# Лабораторная работа № 11

# Комбинированные вычислительные процессы.

Цель работы: Научиться реализовывать алгоритмы комбинированных вычислительных процессов с помощью FreePascal.

## Оборудование: PC, Lazarus

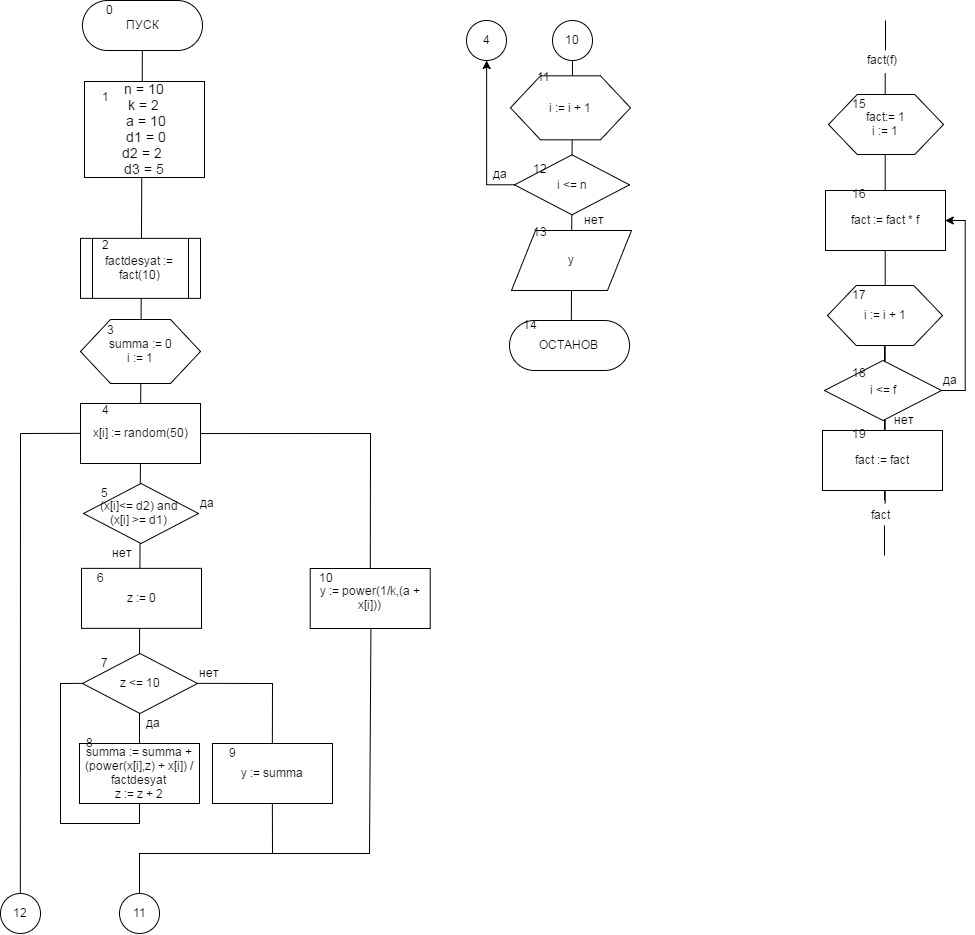
# Задача № 1

Постановка задачи: Дан массив чисел X, который состоит из элементов Xi, где i = 1 ÷ n (шаг по Z равен 2) . Для элементов массива, попавших в заданный диапазон вычислить y.

