Лабораторная работа №5

по курсу "Системы аналитических вычислений"
студент: Ляшун Д. С.

Задание 1 - Нахождение НОД полиномов

Вариант 3

In[1]:=
```
x = .
f := 126*x^6 - 54*x^5 - 28*x^3 + 12*x^2 + 14*x - 6;
g := -42*x^5 + 39*x^4 - 9*x^3 - 189*x + 81;
```

1. Нахождение НОД с помощью расширенного алгоритма Евклида

In[4]:=
```
myBezoutPoly[aa_, bb_] := Module[
            {a = aa, b = bb
                , x0 = 1
                , xx = 0
                , y0 = 0
                , yy = 1
                , q, r
            }
            ,
     While[
                Not[SameQ[b, 0]]
                ,
                    q = PolynomialQuotient[a, b, x];
                    r = PolynomialRemainder[a, b, x];
                    {a, b} = {b, r};
                    {x0, xx} = {xx, (x0 - xx * q) // ExpandAll};
                    {y0, yy} = {yy, (y0 - yy * q) // ExpandAll};

            ];
           list = CoefficientList[a, x];
           x0 /= list[[1]];
           y0 /= list[[1]];
           a /= list[[1]];
            {x0, y0, a}
        ];
res1 = myBezoutPoly[f, g];
{u1, v1} = {Simplify[res1[[1]]], Simplify[res1[[2]]]};
iden1 = Simplify[u1 * f + v1 * g];
Print["u = ", u1];
Print["v = ", v1];
Print["u*f+v*g = ", iden1];
Print[If[iden1 === Simplify[res1[[3]]], "u*f+v*g = gcd(f, g)", "u*f+v*g ≠ gcd(f, g)"]]
```

u = $\dfrac{-509\,443 - 404\,829\, x + 1\,998\,607\, x^2 - 2\,079\,874\, x^3}{2\,168\,259\,612}$

v = $\dfrac{40\,096\,351 - 44\,981\, x + 335\,277\, x^2 + 4\,314\,015\, x^3 - 9\,359\,433\, x^4}{3\,252\,389\,418}$

u*f+v*g = $1 - \dfrac{7\,x}{3}$

u*f+v*g = gcd(f, g)

2. Нахождение НОД с помощью встроенной функции

In[12]:=
```
res2 = PolynomialExtendedGCD [f, g, x];
res2[[2, 1]] /= CoefficientList [res2[[1]], x][[1]];
res2[[2, 2]] /= CoefficientList [res2[[1]], x][[1]];
res2[[1]] /= CoefficientList [res2[[1]], x][[1]];
{u2, v2} = {Simplify[res2[[2, 1]]], Simplify[res2[[2, 2]]]};
iden2 = Simplify[u2 * f + v2 * g];
Print["u = ", u2];
Print["v = ", v2];
Print["u*f+v*g = ", iden2];
Print[If[iden2 === Simplify[res2[[1]]], "u*f+v*g = gcd(f, g)", "u*f+v*g ≠ gcd(f, g)"]]
```

$$u = \frac{-509\,443 - 404\,829\, x + 1\,998\,607\, x^2 - 2\,079\,874\, x^3}{2\,168\,259\,612}$$

$$v = \frac{40\,096\,351 - 44\,981\, x + 335\,277\, x^2 + 4\,314\,015\, x^3 - 9\,359\,433\, x^4}{3\,252\,389\,418}$$

$$u*f+v*g = 1 - \frac{7\,x}{3}$$

u*f+v*g = gcd(f, g)

In[22]:=
```
Print[If[Simplify[res2[[1]]] === Simplify[res1[[3]]],
   "gcd1(f,g) = gcd2(f,g)", "gcd1(f,g) ≠ gcd2(f, g)"]];
```

gcd1(f,g) = gcd2(f,g)

