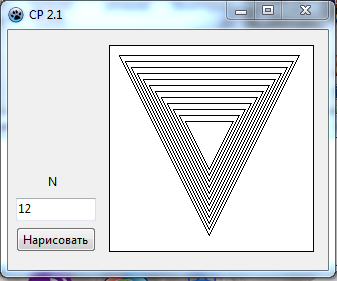
**Самостоятельная работа 2**

**Инвариантная часть**

**Задание 1:**

Разработать математическую модель и программную реализацию задачи по вычерчиванию произвольного числа треугольников, располагаемых внутри друг друга.

**Скриншот:**



**Код процедуры:**

procedure TForm1.Button1Click(Sender: TObject);

const

n = 4;

l = 2;

var

p: array [1..n, 1..l] of integer = ((10, 10), (190, 10), (100, 190), (10, 10));

i, j, k, g, num: integer;

begin

Image1.Canvas.Rectangle(0, 0, Image1.Width, Image1.Height);

num := StrToInt(Edit1.Text);

for k := 1 to num do

begin

Image1.Canvas.MoveTo(p[1, 1], p[1, 2]);

for g := 2 to n do

Image1.canvas.lineto(p[g, 1], p[g, 2]);

for i := 1 to n do

for j := 1 to l do

begin

if (((i = 1) and (j = 1)) or ((i = 1) and (j = 2)) or

((i = 2) and (j = 2)) or ((i = 4) and (j = 1)) or

((i = 4) and (j = 2))) then

p[i, j] := p[i, j] + 6;

if (((i = 2) and (j = 1)) or ((i = 3) and (j = 2))) then

p[i, j] := p[i, j] - 6;

end;

end;

end;

**Задание 2:**

Разработайте программу для вычерчивания следующей последовательности отрезков прямых линий:

из точки (1.0, 6.0) в точку (1.0, 1.0);

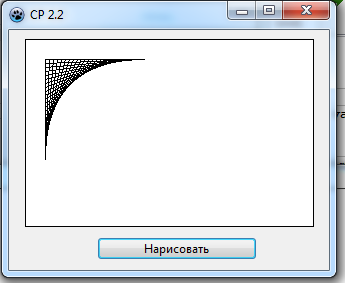
из точки (1.0, 5.8) в точку (1.2, 1.0);

из точки (1.0, 5.6) в точку (1.4, 1.0);

.....

из точки (1.0, 1.0) в точку (6.0, 1.0);

**Скриншот:**



**Код процедуры:**

procedure TForm1.Button1Click(Sender: TObject);

var x,xi,y,yi:integer;

begin

x := 10;

y := 60;

xi := 10;

yi := 10;

Image1.Canvas.Rectangle(0, 0, Image1.Width, Image1.Height);

while(xi <= 60) do

begin

Image1.Canvas.MoveTo(x\*2,y\*2);

Image1.Canvas.LineTo(xi\*2,yi\*2);

xi := xi + 2;

y := y - 2;

end;

end;

**Задание 3:**

Разработать математическую модель и программную реализацию задачи по вычерчиванию 30 стрелок на окружности с центром в (x,y).

**Скриншот:**



**Код процедуры:**

{ TForm1 }

procedure TForm1.Button1Click(Sender: TObject);

var

h,w,x,xi,x3,x4,y,yi,y3,y4,radius,xs,ys,dx1,dx2,dy1,dy2: integer;

a,ai,p,p1,p2,angle: real;

begin

Image1.Canvas.Rectangle(0, 0, Image1.Width, Image1.Height);

w:=Image1.Width;

h:=Image1.Height;

a:=pi/15;

radius:=round(w/10+w/8);

xs:=round(w/2);

ys:=round(h/2);

Image1.Canvas.MoveTo(xs,0);

Image1.Canvas.LineTo(xs,Image1.Height);

Image1.Canvas.MoveTo(0,ys);

Image1.Canvas.LineTo(Image1.Width,ys);

xi:=xs+radius;

yi:=ys;

ai:=a;

while ai<=2\*pi+a do

begin

x:=xi;

y:=yi;

xi:=round(xs+radius\*cos(ai));

yi:=round(ys+radius\*sin(ai));

Image1.Canvas.MoveTo(x,y);

Image1.Canvas.LineTo(xi,yi);

p1:=y-yi;

p2:=x-xi;

if p2=0 then

p:=pi/2

else p:=p1/p2;

angle:=pi/3-ArcTan(abs(p));

dx1:=round(10\*sin(angle));

dy1:=round(10\*cos(Angle));

angle:=angle+pi/3;

dx2:=round(10\*sin(angle));

dy2:=round(10\*cos(angle));

if p1<0 then

begin

if p2<0 then

begin

x3:=x+dx1;

y3:=y+dy1;

x4:=x+dx2;

y4:=y+dy2;

end

else begin

x3:=x-dx1;

y3:=y+dy1;

x4:=x-dx2;

y4:=y+dy2;

end

end

else begin

if p2<0 then

begin

x3:=x+dx1;

y3:=y-dy1;

x4:=x+dx2;

y4:=y-dy2;

end

else begin

x3:=x-dx1;

y3:=y-dy1;

x4:=x-dx2;

y4:=y-dy2;

end

end;

Image1.Canvas.MoveTo(x,y);

Image1.Canvas.LineTo(x3,y3);

Image1.Canvas.MoveTo(x,y);

Image1.Canvas.LineTo(x4,y4);

ai:=ai+a;

end;

end;