1. **Bisekcijos metodu rasti funkcijos f(x) minimumo tašką atkarpoje [a; b];**

a=-2;

b=0;

while(a~=b)

f =inline('x.^4-5\*x.^3+2\*x.^2+20\*x-24')

X = [a; (3\*a+b)/4; (a+b)/2; (a+3\*b)/4; b]

F=f(X)

k = find(F == min(F))

a=X(k-1)

b=X(k+1)

end

1. **Auksinio pjūvio metodu rasti funkcijos f(x) minimumo tašką atkarpoje [a; b];**

a=-2;

b=0;

while(a~=b)

f =inline('x.^4-5\*x.^3+2\*x.^2+20\*x-24')

alpha = (sqrt(5)-1)/2;

X = [a; a+(1-alpha)\*(b-a); a+alpha\*(b-a); b]

F=f(X)

k=find(F == min(F))

a=X(k-1)

b=X(k+1)

end

1. **Niutono metodu rasti funkcijos f(x) minimumo tašką nuo pradinio artinio x0 = b**

syms x

f = x^4-5\*x^3+2\*x^2+20\*x-24;

x=0;

f1=diff(f);

f2=diff(f,2);

i=1;

while abs(f1) > 0.00001

f1x = double(subs(f1))

f2x = double(subs(f2))

x = x - f1x/f2x

i=i+1

end

1. **a, b, c punkte gautame minimumo taške (taškuose) patikrinti būtiną ir pakankamą minimumo sąlyga.**

//butina salyga

x(min) = -0,9321

>> f = x^4-5\*x^3+2\*x^2+20\*x-24;

>> f1=diff(f)

f1 =

4\*x^3 - 15\*x^2 + 4\*x + 20

>> a=4\*(-0.9321)^3 - 15\*(-0.9321)^2 + 4\*(-0.9321) + 20

a = 1.7112e-004

//pakankama salyga

>> f2=diff(f,2)

f2 =

12\*x^2 - 30\*x + 4

>> b=12\*(-0.9321)^2 - 30\*(-0.9321) + 4

b = 42.3887

1. **Įvykdyti minimumo paiešką antigradientinio nusileidimo metodu funkcijai f(x, y) nuo pradinio taško (x0, y0) = (9, 9).**

clc; clear;

syms x y

format short

h = 0.001;

f=3\*x^3+x^2\*y+2\*x\*y^2+y^3-70\*x-80\*y;

fx=simplify(diff(f,x));

fy=simplify(diff(f,y));

gradientas=[fx, fy];

x\_pr = [9, 9];

X(1,:) = x\_pr;

for i = 2:400

gr = subs(gradientas,{x,y},x\_pr);

x\_pr = x\_pr-h\*gr;

duom =[i, gr, x\_pr]

X(i,:) = x\_pr;

if i>1 && abs(X(i,1)-X(i-1,1))<0.00001

break;

end;

end

1. **Įvykdyti minimumo paiešką greičiausio nusileidimo metodu funkcijai f(x, y) nuo pradinio taško (x0, y0)= (9, 9).**

clc; clear;

syms x y

format short

f=3\*x^3+x^2\*y+2\*x\*y^2+y^3-70\*x-80\*y;

fx=simplify(diff(f,x));

fy=simplify(diff(f,y));

gradientas=[fx, fy];

x\_pr = [9, 9];

X(1,:) = x\_pr;

for i = 2:400

gr = subs(gradientas,{x,y},x\_pr);

syms g

fg = expand(subs(f, {x, y}, x\_pr - g\*gr));

fgg = diff (fg, g);

h0 = double(solve(fgg));

h0 = real(min(h0(find(h0(find(abs(imag(h0))<0.000001))>0))));

x\_pr = x\_pr-h0\*gr;

duom =[i, gr, x\_pr, h0]

X(i,:) = x\_pr;

if i>1 && abs(X(i,1)-X(i-1,1))<0.00001

break;

end;

end

1. **Įvykdyti minimumo paiešką Niutono metodu funkcijai f(x, y) nuo pradinio taško (x0, y0) = (9,9);**

clc; clear;

syms x y

format short

f=3\*x^3+x^2\*y+2\*x\*y^2+y^3-70\*x-80\*y;

fx = simplify(diff(f, x));

fy = simplify(diff(f, y));

fxx = simplify(diff(fx, x));

fyy = simplify(diff(fy, y));

fxy = simplify(diff(fy, x));

gradientas = [fx, fy];

hesse = [fxx, fxy; fxy, fyy];

x\_pr = [9, 9];

X(1,:) = x\_pr;

for i = 2:400

gr = subs(gradientas,{x,y},x\_pr);

H1 = inv(subs(hesse),{x,y},x\_pr))

x\_pr = x\_pr - gr\*H1;

duom = [i, gr, x\_pr]

X(i,:) = x\_pr;

if i>1 && abs(X(i,1)-X(i-1,1))<0.0001

break;

end;

end

[x,y]=meshgrid(-10:.2:10);

z=3\*x^3+x^2\*y+2\*x\*y^2+y^3-70\*x-80\*y;

figure(1)

hold off

contour(x,y,z)

[dx,dy]=gradient(z,0.1,0.1);

hold on

quiver(x,y,dx,dy)

XX = X(:,1)'; YY=X(:,2)';

plot(XX,YY,'-ro')

**a, b, c punkte gautame minimumo taške (taškuose) patikrinti būtiną ir pakankamą minimumo sąlyga.**

//butina salyga

syms x y

f=3\*x^3+x^2\*y+2\*x\*y^2+y^3-70\*x-80\*y;

fx = simplify(diff(f, x));

fy = simplify(diff(f, y));

gradientas = [fx, fy];

x\_min = [1.5891, 4.1317];

gr = subs(gradientas, {x, y}, x\_min)

gr =1.0e-003 \* 0.4080 1.0e-003 \* 0.8114

//pakankama salyga

fxx = simplify(diff(fx, x));

fyy = simplify(diff(fy, y));

fxy = simplify(diff(fy, x));

hesse = [fxx, fxy; fxy, fyy];

H = subs(hesse, {x, y}, x\_min)

det(H)

H =

36.8672 19.7050

19.7050 31.1466

ans =

760.0009