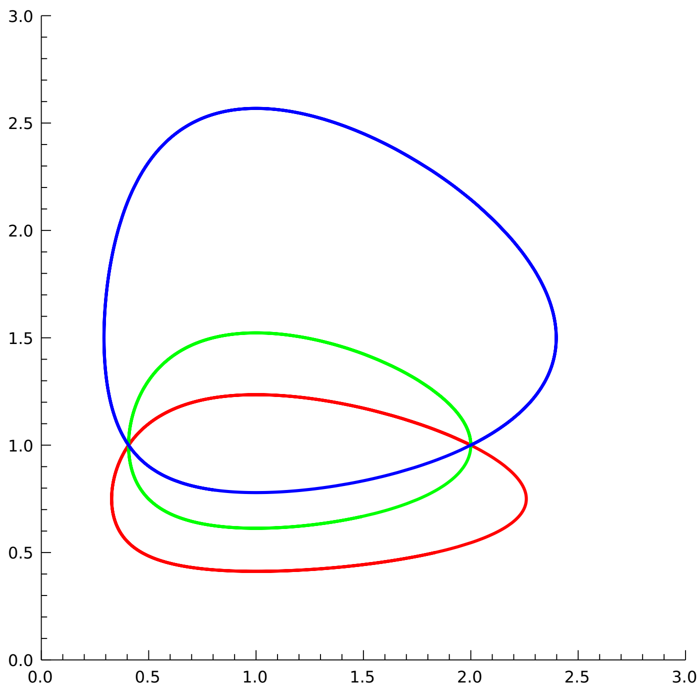
Как видно, тождество Безу ***u*f+v*g = gcd(f, g)*** выполняется, и НОД, посчитанные с помощью встроенного метода и собственного, совпадают.

Задание 2

1. Модель "Хищник-жертва"

In[23]:=
```
x = .;
y = .;
plt = {};
colors = {Red, Green, Blue};
For[{b = 4, i = 1}, b > 1, {b--, ++i},
{a, c, d} =  {3, 1, 1};
eq1[t_] := x'[t] == (a - b y[t]) x[t];
eq2[t_] := y'[t] == (-c + d x[t]) y[t];
sol =
  NDSolve[{ eq1[t], eq2[t], x[0] == 2, y[0] == 1}, {x, y}, {t, 0, 7}, MaxSteps → 3000];
plt = Append[plt, ParametricPlot [Evaluate[{x[t], y[t]} /. sol],
    {t, 0, 7}, PlotRange → {{0, 3}, {0, 3}}, PlotStyle → {colors[[i]]}]] ]
Show[
 plt]
```

Out[28]=



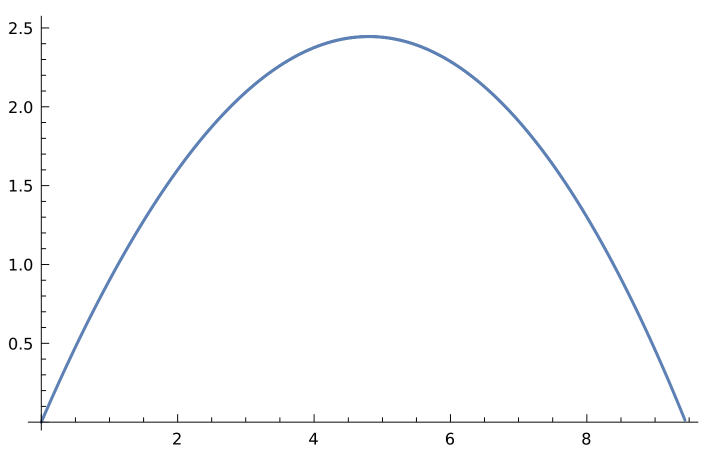
2. Падение тела, брошенного под углом к горизонту

In[87]:=

```
tmax = 1.41;
alph = Pi / 4;
v0 = 10;
 y0 = {0, 0, v0 * cos (alph), 0, v0 * sin (alph)};
k = 0.01;
g = 9.81;
sol = NDSolve[
  {
   y1 '[t] == y2[t],
   y2 '[t] == -k * y2[t] * Sqrt[y2[t] ^ 2 + y4[t] ^ 2],
   y3 '[t] == y4[t],
   y4 '[t] == -k * y4[t] * Sqrt[y2[t] ^ 2 + y4[t] ^ 2] - g,
   y1[0] == 0,
   y2[0] == v0 * Cos[alph],
   y3[0] == 0,
   y4[0] == v0 * Sin[alph]
  },
  {y1, y2, y3, y4},
  {t, 0, tmax},
  MaxSteps → 10 000
];
dataset = Table[
 {y1[t], y3[t]} /. sol[[1]],
 {t, 0, tmax, 0.001}
];

ListPlot[dataset]
```

