***Лабораторная работа №2.***

**Тема: Детерминированные циклические вычислительные процессы с управлением по аргументу.**

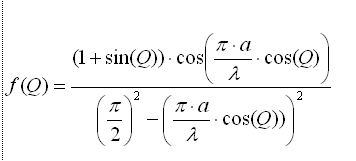
**Цель: научиться реализовывать алгоритмы решения определенного вида задач, где результат изменяется в зависимости от аргумента.**

**Используемое оборудование: ПК, среда разработки Lazarus, draw.io.**

**Задание 1.**

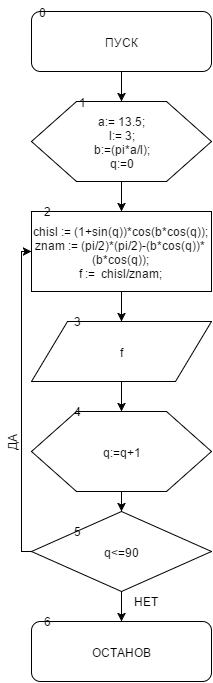
Постановка задачи: Рассчитать значения для построения диаграммы направленности антенны в вертикальной плоскости.

Математическая модель:



Q меняется в диапазоне от 0 до 90 градусов с шагом 1 градус,  *a*= 13.5,  = 3 см

Блок-схема:



Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя идентификатора | Тип идентификатора | Смысл идентификатора |
| f | real | делимое |
| a | real | делитель |
| l | real | лямбда |
| chisl | real | числитель |
| znam | real | знаменатель |
| q | integer | Градусы наклона |
| b | real | результат деления(частное) |

Код программы:

**program** zadanie1;

**var**

q : integer;

f, a, l, chisl, znam,b : real;

**begin**

a:= 13.5;

l:= 3;

b:=pi\*a/l;

**for** q := 0 **to** 90 **do begin**

chisl := (1+sin(q))\*cos(b\*cos(q));

znam := (pi/2)\*(pi/2)-(b\*cos(q))\*(b\*cos(q));

f := chisl/znam;

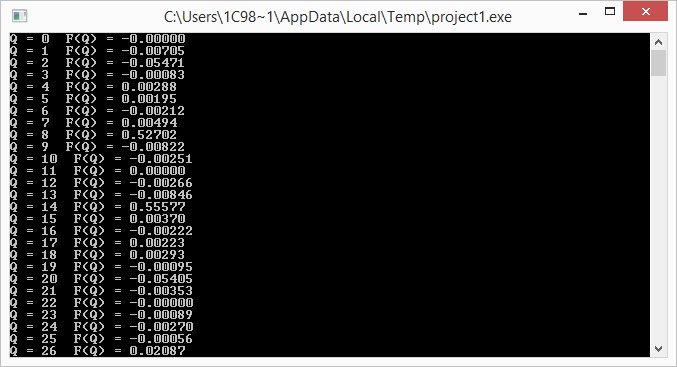
writeln('Q = ',q,' F(Q) = ',f:3:5);

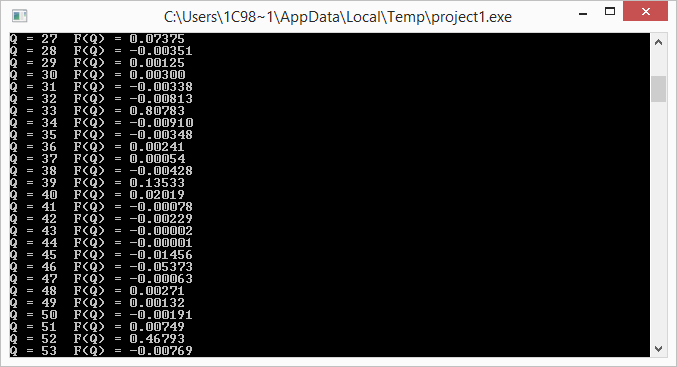
**end**;

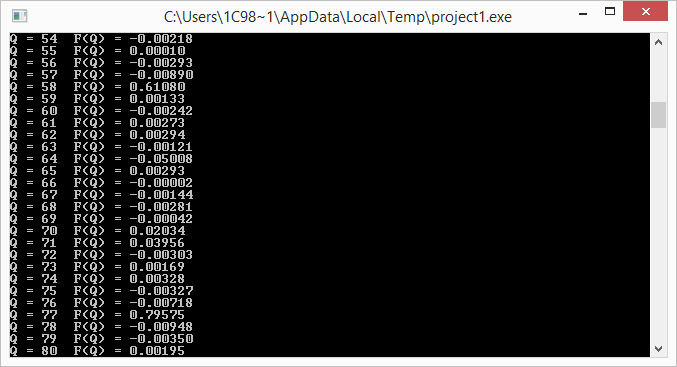
readln();

