

УВАЖАЕМЫЕ СТУДЕНТЫ!

ВАМ НЕОБХОДИМО ВЫПОЛНИТЬ СЛЕДУЮЩЕЕ:

1. Ознакомиться с теорией и законспектировать лекцию не меньше трех листов.
 2. Предоставит отчет конспекта лекции прислать в виде скриншота в течении трех дней .
- Отправить преподавателю на почту v.vika2014@mail.ru и указать свою Ф.И.О, группу, и название дисциплины

Тема: Запись вспомогательных алгоритмов на языке

Планируемые образовательные результаты:

— предметные – представления о способах записи вспомогательных алгоритмов в языке Паскаль;

— метапредметные – умение самостоятельно планировать пути достижения целей;

умение соотносить свои действия с планируемыми результатами, осуществлять контроль своей деятельности, определять способы действий в рамках предложенных условий, корректировать свои действия в соответствии с изменяющейся ситуацией;

умение оценивать правильность выполнения учебной задачи; — личностные – алгоритмическое мышление, необходимое для профессиональной деятельности в современном обществе;

представление о программировании как сфере возможной профессиональной деятельности.

Ключевые слова

- подпрограмма
- процедура
- функция
- рекурсивная функция

Подпрограммы

Запись вспомогательных алгоритмов в языках программирования осуществляется с помощью **подпрограмм** .

Структура описания подпрограммы аналогична структуре главной программы. Описание подпрограммы начинается с заголовка и заканчивается оператором **end**.

Подпрограмма

Процедура

Функция

Процедуры

Процедура - подпрограмма, имеющая произвольное количество входных и выходных данных.

Входные параметры: переменные, константы, выражения

procedure (;

var :);

begin

end ;

Выходные параметры

Для вызова процедуры достаточно указать её имя со списком фактических параметров.

b then a:=a-b else b:=b-a; c:=a end; " width="640"

Алгоритм Евклида

Алгоритм Евклида

Процедура для нахождения НОД

procedure nod (a, b: integer; var c: integer);

begin

while ab do

if ab **then** a:=a-b **else** b:=b-a;

c:=a

end;

Варианты вызова процедуры

в качестве параметров-значений

использованы константы

```
nod (36, 15, z)
```

```
nod ( x , y , z)
```

в качестве параметров-значений

использованы имена переменных

```
nod ( x + y , 15, z)
```

в качестве параметров-значений

использованы выражение и константа

!

Между фактическими и формальными параметрами должно быть полное соответствие по количеству, порядку следования и типу.

```
b then a:=a-b else b:=b-a; c:=a end; begin x:=m[1]; for i:= 2 to 6 do begin
y:=c[i]; nod (x, y, z); x:=z end ; writeln ( 'НОД=', nod ) end . Заголовок главной
программы Описание констант Раздел описания переменных Раздел
описания подпрограммы Раздел описания операторов главной программы "
```

width="640"

Программа с процедурой

```
program n_20;
```

```
const m: array [1..6] of integer =(16, 32,
```

```
40, 64, 80, 128);
```

```
var i: integer;
```

```
procedure nod (a, b: integer;
```

```
var c:integer);
```

```
begin
```

```
while ab do
```

```
if ab then a:=a-b else b:=b-a;
```

```
c:=a
```

```

end;
begin
x:=m[1];
for i:= 2 to 6 do
begin
y:=c[i];
nod (x, y, z);
x:=z
end ;
writeln ( 'НОД=', nod )
end .

```

Заголовок главной программы

Описание констант

Раздел описания переменных

Раздел описания

подпрограммы

Раздел описания операторов

главной программы

Функции

Функция - подпрограмма, имеющая единственный результат, записываемый в ячейку памяти, имя которой совпадает с именем функции.

Перечень формальных параметров и их типов

```
function (); ;
```

```
begin
```

```
;
```

```
:=
```

```
end;
```

Тип результата

В блоке функции обязательно должен присутствовать оператор **:=**.

!