Out[95]=



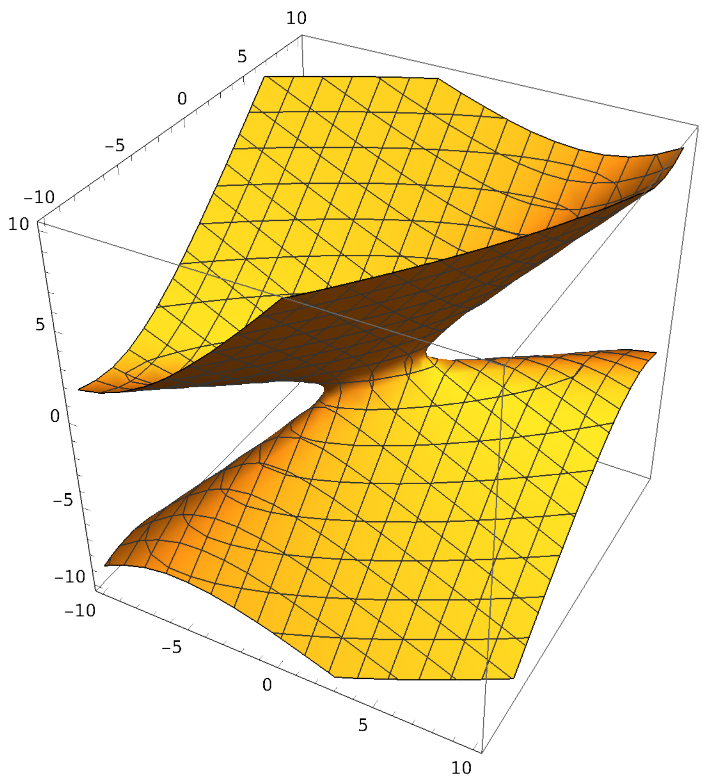
Задание 3 - Упрощение уравнения поверхности второго порядка в пространснтве

Вариант 3

In[38]:=
```
x = .
y = .
z = .
u = 7*x^2 - 14*x*y + 9*y^2 + 8*x*z - 11*z^2 + x + y + z - 10;
Print["u(x, y, z) = ", u // TraditionalForm];
graphic1[a_, b_, c_] := u /. {x → a, y → b, z → c};
ContourPlot3D[graphic1[x, y, z] == 0, {x, -10, 10}, {y, -10, 10}, {z, -10, 10}]
```

u(x, y, z) = $7 x^2 - 14 x y + 8 x z + x + 9 y^2 + y - 11 z^2 + z - 10$

Out[44]=



In[45]:=
```
A := {{7, -7, 4},
     {-7, 9, 0},
    {4, 0, -11}};
B := {0.5, 0.5, 0.5};
X := {x, y, z};
a0 := -10;
Print["A = ", MatrixForm[A]];
Print["B = ", MatrixForm[B]];
Print["X = ", MatrixForm[X]];
Print["a0 = ", a0];
sol = LinearSolve[A, -B];
Print["x = ", sol[[1]], ", y = ", sol[[2]], ", z = ", sol[[3]]];
```

$$A = \begin{pmatrix} 7 & -7 & 4 \\ -7 & 9 & 0 \\ 4 & 0 & -11 \end{pmatrix}$$

$$B = \begin{pmatrix} 0.5 \\ 0.5 \\ 0.5 \end{pmatrix}$$

$$X = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$$

a0 = -10

x = -0.355705 , y = -0.332215 , z = -0.0838926

Поскольку решение системы ***Ax+B=0*** существует, то данная система является совместной, что приводит к 1-му случаю решения:

In[55]:=
```
x =.
a0New = sol.B + a0;
Print["new_a0 = ", a0New];
{eigenvals1, eigenvectors1} = Eigensystem[A];
N[eigenvals1]
```

new_a0 = -10.3859

Out[59]= {15.3383, -11.9624, 1.62413}

In[60]:=
```
AE := A - x*IdentityMatrix[3];
det := Det[AE];
eigenvals2 := Solve[det == 0, x];
N[eigenvals2]
```

Out[63]= {{x → -11.9624}, {x → 1.62413}, {x → 15.3383}}

In[64]:=

```
In[65]:= sortedEigenvals1 = Sort[eigenvals1];
sortedEigenvals2 = Sort[x /. eigenvals2];
sortedEigenvals1 == sortedEigenvals2
```

```
Out[67]= True
```

Как видно, собственные значения матрицы ***A***, найденные с помощью **Solve** и встроенной функции **Eigenvalues**, совпадают.