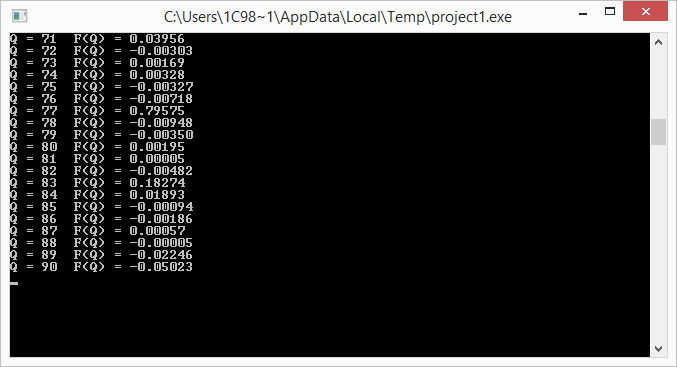
**end**.

Результат работы программы:



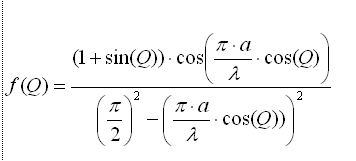






Анализ результатов вычисления:

Наша программа выводит на экран значения для построения диаграммы направленности антенны в вертикальной плоскости в соответствии с данной формулой.

.

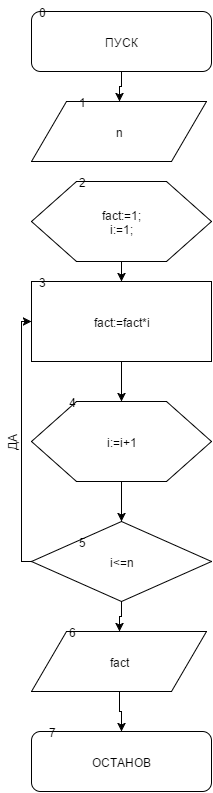
Вывод: Мы написали программу для вычисления значений для построения диаграммы направленности антенны в вертикальной плоскости.

**Задание 2.**

Постановка задачи: Вычислить n!, где n вводится с клавиатуры.

Мат. Модель: n!=1\*2\*3...\*n

Блок-схема:



Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| ИМЯ | ТИП | СМЫСЛ |
| i | integer | Переменная-счетчик |
| n | integer | Вводимая переменная |
| fact | longint | Искомое значение |

Код программы:

**program** zadaniefactorial;

**var**

i,n: integer;

fact: longint;

**begin**

Writeln('Vvedite n');

Readln(n);

fact:=1;

**for** i:=1 **to** n **do**

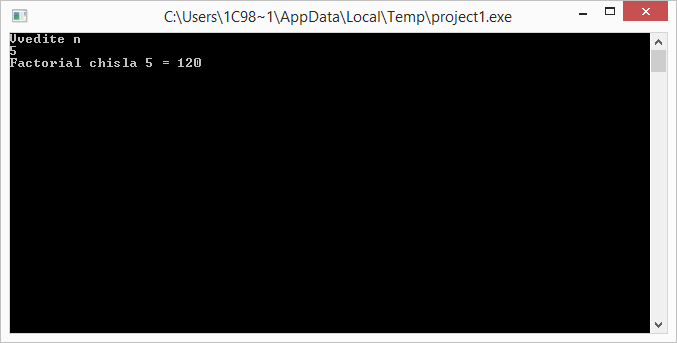
fact:=fact\*i;

Writeln('Factorial chisla',' ', n,' ','=',' ',fact);

Readln();

**end**.

Результат работы программы:



Анализ: наша программа, входя в цикл, на выходе выдает значение факториала числа n, которое изначально вводится пользователем с клавиатуры.

Вывод: Мы написали программу, считающую факториал числа, которое пользователь вводит с клавиатуры.

