Лабораторная работа №7.

**Тема:**

**Цель:**

**Используемое оборудование: ПК, среда программирования Lazarus.**

**Задание 1.**

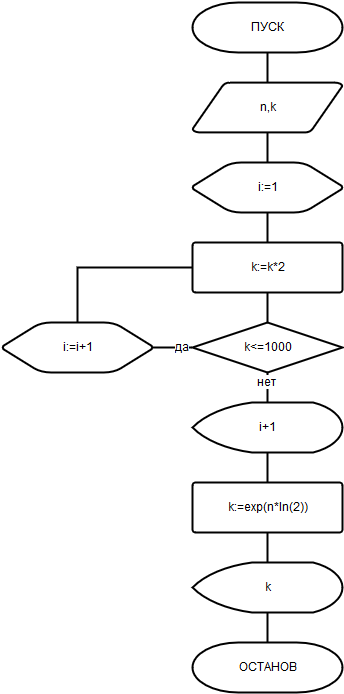
1. Постановка задачи: Вычислить 2 в степени n и при этом определить первое значение степени, при котором результат будет превышать значение 1000.

Для данной задачи написать максимально возможное количество вариантов программ, используя разные виды циклов.

1. Математическая модель:

x=2^n>1000 or <1000

1. Блок-схема:



1. Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| A | Real | Нижний предел интегрирования |
| B | Real | Верхний предел интегрирования |
| H | Real | Шаг |
| z | Real | Сумма |
| K | Real | Вспомогательная переменная |
| N | Real | Количество отрезков |
| X | Real | Переменная |
| Y | real | Функция |

1. Код программы:

1) program zadanie1\_var1;

var

n,i:integer;

k:real;

begin

writeln('vvedite stepen');

readln(n);

k:=2;

for i:=1 to n

begin

k:=2\*k;

if k>1000 then

begin

writeln('stepen = ', i+1);

break;

end;

end;

k:=exp(n\*ln(2));

writeln('k = ',k:2:0);

readln()

end.

2) program zadanie1\_var2;

var

n,i,a:integer;

k:real;

begin

writeln('vvedite stepen');

readln(n);

k:=2;

i:=1;

while i<n do

begin

k:=2\*k;

if (k>1000) then

begin

writeln('stepen = ', i+1);

break;

end;

i:=i+1;

end;

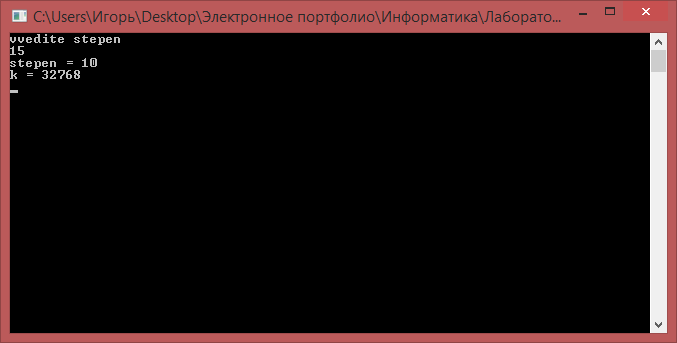
k:=exp(n\*ln(2));

writeln('k = ',k:2:0);

readln()

end.

1. Результат выполнения программы:



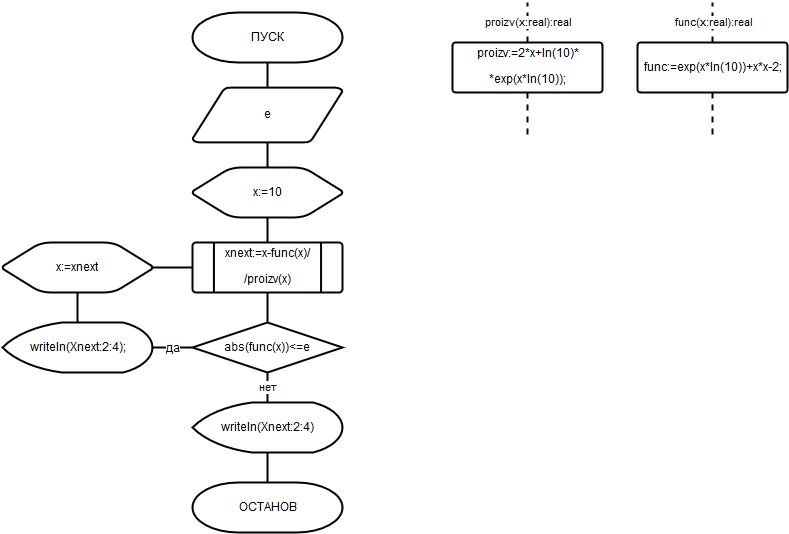
1. Анализ результатов вычисления: Программа выдает значение k=2n и степень, при которой k>1000
2. Вывод: Программа работает корректно

