**Лабораторная работа № .**

Тема: процедуры и функции.

Цель: научиться реализовывать вычисления посредством процедур и функций в среде программирования Lazarus.

Используемое оборудование: ПК, среда программирования Lazarus, сайт draw.io

*Далее к каждому заданию будет начинаться с пункта №4(постановка задачи).*

**Задание 1.**

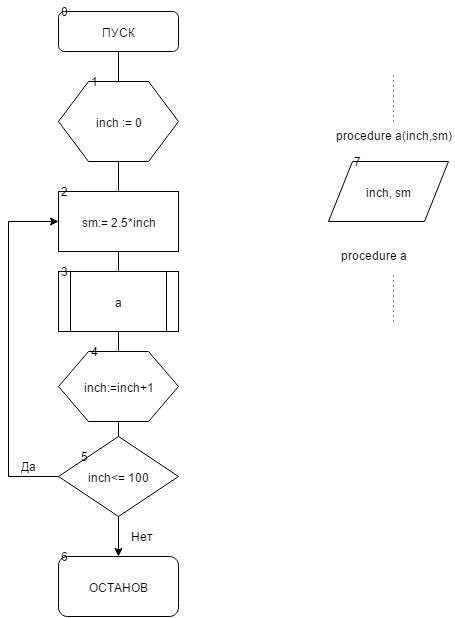
**Постановка задачи:**

Перевести дюймы в сантиметры от 0 до 100 дюймов. (1 дюйм=2,5 см). Результаты вывести в виде таблицы. Операторы для формирования вывода таблицы оформить в виде пользовательской **процедуры.**

**Математическая модель:**

1 дюйм = 2.5 см

**Блок-схема:**



**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| inch | integer | Количество дюймов, параметр цикла |
| sm | real | Количество сантиметров |
| a | - | Процедура вывода данных на экран |

**Код программы:**

program zadanie1;

var

inch:integer;

sm:real;

procedure a(inch:integer;sm:real);

begin

writeln('Dlya ',inch, ' dyima(-ov) = ','Kolichestvo sm = ',sm:2:1);

end;

begin

for inch:= 0 to 100 do begin

sm := 2.5\*inch;

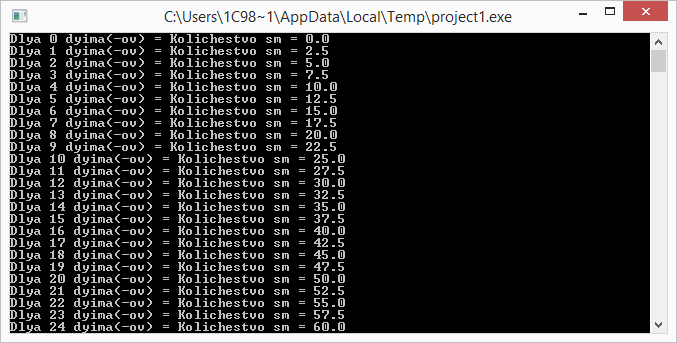
a(inch,sm);

end;

readln;

end.

**Результат работы программы:**



**Анализ:** Мы написали программу, подсчитывающую количество сантиметров для дюймов в диапазоне от 0 до 100, используя пользовательскую процедуру для вывод каждого значения дюйма и эквивалентного ему значения сантиметра на экран.

**Вывод:** Мы написали программу, подсчитывающую количество сантиметров для дюймов в диапазоне от 0 до 100, используя информацию о том, что 1 дюйм = 2.5 см.

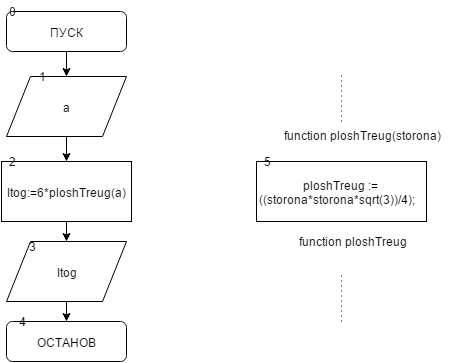
**Задание 2.**

**Постановка задачи:**

Вычислить площадь правильного шестиугольника со стороной а, используя функцию вычисления площади треугольника

**Математическая модель:**

**Блок-схема:**



**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| ploshTreug | real | Функция, считающая площадь треугольника через сторону а |
| a | real | Сторона правильного треугольника, вводимая пользователем с клавиатуры |
| Itog | real | Итоговая площать шестиугольника |
| storona | real | Параметр функции |