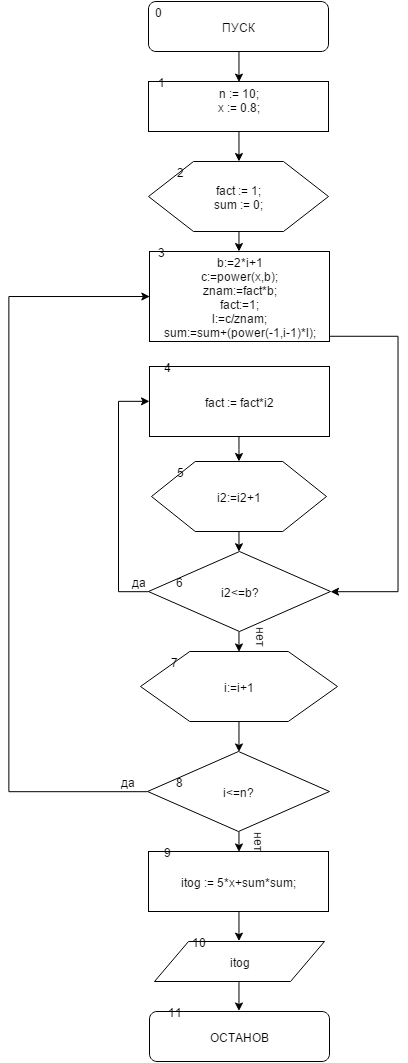
**Задание 3.**

Постановка задачи: вычислить значение данного выражения 

с учетом того, что n=10, x=0.8

Мат. Модель: 

Блок-схема:



Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| b | integer | Переменная, содержащая часть формулы |
| i | integer | Переменная-счетчик |
| i2 | integer | Переменная-счетчик |
| n | integer | Заданная переменная |
| x | real | Заданная переменная |
| sum | real | Переменная суммы |
| itog | real | Итоговая переменная |
| c | real | Числитель |
| znam | real | Знаменатель |
| fact | real | Факториал |
| l | real | Переменная, содержащая часть формулы |

**program** LR2Task3\_3;

**uses** math;

**var**

b,i,i2,n:longint;

x,sum,itog,c,znam,fact,l:real;

**begin**

n := 10;

x := 0.8;

fact := 1;

sum := 0;

**for** i:= 1 **to** n **do begin**

b := 2\*i-1;

**for** i2:= 1 **to** b **do begin**

fact := fact\*i2;

**end**;

c:=power(x,b);

znam:=fact\*b;

fact:=1;

l:=c/znam;

sum:=sum+(power(-1,i-1)\*l);

Writeln('Dlya i=',i,' summa = ',sum:1:10);

**end**;

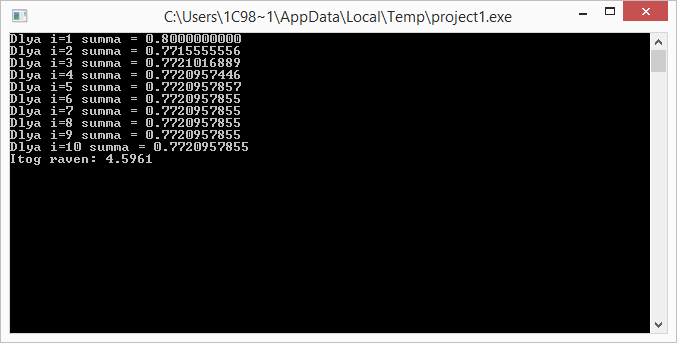
itog := 5\*x+sum\*sum;

writeln('Itog raven: ',itog:2:4);

readln();

**end**.

Результаты работы:

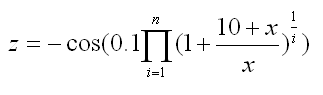


Анализ: мы научились считать значение выражения

 для x=0.8 и n=10;

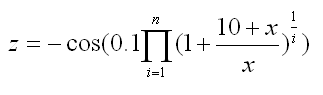
Вывод: мы написали программу, которая высчитывает значение z.

