

```

program Plaukia_Valtis_Ezere;
uses Graph, crt;
var gd, gm, y, x, m, yl, xl, greitis, i, taskox, taskoy: integer;
korpusas: array[1..10] of integer;
begin
  WriteLn('Kitame lange pliaukios valtis!');
  Randomize;
  Write('Įveskite norimą valties greitį nuo 1 iki 1200: '); ReadLn(greitis);
  InitGraph (gd, gm, '');
  WriteLn('Kitame lange pliaukios valtis!');

  //Žolės piešimas.
  SetColor(2);
  setfillstyle(1, green);
  bar(0, 0, 1700, 286); //Ši funkcija reikalinga norint nuspalvinti viršutinę dalį žaliai
- lyg žolė.

  y:=1; //Bus reikalinga, kad y būtų 1, tad nustatau tokia jo reikšmę.
  SetColor(0);
  SetLineStyle(0, 3, 1);
  while y<286 do
  begin
    x:=Random(1700); //Atsitiktinai renku x koordinatą
    m:=y;
    y:=Random(286); //Atsitiktinai renku y koordinatą //Apskritai, visose
šiose eilutėse "auginu žolę"
    Moveto(x, y);
    Lineto(x-2, y-20); //Nupiešia vieną iš žolę sudarančių linijų.
    xl:=x-5;
    yl:=y-20;
    Lineto(xl-2, yl+20); //Piešia kita žolę sudarančią liniją.
    y:=m+1;
  end;

  //Valties piešimas ir judėjimas, vandens piešimas.
  x:=130; y:=672; //Valties tuščiosios dalies centro koordinatės
  yl:=750; //Apatinio dešinio valties kampo koordinatės.
  i:=1; //Kodėl nustatau tokia i reikšmę pamatysite kiek vėliau.
  repeat //Ji reikalinga tam, kad kartojama būtų amžinai;
  while x<1700 do
  begin
    setfillstyle(1, blue);
    bar(0, 286, 1700, 1200); //Ši funkcija reikalinga norint vėl užspalvinti valtį, kad
neliktų valties "Dėmių";
    taskoy:=1;
    while taskoy<1000 do
    begin
      m:=taskoy;
      taskoy:=(Random(913)+287); //Sudėlioju taškiukus vandens paviršiuje,
kurie vėliau sukels raibuliuojančio vandens išpūdį
      taskox:=Random(1700);
      PutPixel(taskox, taskoy, cyan);
      taskoy:=m+1;
    end;
    Setcolor(0);
    SetFillStyle(1, 6);
    //Valties korpusas//;
    SetLineStyle(0, 4, 1);
    korpusas[1]:=x-180;
    korpusas[2]:=y;
    korpusas[3]:=x-100;
    korpusas[4]:=yl;
    korpusas[5]:=x+100; // Čia parenkami taškai nubrėžti daugiakampiui - tai
turėtų būti trapecija, tai yra, valties korpusas.
    korpusas[6]:=yl;
    korpusas[7]:=x+180;
    korpusas[8]:=y;
    korpusas[9]:=x-182;
  end;
end;
end.

```

```

korpusas[10]:=y;
setcolor(brown); Setfillstyle(1, brown);
Drawpoly(5, korpusas);
FillPoly(5, korpusas); //Piešiamas ir nuspalvinamas valties korpusas.
SetColor(black);
SetLineStyle(0, 4, 1);
FillEllipse(x, y, 180, 30); //Piešiu valties tuščiąją dalį
MoveTo(x-180, y); //Valties korpusą apvedžiosiu juoda linija paruošiu tam kursorių
LineTo(x-100, yl); LineTo(x+100, yl); LineTo(x+180, y);
//Dabar reikia parašyti valties judėjimą//;

x:=x+greitis; //Tai eilutė nuo kurios priklauso valties greitis.
delay(200); // Kad valtis mažiau mirgėtų plaukdama.
end;
if x>=1700 then //Ši funkcija apgręš valtį, kai ji pasieks dešinį ekrano
kraštą.
begin
  while x>1 do
  begin
    setfillstyle(1, blue);
    bar(0, 286, 1700, 1200); //Ši funkcija reikalinga norint vėl užspalvinti valtį, kad
neliktų valties "Dėmių";
    taskoy:=1;
    while taskoy<1000 do
    begin
      m:=taskoy;
      taskoy:=(Random(913)+287); //Sudėlioju taškiukus vandens paviršiuje,
kurie vėliau sukels raibuliuojančio vandens išpūdį
      taskox:=Random(1700);
      PutPixel(taskox, taskoy, cyan);
      taskoy:=m+1;
    end;
    Setcolor(0);
    SetFillStyle(1, 6);
    //Valties korpusas
    SetLineStyle(0, 4, 1);
    korpusas[1]:=x-180;
    korpusas[2]:=y;
    korpusas[3]:=x-100;
    korpusas[4]:=yl;
    korpusas[5]:=x+100; // Čia parenkami taškai nubrėžti daugiakampiui - tai
turėtų būti trapecija, tai yra, valties korpusas.
    korpusas[6]:=yl;
    korpusas[7]:=x+180;
    korpusas[8]:=y;
    korpusas[9]:=x-180;
    korpusas[10]:=y;
    setcolor(brown); Setfillstyle(1, brown);
    Drawpoly(5, korpusas);
    FillPoly(5, korpusas);
    SetColor(black);
    SetLineStyle(0, 4, 1);
    FillEllipse(x, y, 180, 30); //Piešiu valties tuščiąją dalį//;
    MoveTo(x-180, y); //Valties korpusą apvedžiosiu juoda linija paruošiu tam kursorių
    LineTo(x-100, yl); LineTo(x+100, yl); LineTo(x+180, y);
    //Dabar reikia parašyti valties judėjimą//;

    x:=x-greitis; //Tai eilutė nuo kurios priklauso valties greitis.
    delay(2);
  end;
end;
until i=0; // Taip nebus niekada, tad kartojama - amžinai;
ReadLn;
end.

```