**Код программы:**

program zadanie2;

var Itog, a : real;

function ploshTreug(storona:real):real;

begin

ploshTreug := ((storona\*storona\*sqrt(3))/4);

end;

begin

Writeln('Vvedite storony treygolnika') ;

readln(a);

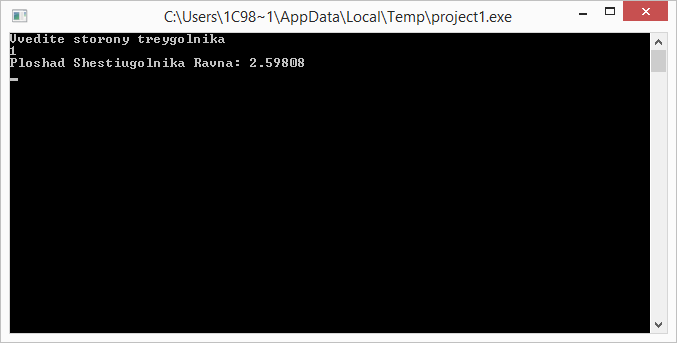
Itog := 6\*ploshTreug(a);

writeln('Ploshad Shestiugolnika Ravna: ',Itog:2:5);

readln;

end.

**Результат работы программы:**



**Анализ:** Мы написали программу, которая считает площадь правильного шеустиугольника через его сторону a, которую пользователь вводит с клавиатуры. Формула, используемая для подсчета значений площади:

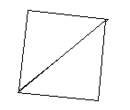
**Вывод:** Мы написали программу, которая считает площадь правильного шеустиугольника через его сторону a.

**Задание 3.**

**Постановка задачи:**

Вычислить площадь фигуры, заданной сторонами и диагональю. Фигура не является прямоугольником, а треугольники, которые ее составляют, не являются прямоугольными.

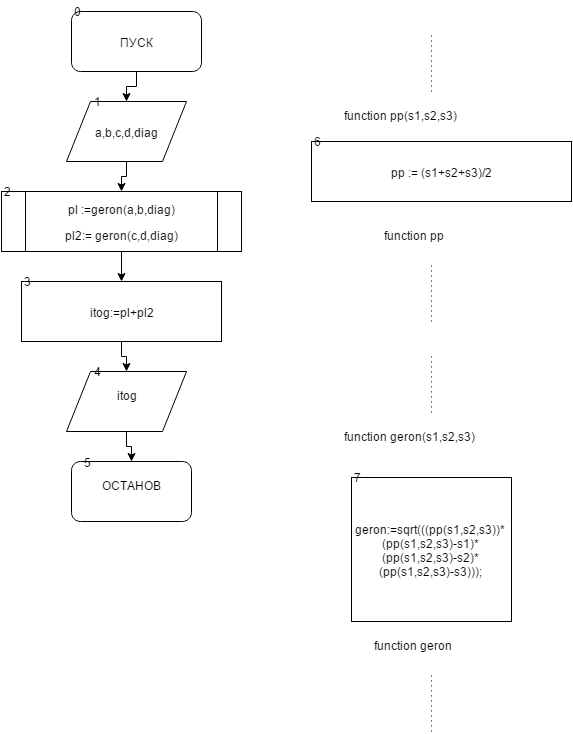
**Математическая модель:**



(Формула Герона для вычисления площади одного из треугольников)

p – полупериметр, a,b,c – стороны треугольника

**Блок-схема:**



**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| a | real | Сторона AB |
| b | real | Сторона BC |
| c | real | Сторона CD |
| d | real | Сторона DA |
| pl | real | Площадь треуг-ка ABC |
| pl2 | real | Площадь треуг-ка ACD |
| itog | real | Общая площадь фигуры |
| pp | real | Функция, считающая полупериметр |
| geron | real | Функция, считающая площадь треугольника по формуле Герона |
| s1 | real | Параметр функции(первая сторона) |
| s2 | real | Параметр функции(вторая сторона) |
| s3 | real | Параметр функции(третья сторона) |
| diag | real | Диагональ фигуры |

**Код программы:**

program zadanie3;

var

a,b,c,d,diag:real;

pl,pl2,itog:real;

function pp(s1,s2,s3:real):real;

begin

pp:=((s1+s2+s3)/(2));

end;

function geron(s1,s2,s3:real):real;

begin

geron:=sqrt(((pp(s1,s2,s3))\*(pp(s1,s2,s3)-s1)\*(pp(s1,s2,s3)-s2)\*(pp(s1,s2,s3)-s3)));

end;

begin

Writeln('Vvedite storony AB');

readln(a);

Writeln('Vvedite storony BC');

readln(b);

Writeln('Vvedite storony CD');

readln(c);