**Задание 2.**

1. Постановка задачи: Решить нелинейное уравнение методом Ньютона.
2. Математическая модель:

task26.gif(439 bytes)

1. Блок-схема:



1. Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| e | Real | Точность |
| x | Real | Переменная |
| x0 | Real | Первая переменная |
| xnext | Real | Следующая переменна |

1. **Код программы:**

program zad2;

var

e,x,x0,xnext:real;

function func(x:real):real;

begin

func:=exp(x\*ln(10))+x\*x-2;

end;

function proizv(x:real):real;

begin

proizv:=2\*x+ln(10)\*exp(x\*ln(10));

end;

begin

e:=0.000001;

x0:=10;

x:=x0;

repeat

xnext:=x-func(x)/proizv(x);

x:=xnext;

writeln(Xnext:2:4);

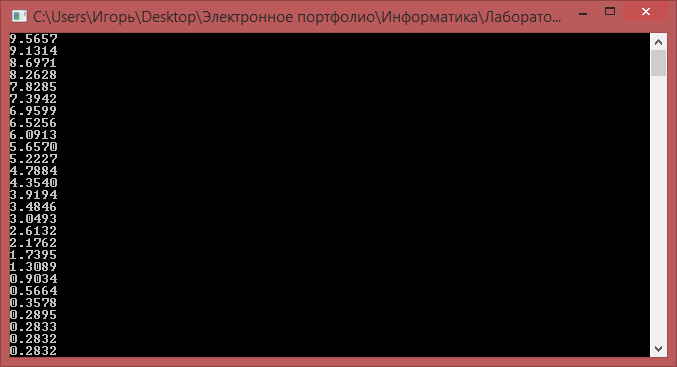
until abs(func(x))<=e;

writeln(Xnext:2:4);

readln;

end.

1. Результат выполнения программы:



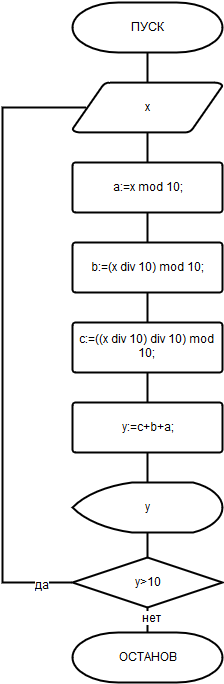
1. Анализ результатов вычисления: Программа находит корни нелинейного уравнения методом Ньютона
2. Вывод: Программа работает корректно

**Задание 3.**

1. Постановка задачи: С клавиатуры вводится трехзначное число, считается сумма его цифр. Если сумма цифр числа больше 10, то вводится следующее трехзначное число, если сумма меньше либо равна 10 – программа завершается
2. Математическая модель:

sum=(n mod 10) + (n div 100) +(( n mod 100) div 10)

1. Блок-схема:

****

1. Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| a | Integer | Третья цифра числа х |
| b | Integer | Вторая цифра числа х |
| с | Integer | Первая цифра числа х |
| x | Integer | Вводимое число |
| y | Integer | Сумма |

1. Код программы:

program zad3;

label

goback;

var

x,a,b,y,c:integer ;

begin

goback:writeln('vvedite chislo');

readln(x);

a:=x mod 10;

b:=(x div 10) mod 10;

c:=((x div 10) div 10) mod 10;

y:=c+b+a;

