

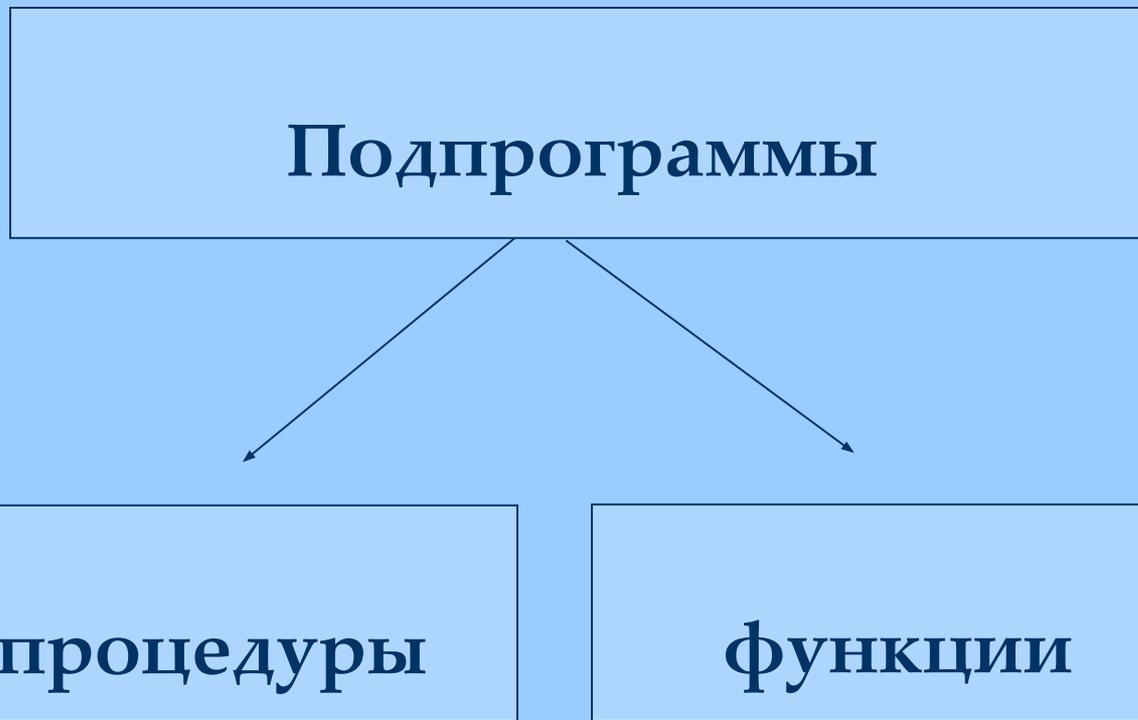
Подпрограммы в Паскале

Назначение

- При разработке программы иногда появляются повторяемые группы действий или возникает необходимость расчленить программу на функциональные модули, сделать ее структуру иерархической. Для этого во всех языках программирования существуют средства организации подпрограмм.
- В Паскале подпрограмма является частью основной программы, ее описание располагается между разделом `var` главной программы и ее программным блоком (первым `begin`). Подпрограмм может быть несколько, их описания располагаются в произвольном порядке одно за другим.

Подпрограмма — это специальным образом оформленный алгоритм, который может многократно использоваться при решении более общей задачи.





В Паскале различают два вида подпрограмм: ***процедуры*** и ***функции***. Основное различие между ними заключается в том, что процедура получает в результате своей работы любое количество данных, а функция — только одно значение.

Функция

- Это последовательность операторов, имеющая имя и результат
- Обрабатывает данные, предназначенные ей из главной программы, и затем возвращает полученный результат
- Функция называется с помощью указателя.
Указатель- это **имя функции**, после которого в круглых скобках перечисляются аргументы функции.

[Далее](#)

Функции Pascal можно разделить на три группы:

- стандартные функции ($\text{abs}(x)$, $\text{sqrt}(x)$, $\text{sqr}(x)$, $\text{sin}(x)$, $\text{cos}(x)$, и т.д.);
- функции программиста (объявлять свою собственную функцию и дальнейшем использовать её так же как и стандартную);
- библиотечные функции (стандартные библиотечные модули).

Меню

Структура функции

Function <имя> (<параметры>):<тип результата>;

const ...;

.....

var ... ;



Блок описания локальных переменных

Begin

<операторы>

имя:= выражение;

End;

В разделе операторов должен находиться, хотя бы один оператор, присваивающий имени функции значение.

Объявление переменных

- **Глобальные переменные** - переменные, объявленные в основной программе, доступны всем операторам программы, а также операторам процедур и функций.
- **Локальные переменные** - переменные, объявленные в процедуре или функции. Они доступны только операторам процедур или функций.

[Далее](#)

Пример:

```
program primer1;
```

```
var
```

```
    r, c, q : real;
```



*Глобальные
переменные*

```
function inper ( a: real; b: real) :real;
```

```
var
```

```
    x,y: real;
```



*Локальные
переменные*

```
begin
```

```
<операторы функции>;
```

```
end;
```

```
begin
```

```
(основная программа)
```

```
end.
```

Назад

Примеры:

- Вычисление функции $\sqrt{n} + n$
- Вычисление скорости свободного падения тела.
- Вычисление **Вычисление**
 n **Вычисление** n -й степени числа.

