

Pascal – язык структурного программирования. Ф У Н К Ц И И.

Учитель Комкова М.С.

Гимназия МИИТ

г. Москва

Понятие функции.

* **ФУНКЦИЯ** - это подпрограмма, результат выполнения которой есть единственное скалярное значение, присваиваемое имени этой функции. Функция - частный случай процедур.

* **ОТЛИЧИЯ от процедур:**

- * 1. В описании указывается тип результата функции.
- * 2. Результат выполнения функции - **ОДНО** значение.
- * 3. Имени функции обязательно присваивается результат.
- * 4. Результат передается в программу как значение имени этой функции.
- * 5. Функция записывается в выражениях, а не отдельной командой как процедура.

Структура функций

- * **FUNCTION** имя (список формальных параметров): тип;
- * [var локальные переменные: тип;]
- * **BEGIN**
- * блок функции
- * **END;**
- * Допускается описание функции без параметров:
- * **FUNCTION** имя : тип;
- * ...

* **Пример 1.**

* Найти сумму чисел ряда 1,2,3,..., используя функцию.

* **Program Summa; uses crt;**

* **var n:integer;** { n - глобальная переменная }

* **function Sum(m: integer): integer;** { m - параметр-значение функции }

* **var i,s: integer;** { i,s - локальные переменные }

* **begin**

* **s:=0;**

* **for i:=1 to m do s:=s+i;**

* **Sum:=s;** {- присваивание имени функции значения }

* **end;**

* **Begin**

* **clrscr;**

* **write('До какого числа суммировать? '); readln(n);**

* **write('Сумма чисел ряда S=',Sum(n));** { -вызов функции Sum с
парам. }

* **readkey;**

* **End.**

* **Пример 2.**

* Составить программу вычисления степени числа X с целым показателем n , используя функцию без параметров.

* **Program Step; uses crt;**

* **var n:integer;**

* **x,y:real;**

* **function St:real;**

* **var i: integer;**

* **begin**

* **y:=1;**

* **for i:=1 to n do y:=y*x;**

* **St:=y**

* **end;**

* **Begin**

* **clrscr;**

* **write('Введите число :'); readln(x);**

* **write('Введите степень :'); readln(n);**

* **if n<0 then begin n:=-n; x:=1/x end;**

* **write('Число ',x,' в степени ',n,' равно ',St)**

* **readkey;**

* **End.**

СИМВОЛЬНЫЕ И СТРОКОВЫЕ ФУНКЦИИ

- * Различают величины:
 - * 1. **Константы** - последовательность символов в апострофах.
 - * 2. **Символьные переменные** типа **CHAR** (кэр, чар) - переменные принимают значения символов: ...var a:char;...a:='м';...
 - * 3. **Строковые переменные** типа **STRING** - цепочки символов:..var a:string;
- * Максимальное число символов в цепочке равно 255.

* СИМВОЛЬНЫЕ И СТРОКОВЫЕ ФУНКЦИИ

- * 1. **LENGTH(S)** -(лэнкс) - определяет и присваивает ДЛИНУ строковой величины S (количество символов, включая пробел). Результат – целое число.
- * 2. **CONCAT(S1,S2)** -(конкат) - ОБЪЕДИНЕНИЕ (склеивание) символьных и строковых величин S1и S2. Конкатенация производится также с помощью знака + ,например S:='Ва'+ 'ся' Результат - строка 'Вася'.
- * 3. **POS(ST,S)** -(позишэн) - определяет и присваивает НОМЕР ПОЗИЦИИ, с которой подстрока ST входит в строку S. Результат - число, например `pos('о','колесо')` =2.
- * 4. **COPY(S,P,D)** -(копи) - ВЫБИРАЕТ часть строковой величины S,начиная с позиции P, длиной D символов.
- * Результат - выбранная строка или один символ.

* Примеры:

```
* Program Primer1; Uses crt;          Program Primer2; Uses crt;
* var b: string;                      var a,b,c: string;
* y: integer;                          Begin
* Begin                                clrscr;
* clrscr;                               a:=' погода';
* b:='информатика';                    b:=' хорошая';
* y:=length(b);                        c:=a+b;
* writeln('Длина строки =',y);         writeln(c)
* End.                                  End.
```

```
* Program Primer3; Uses crt;          Program Primer4; Uses crt;
* var a: string;                      var a,b: string;
* y: integer;                          Begin
* Begin                                clrscr;
* clrscr;                               a:='информатика';
* a:='информатика';                    b:=copy(a,3,5);
* writeln(a);                           writeln(a);
* y:=pos('р',a);                        writeln(b) { форма }
* writeln('р на ',y,' месте');          End.
* End.
```

СИМВОЛЬНЫЕ И СТРОКОВЫЕ ФУНКЦИИ

- * **CHR(X)** - ПРИСВАИВАЕТ **СИМВОЛ** соответствующий коду X.
- * CHR(65) соответствует латинской заглавной букве A.
- * **ORD(S)** -ПРИСВАИВАЕТ **НОМЕР** символа из упорядоченной последовательности символов (см. ASCII), например: ORD('B')=66.

ПРОЦЕДУРЫ НАД СТРОКОВЫМИ ВЕЛИЧИНАМИ

- * 1. **DELETE(S,POS,NUM)** -(дилит) - УДАЛЯЕТ из строки S участок, содержащий NUM символов с позиции POS.
- * 2. **INSERT(OBJ,S,POS)** -(инсет) - ВСТАВИТЬ строку OBJ (переменную или константу) в строку S(переменная) с позиции POS (целое).
- * ПРОЦЕДУРЫ ПРЕОБРАЗОВАНИЯ ТИПОВ:
- * 3. **STR(VALUE,ST)** -(стр) - ПРЕОБРАЗУЕТ числовое значение в СТРОКУ символов и запоминает результат в строке ST.
- * 4. **VAL(ST,VAR,CODE)** -(вэл) - ПРЕОБРАЗУЕТ строковую величину ST в ЧИСЛОВУЮ величину (в зависимости от VAR). CODE - целая величина. Code=0 ,если нет ошибки, иначе Code равно Pos ошибочного символа, начиная с апострофа.

* Примеры:

```
* Program Primer5; Uses crt;          Program Primer6; Uses crt;
*   var a: string;                   var a: string;
* Begin                               Begin
*   clrscr;                           clrscr;
*   a:='Информатика';                 a:='Инка';
*   writeln(a);                       writeln(a);
*   delete(a,3,7);                    insert('формати',a,3);
*   writeln(a);                       writeln(a)
* End.                                 End.
```

```
* Program Primer7; Uses crt;         Program Primer8; Uses crt;
*   var a,b: string;                 var a: string;
*       c: integer;                  r,x,y: integer;
* Begin                               Begin
*   clrscr;                           clrscr;
*   c:=100;                            a:='100 лет';
*   a:='Цена руб.';                   val(a,x,r);
*   str(c:4,b);                        y:=5*x;
*   insert(b,a,5);                     write('Y=',y);
*   writeln(a);                        readkey;
* End.                                 End.
```