

## 1. Строки

Строка – это последовательность символов. Количество символов в строке может меняться от 0 до 255. Переменная строкового типа описывается или через описание типа, или непосредственно в разделе описания переменных. Для описания переменной строкового типа используется служебное слово *string*. В квадратных скобках указывается длина.

```
type slovo = string[50];
var x,y:slovo; var x:string[50];
```

В памяти ЭВМ под строку отводится максимальная длина + 1 байт. Этот байт располагается в начале слова и указывает длину строки.

В рассмотренных примерах под переменные *x*, *y* отводится по 51 байту. Если *m* – текущая длина строки, а *b* – начальный адрес, с которого начинает располагаться строка *k*, то в ячейке *b* будет число *m*, *b+1* – 1-й символ, *b+2* – 2-й символ, *b+m* – *m*-й символ, *b+m+1* – свободная ячейка, *b+k* – свободная ячейка.

Над строковыми данными можно выполнять операции:

1) сложение (сцепление, конкатенация);

2) отношение (равно, не равно, <, >, <=, >=). Эти операции выполняются после операции сцепления. Операции отношения результатом имеют истинно или ложно.

Если длина строкового выражения превышает максимальную длину, отведенную для переменной, то лишние символы отбрасываются.

Для присваивания строковой переменной результата строкового выражения используется оператор присваивания.

```
A=string[4];
A:='не'+'книга';
```

Результат: 'некн'.

В выражение могут входить не только строковые, но и символьные величины.

```
a,c:string[5];
b,d:char;
a:='лон'; b:='с'; c:=b+a 'слон';
d:=a; (зависнет)
```

К отдельному символу можно обратиться по номеру:

```
a:='книга';
a:=a[3];
Ответ: 'и'.
a:='книга';
a:=a[2]+a[3]+a[1]+a[5];
```

Ответ: 'ника'.

В работе со строками можно использовать процедуры:

1) *delete (str, poz, n)* – из строки *str* удалить *n* символов с позиции *poz*: *delete (a,2,1)*.

```
program ad;
var a:string[10];
begin
  writeln('введите a'); readln(a);
  delete(a,3,4; writeln(a);
```

Ответ: кн

2) *insert (str1, str2, poz)*.

Вставка строки *str1* в строку *str2*, начиная с позиции *poz*.

```
begin
  writeln('введите a'); readln(a);
  writeln('введите c'); readln(c);
  insert(c,a,3); writeln(a);
```

Если в этом фрагменте переменная *a* имеет *string[2]*, то будет ошибка. Лучше не писать *read (a,c) (a,c:char)*, т. к. при вводе мы должны отделять пробелом, а пробел – строковая величина, и машина не воспринимает *c*.

3) *str(ib, st)*. Преобразует числовую величину *ib* в строковую *st*. Числовая величина *ib* может указываться в формате:

```
str(-40,a);
str(-40:6,a);
a='-40';
a=' -40';
```

4) *val(str;per;cod)*. Строковую переменную *str* переводит в числовую переменную *per* (*cod* – номер позиции; *str* (исходная строковая величина) не должна содержать пробелов в начале и в конце; *per* – переменная вещественного, целочисленного или байтового типа (целое число от 0 до 255 и длиной 1 байт); *cod* – целочисленная переменная).

Если процедура выполнена, то значение *cod* равно 0, в противном случае это номер позиции, в которой произошла ошибка. При этом значение *per* не определено.

```
program cd;
var
  a:string[6]; k,m:integer; d:real;
begin
  val('400',k,m);
  writeln(k-1, ,m);
```

Ответ:399 0.

```
a:='12.2';
val(a,d,m) val(a,k,m);
writeln(d);. {Здесь будет ошибка: несоответствие}
```

Ответ: 1,22000E+1 типов.

```
a:='1 2';
val(k,k,m); writeln(k, , m);
```

На экране вместо *k* будет любое число, а вместо *m* будет число 2.

#### **Функции для работы со строками.**

1) *length(str)* – длина строковой переменной. Примеры: *length(a)*; *length('123')*;

Ответ:3.

2) *pos(str1, str2)*. Обнаруживает первое появление подстроки *str1* в строке *str2*. Значение – номер позиции, с которой входит *str1*. Если вхождения нет, то значение функции равно 0. Лучше использовать переменную целого типа.

#### **Пример:**

```
var a:string[10]; i:integer;
begin
  a:='wapa'; i:=pos('p',a);
  write(i);
```

Ответ: 3

3) *copy(str,poz, n)*. Вырезает из строки *str* подстроку длиной *n* символов, начиная с позиции *poz*. Следует обратить внимание на то, что тип результата у этой функции *string*.

