



# **Организация ввода и вывода данных**

# Вывод данных

Вывод данных из оперативной памяти на экран монитора:

**write** (<выражение 1> ,< выражение 2> , ...,< выражение N>)

СПИСОК ВЫВОДА

**Выражения** - символьные, числовые, логические,  
в том числе переменные и константы

**Пример:**

**write** ('s=', s);      Для s=15 на экране будет: s=15.



**Информация в кавычках выводится на экран  
без изменений**

# Варианты организации вывода

Вариант организации вывода	Оператор вывода	Результат
Без разделителей	<code>write (1, 20, 300);</code>	120300
Добавить разделители – запятые	<code>write (1, ',', 20, ',', 300);</code>	1, 20, 300
Добавить разделители – пробелы	<code>write (1, ' ', 2, ' ', 300);</code>	1 20 300

# Формат вывода

**Формат вывода** позволяет установить количество позиций на экране, занимаемых выводимой величиной.

**write (s:x:y)**

**x** - общее количество позиций, отводимых под число;

**y** - количество позиций в дробной части числа.

Оператор вывода	Результат выполнения оператора
<b>write ('s=', s:2:0);</b>	s=15
<b>write ('s=', s:3:1);</b>	s= 15.0
<b>write ('s=', s:5:2);</b>	s= 15.00

**writeln** - вывод с новой строки!

# Вывод данных с новой строки

**writeln** (<выражение 1> ,< выражение 2> , ...,< выражение N>)

СПИСОК ВЫВОДА

**Выражения** - символьные, числовые, логические,  
в том числе переменные и константы

***Пример:***

```
writeln ('s=', s);
```

Для s=15 на экране будет:

s=15

и курсор переместится на новую строку



# Ввод данных с клавиатуры

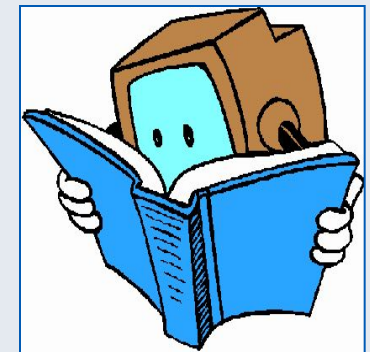
Ввод в оперативную память значений переменных :

**read** (<имя переменной1>, ..., <имя переменной N> );

список ввода

Выполнение оператора **read**:

- 1) компьютер переходит в режим ожидания данных:
- 2) пользователь вводит данные с клавиатуры:
  - *несколько значений переменных числовых типов могут вводиться через пробел или через запятую;*
  - *при вводе символьных переменных пробел и запятую ставить нельзя;*
- 3) пользователь нажимает клавишу **Enter**.



# Ввод данных с клавиатуры

**!** *Типы вводимых значений* должны *соответствовать типам переменных*, указанных в разделе описания переменных.

```
var i,j:integer; x:real; a:char;  
read (i, j, x, a);
```

варианты ввода данных:

```
1 0 2.5 A<Enter>          1<Enter>  
                           0<Enter>  
                           2.5<Enter>  
                           A<Enter>
```

Uses Crt;

```
const pi=3.14;
```

```
var r, c, s: real;
```

```
begin
```

```
  writeln('Вычисление длины окружности и площади круга');
```

```
  write('Введите r>>');
```

```
  readln(r);
```

```
  c:=2*pi*r;
```

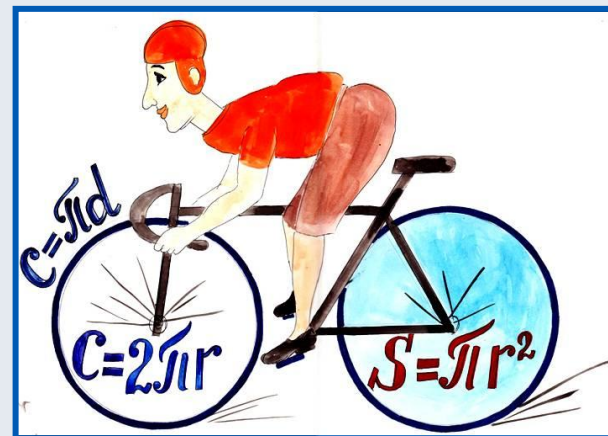
```
  s:=pi*r*r;
```

```
  writeln('c =', c:6:4);
```

```
  writeln('s=', s:6:4);
```

```
  readln
```

```
end.
```



Результат работы программы:

```
Turbo Pascal 7.0
File Edit Search Run Compile Debug Tools Options W
[ ] \DOCUME~1\WUJU\PULPIT\45.PAS

Вычисление длины окружности и площади круга
Введите r>> 8.5
c =53.3800
s =226.8650
```



# Вопросы и задания

Какой тип имеет переменная, если после выполнения программы выводится значение 20, а не 30?

Запишите оператор, обеспечивающий ввод в программу вограниченного времени, а оператор, обеспечивающий ввод произвольного времени.

Запишите оператор, обеспечивающий ввод в программу вограниченного времени, а оператор, обеспечивающий ввод произвольного времени.

Какие числа будут выведены на экран компьютера?

- а) 20 10 30
- б) 30 20 10
- в) write (a = 2 a)
- г) 10 30 20

# Домашнее задание

Постройте блок-схему и напишите программу вычисления площади ( **$S$** ) прямоугольника по двум его сторонам ( **$a$**  и  **$b$** ). В программе предусмотреть ввод исходных данных с клавиатуры.