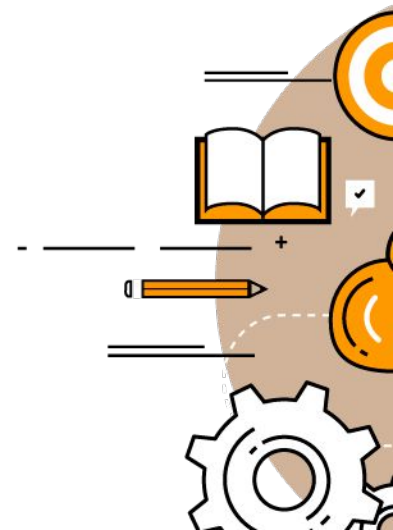
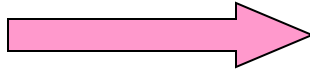




# Алгоритм ветвление

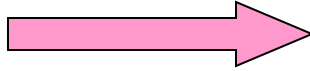


**Начало**



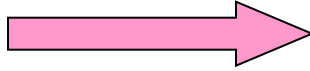
**Начало алгоритма,  
вход в программу**

**Ввод/вывод  
данных**



**Ввод исходных данных  
или вывод результата**

**Команда  
(действие)**



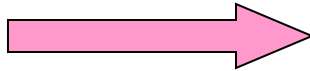
**Выполнение действий**

**Условие**



**Проверка логического  
условия**

**Счетчик  
цикла**



**Количество  
повторов цикла**

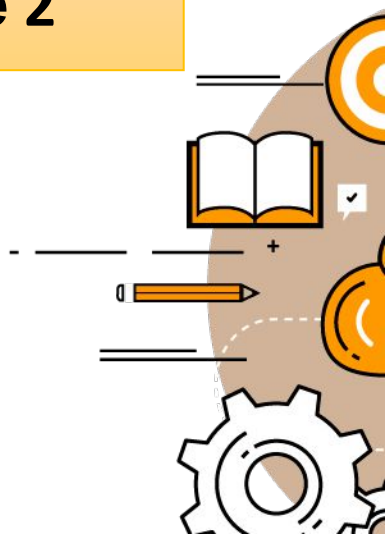
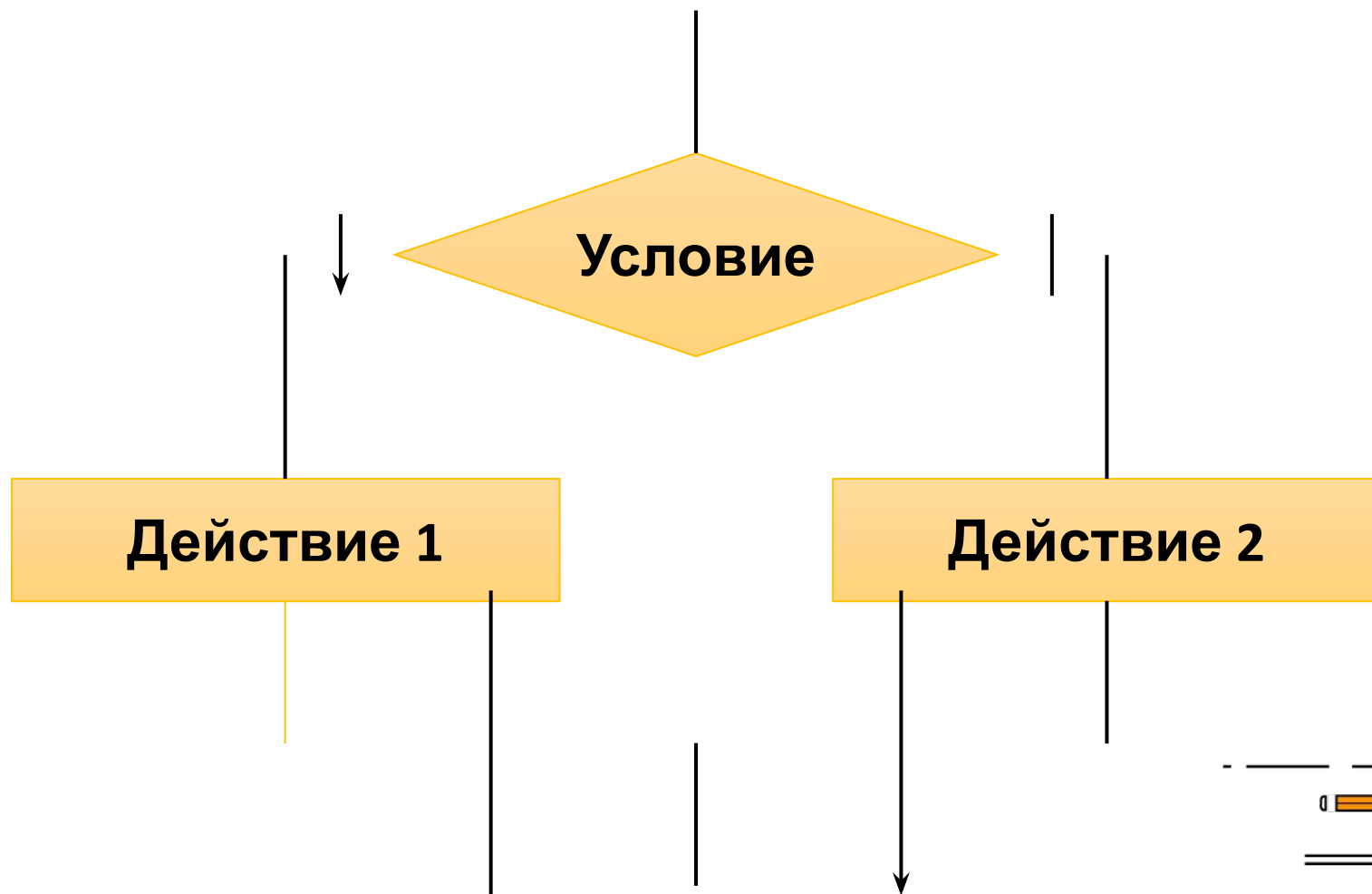
**Конец**



