

Vilniaus Mykolo Biržiškos gimnazija

Informacinių technologijų darbas

KAMUOLYS

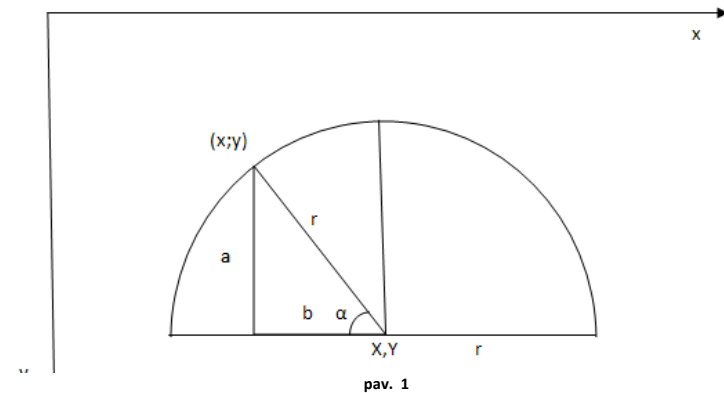
(Programavimas „Pascal“ kalba)

Užduotis

Parašykite programą, kuri sukurtų tokį vaizdą – ekraną padalintą į dvi dalis, vaizduojančias žemę ir dangų, viršutinėje dalyje naudodami apskritimo funkcijas nupieškite saulę. Naudodamiesi linijos ir apskritimo funkcijomis nupieškite du stilizuotus žmones, naudodami apskritimo funkciją nupieškite kamuolį. Panaudokite judesio algoritmą, kad kamuolys judėtų tarp žmonių, tarsi jie jį mėtytų.

Idėja

1. Kamuolio judėjimo trajektorija turi būti lankas.



Šiam judėjimui sugalvojau algoritmą.

Kamuolio padėties taško koordinatė x lygi iš vidurio taško koordinatės X atimtai b reikšmei, kuri randama spindulį (r) dauginant iš kosinuso iš alfa (α). Padėties taško koordinatė y randama iš vidurio taško koordinatės Y atimtai a reikšmei, kuri randama spindulį (r) dauginant iš sinuso iš alfa (α).

$$x = X - (r * \cos \alpha)$$

$$y = Y - (r * \sin \alpha)$$

Belieka tik cikle didinti kampą α ir keisis kamuolio padėtis.

2. Kamuolys turi nesustojamai keisti kryptį.

Šiai užduočiai taip pat reikėjo mažo algoritmo. Kai tik α (alfa) reikšmė pasiekia 180° , kintamojo (h) reikšmė tampa lygi 1 ir sąlygos sakiniu nurodoma, kad kol h reikšmė lygi 1 judėjimo kryptis (α) mažėja. Kai tampa lygi 0, kintamasis h taip pat prilyginamas 0 ir sąlygos sakiniu nurodoma, kad kol h reikšmė lygi 0, α reikšmė didės.

```
if a = 180
    then h:=1;

if a = 0
    then h:=0;
if h = 0
    then a:=a + 1
    else a:=a - 1;
```

3. Žmogeliukų rankos

Kamuoliui pasiekiant tam tikrą tašką, prie žmogaus kūno pasirodo elipsės formos ranka, truputį užsilaiko ir dingsta. Tai padaroma paprasčiausiai nurodant, jog ranka atsirasti turi kai $\alpha < 10^\circ$, arba kai $\alpha > 170^\circ$. Kadangi rankos neužsiliaika ir pasiekus tam tikrą tašką dingsta (vizualiai atrodo negražiai), teko sugalvoti jų užsilaikymo algoritmą.

Vieno žmogaus rankos algoritmas. Kol $\alpha < 10^\circ$ kintamasis q prilyginamas 30. Kai tik α tampa didesnė už 10° iš q reikšmės kiekvieną ciklą vis atimamas 1, taip ranka piešiama ir trinama kiekvieną ciklą, kol q reikšmė didesnė už 0.

