

Оператор безусловного перехода **GOTO**

Оператор **GOTO** служит
для безусловных
переходов в программах

**Оператор безусловного
перехода**

Общий вид оператора:

GOTO метка;

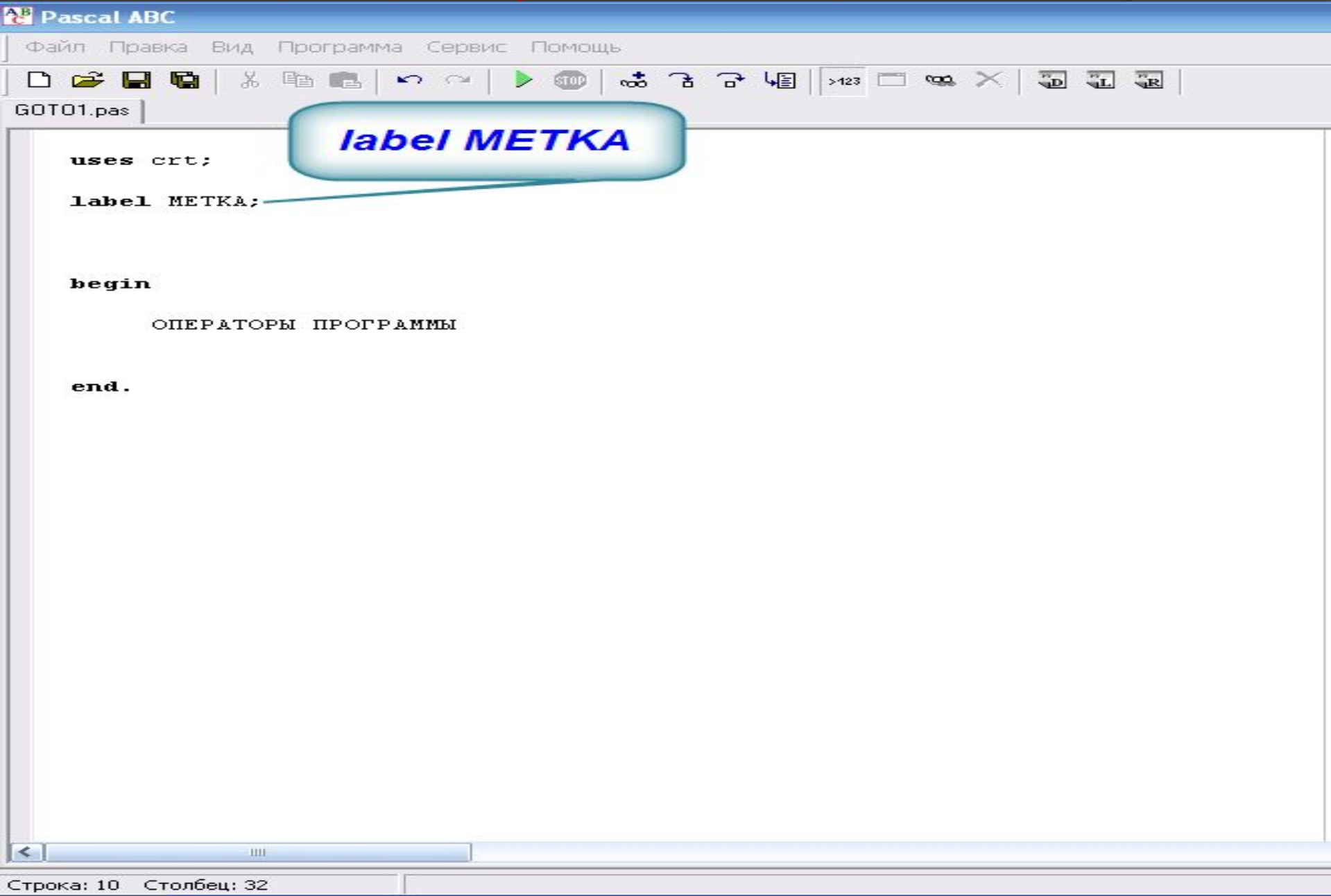
**метка представляет собой число
от 1 до 9999 или
последовательность латинских
букв и цифр**

**Оператор безусловного
перехода**

goto

Метка объявляется в
разделе описания меток,
который следует после
раздела описания модулей

Оператор безусловного перехода GOTO



The screenshot shows a Pascal ABC IDE window titled "Pascal ABC". The menu bar includes "Файл", "Правка", "Вид", "Программа", "Сервис", and "Помощь". The toolbar contains various icons for file operations and execution. The main editor area displays the following Pascal code:

```
GOTO1.pas  
  
uses crt;  
  
label МЕТКА;  
  
begin  
    ОПЕРАТОРЫ ПРОГРАММЫ  
  
end.
```

A blue callout box with the text "label МЕТКА" points to the label declaration in the code. The status bar at the bottom indicates "Строка: 10" and "Столбец: 32".