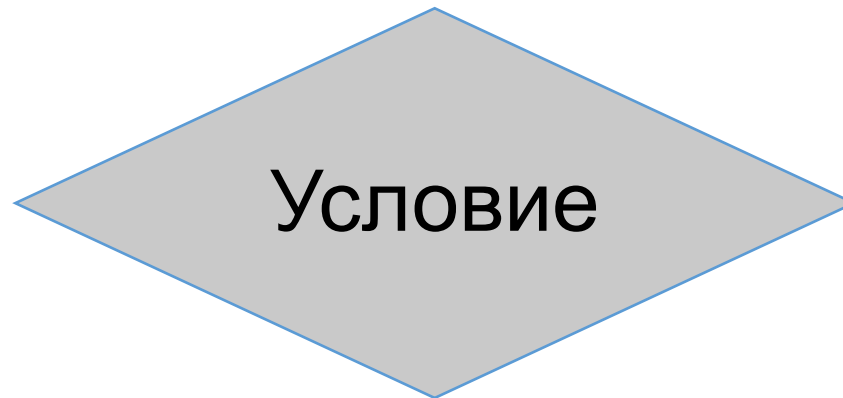
**Конец алгоритма,  
выход из программы**



# Общий вид блок-схем алгоритмической структуры “ветвление”



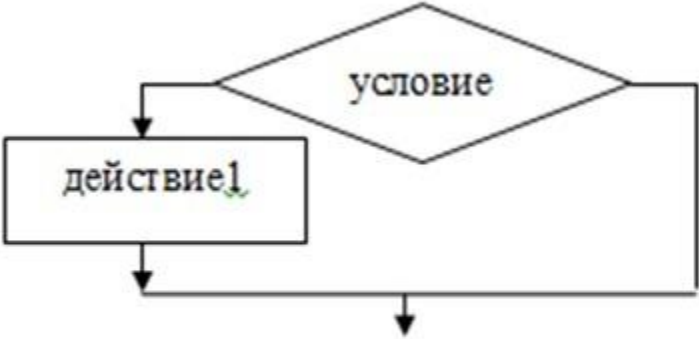

- **Разветвляющимся** называется алгоритм в котором порядок выполнения действий зависит от некоторого условия.