Для вызова функции её имя со списком параметров можно в любом выражении, в условиях, в операторе **write** главной программы.

```
y then max:=x else max:=y; end; begin readln (a, b, c, d); f:= max(max(a, b),
max(c, d)); writeln ('f=', f); end. Заголовок главной программы Описание
переменных Раздел описания подпрограммы Раздел операторов главной
программы " width="640"
```

Функция поиска максимального из 2-х

```
program n_20;
var a, b, c, d, f: integer;
function max (x, y: integer): integer;
begin
if xy then max:=x else max:=y;
end;
begin
readln (a, b, c, d);
f:= max(max(a, b), max(c, d));
writeln ('f=', f);
end.
```

Заголовок главной программы

Описание переменных

Раздел

описания

подпрограммы

Раздел операторов

главной

программы

Последовательность Фибоначчи

В январе Саше подарили пару новорождённых кроликов. Через два месяца они дали первый приплод - новую пару кроликов, а затем давали приплод по паре кроликов каждый месяц.

Каждая новая пара также даёт первый приплод (пару кроликов) через два месяца, а затем - по паре кроликов каждый месяц. Сколько пар кроликов будет у Саши в декабре?

Числа 1, 1, 2, 3, 5, 8, ... образуют так называемую **последовательность Фибоначчи**, названную в честь итальянского математика, впервые решившего соответствующую задачу ещё в начале XIII века.

= 3 . " width="640"

Математическая модель

Пусть $f(n)$ количество пар кроликов в месяце с номером n .

По условию задачи:

$$f(1) = 1,$$

$$f(2) = 1,$$

$$f(3) = 2.$$

Из двух пар, имеющих в марте, дать приплод в апреле сможет только одна: $f(4) = 3$.

Из пар, имеющих в апреле, дать приплод в мае смогут только пары, родившиеся в марте и ранее:

$$f(5) = f(4) + f(3) = 3 + 2 = 5.$$

В общем случае:

$$f(n) = f(n-1) + f(n-2), n \geq 3.$$

Функция

function f (n: integer): integer;

begin

if (n=1) or (n=2) **then** f:=1

else f:=f(n-1)+f(n-2)

end;

Полученная функция **рекурсивная** - в ней реализован способ вычисления очередного значения функции через вычисление её предшествующих значений.

Самое главное

Запись вспомогательных алгоритмов в языках программирования осуществляется с помощью **подпрограмм**. В Паскале различают два вида подпрограмм: процедуры и функции.

Процедура - подпрограмма, имеющая произвольное количество входных и выходных данных.

Функция - подпрограмма, имеющая единственный результат, записываемый в ячейку памяти, имя которой совпадает с именем функции.

Вопросы и задания

Напишите программу перестановки значений переменных a, b, c в порядке возрастания, т. е. так, чтобы a . Используйте процедуру **swap**.

```
procedure swap (var x, y: integer);
```

```
var m: integer;
```

```
begin
```

```
m:=x;
```

```
x:=y;
```

```
y:=m
```

```
end;
```

Исходные данные вводятся с клавиатуры.

Для чего используются подпрограммы?

В чём основное различие процедур и функций?

Напишите программу вычисления наименьшего общего кратного следующих четырёх чисел: 36, 54, 18 и 15.

Используйте процедуру вычисления наибольшего общего делителя двух чисел.

Напишите функцию, вычисляющую площадь треугольника по целочисленным координатам его вершин.

С помощью этой функции вычислите площадь четырёхугольника по координатам его вершин.

Видоизмените программу сортировки массива выбором так, чтобы в ней использовалась процедура выбора наибольшего элемента массива.

Напишите программу поиска наибольшего из четырёх чисел с использованием подпрограммы поиска наибольшего из трёх чисел.

Напишите функцию, вычисляющую длину отрезка по координатам его концов. С помощью этой функции напишите программу, вычисляющую периметр треугольника по координатам его вершин.

Напишите программу вычисления выражения:

$$s = 1! + 2! + 3! + \dots + n!$$

Используйте функцию вычисления факториала.

Напишите программу вычисления выражения:

$$s = x^3 + x^5 + x^n,$$

где x и n вводятся с клавиатуры.

Используйте функцию вычисления степени.

Пример входных данных

1 2 3

Пример выходных данных

2 1 3

1 2 3

3 1 2

1 2 3

2 3 1

1 2 3

1 2 3

Опорный конспект

Запись вспомогательных алгоритмов в языках программирования осуществляется с помощью **подпрограмм**.

Подпрограмма

Функция

Процедура

Подпрограмма, имеющая единственный результат, записываемый в ячейку памяти, имя которой совпадает с именем функции.

Подпрограмма, имеющая произвольное количество входных и выходных данных.