Writeln('Vvedite storony DA');

readln(d);

Writeln('Vvedite diagonal');

readln(diag);

pl:=geron(a,b,diag);

pl2:=geron(c,d,diag);

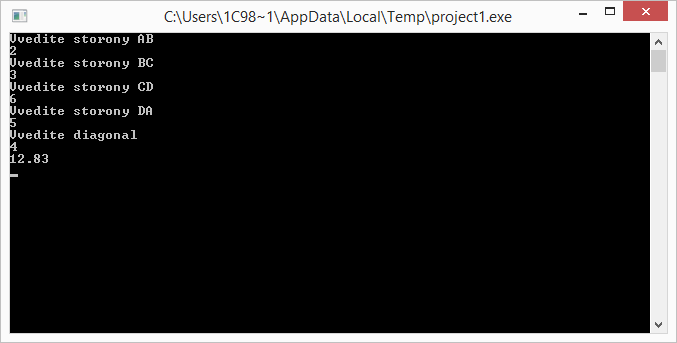
itog:=pl+pl2;

Writeln(itog:2:2);

readln();

end.

**Результат работы программы:**

****

**Анализ:** Мы написали программу, вычисляющую общую площадь фигуры, разделенной на два треугольника диагональю, путем вычисления площади каждого из треугольников и их суммирования.

При вычислении площади каждого из треугольников использовалась формула Герона:

, где

p – полупериметр треугольника

a – первая сторона треугольника

b – вторая сторона треугольника

c – третья сторона треугольника(в нашем конкретном случая третьей стороной для обоих треугольников являлась диагональ)

**Вывод:**

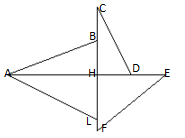
Мы написали программу, вычисляющую общую площадь фигуры, разделенной на два треугольника диагональю, путем вычисления площади каждого из треугольников и их суммирования.

**Задание 4.**

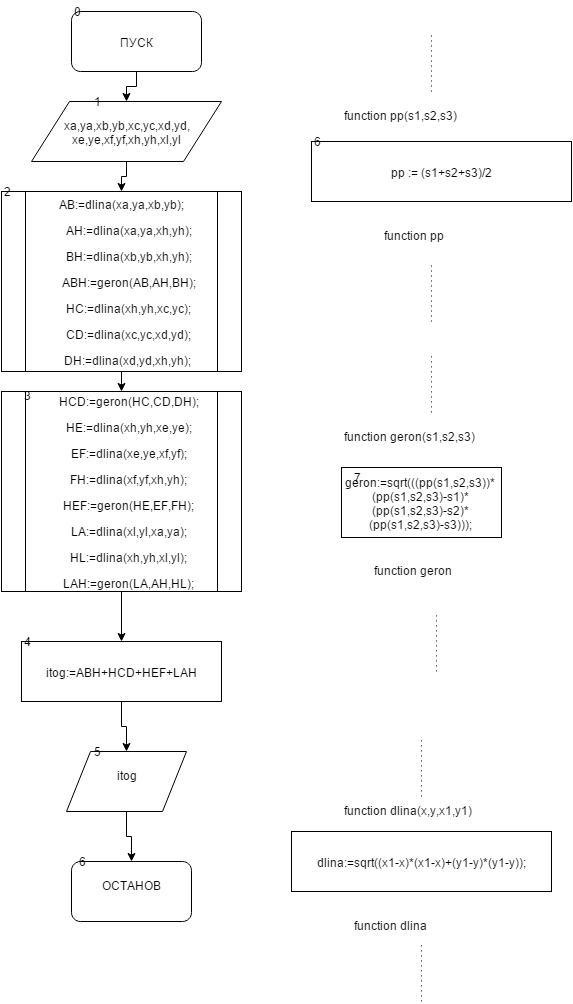
**Постановка задачи:**

Вычислить площадь фигуры, которая задана координатами точек. Треугольники, из которых состоит фигура, не являются прямоугольными.