## Математическая модель:

## 

## Блок-схема



## Список идентификаторов (обозначение переменных):

*Таблица 1*

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| n,k,a | Константы | integer |
| d1,d2,d3 | Ограничители | integer |
| x | Массив | integer |
| i | Итератор | integer |
| z | Аргумент цикла | integer |
| factdesyat | Факториал 10 | integer |
| y | Искомое y | real |
| summa | Сумма | real |
| f | Локальная переменная принимаемая функцией | integer |
| factorial | Локальная переменная, для расчета факториала | integer |
| fact | Функция для расчета факториала | integer |

Код программы:

program zadacha1;

uses Math;

const

n = 10;

k = 2;

a = 10;

d1 = 0;

d2 = 2;

var

y,summa:real;

i,z,factdesyat:integer;

x:array [1..n] of integer;

Function fact(f:integer) : integer;

var factorial,i:integer;

begin

factorial := 1;

for i := 1 to f do

begin

factorial := factorial \* f;

end;

fact := factorial;

end;

begin

Randomize;

summa := 0;

factdesyat := fact(10);

for i := 1 to n do

begin

x[i] := random(5);

if (x[i]<= d2) and (x[i] >= d1) then y := power(1/k,(a + x[i]))

else

begin

z := 0;

while z <= 10 do

begin

summa := summa + (power(x[i],z) + x[i]) / factdesyat;

z := z + 2;

end;

y := summa;

end;

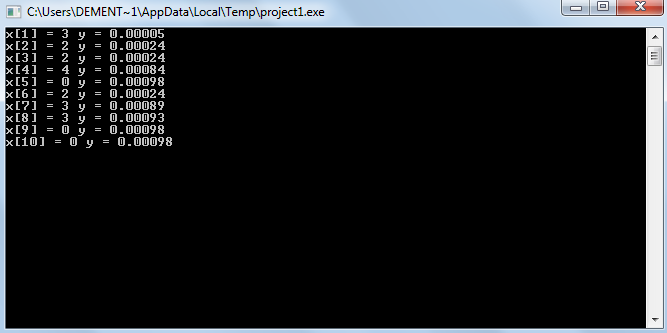
writeln('x[',i,'] = ', x[i], ' y = ', y:3:5);

end;

readln();

end.

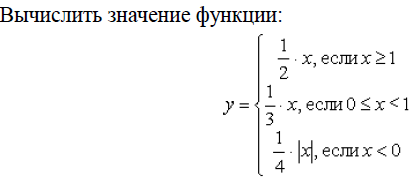
## Результаты выполненной работы:



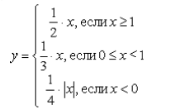
Анализ результатов вычисления: Программа вычисляети выводит на экранy, который был вычислен для элементов массива, которые попали в заданный диапазон значений.

# Задача №2

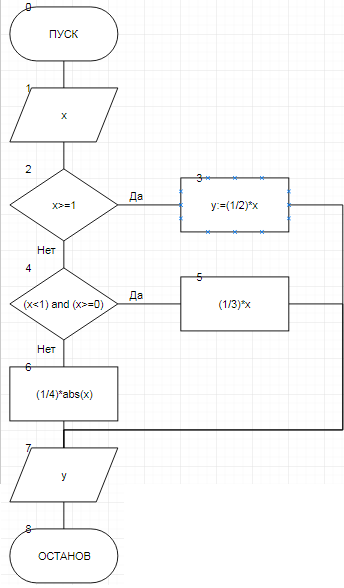
Постановка задачи:



## Математическая модель:

****

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 2*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| x | Переменная y | real |
| y | Переменная x | real |

## Код программы:

program zadacha2;

var

y,x:real;

begin

writeln('Vvedite x');

readln(x);

if x>=1 then

y:=(1/2)\*x else

if (x<1) and (x>=0) then

y:=(1/3)\*x else

y:=(1/4)\*abs(x);

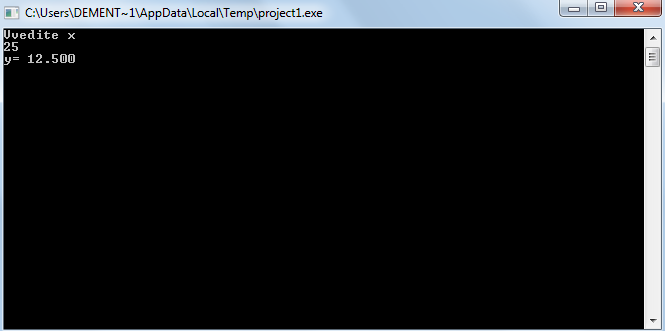
writeln(y=: ',y:1:3);

readln();

end.