#### **Пример:**

```
1) var a,b:string[20];
begin
  a:='informatika'; b:=copy(a,3,5);
  writeln(b);
```

Ответ: forma

```
2) var a,b,c,d:string[20];
begin
  a:='informatika'; b:=copy(a,9,3);c:=copy(a,5,1);
  d:=b+c;
  writeln(d);
```

Ответ: ikar

3) *uppercase(c)*. Преобразует малую букву в большую. Следует обратить внимание на то, что тип аргумента *c* у этой функции *char*.

#### **Пример:**

```

var a:string[20];c:char;
begin
  a:='informatika'; c:=upcase(a[3]);
  writeln(c);

```

Ответ: F

**Задача.** Распечатать слово по буквам.

```

program buk;
const n=10;
var a:string[n]; i:integer; c:char;
BEGIN
  writeln('введите a'); readln(a); если обратно
  for i:=1 to length(a) do for i:=length(a) downto 1 do
    begin c:=a[i];
      writeln(c);
    end;

```

END.

**Задача.** Узнать, сколько раз в слове *b1* встречается 3-я буква *a1*.

```

program ser;
const n=10;
var a1:string[n]; b1:string[n]; i,s:integer; c:string[1];
BEGIN
  writeln ('введите a1');readln(a1);  writeln('введите b1'); readln(b1);
  s:=0; c:=a1[3];
  for i:=1 to length (b1) do
    if c=copy (b1,i,1) then s:=s+1;
  writeln('s=',s);

```

END.

**Задача.** В данном слове заменим букву "a" буквой "c":

```

program slr;
var s,t:string; i:integer;
BEGIN
  writeln('Введите слово'); readln(s);
  for i:=1 to length(s) do
    if s[i]='a' then s[i]='c';
  writeln(' ответ ',s);

```

END.

**Задача.** В данном слове заменим сочетание букв "ab" многоточием:

```

program slr;
var s,t:string; i:integer;
BEGIN
  writeln('Введите предложение'); readln(s);
  t:=''; i:=1;
  while i<=length(s) do
    if copy(s,i,2)='ab' then begin t:=t+'...'; i:=i+2 end
    else begin t:=t+s[i]; i:=i+1 end;
  writeln(' ответ ',t); END.

```

**Задача.** Определите сумму цифр натурального числа.

```

program ghit;
const N=30;
var S:string[N]; i,K,f,j:integer;
BEGIN
  K:=0; writeln(' Уведзіце лік); readln(S);
  for i:=1 to length(s) do begin

```

```

val(s[i],f,j);    -вырезаем один символ из строки s- это s[i] и переводим его в число
K:=K+f; end;
writeln(K);

```

END.

**Задача 5.** Задан текст до 10 слов, в котором слова (от 1 до 8 букв) могут быть разделены одной или несколькими звездочками. Удалите все звездочки в начале и в конце текста.. Слова, состоящих из трех букв, замените на слово "три":

```

program gfh;
const s='****a*bc***efg**k*';
var t,w,b:string;
    i,k,n:integer; sl:array[1..10] of string[8];
BEGIN
    t:=s;
    writeln('-----исходный текст-----'); writeln(s);
    while(t[1]='*') and (length(t)>0) do delete(t,1,1);      {удаляем все звездочки в начале
текста}
    while (t[length(t)]='*') do delete(t,length(t),1);      {удаляем все звездочки в конце
текста}
    k:=0; {счетчик звездочек между словами}
    n:=0; {счетчик слов}
    b:="";
    t:=t+'*'; {приклеиваем к концу текста * , чтобы затем получить последнее
слово}
    {разделяем текст на слова, т.е. получаем массив слов из текста}
    for i:=1 to length(t) do
        if t[i]<>'*' then begin b:=b+t[i]; k:=0; end
        else begin k:=k+1;
                                if k=1 then begin n:=n+1; sl[n]:=b; b:=""; end;
                                end;
    writeln('-----вывод всех слов на экран-----');
    for i:=1 to n do writeln(sl[i]);
    {заменяем слова}
    t:="";
    for i:=1 to n do
        if length(sl[i])=3 then t:=t+'три'+'*'
        else t:=t+sl[i]+'*';
    {удаляем звездочку в конце текста}
    delete(t,length(t),1);
    writeln('-----ответ-----');
    writeln(t);

```

END.