**Математическая модель:**



**Блок-схема:**



**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| x\* | Real | Переменная x\*, где \*-буквa a/b/c/d/e/h/l/f — первая координата точки A/B/C/D/E/H/L/F соответственно |
| y\* | Real | Переменная y\*, где \*-буквa a/b/c/d/e/h/l/f — вторая координата точки A/B/C/D/E/H/L/F соответственно |
| AB,AH,BH,HC,CD,DH,HE,EF,FH,LA,HL | Real | Стороны треугольников |
| ABH,HCD,HEF,LAH | Real | Площади треугольников |
| pp | Real | Функция, считающая полупериметр треугольника |
| geron | Real | Функция, считающая площадь треугольника по формуле Герона |
| dlina | Real | Функция, считающая длину отрезка |
| s1 | Real | Параметр функции. Первая сторона |
| s2 | Real | Параметр функции. Вторая сторона |
| s3 | real | Параметр функции. Третья сторона |
| x | real | Параметр функции. Координата х начала отрезка |
| y | real | Параметр функции. Координата у начала отрезка |
| x1 | real | Параметр функции. Координата х конца отрезка |
| y1 | real | Параметр функции. Координата у конца отрезка |
| itog | real | Сумма площадей вех треугольников |

**Код программы:**

program zadanie4;

var

xa,ya,xb,yb,xc,yc,xd,yd,xe,ye,xf,yf,xh,yh,xl,yl:real;

AB,AH,BH,HC,CD,DH,HE,EF,FH,LA,HL:real;

ABH,HCD,HEF,LAH,itog:real;

function pp(s1,s2,s3:real):real;

begin

pp:=((s1+s2+s3)/(2));

end;

function geron(s1,s2,s3:real):real;

begin

geron:=sqrt(((pp(s1,s2,s3))\*(pp(s1,s2,s3)-s1)\*(pp(s1,s2,s3)-s2)\*(pp(s1,s2,s3)-s3)));

end;

function dlina(x,y,x1,y1:real):real;

begin

dlina:=sqrt((x1-x)\*(x1-x)+(y1-y)\*(y1-y));

end;

begin

Writeln('Vvedite koordinaty tochki A');

Readln(xa,ya);

Writeln('Vvedite koordinaty tochki B');

Readln(xb,yb) ;

Writeln('Vvedite koordinaty tochki C');

Readln(xc,yc) ;

Writeln('Vvedite koordinaty tochki D');

Readln(xd,yd) ;

Writeln('Vvedite koordinaty tochki E');

Readln(xe,ye) ;

Writeln('Vvedite koordinaty tochki F');

Readln(xf,yf);

Writeln('Vvedite koordinaty tochki H');

Readln(xh,yh) ;

Writeln('Vvedite koordinaty tochki L');

Readln(xl,yl) ;

AB:=dlina(xa,ya,xb,yb);

AH:=dlina(xa,ya,xh,yh);

BH:=dlina(xb,yb,xh,yh);

ABH:=geron(AB,AH,BH);

HC:=dlina(xh,yh,xc,yc);

CD:=dlina(xc,yc,xd,yd);

DH:=dlina(xd,yd,xh,yh);

HCD:=geron(HC,CD,DH);

HE:=dlina(xh,yh,xe,ye);

EF:=dlina(xe,ye,xf,yf);

FH:=dlina(xf,yf,xh,yh);

HEF:=geron(HE,EF,FH);

LA:=dlina(xl,yl,xa,ya);

HL:=dlina(xh,yh,xl,yl);

LAH:=geron(LA,AH,HL);

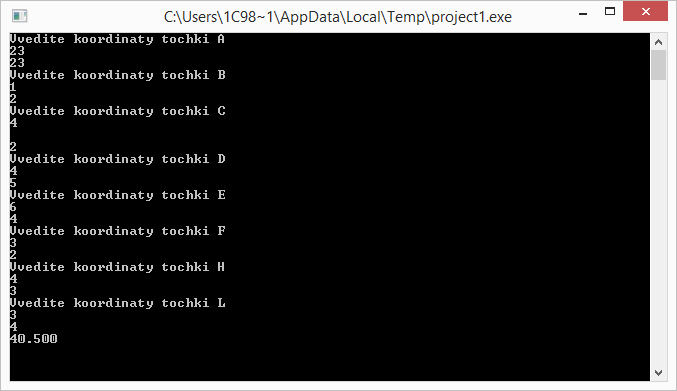
itog:=ABH+HCD+HEF+LAH;

Writeln(itog:2:3);

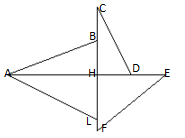
readln();

end.

**Результат работы программы:**



**Анализ:** Мы написали программу, реализующую вычисление площади данной фигуры



посредством суммирования площадей треугольников ABH,HCD,HEF,LAH.