## 

## Результаты выполненной работы:



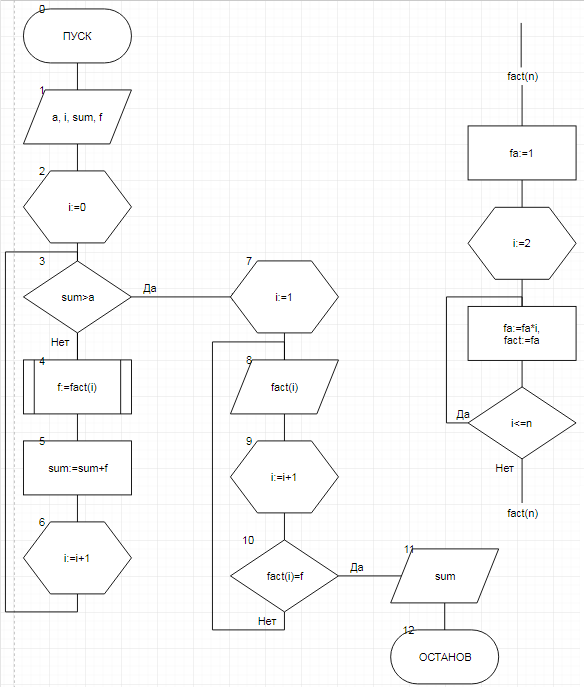
## Анализ результатов вычисления: Программа вычисляет по формуле y и выводит ответ на экран.

# Задача №3

Постановка задачи: Составить программу подсчета суммы факториалов целых чисел, где сумма не превышает число А, которое вводится с клавиатуры. На экран вывести сумму и все слагаемые.

## Математическая модель:

## Блок-схема:



## Список идентификаторов (обозначение переменных):

*Таблица 3*

|  |  |  |
| --- | --- | --- |
| Имя | Смысл | Тип |
| a | Вводимое число | longint |
| f | Факториал | longint |
| sum | Сумма факториалов | longint |
| i | Параметр цикла | integer |
| fact(n) | Функция вычисления факториала | longint |
| fa | Промежуточная переменная для вычисления факториала | longint |

## Код программы:

program zadanie3;

var

a,f,sum:longint;

i:integer;

function fact(n:integer):longint;

var

i:integer;

fa:longint;

begin

fa:=1;

for i:=2 to n do

fa:=fa\*i;

fact:=fa;

end;

begin

readln(a);

i:=1;

sum:=0;

f:=1;

while(sum<=a) do

begin

f:=fact(i);

sum:=sum+f;

i:=i+1;

end;

sum:=sum-f;

i:=1;

repeat

writeln(i,' slagaemoe=',fact(i));

i:=i+1;

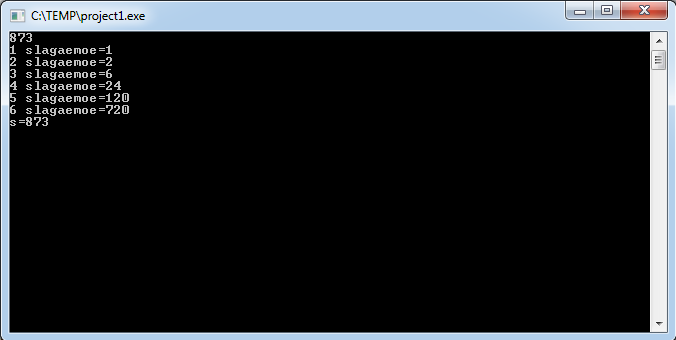
until fact(i)=f;

writeln('s=',sum);

readln;

end.

## Результаты выполненной работы:



## Анализ результатов вычисления:

Программа получает число и считает факториал, обращаясь к функции. После этого, т.к. изначально неизвестно какое число могло быть введено пользователем, идем перерасчет факториала и уменьшение суммы, с их выводом на экран.

# Вывод.

Таким образом, были изучены методы реализации комбинированных вычислительных процессов средствами Free Pascal.