**Задание 4.**

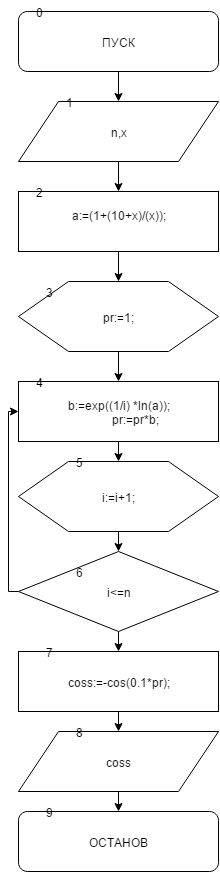
Постановка задачи: написать программу, вычисляющую значение переменной z в заданной формуле 

И протестировать ее на значениях n=40 и x=5;

Мат. Модель:



Блок-схема:



Список переменных:

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Значение |
| i | integer | Переменная-счетчик |
| n | integer | Значение, вводимое с клавиатуры |
| x | real | Значение, вводимое с клавиатуры |
| coss | real | Косинус |
| pr | real | Произведение части формулы |
| a | real | Переменная, считающая значение вне цикла |
| b | real | Степень |

Код программы:

**program** zad;

**var**

coss, pr,x,a,b:real;

i,n:integer;

**begin**

Writeln('Vvedite n');

Readln(n);

Writeln('Vvedite x');

Readln(x);

a:=(1+(10+x)/(x));

pr:=1;

**For** i:=1 **to** n **do**

**begin**

b:=exp((1/i) \*ln(a));

pr:=pr\*b;

**end**;

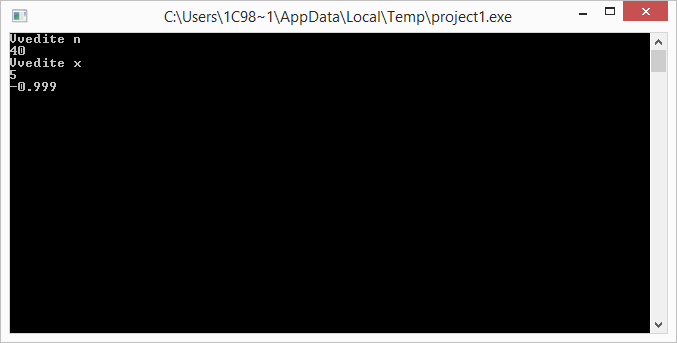
coss:=-cos(0.1\*pr);

Writeln(coss:2:3);

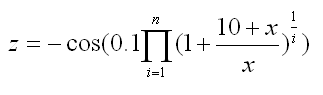
Readln();

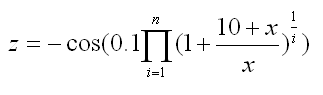
**end**.

Результаты работы:



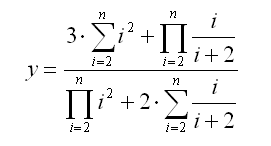
Анализ: наша программа сначала читает вводимые с клавиатуры n и x, а потом, с помощью цикла, считает искомый результат z по формуле



Вывод: мы написали программу, считающую переменную z по формуле

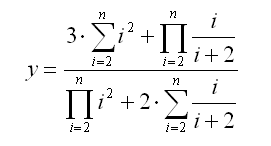
**Задание 5.**

Постановка задачи: написать программу, которая позволяет пользователю ввести число n и по данной формуле

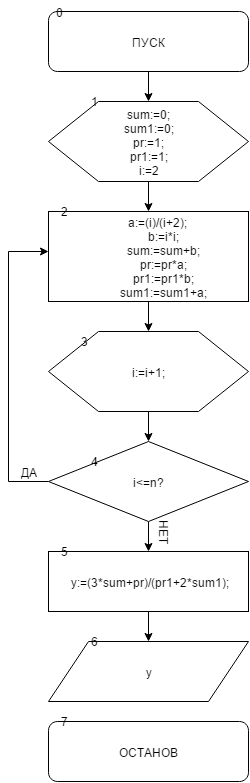


вычислить значение y, вывести его на экран.

Мат. Модель:



Блок-схема:



Список переменных:

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| a | real | Фрагмент вычислений в цикле |
| b | real | Фрагмент вычислений в цикле |
| i | integer | Переменная-счетчик |
| n | integer | Переменная-счетчик |
| sum | real | Фрагмент вычислений в цикле |
| sum1 | real | Фрагмент вычислений в цикле |
| pr | real | Фрагмент вычислений в цикле |
| pr1 | real | Фрагмент вычислений в цикле |
| y | real | Искомое значение при заданном n |

Код программы:

**program** zad5;

**var**

a,b,sum,sum1,pr,pr1,y:real;

i,n: integer;

**begin**

Writeln('Vvedite n');

readln(n);

sum:=0;

sum1:=0;

pr:=1;

pr1:=1;