Далее необходимо найти собственные векторы матрицы ***A***, которые будут являться направляющими векторами для осей в новой системе координат:

```
In[68]:= x =.
y =.
z =.
lambdas = x /. eigenvals2;
eigenvectors2 = {};
For[i = 1, i ≤ Length[lambdas], ++i,
myD = AE /. x → lambdas[[i]];
myD = myD.X;
res = Solve[myD == {0, 0, 0}];
res2 = {y, z} /. res[[1]];
res2 = res2 /. x → 1;
res2 = Prepend[res2, 1];
eigenvectors2 = Append[eigenvectors2, res2]];
```

```
In[74]:= Print["Найденные собственные векторы: "];
For[i = 1, i ≤ Length[lambdas], ++i,
  Print["Для с.з. ", N[lambdas[[i]]], " : v = ", MatrixForm[N[eigenvectors2[[i]]]]]];

Print["Соственные векторы, найденные встроенным методом: "];
For[i = 1, i ≤ Length[eigenvals1], ++i,
  Print["Для с.з. ", N[eigenvals1[[i]]], " : v = ", MatrixForm[N[eigenvectors1[[i]]]]]];
```

Найденные собственные векторы :

Для с.з. -11.9624 : $v = \begin{pmatrix} 1. \\ 0.333931 \\ -4.15622 \end{pmatrix}$

Для с.з. 1.62413 : $v = \begin{pmatrix} 1. \\ 0.949041 \\ 0.316854 \end{pmatrix}$

Для с.з. 15.3383 : $v = \begin{pmatrix} 1. \\ -1.1044 \\ 0.15187 \end{pmatrix}$

Соственные векторы , найденные встроенным методом :

Для с.з. 15.3383 : $v = \begin{pmatrix} 6.58457 \\ -7.272 \\ 1. \end{pmatrix}$

Для с.з. -11.9624 : $v = \begin{pmatrix} -0.240603 \\ -0.0803448 \\ 1. \end{pmatrix}$

Для с.з. 1.62413 : $v = \begin{pmatrix} 3.15603 \\ 2.9952 \\ 1. \end{pmatrix}$

In[78]:=
```
newU = eigenvals1[[1]] x1^2 + eigenvals1[[2]] y1^2 + eigenvals1[[3]] z1^2 + a0New;
newU /= a0New;
newU  = newU // Simplify;
Print["Получено  каноническое  уравнение
    в  новой  системе  координат  u(x1, y1, z1) = ", newU];
```

Получено каноническое уравнение в новой системе координат $u(x1, y1, z1) = 1. - 1.47684\ x1^2 + 1.15179\ y1^2 - 0.156378\ z1^2$

Для построения графика функции необходимо найти каноническое уравнение относительно старой системы координаты. Для этого необходимо произвести сдвиг по каждой координате на значения найденного решения системы ***Ax+B=0*** которое является координатами вектора ***OO'*** перехода из старой системы координаты в новую.

In[82]:=
```
newUInOld := eigenvals1[[1]] * (x - sol[[1]])^2 +
   eigenvals1[[2]] * (y - sol[[2]])^2 + eigenvals1[[3]] * (z - sol[[3]])^2 + a0New;
newUInOld /= a0New;
Print["Получено  каноническое  уравнение
    в  старой  системе  координат  u(x, y, z) = ", N[newUInOld]];
```

Получено каноническое уравнение в старой системе координат $u(x, y, z) = -0.0962843 \left(-10.3859 + 15.3383\ (0.355705 + x)^2 - 11.9624\ (0.332215 + y)^2 + 1.62413\ (0.0838926 + z)^2\right)$

Полученный график поверхности:

In[85]:=
```
graphic2[a_, b_, c_] := newUInOld /. {x → a, y → b, z → c};
ContourPlot3D[graphic2[x, y, z] == 0, {x, -10, 10}, {y, -10, 10}, {z, -10, 10}]
```

Out[86]=