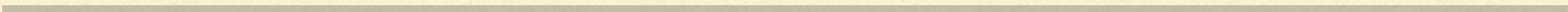

ПРОЦЕДУРЫ И ФУНКЦИИ В ПАСКАЛЬ

Процедура (подпрограмма)

Самостоятельная часть программы,
которую можно неоднократно вызывать из тела основной программы

Синтаксис

```
procedure имя процедуры (формальные параметры);  
    раздел описаний процедуры  
begin  
    исполняемая часть процедуры  
end;
```



ПРОЦЕДУРЫ И ФУНКЦИИ В ПАСКАЛЬ

Функция

Самостоятельная часть программы,
которая возвращает программе какое-либо значение.

Функцию можно неоднократно вызывать из тела основной программы
в операторах присвоения или условиях.

Синтаксис

```
function имя функции (формальные параметры): тип результата;  
    раздел описаний функции  
begin  
    исполняемая часть функции  
    Имя функции := значение функции;  
end;
```

ПРОЦЕДУРЫ И ФУНКЦИИ В ПАСКАЛЬ

Параметры процедур и функций

```
procedure имя (входные переменные: тип; var изменяемые переменные: тип);  
    var  
        внутренние переменные: тип переменных;  
begin  
    исполняемая часть процедуры  
end;
```

Входные переменные — переменные, требуемые для работы процедуры или функции, которые должны быть сообщены ей программой. Даже если эти переменные будут изменены процедурой, в программу они не передадутся.

Изменяемые переменные — переменные, требуемые для работы процедуры или функции, которые должны быть сообщены ей программой. Данные переменные передаются обратно в программу.

Внутренние переменные — переменные, требуемые для работы процедуры, которые существуют только внутри процедуры или функции. Данные переменные не передаются в программу.

Описание переменных в процедуре или функции независимы от переменных, с которыми их вызывают.

*К предыдущем примере, в описании использовались переменные **a** и **b**, а при вызове **x** и **y**.*

ПРОЦЕДУРЫ И ФУНКЦИИ В ПАСКАЛЬ

Пример программы с процедурой

Program maximum;

var x, y, max: integer;

procedure MaxNum(a, b: integer; var max: integer); **Заголовок процедуры**

begin

if a>b then max:=a

else max:=b;

end;

begin

readln(x, y);

MaxNum(x, y, max); **Вызов процедуры**

write(max);

end.

} Тело процедуры

} Тело программы

Данная программа запрашивает у пользователя два числа, вызывает подпрограмму MaxNum, которая сравнивает эти числа и возвращает большее из них в переменной max, после чего результат выводится на экран.

ПРОЦЕДУРЫ И ФУНКЦИИ В ПАСКАЛЬ

Пример программы с функцией

Program maximum;

var x, y, max: integer;

function MaxNum(a, b: integer): integer;

Заголовок функции

begin

if a > b then MaxNum := a

else MaxNum := b;

} Тело функции

end;

begin

readln(x, y);

max := MaxNum(x, y); **Вызов функции**

} Тело программы

write(max);

end.

Данная программа запрашивает у пользователя два числа и в переменную max записывает значение функции MaxNum(x, y), после чего результат выводится на экран.

Функция всегда чему-то равна, поэтому она используется в операторах присваивания или в качестве условия.

ПРОЦЕДУРЫ И ФУНКЦИИ В ПАСКАЛЬ

Использование в процедурах и функциях массивов
на примере

Program massiv;

const n=10; **Задание константы — размерности массива**

var i: integer; mas: array [1..n] of integer; **Описание массива в программе**

Procedure inicial(n: integer; var mas: array of integer;)

**Описание массива
как изменяемого
параметра в процедуре**

var i: integer; **Внутренняя переменная процедуры**

begin

for i:=1 to n do read(mas[i]); **Цикл инициализации массива**

end;

begin

inicial(n, mas); **Вызов процедуры и передача ей массива и его размера**

for i:=1 to n do writeln(mas[i]); **Цикл вывода массива на экран**

end.

Данная программа инициализирует массив mas
с помощью подпрограммы inicial и выводит его на экран.

При описании массива, как переменной в процедуре, его размерность не указывается

ПРОЦЕДУРЫ И ФУНКЦИИ В ПАСКАЛЬ

Стандартные функции

В Pascal много стандартных функций, которые не надо описывать самому.

Функция	Назначение	Тип результата
<code>abs (x)</code>	абсолютное значение аргумента (модуль)	<i>тот же, что и у аргумента</i>
<code>sqr (x)</code>	квадрат аргумента	<i>тот же, что и у аргумента</i>
<code>sqrt (x)</code>	квадратный корень аргумента	<i>real</i>
<code>cos (x)</code>	косинус аргумента	<i>real</i>
<code>sin (x)</code>	синус аргумента	<i>real</i>
<code>arctan (x)</code>	арктангенс аргумента	<i>real</i>
<code>exp (x)</code>	экспонента аргумента	<i>real</i>
<code>ln (x)</code>	натуральный логарифм	<i>real</i>
<code>int (x)</code>	целая часть числа	<i>real</i>
<code>frac (x)</code>	дробная часть числа	<i>real</i>
<code>chr(x)</code>	возвращает символ по его коду	<i>char</i>
<code>ord(c)</code>	возвращает код символа	<i>byte</i>
<code>pos(st,s)</code>	номер, с которого строка <code>s</code> встречается в строке <code>st</code>	<i>integer</i>
<code>length(s)</code>	длина строки	<i>integer</i>
<code>copy(s,x,y)</code>	возвращает <code>y</code> символов, начиная с <code>x</code> -го из строки <code>s</code>	<i>string</i>
<code>delete(s,x,y)</code>	возвращает строку <code>s</code> без <code>y</code> символов, начиная с <code>x</code> -го	<i>string</i>
<code>random(x)</code>	возвращает случайное число от 0 до <code>x</code>	<i>integer</i>

ПРОЦЕДУРЫ И ФУНКЦИИ В ПАСКАЛЬ

Стандартные процедуры

Процедуры	Назначение
<code>delete(s,x,y)</code>	возвращает строку <code>s</code> без <code>y</code> символов, начиная с <code>x</code> -го
<code>insert(s1,s2,x)</code>	вставляет строку <code>s1</code> в строку <code>s2</code> , начиная с <code>x</code> -го символа
<code>str(x,s)</code>	преобразует число <code>x</code> в строку <code>s</code>
<code>val(s,x,code)</code>	преобразует строку <code>s</code> число <code>x</code> , <code>code</code> – номер символа с которого произошла ошибка
<code>randomize</code>	запускает датчик случайных чисел
<code>inc(x)</code>	увеличивает значение <code>x</code> на единицу
<code>dec(x)</code>	уменьшает значение <code>x</code> на единицу
<code>inc(x,a)</code>	увеличивает значение <code>x</code> на <code>a</code>
<code>dec(x,a)</code>	уменьшает значение <code>x</code> на <code>a</code>
<code>break</code>	возвращает строку <code>s</code> без <code>y</code> символов, начиная с <code>x</code> -го
<code>continue</code>	вставляет строку <code>s1</code> в строку <code>s2</code> , начиная с <code>x</code> -го символа
<code>writeln</code>	переводит каретку на экране на следующую строку