A decorative border surrounds the page, featuring a mix of black, white, and yellow floral motifs, including daisies and stylized flowers, intertwined with black and yellow swirls and circular patterns. The background is a light green textured surface with black splatters and swirls.

Автор: Зубкова Екатерина Александровна  
ГБОУ СОШ №1378 г. Москва

# ОПЕРАТОР ПЕРЕХОДА GOTO

## ЦИКЛ МЕТКИ

Язык программирования Pascal  
9 класс

# ПОНЯТИЕ ЦИКЛА

Цикл – главное средство заставить компьютер много раз сделать одно и тоже или похожее.

С помощью цикла можно создать повторяющиеся узоры (рис.1), эффект движения, выполнить расчеты по таблицам и многое другое.



Рис.1. Образец циклических композиций



Рассмотрим фрагмент программы:

```
Write ('Это ');  
Write ('тело ');  
Write ('цикла');  
Write (' ');
```



В случае, если эта часть программы выполнялась бы бесконечно, то на экране мы бы увидели:

*Это тело цикла      Это тело цикла      Это тело цикла      ...*

Задача: как изменить порядок выполнения операторов, чтобы после оператора Write (' ') выполнялся оператор Write ('Это ') и программа выводила бы нужную нам строчку?

# ОПЕРАТОР ПЕРЕХОДА

GOTO – оператор перехода (читается – «гоуту», переводится как «иди к»);

Метка – это произвольное имя или произвольное не слишком большое целое положительное число.

Пример программы:

```
LABEL m1;
```

```
BEGIN
```

```
  m1: Write ('Это');
```

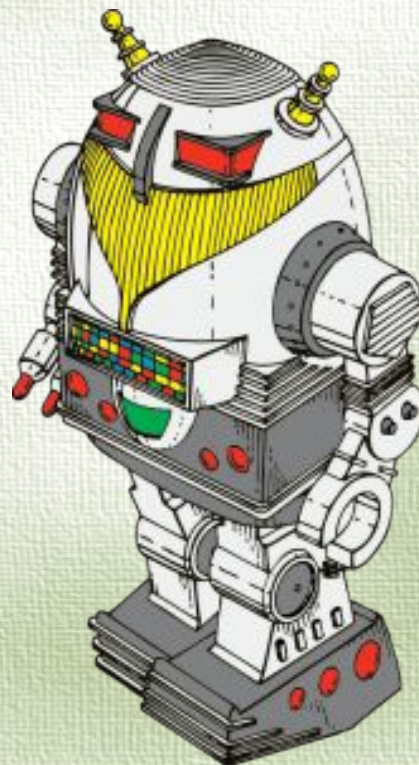
```
      Write ('тело');
```

```
      Write ('цикла');
```

```
      Write ('');
```

```
      Goto m1
```

```
END.
```



# ПРАВИЛА РАБОТЫ С ОПЕРАТОРОМ GOTO

- Оператор GOTO можно писать в любых местах программы, и метку можно ставить перед любым оператором, заставляя компьютер таким образом перескакивать в программе отсюда угодно куда угодно;
- Метка должна отделяться от оператора двоеточием;
- Метки, встречающиеся в программе, должны быть описаны выше BEGIN после слова LABEL (читается – «лэйбл», переводится как «метка»).



# ПРИМЕР ИСПОЛЬЗОВАНИЯ ОПЕРАТОРА GOTO

```

LABEL 8;
VAR a, k: integer;
BEGIN
  k:=6;
  a:=100;
  Goto 8;
  a:=a+k;
  k:=2*k;
  WriteLn (a);
  8: a:=a+1;
  k:=k+10;
  WriteLn (k,' ',a);
END.
```



Результат работы программы: **16 101**

# САМОСТОЯТЕЛЬНАЯ РАБОТА НА КОМПЬЮТЕРЕ

Написать программы для выполнения  
следующих заданий:

1. Бесконечно печатать букву А:  
АААААААААА.....
2. Бесконечно печатать 1000 999 998 997  
996.....
3. Бесконечно печатать 100 50 25 12.5..... с  
восьмью десятичными знаками.



# ВЫХОД ИЗ ЦИКЛА С ПОМОЩЬЮ IF

## *Вариант 1*

```
LABEL m;  
VAR f: integer;  
BEGIN  
    Write ('Начало счета ');  
    f:=3;  
m: Write (f,' ');  
    f:=f+2;  
    if f<=9 then goto m;  
    Write ('Конец счета');  
END.
```



## *Вариант 2*

```
LABEL m1, m2;  
VAR f: integer;  
BEGIN  
    Write ('Начало счета ');  
    f:=3;  
m1: Write (f,' ');  
    f:=f+2;  
    if f>9 then goto m2  
        else goto m1;  
m2: Write ('Конец счета');  
END.
```



# ВЫХОД ИЗ ЦИКЛА С ПОМОЩЬЮ IF

*Вариант 3*

```
LABEL m1, m2, m3;  
VAR f: integer;  
BEGIN  
    Write ('Начало счета ');  
    f:=3;  
m1: if f<=9 then goto m3  
     else goto m2;  
m3: Write (f, ' ');  
     f:=f+2;  
     goto m1;  
m2: Write ('Конец счета');  
END.
```



# РЕШИ САМОСТОЯТЕЛЬНО

Написать программы решения следующих задач:

1. Напечатать 1 2 3 4 ... 99 100;

2. Напечатать 100 99 ... 3 2 1;

3. Напечатать 1 2 3 4 ... 99 100 99 ... 3 2 1.

