

*Символьные и
строковые
величины*

* Команды ввода

- * `Read(x,y)` - ввод значений переменных x , y . Числовые значения могут быть введены через пробел.
- * `Readln(x,y)` - ввод значений переменных x , y . Числовые значения вводятся по одному в строке (после каждого значения нажимается клавиша Enter).

* Команды вывода

- * `Write(x,y)` - вывод значений переменных `x`, `y`. Разделитель между значениями не ставится.
- * `Writeln(x,y)` - вывод значений переменных `x`, `y`. После вывода всех значений курсор переводится в следующую строку.

* Команды вывода

* `Writeln('введите два числа')` - вывод строковой константы, значение которой записано в апострофах.

* Команда ветвления

Полная форма

If <условие> then
 <команда 1>

Else

 <команда 2>;

Сокращенная форма

If <условие> then
 <команда 1>;

* Команда цикла

* Цикл с
предусловием

While <условие> **do**
 <команда>

* Цикл с
параметром

For $i:=A$ **to** B **do**
 <команда>

* Описание символов и строк

Тип данных	Описание
Char	Хранит один символ кодовой таблицы
String	Хранит значение строки символов кодовой таблицы

* Значения строковых и символьных констант записываются в апострофах:

* C:='\$';

* S:='Привет';

* Writeln ('сумма чисел равна');

* If c='+' then ...

* For c:='a' to 'z' do ...

*Задание 1

*Написать программу,
которая выведет на экран
слово Привет.

*Разбор задания 1

Поскольку программа должна вывести постоянную величину (слово Привет), то опишем ее как константу с именем slovo, а затем выведем значение с помощью команды write.

Программа:

```
const slovo='Привет';  
begin  
  writeln(slovo);  
end.
```

Команда write может выводить также значения и неописанных констант. Можно изменить в программе команду write на следующую:

```
writeln('Привет');
```

и удалить строку `const slovo='Привет';`

*Задание 2

Напишите программу, которая спросит имя пользователя, а затем поздоровается с ним.

*Разбор задания 2

Программа:

```
var imja:string;  
begin  
  writeln ('Как тебя зовут? ');  
  readln(imja);  
  writeln ('Привет, ', imja);  
end.
```

Тестирование

Если запустить программу и ввести значение - Маша, то результат должен быть следующим:

Привет, Маша

Правило сравнения СИМВОЛОВ

Символы сравниваются в порядке, называемом **лексикографическим**.

Это порядок расположения символов в кодовой таблице.

Каждому символу соответствует числовое значение - код символа. Именно эти значения и сравниваются при сравнении символов.

Больше будет тот символ, код которого больше.

Символы русского алфавита, набранные в одном регистре упорядочены в алфавитном порядке;

'Б' < 'Д'; 'п' < 'т';

Символы латинского алфавита, набранные в одном регистре упорядочены в алфавитном порядке;

'S' < 'Z'; 'f' < 'v'

Большие буквы русского алфавита меньше
маленьких букв русского алфавита;

'Б' < 'б'; 'Я' < 'а'

Большие буквы латинского алфавита меньше
маленьких букв латинского алфавита;

'R' < 'r'; 'Z' < 'a'

Символы цифр расположены в порядке возрастания их числовых значений;

'1' < '5'; '0' < '9'

Символы цифр меньше символов любых букв;

'1' < 'r'; '2' < 'W'; '5' < 'ж'; '9' < 'Я'

Символы латинских букв меньше символов русских букв;

'z' < 'б'; 'W' < 'Ц'; 'D' < 'ж'; 'q' < 'Я'

Символы различных знаков могут быть как больше букв (цифр), так и меньше

Правило сравнения строк

Строки сравниваются в лексикографическом порядке посимвольно.

- * Сравнивают первые символы, больше будет та строка, первый символ которой больше;
- * Если первые символы равны, то сравнивают вторые, третьи и т.д.
- * Если все символы двух строк равны, то строки равны.
- * Если одна из строк является началом другой, то вторая строка больше.

Примеры сравнения строк

Первая строка	Вторая строка	Результат
'Четверг'	'Пятница'	Первая больше
'Понедельник'	'Пятница'	Вторая больше
'111111'	'111111'	Равны
'Лес'	'Леска'	Вторая больше

Программа сравнения двух СИМВОЛОВ

```
var a,b:char;  
begin  
  writeln('введите первый символ');  
  readln(a);  
  writeln('введите второй символ');  
  readln(b);  
  if a>b then  
    writeln(a, '>', b)  
  else  
    if a<b then  
      writeln(a, '<', b)  
    else  
      writeln(a, '=', b)  
end.
```

Функции для работы с символами

Название функции	Действие	Тип аргументов	Тип результата
Chr(x)	Определяет символ по его номеру в кодовой таблице	Word	Char
Ord(c)	Определяет код символа в кодовой таблице	Char	Word

Задача 3

Определить код
введенного символа.

