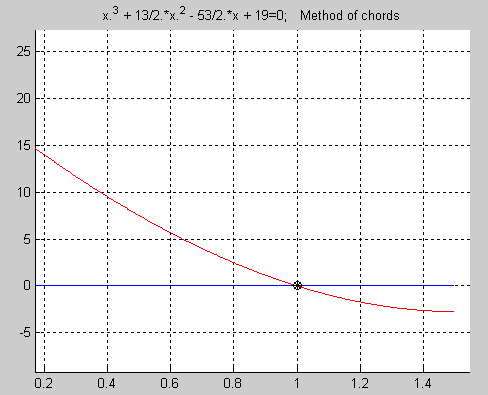
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **grupė** | **Pavardė vardas** | **Savarankiško darbo nr.** | **Lygčių nr.** | **Sprendimo metodas** | |
| **Daugianario**  **stygų** | **Lygčių sistemos LU skaidos** |
| IFE-0 | Juozapavičius Aidas | 1 | **16** | x3+13/2\*x2-53/2\*x+19 |  |

1. ***stygų metodas***



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| metodas | | lygtis | | | |  | | --- | | **Šaknų intervalo įverčiai** | | |
| stygų | | x3+13/2\*x2-53/2\*x+19 | | |  | |
| šaknis | Intervalas  arba artinys | | tikslumas | Iteracijų skaičius | Iteracijų pabaigos sąlygos | roots funkcijos rezultatas |
|  | 3 | | 3.69472e-010 | 20 | 1e-9 | 2 |

**Programos kodas:**

%

% Vienos lygties sprendimas: stygu metodas

%

function stygu\_metodas

clc,close all

%------------------------ PRADINIAI DUOMENYS ---------------------------

f='x.^3 + 13/2.\*x.^2 - 53/2.\*x + 19'

% range=[-19.5,-18.5];

% range=[-18.5,-17.5];

range=[0.5,1.5];

% range=[2.5,3.5];

eps=1e-9; % parenkame tikslumo reiksme

nitmax=100;% parenkame didziausia leistina iteraciju skaiciu

method='chords';

% braizomas funkcijos grafikas

npoints=1000; x=range(1): (range(2)-range(1))/(npoints-1) :range(2);

figure(1); grid on; hold on;

str=[f,'=0; Method of ',method]; title(str);

plot(x,eval(f),'r-');

plot(range,[0 0],'b-');

%------------------------ SPRENDIMAS -----------------------------------

xn=range(1);xn1=range(2);prec=1;

nit=0;

while prec > eps

nit=nit+1;

if nit > nitmax, fprintf('Virsytas leistinas iteraciju skaicius');break;end

plot(xn,0,'mp');h = findobj(gca,'Type','line');h1=h(1); % paskutinio grafinio objekto valdiklis irasomas handle masyvo priekyje

plot(xn1,0,'cp');h = findobj(gca,'Type','line');h2=h(1);

% pasirenkame, ar spresti pusiaukirtos, ar stygu metodu:

if strcmp(method,'chords') % stygu metodas

x=xn;fxn=eval(f);x=xn1;fxn1=eval(f);

k=abs(fxn/fxn1);xmid=(xn+k\*xn1)/(1+k);

plot(xmid,0,'gs');plot([xn,xn1],[fxn,fxn1],'g-');h = findobj(gca,'Type','line');h3=h(1:2);

else, fprintf('neaprasytas metodas \n');

end

x=xmid;fxmid=eval(f);

% jeigu pradzioje tikriname kairi taska

x=xn;fxn=eval(f);

if sign(fxmid) == sign(fxn), xn=xmid;

else, xn1=xmid;

end

pause(1)

delete(h1);delete(h2);delete(h3);

prec=abs(fxmid);

fprintf(1,'iteracija %d tikslumas= %g \n',nit,prec);

end

plot(xmid,0,'k\*');plot(xmid,0,'ko');

fprintf(1,'\n tikslumas pasiektas, saknis xmid=%g\n\n',xmid);

% ................................................................................

disp('.... Patikriname saknies reiksme, naudodami MATLAB funkcija fsolve: ....')

fprintf('\n')

% fsolve iejimo pirmu parametru gali buti simboliu masyvas(char), kuriame irasyta lygties funkcijos israiska.

% Lygties funkcijos argumentas yra simbolis x.

% Antras parametras yra x reiksme. x gali buti vektorius.

x0=-20; sprendinys = fsolve(char(f),x0); fprintf(1,'\npradinis artinys x0=%g, sprendinys=%g',x0,sprendinys);

x0=-17; sprendinys = fsolve(char(f),x0); fprintf(1,'\npradinis artinys x0=%g, sprendinys=%g',x0,sprendinys);

x0=1; sprendinys = fsolve(char(f),x0); fprintf(1,'\npradinis artinys x0=%g, sprendinys=%g',x0,sprendinys);

x0=3; sprendinys = fsolve(char(f),x0); fprintf(1,'\npradinis artinys x0=%g, sprendinys=%g',x0,sprendinys);

fprintf(1,'\n');

end

**2. LU skaidos metodas**

**Lygčių sistema:**

**Rezultatų vektorius**

b =

36.0000

-20.5000

71.0000

-4.0000

Buvo atlikti tiesioginis ir atvirkštinis žingsniai

**Gautas atsakymas**

b1 =

4.0000

-3.5000

6.0000

1.5000

**Gauta liekana (AX - B)**

liekana =

1.0e-014 \*

0

0

0

0.3553

**Patikriname su linsolve**

X =

4.0000

-3.5000

6.0000

1.5000

**Programos kodas:**

% LU skaida

function laoras\_15\_LU\_skaida

clc, close all, clear all

A=[6 0 1 4;

2 9 1 -2;

1 1 12 -1;

0 2 -3 14]

b=[36;-20.5;71;-4]

A1=A; b1=b;

n=size(A, 1)

P=[1:n]

%tiesioginis zingsnis

for i=1:n-1

[a, iii]=max(abs(A1(i:n,i)));

A1([i,iii+i-1], :)=A1([iii+i-1,i], :);

P([i,iii+i-1])=P([iii+i-1,i])

for j=i+1:n

r = A1(j, i)/A1(i, i);

A1(j,i+1:end)=A1(j,i+1:end)-A1(i,i+1:end)\*r;

A1(j,i)=r;

end

end

%atvirkstinis zingsnis

b1=b1(P);

for i=2:n

b1(i)=(b1(i)-A1(i, 1:i-1)\*b1(1:i-1));

end

for i=n:-1:1

b1(i)=(b1(i)-A1(i, i+1:n)\*b1(i+1:n))/A1(i,i);

end

b1

% tikslumo tikrinimas

liekana=A\*b1-b;

liekana

%tikrinimas su MatLab funkcija

X = linsolve(A,b)

end