В Паскале ветвление организуется с помощью двух операторов: **IF** и **CASE**.



# Оператор IF

Не полная форма	Полная форма
Если...то...	Если... то... иначе...
	
<b>На языке Паскаль команда ветвления записывается сл. образом:</b>	
<code>if&lt;условие&gt; then&lt;оператор1&gt;;</code>	<code>if&lt;условие&gt; then&lt;оператор1&gt; else&lt;оператор2&gt;;</code>



Составить алгоритм планирования выходного дня студентом: если будет хорошая погода, он пойдет гулять, а если плохая – будет писать реферат.

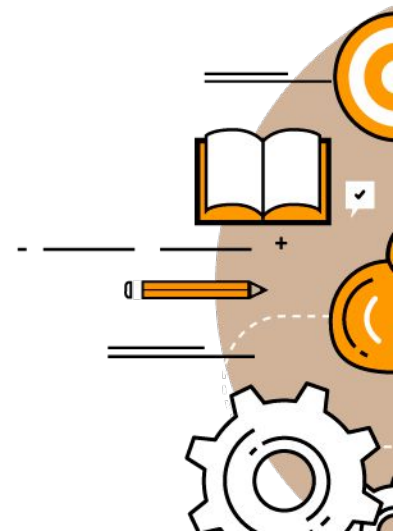
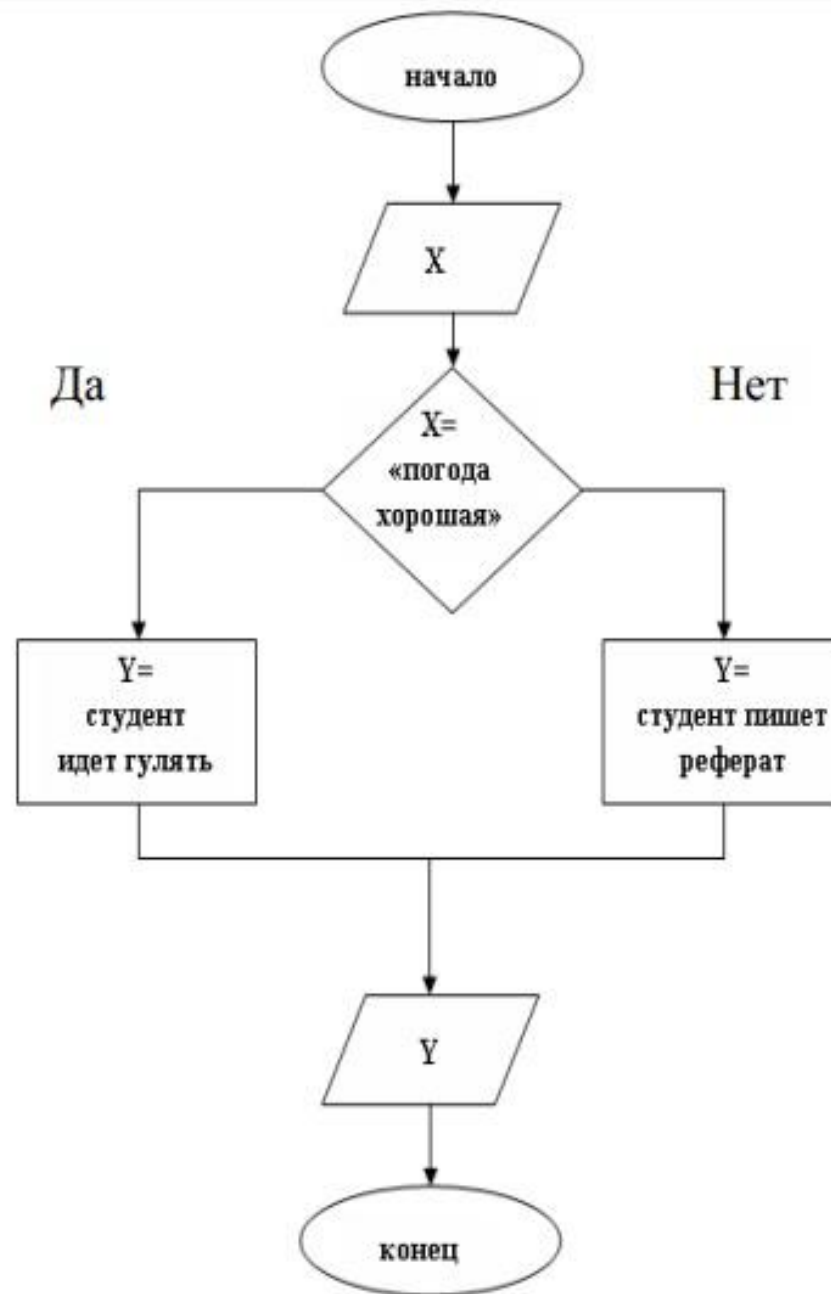
Входные данные:  $x$  (информация о погоде);