Programa

```
program Kamuolys;
uses Graph, Crt;
var gd, gm, //numatytieji kintamieji
    q, m, //kintamieji ranku trinimo uzdelsimui
    x,y, //koordinates nusakantys kintamieji
    h:integer; //kintamasis padedantis nurodyti kamuolio judejimo krypti
    a:real; //kintamasis laipsniams
//-----
procedure saulute(x:integer); // keiciant x, galima keisti saules padeti (horizontalia)
begin
    SetFillStyle(1, 14);
    FillEllipse(x, 100, 50, 50); //saule
    SetColor(0); //kaire akis
    Circle (x - 20, 80, 11);
    SetFillStyle(1, 15);
    FillEllipse(x - 20, 80, 10, 10);
    SetFillStyle(1, 0);
    FillEllipse(x - 20, 80, 4, 4);
    SetColor(0); //desine akis
    Circle (x + 20, 80, 11);
    SetFillStyle(1, 15);
    FillEllipse(x + 20, 80, 10, 10);
    SetFillStyle(1, 0);
    FillEllipse(x + 20, 80, 4, 4);
    SetColor(0); //lupos
    Arc (x, 100, 25, 155, 10);
    SetColor(0);
    Arc (x, 100, 205, 335, 25);
    SetColor(0);
    Arc (x, 110, 205, 335, 20);
    SetFillStyle(1, 4);
    FloodFill(x, 126, 0);
end;
//-----
procedure fonas (riba:integer); //galima reguliuoti ribos tarp dangaus ir zoles auksti
var Xmax, Ymax :integer;
begin
    Xmax:= getMaxX;
    Ymax:= getMaxY;
    SetFillStyle (1, 3); //dangus
    Bar(0, 0, Xmax, riba);
    SetFillStyle(1, 2);
    Bar(0, riba, Xmax, Ymax); //zole
```

```

end;
//-----
procedure zmogus (r:integer); //keiciant r galima keisti zmogaus padeti (horizontaliai)
begin
SetFillStyle(1, 0);
Bar(125 + r, 380, 200 + r, 500);
SetColor(2);
Line(125 + r, 400, 162 + r - 10, 500);
Line(162 + r + 10, 500, 200 + r, 400);
SetColor(0);
FillEllipse(162 + r, 360, 20, 20);
SetFillStyle(1, 2);
FloodFill(126 + r, 403, 2);
FloodFill(163 + r + 10, 500, 2);
end;
//-----
procedure judejimas(x0, y0, r:integer; a:real; var x, y:integer); { X ir Y nuliniai tai judejimo lanku centro
koordinates, r-lanko skersmuo, a-laipsniai, x ir y isvesties koordinates}
begin
x:=Round(x0 - (r * Cos (Pi * a / 180))); //kamuolio judejimo lanku algoritmas
y:=Round(y0 - (r * Sin (Pi * a / 180)));
end;
//-----
procedure kamuoliukas(t, u:integer);
begin
SetFillStyle(1, 8); //kamuolio piesimas
FillEllipse(t, u, 30, 30);
Delay(10);
SetColor(3); //kamuolio trynimas
SetFillStyle(1, 3);
FillEllipse(t, u, 30, 30);
end;
//-----
procedure ranka(a:real; vt, q:integer; var o:integer);
begin
if q > 0
then
begin
SetFillStyle(1, 0);
FillEllipse(vt, 392, 10,5);
end;
if q = 0
then
begin
SetFillStyle(1, 3);
FillEllipse(vt, 392, 10, 5);
end;
end;

```

```

o:=q - 1;
end;
//-----
begin
InitGraph(gd, gm, "");
fonas(400);
saulute(200);
zmogus(75);
zmogus(500);
While not keypressed do
begin
judejimas(450, 360, 150, a, x, y);
kamuoliukas(x, y);

if a = 180 //judejimo kryptis
then h:=1;
if a = 0
then h:=0;
if h = 0
then a:=a + 1
else a:=a - 1;

if a<10 //q ir m - kintemieji, kurie uzdelsina ranku dingimo laika
then q:=30;
if a>170
then m:=30;

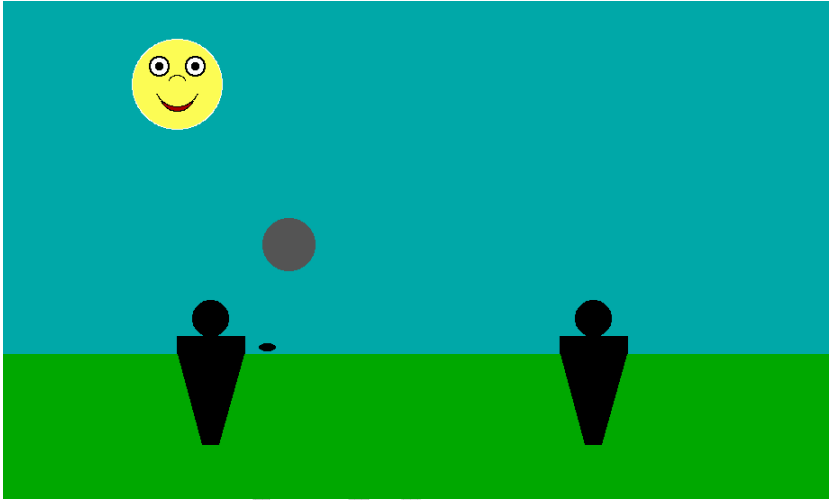
ranka(a, 300, q, q);
ranka(a, 600, m, m);
end;
Readln;
end.

```

Išvados

Išmokau grafiniam vaizde išgauti geometrines figūras, panaudoti judėjimo sąlygas.

Programos išvesties į ekraną vaizdas



Turinys

KAMUOLYS	1
Užduotis	2
Idėja	2
1. Kamuolio judėjimo trajektorija turi būti lankas.	2
2. Kamuolys turi nesustojamai keisti kryptį.	3
3. Žmogeliukų rankos	3
Programa	4
Išvados	7
Programos išvesties į ekraną vaizdas	7