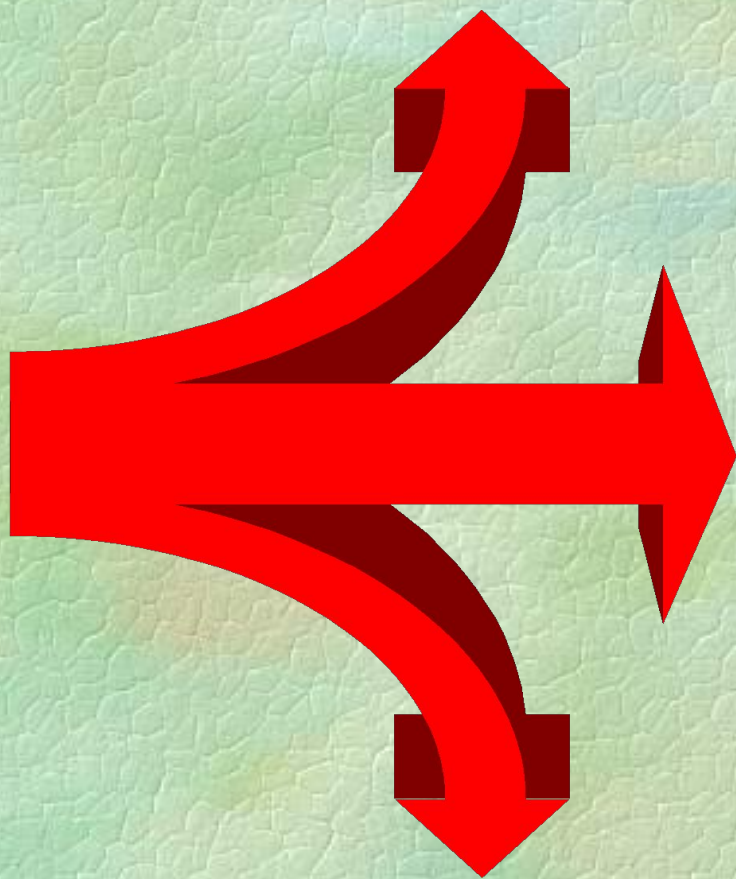
IF (*условие*) **THEN** (*операторы*) **ELSE** (*операторы*);

условие - записывается с помощью знаков сравнения
и принимает два значения: *да* или *нет*