Выходные данные:  $y$  (результат прошедшего выходного дня).

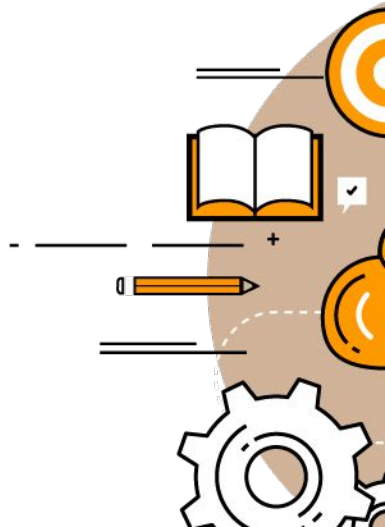
### Псевдокод

1. Ввод  $X$  (информация о погоде в выходные дни)
2. Проверка условия:  
**Если**  $X$ =«Погода хорошая»  
    **то**  $Y$ =«студент идет гулять»,  
    **иначе**  $Y$ =«студент пишет реферат»  
**конец если**
3. Вывод  $Y$
4. Конец



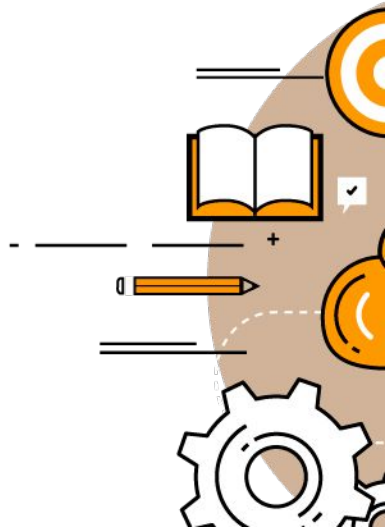
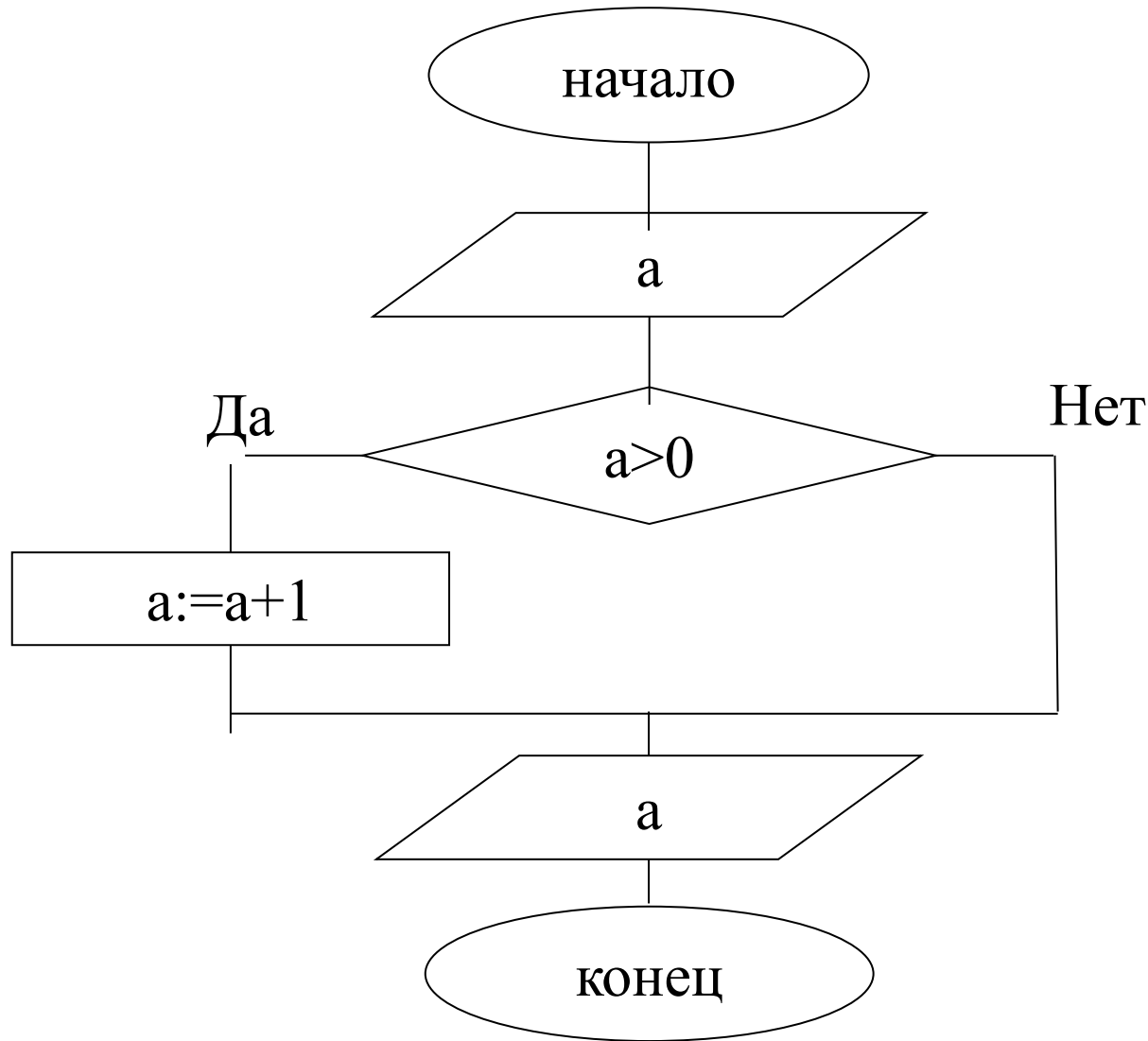


```
program Student;  
var x, y;  
begin  
writeln('Хорошая погода?');  
read ({Да});  
if Y:={Да} then Y:={Студент идет гулять}  
      else Y:={Студент пишет реферат};  
writeln ('Как студент провел свой выходной?',  
Y);  
end.
```





**Задача №1.** Дано целое число. Если оно является положительным, то прибавить к нему 1; в противном случае не изменять его. Вывести полученное число.



# Решение:

```
Program zadacha;
```

```
Var
```

```
a :integer;
```

```
Begin
```

```
Writeln('введите целое число a ');
```

```
readln(a);
```