Оператор безусловного перехода GOTO

Метка записывается
перед помечаемым
оператором и
отделяется от него
двоеточием

Оператор безусловного перехода GOTO

Pascal ABC

Файл Правка Вид Программа Сервис Помощь



GOTO1.pas

label МЕТКА

```
uses crt;
```

```
label МЕТКА;
```

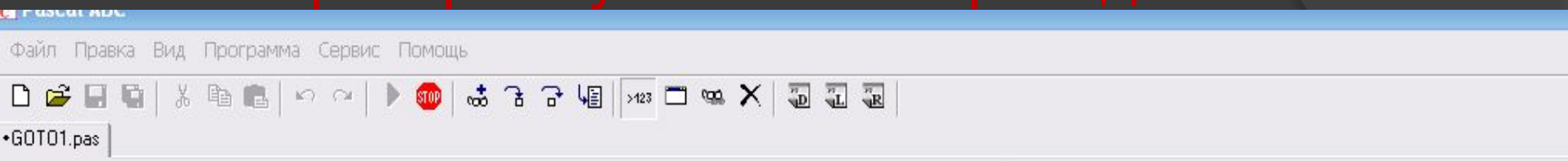
```
begin
```

```
    МЕТКА: writeln ('Информатика');
```

```
    goto МЕТКА;
```

```
end.
```

Оператор безусловного перехода GOTO



```
uses crt;
```

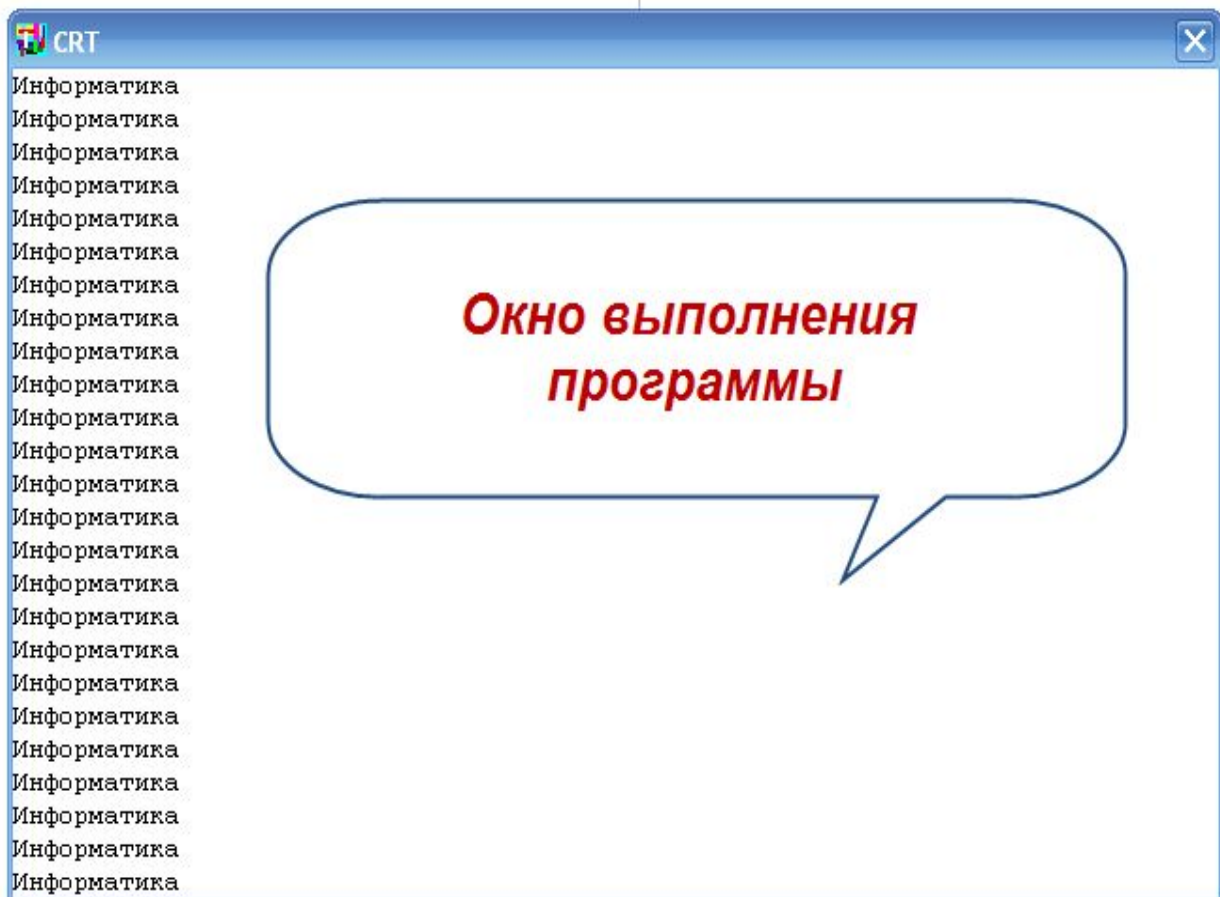
```
label METKA;
```

```
begin
```

```
    METKA:writeln ('Информатика');
```

```
    goto METKA;
```

```
end.
```