Меню

Вычисление функции

$$\sqrt{n} + n$$

```
program f1;
```

```
var n,x: real;
```

```
function fun(a:real):real;
```

```
var y:real;
```

```
begin
```

```
y:=sqrt(a)+a;
```

```
fun:=y;
```

```
end;
```

```
begin
```

```
write('n='); readln(n);
```

```
x:=fun(n);
```

```
writeln('x=',x:3:5);
```

```
readln;
```

```
end.
```

[Назад](#)

Вычисление скорости свободного падения тела.

```
program f2;  
var t,v: real;  
function fun(t:real):real;  
  begin  
  fun:=9.8*t;  
  end;
```



```
  begin  
  write('t='); readln(t);  
  v:=fun(t);  
  writeln('v=',v:3:5);  
  readln;  
  end.
```

[Назад](#)

Вычисление n-й степени числа.

```
program f2;

var n: integer;
    z,x: real;
function
step(a:integer;x:real):real;
var y:real;
    i:integer;
    begin
y:=1;
for i:=1 to a do
begin y:=y*x;
step:=y; end;
end;

begin
write(`степень='); readln(n);
write('x='); readln(x);
z:=step(n,x);
writeln('z=',z:3:5);
readln;
end.
```

[Назад](#)

Задания:

1. Рассчитать значение $x = \frac{\sqrt{6} + 6}{2} + \frac{\sqrt{13} + 13}{2} + \frac{\sqrt{21} + 21}{2}$

2. Рассчитать значение $x = \frac{1 + \sin 1}{3} + \frac{5 + \sin 5}{3} + \frac{3 + \sin 3}{3}$

3. Даны шесть различных чисел. Определить максимальное из них.

4. Написать программу свободного падения, выводящую таблицу скорости за первые 10 секунд с шагом 0,5 секунд

Меню

Диктант по теме: «Функции в Паскале»

1. Какую подпрограмму можно назвать функцией?
2. Объясните назначение локальных и глобальных переменных.
3. Что такое указатель?
4. Перечислите группы функций.
5. Каким образом происходит обращение к функции?



Задания для самостоятельной работы

1. Составить программу для вычисления суммы членов геометрической прогрессии.
2. Даны стороны двух треугольников. Найти сумму их периметров и сумму их площадей.
3. Даны координаты точек прямоугольника. Определить, является ли он квадратом, ромбом, трапецией или параллелограммом, и вычислить, исходя из этого, площадь фигуры.
4. Найти все простые числа из заданного интервала натуральных чисел $[N1, N2]$.

Вычисление функции

$$\frac{\sqrt{6}+6}{2} + \frac{\sqrt{13}+13}{2} + \frac{\sqrt{21}+21}{2}$$

```
program f1;
uses Crt;
var n,x,s: real;
    i:integer;
function fun(a:real):real;
var y:real;
    begin
y:=sqrt(a)+a;
fun:=y;
    end;
begin
s:=0;
for i:=1 to 3 do begin
write('n='); readln(n);
x:=fun(n)/2;
s:=s+x;
    end;
writeln('s=',s:3:5);
readln;
end.
```

Ф
У
Н
К
Ц
И
Я

[Назад](#)

Вычисление функции

$$\frac{1 + \sin 1}{3} + \frac{5 + \sin 5}{3} + \frac{3 + \sin 3}{3}$$

Ф
У
Н
К
Ц
И
Я

```
program f1;  
uses Crt;  
var n,x: real;  
    i:integer;  
function fun(a:real):real;  
var y:real;  
    begin  
        y:=a+sin(a);  
        fun:=y;  
    end;
```

```
    begin  
        s:=0;  
        for i:=1 to 3 do begin  
            write('n='); readln(n);  
            x:=fun(n)/3;  
            s:=s+x;  
        end;  
        writeln('s=',s:3:5);  
        readln;  
    end.
```

[Назад](#)

Составить таблицу скорости свободного падения тела за первые 10 секунд с шагом 0,5 секунд

program f2;

var t,v: real;

n:integer;

function fun(t:real):real;

begin

fun:=9.8*t;

end;

и

я

begin

n:=1;

while n<=10 do begin

write('t='); readln(t);

v:=fun(t);

writeln('t=';t'v=',v:3:5);

end;

readln;

end.

[Назад](#)

Ответы на диктант «Функции в Паскале»

1. Это последовательность операторов, имеющая имя и результат.
2. Глобальные переменные - переменные, объявленные в основной программе, доступны всем операторам программы, а так же операторам процедур и функций. Локальные переменные - переменные, объявленные в процедуре или функции. Они доступны только операторам процедур или функций.
3. Это имя функции.
4. Стандартные, функции программиста библиотечные.
5. В основной программе указывается имя функции и аргумент



Процедура – это независимая именованная часть программы, которую можно вызвать по имени для выполнения определенных действий.

Структура процедуры:

```
procedure <ИМЯ> (список формальных параметров);  
const  
type  
var  
begin  
  <операторы>;  
end;
```

Вызов процедуры – это упоминание ее имени в тексте основной программы.

PROGRAM PRIMER;

VAR A,B,S: REAL;

PROCEDURE SUMMA (X,Y:REAL; VAR S:REAL);

BEGIN

S:=X+Y;

END;

BEGIN

WRITELN('A=B='); READLN(A,B);

SUMMA(A,B,S);

WRITELN('S=',S, ' ',A, ' ', B);

END.