**Вывод:** Мы написали программу, реализующую вычисление данной нам фигуры с ипользование пользовательских функций.

**Задание 5.**

**Постановка задачи:**

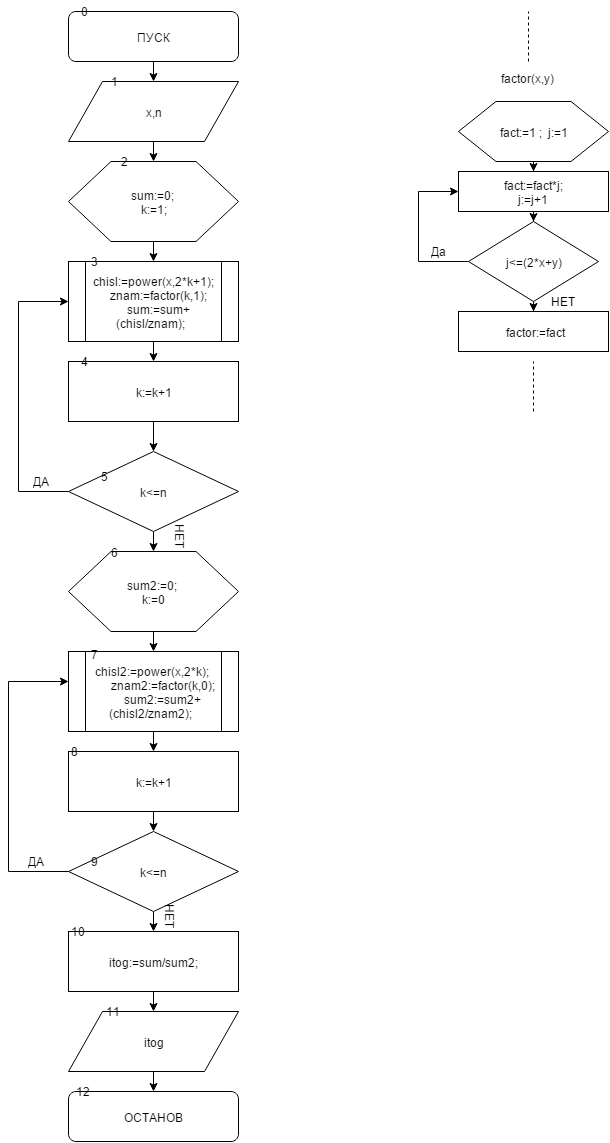
Вычислить



**Математическая модель:**



**Блок-схема:**

**Список идентификаторов:**

|  |  |  |
| --- | --- | --- |
| Имя | Тип | Смысл |
| chisl | real | Числитель числителя |
| chisl2 | real | Числитель знаменателя |
| znam | real | Знаменатель числителя |
| znam2 | real | Знаменатель знаменателя |
| sum | real | Сумма в числителе |
| sum2 | real | Сумма в знаменателе |
| k | integer | Параметр цикла |
| n | longint | Вводимое значение |
| x | longint | Вводимое значение |
| itog | real | Итог |

**Код программы:**

program zadanie5;

uses math;

var chisl,znam,sum,chisl2,znam2,sum2,itog:real;

k:integer;

x,n:longint;

function factor(x,y:longint):longint;

var fact:longint;

j:integer;

begin

fact:=1 ;

for j:=1 to (2\*x+y) do

fact:=fact\*j;

factor:=fact;

end;

begin

Writeln('Vvedite x');

Readln(x);

Writeln('Vvedite n');

Readln(n);

sum2:=0;

sum:=0;

for k:=1 to n do begin

chisl:=power(x,2\*k+1);

znam:=factor(k,1);

sum:=sum+(chisl/znam);

end;

sum2:=0;

for k:=1 to n do begin

chisl2:=power(x,2\*k);

znam2:=factor(k,0);

sum2:=sum2+(chisl2/znam2);

end;

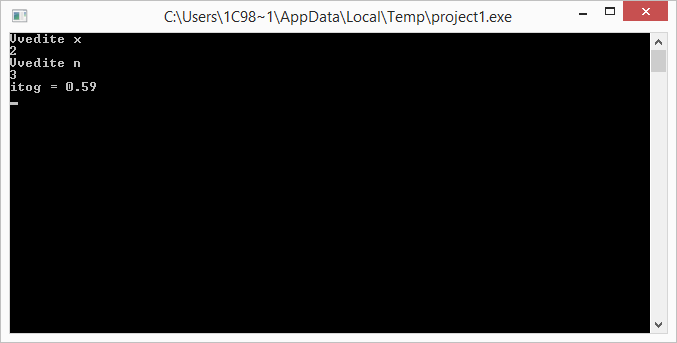
itog:=sum/sum2;

writeln('itog = ',itog:2:2);

readln();

end.

**Результат работы программы:**



**Анализ:** Мы написали программу, вычисляющую значение t, зависящее от х по формуле



**Вывод: Мы написали программу, вычисляющую значение t, зависящее от х по формуле, данной в задании №5.**