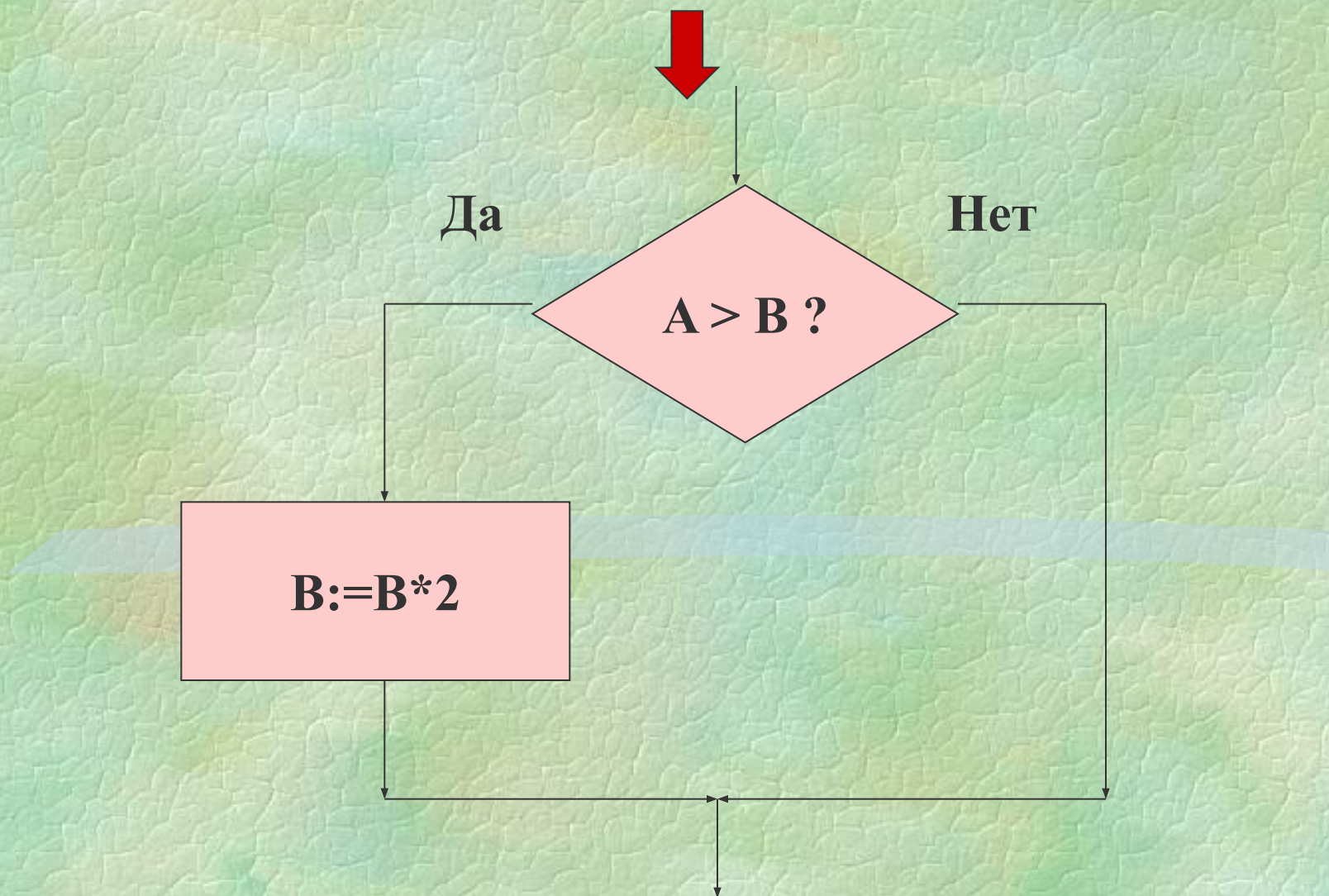
```
IF A>B THEN max:=A ELSE max:=B;
```

Сложные условия составляют из простых с помощью
логических операций (*OR, AND, NOT*)

```
IF (A>B)AND(A>C) THEN max:=A ELSE max:=100;
```

**Ветвление в
неполном
варианте**



Оператор ветвления в неполном варианте

IF (*условие*) **THEN** (*операторы*);

IF $A > B$ **THEN** $B := B * 2$;

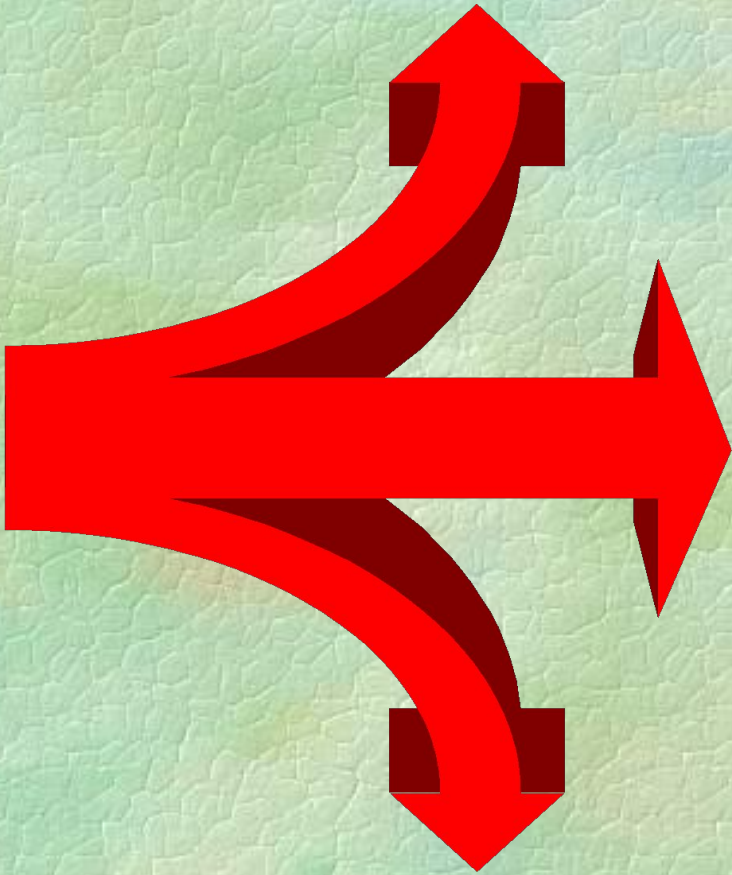
Оператор передачи управления

GOTO метка;

метка - обязательно должна быть описана в блоке описания.
В тексте программы ставится в той строке,
на которую передается управление. После метки
ставится двоеточие.

```
Program neo;  
  uses Graph;  
  var x:integer;  
  label 1;  
begin  
  ...  
  goto 1;  
  ...  
  
1: x:=10;
```


**Программа,
рисующая 20
случайных
окружностей в
правой половине
экрана**




```
Program krugi;  
  Uses Graph;  
  Var x,y,r,z,i,grDriver,grMode:integer;  
  Label 1;  
Begin  
  grDriver:=detect;  
  initGraph(grDriver,grMode,'d:/tp/bgi');  
  randomize;  
  for i:=1 to 20 do  
  begin  
    z:=random(15); y:=random(480); r:=random(100);  
    setcolor(z);  
1:   x:=random(640);  
    if x>340 then circle(x,y,r) else goto 1;  
  end;  
  readln;  
  closeGraph;  
end.
```


Домашнее задание:

п. 7.9 – читать,
составить программу,
рисующую 10 случайных
окружностей в левой
нижней четверти экрана