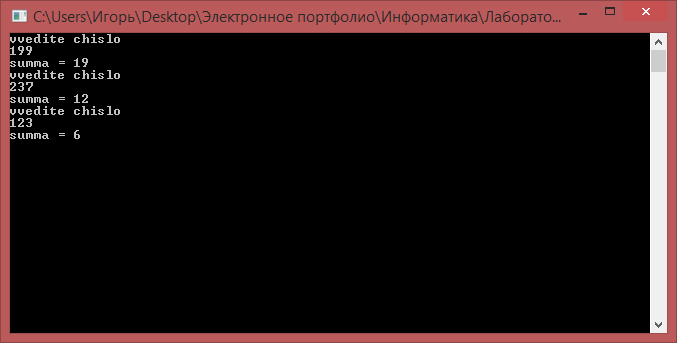
writeln('summa = ',y);

If y>10 then goto goback;

readln

end.

1. Результат выполнения программы:



1. Анализ результатов вычисления: Программа считает сумму цифр трехзначного числа х и сравнивает её с 10. При y>10 программа принимает на ввод ещё одно число, а при y<10 программа прекращает свое выполнение.
2. Вывод: Программа работает корректно

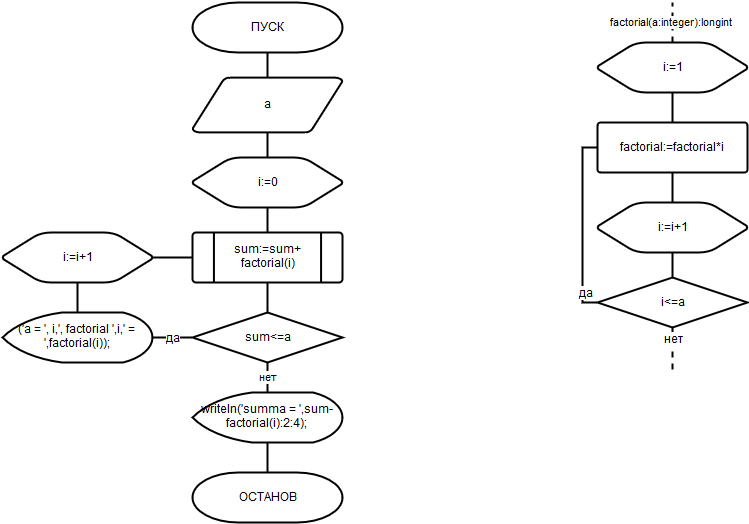
**Задание 4.**

1. 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.
2. Математическая модель:

f=1\*2\*3…\*n

sum = sum+f

1. Блок-схема:



1. Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| a | Real | Вводимое число |
| sum | Real | Сумма факториалов |
| i | Integer | Параметр цикла |

1. Код программы:

program zad4;

var

a,sum:real;

i:integer;

function factorial(a:integer):longint;

var

i:integer;

begin

factorial:=1;

for i:=1 to a do

factorial:=factorial\*i;

end;

begin

writeln('vvedite a');

readln(a);

sum:=0;

for i:=0 to a do

begin

sum:=sum+factorial(i);

If sum>a then break

else writeln('a = ', i,', factorial ',i,' = ',factorial(i));

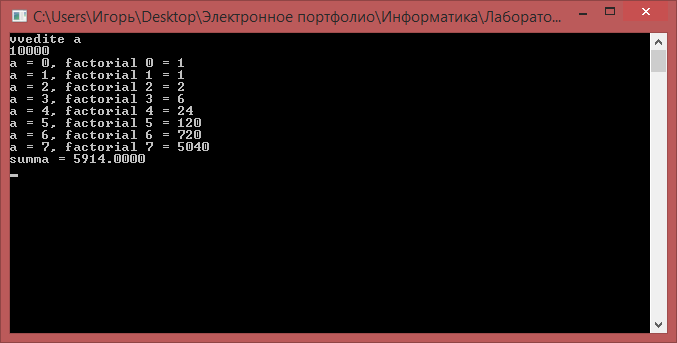
end;

writeln('summa = ',sum-factorial(i):2:4);

readln

end.

1. Результат выполнения программы:



1. Анализ результатов вычисления: Программа считает сумму факториалов целых чисел от 0 до заданного числа.
2. Вывод: Программа работает корректно