**For** i:=2 **to** n **do**

**begin**

a:=(i)/(i+2);

b:=i\*i;

sum:=sum+b;

pr:=pr\*a;

pr1:=pr1\*b;

sum1:=sum1+a;

**end**;

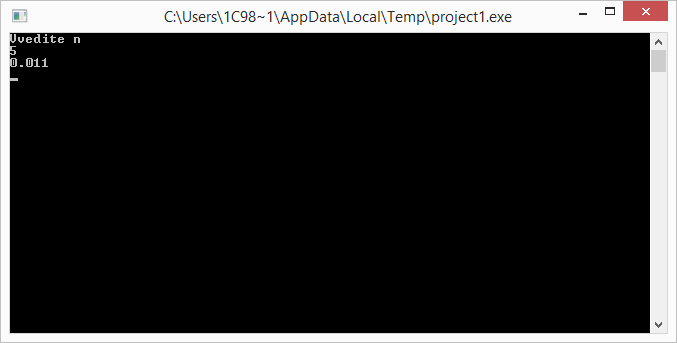
y:=(3\*sum+pr)/(pr1+2\*sum1);

Writeln(y);

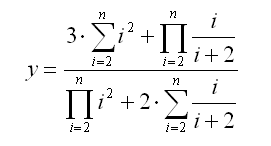
readln();

**end**.

Результаты работы:



Анализ: с помощью цикла мы разработали программу, которая вычисляет значение y по формуле



Вывод: Мы написали программу для вычисления значения выражения с n, вводимым пользователем.

**Задание 6.**

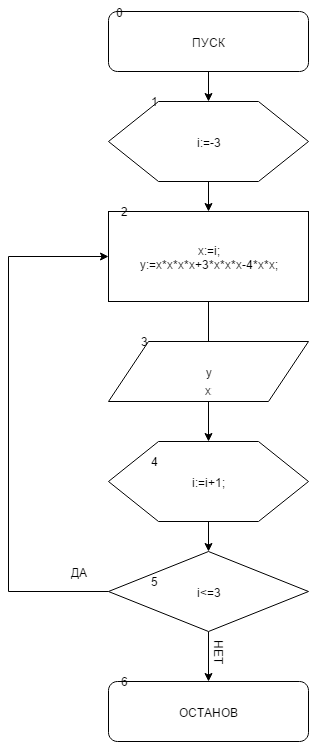
Постановка задачи: протабулировать функцию

 на интервале [-3, 3] с шагом 1. Результаты оформить в таблицу.

Мат. Модель:



Блок-схема:



Список идентификаторов:

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| i | integer | Переменная-счетчик |
| x | real | Значение, изменяющееся в интервале [-3;3] |
| y | real | Искомое значение |

Код программы:

**program** zadanie6;

**var**

i:integer;

x,y: real;

**begin**

**For** i:=-3 **to** 3 **do**

**begin**

x:=i;

y:=x\*x\*x\*x+3\*x\*x\*x-4\*x\*x;

Writeln('Dlya x=',x:2:3,' ', 'y= ', y:2:3);

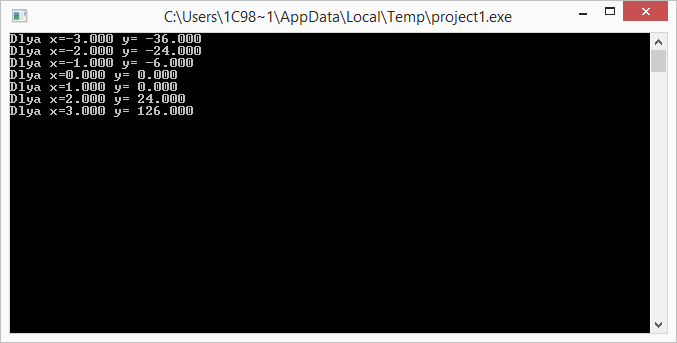
**end**;

readln();

**end**.

Результат работы программы:

|  |  |
| --- | --- |
| Значения x | Значения y |
| **-3** | **-36** |
| **-2** | **-24** |
| **-1** | **-6** |
| **0** | **0** |
| **1** | **0** |
| **2** | **24** |
| **3** | **126** |



Анализ:

мы разработали программу, которая табулирует функцию y.

Вывод: мы написали программу, которая в соответствие с формулой  считает значения y. Х принадлежит промежутку [-3;3].