



ОРГАНИЗАЦИЯ ВВОДА И ВЫВОДА ДАННЫХ

НАЧАЛА ПРОГРАММИРОВАНИЯ

8 класс

Ключевые слова

- оператор вывода `writer`
- формат вывода
- оператор ввода `read`



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).



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
Разделители – запятые	<code>write (1, ', ', 20, ', ', 300)</code>	1, 20, 300
Разделители – пробелы	<code>write (1, ' ', 2, ' ', 3)</code>	1 20 300

Формат вывода

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

write (s:x:y)

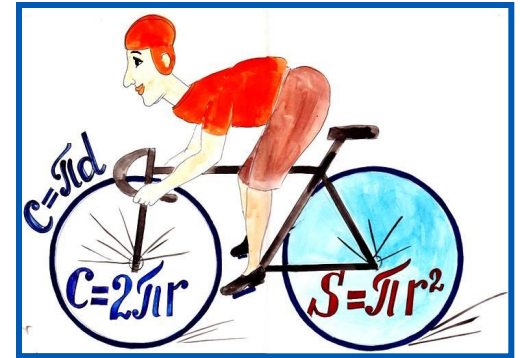
x - общее количество позиций, отводимых под число;

y - количество позиций в дробной части числа.

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

writeln - вывод с новой строки!

Первая программа



```
program n_1;  
  const pi=3.14;  
  var r, c, s: real;  
begin  
  r:=5.4;  
  c :=2*pi*r;  
  s:=pi*r*r;  
  writeln ('c =', c:6:4);  
  writeln ('s=', s:6:4)  
end.
```

A screenshot of the Pascal ABC IDE. The window title is "Pascal ABC". The menu bar includes "Файл", "Правка", "Вид", "Программа", "Сервис", and "Помощь". The toolbar contains icons for file operations, editing, and execution. The file name is "Program1.pas". The code editor shows the same Pascal program as in the previous block. The output window at the bottom shows the results of the program execution.

```
program n_1;  
  const pi=3.14;  
  var r, c, s: real;  
begin  
  r:=5.4;  
  c :=2*pi*r;  
  s:=pi*r*r;  
  writeln ('c =', c:6:4);  
  writeln ('s=', s:6:4)
```

c =33.9120
s=91.5624

Ввод данных с клавиатуры

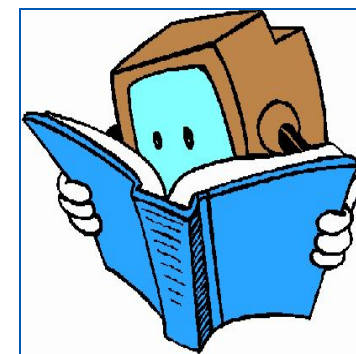
Ввод в оперативную память значений переменных :

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 организации входного потока:

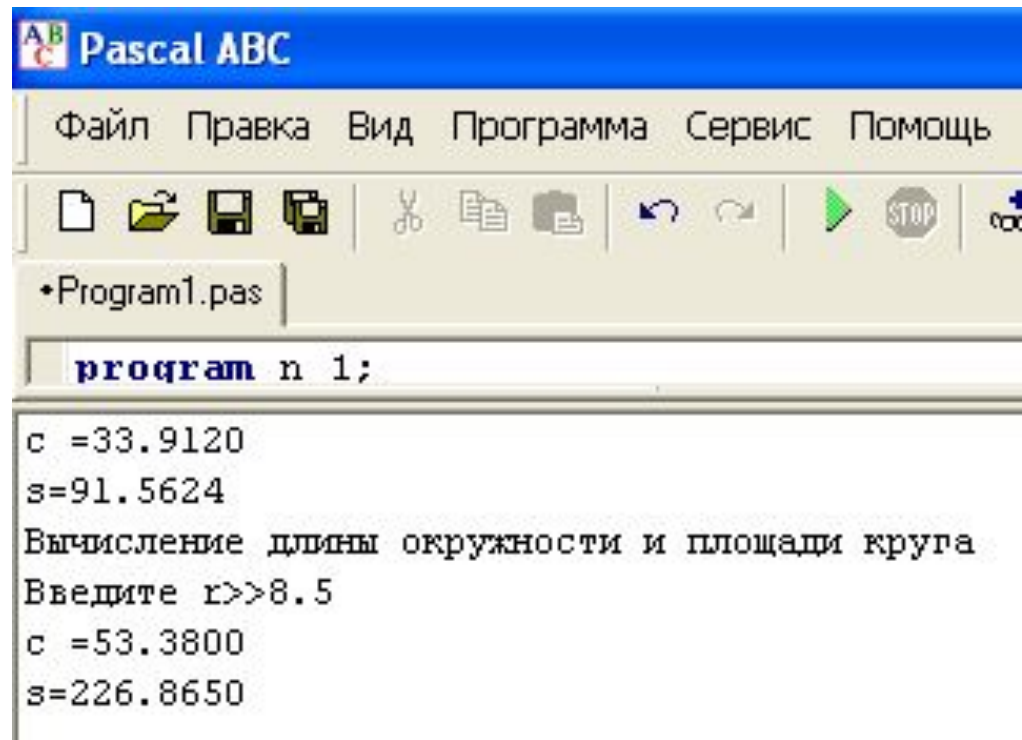
```
1 0 2.5 A<Enter>    1,0 <Enter>    1<Enter>  
                2.5, A<Enter>    0<Enter>  
                        2.5<Enter>  
                        A<Enter>
```

После выполнения оператора **readln** курсор переходит на новую строку.



Улучшенная программа

```
program n_1;  
  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:7:4)  
end.
```



Самое главное

Для **ввода** в оперативную память значений переменных используются операторы ввода ***read*** и ***readln***.

Для **вывода данных** из оперативной памяти на экран монитора используются операторы вывода ***write*** и ***writeln***.

Ввод исходных данных и вывод результатов должны быть организованы ***понятно и удобно***; это обеспечивает дружелюбность пользовательского интерфейса.



Вопросы и задания

Какой тип имеет переменная f , если после выполнения
оператора `write(f)` на экран было выведено следующее
прочитайте операторы, отвечающий вводу
Залишите оператор ввода, соответствующий вводу
Какой тип имеет переменная f , если после выполнения

- а) `write(25 a)`
а) `write(25 a)`
в) `write(25 a)`
в) `write(25 a)`

Опорный конспект

Ввод в оперативную память значений переменных выполняют операторы ввода ***read*** и ***readln***.

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

список ввода

Вывод данных из оперативной памяти на экран монитора выполняют операторы вывода ***write*** и ***writeln***.

write <выражение 1> ,< выражение 2> , ...,< выражение N>

список вывода

Источники информации

1. http://files.school-collection.edu.ru/dlrstore/d1a6e9b7-5eda-4be9-bff2-3197b9f145e7/9_77.swf - команда ввода-вывода