Оператор безусловного перехода GOTO

Составить программу для
вычисления:

$$x := 12 * b / (2 - a)$$

Оператор безусловного перехода GOTO

```
uses crt;  
var a,b,x:real;  
  
begin  
  read (a,b);  
  if (a=2)then  
    writeln ('Нет решений')  
  else  
    begin  
      writeln ('Есть решения');  
      x:=12*b/(2-a);  
      writeln (x);  
    end;  
end.
```

Оператор безусловного перехода GOTO

The image shows a screenshot of the Pascal ABC IDE. The main window displays the source code of a program named 'If_else.pas'. The code is as follows:

```
uses crt;  
var a,b,x:real;  
  
begin  
  read (a,b);  
  if (a=2) then  
    writeln ('Нет решений')  
  else  
    begin  
      writeln ('Есть решения');  
      x:=12*b/(2-a);  
      writeln (x);  
    end;  
end.
```

A secondary window titled 'CRT - программа завершена' is open, showing the output of the program. The output consists of four lines:

```
1  
2  
Есть решения  
24
```

Оператор безусловного перехода GOTO

The image shows a screenshot of the Pascal ABC IDE. The main window displays a Pascal program named `if_else.pas`. The program uses the `GOTO` operator to skip the `else` branch when `a=2`.

```
uses crt;  
var a,b,x:real;  
  
begin  
  read (a,b);  
  if (a=2) then  
    writeln ('Нет решений')  
  else  
    begin  
      writeln ('Есть решения');  
      x:=12*b/(2-a);  
      writeln (x);  
    end;  
end.
```

An output window titled "CRT - программа завершена" shows the execution results. It displays the input values `2` and `1`, followed by the output `Нет решений` and a horizontal line.

Оператор безусловного перехода GOTO

The image shows a Pascal ABC IDE window titled "Pascal ABC" with a menu bar (Файл, Правка, Вид, Программа, Сервис, Помощь) and a toolbar. The main editor displays the following Pascal code:

```
uses crt;  
  
var a,b,x:real;  
  
label METKA;  
  
begin  
    METKA:read (a,b);  
  
    if (a=2)then  
        begin  
            writeln ('Нет решений');  
  
            writeln ('Введите другое значение a');  
  
            goto METKA;  
  
        end  
    else  
        begin  
            writeln ('Есть решения');  
  
            x:=12*b/(2-a);  
  
            writeln (x);  
  
        end;  
  
end;  
end.
```

Overlaid on the right side of the IDE is a window titled "CRT - программа завершена". This window shows the program's execution output with red arrows indicating the flow of control:

```
2  
1  
Нет решений  
Введите другое значение a  
1  
2  
Есть решения  
24  
-
```

The red arrows show the following execution path: starting at line 1, it goes to line 2, then to the first 'writeln' statement, then to the 'writeln' statement for 'Введите другое значение a', then to the input '1', then to the second 'writeln' statement, then to the calculation and output of '24', and finally to the end of the program. The 'goto METKA;' statement causes a jump from the 'writeln' statement for 'Введите другое значение a' back to the input '1'.

A red callout box with a white background and a drop shadow is positioned over the bottom right of the CRT window. It contains the text:

Окно выполнения программы

Оператор безусловного перехода GOTO

Домашние
задание:

Составит
программу для

вычисления

$$y = 4x / 12b + 10$$

```
uses crt;
label 1;
var a,b,x:real;

begin
  1:read (a,b);

  if (a=2) then
    begin
      writeln ('Нет решений');
      writeln ('Введите ещё раз');
      goto 1;
    end
  else
    begin
      writeln ('Есть решения');
      x:=12*b/(2-a);
      writeln (x);
    end;
end.
```