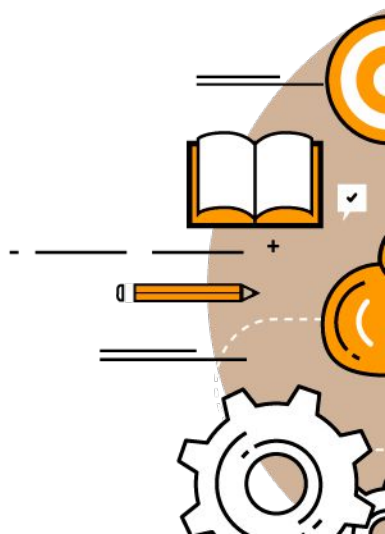
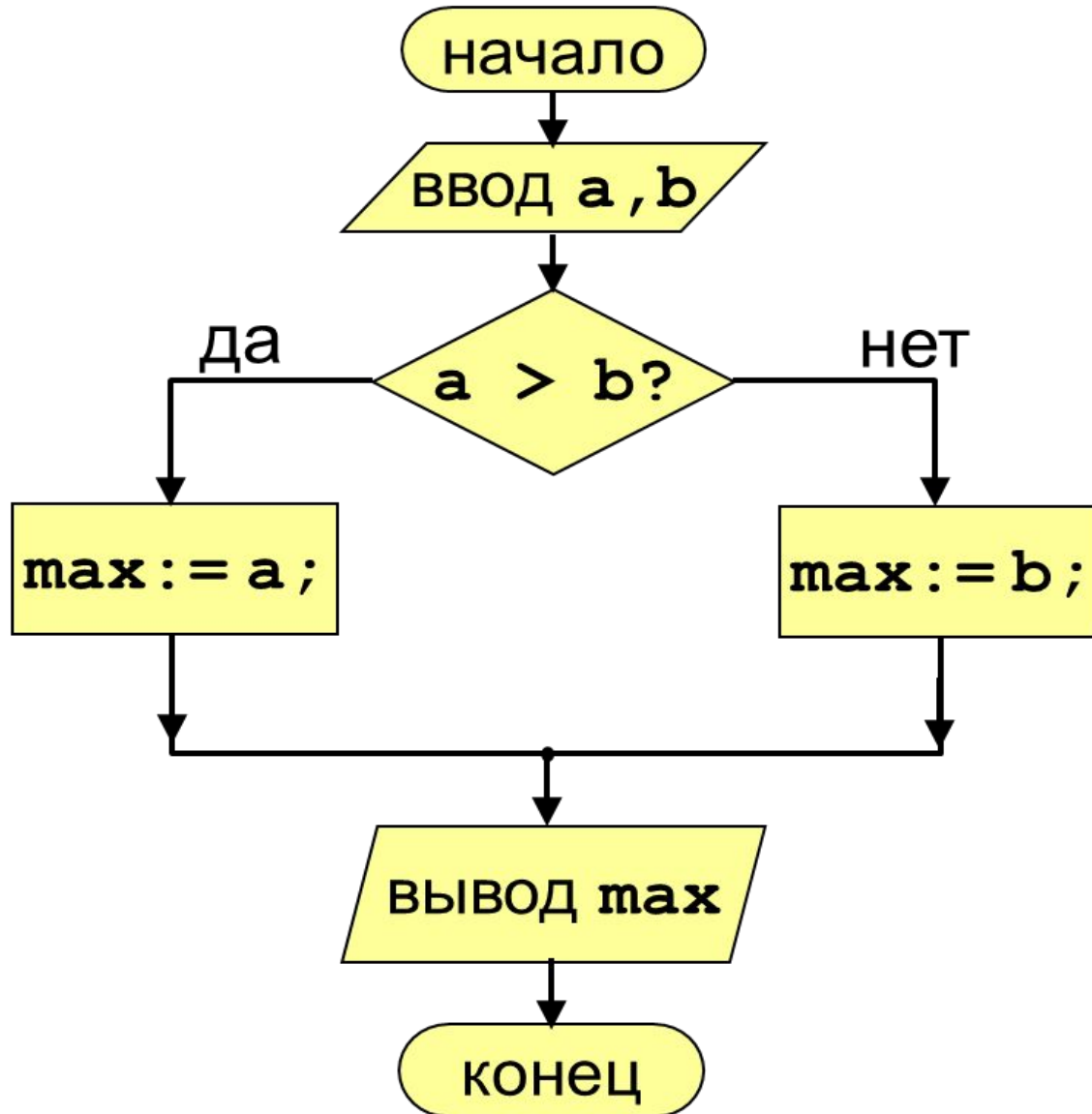
```
if a>0 then a:=a+1;
```

```
Writeln('Полученное число ', a);
```

```
end.
```



**Задача №2.** Дано два целых числа. Вывести на экран наибольшее из них.



# Решение:

```
program primer2;  
var a, b, max: integer;  
begin  
  writeln('Введите два целых числа');  
  read (a, b);  
  if a>b then max:=a  
    else max:=b;  
  writeln ('Наибольшее из введенных чисел  
,max);  
end.
```