Программа:

```
var c:char;  
      x:integer;  
begin  
  writeln('введите символ');  
  readln(c);  
  x:=ord(c);  
  writeln('код=',x);  
end.
```

Функции работы со строками и символами

Название функции	Действие	Тип аргументов	Тип результата
Length (s)	Возвращает длину строки <i>s</i>	string	integer
Pos (s1,s2)	Ищет первое вхождение строки <i>s1</i> в строке <i>s2</i>	<i>s1, s2</i> – string	integer
Uppcase(c)	Преобразует символ в верхний регистр	char	char
LowCase(c)	Преобразует символ в нижний регистр	char	char

Примеры использования функций.

d:=Length ('Светит солнце');	d=13
d:=Length ('Красота');	d=7
c:=Ord('A'); или c:=OrdAnsi('A');	c=65
n:=Chr(63); или n:=ChrAnsi(63);	n='?'
S:='Шла Саша по шоссе' N:=Pos('Саша',S);	N=5
c:=Uppcase('h')	C=H

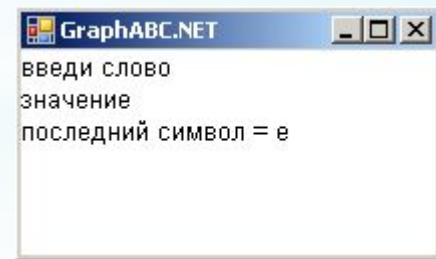
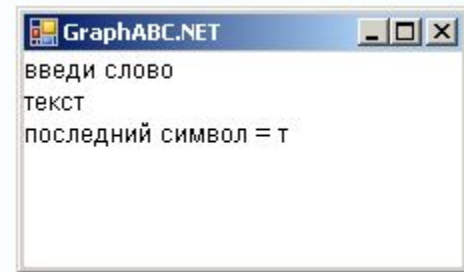
Задача

Дано слово. Вывести на экран его последний символ

Программа

```
var c:char;  
    s:string;  
    n:byte;  
begin  
    writeln('введи слово');  
    readln(s);  
    n:=length(s);  
    c:=s[n];  
    writeln('последний символ =', c);  
end.
```

Результаты



Определение позиции СИМВОЛА

- * Первый символ: $s[1]$;
- * Второй символ: $s[2]$;
- * Предпоследний символ: $s[\text{length}(s)-1]$;
- * Средний символ: $s[\text{length}(s) \text{ div } 2]$ или $s[\text{length}(s) \text{ div } 2+1]$;

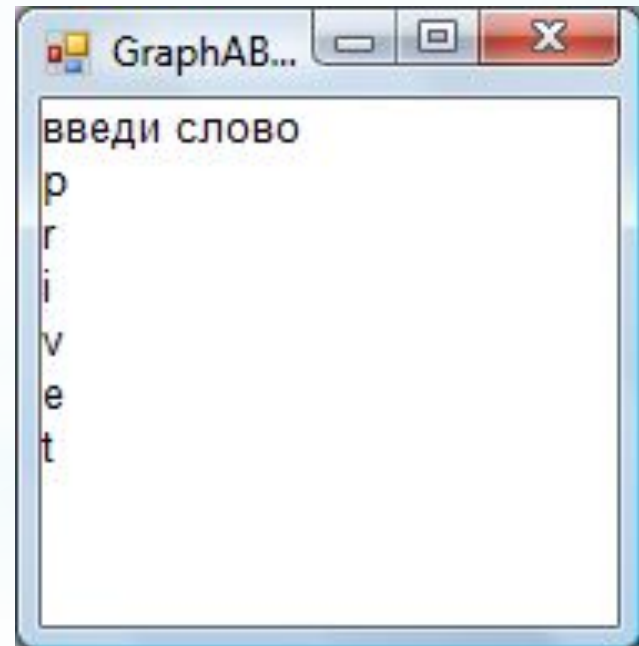
Задача.

Дано слово. Вывести его по одному символу в строке.

Программа

```
uses graphABC;  
var s:string;  
    n,i:byte;  
begin  
    writeln('введи слово');  
    readln(s);  
    n:=length(s);  
    for i:=1 to n do  
        writeln(s[i]);  
end.
```

Результат



В программу внесли следующие изменения:

```
for i:=1 to n do
begin
  write (s[i]);
  if i mod 2 =0 then
    writeln;
end;
```

Как будет теперь выводиться слово?

