%prvi zadatak

function rez = polinomi(P1,P2)

if nargin ~=[1 2]

error('Funkcija ima 1 ili 2 ulazna argumenta');

end

if nargout ~=[0 1]

error('Funkcija ima 0 ili 1 izlaz');

end

s1 = size(P1);

if nargin == 1

if s1(1)>1

error('Ja radim samo sa polinomima');

end

x = -8:0.4:8;

y = polyval(P1,x);

y = y + randn(size(y));

plot(x,y,'gs');

grid on; hold on;

p = polyfit(x,y,length(P1)-1);

plot(x,polyval(p,x),'r');

hold on;

rez = roots(P1);

rez = rez(find(imag(rez)==0));

plot(rez,0,'ko','MarkerSize',12);

title(['Aproksimacija polinomom ' num2str(length(P1)-1) '.reda']);

xlabel('x-osa');

ylabel('y-osa');

elseif nargin == 2

s2 = size(P2);

if s1(1)>1 || s2(1)>1

error('Ja radim samo sa polinomima');

end

proizvod = conv(P1,P2);

[kolicnik,ostatak] = deconv(P2,P1);

red1 = length(P1)-1;

red2 = length(P2)-1;

if red1 == red2

razlika = P1 - P2;

elseif red1 < red2

diff = red2-red1;

P1 = [zeros(1,diff) P1];

razlika = P2 - P1;

elseif red2 < red1

diff = red1-red2;

P2 = [zeros(1,diff) P2];

razlika = P1 - P2;

end

zbir = P1 + P2;

disp(['Zbir = ' num2str(zbir)]);

disp(['Razlika = ' num2str(razlika)]);

disp(['Proizvod = ' num2str(proizvod)]);

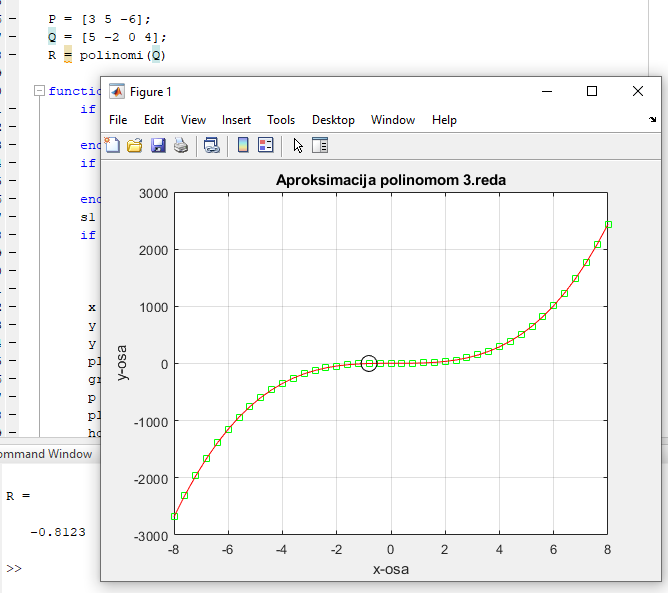
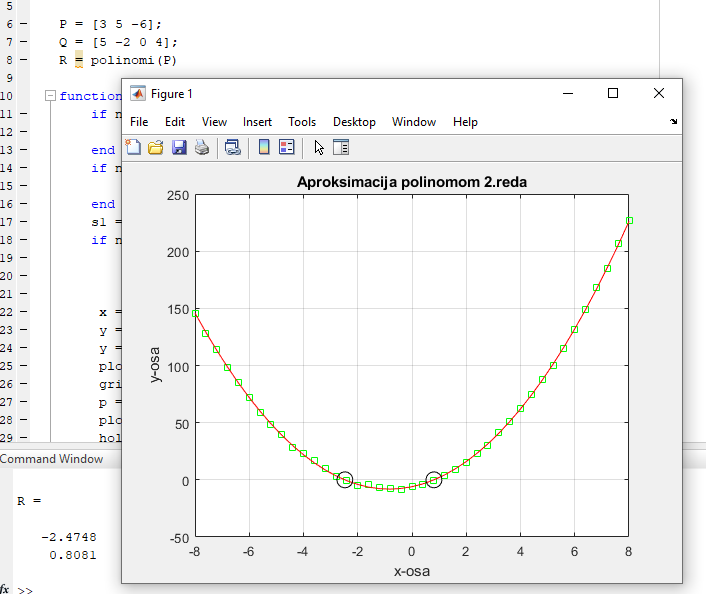
disp(['Kolicnik = ' num2str(kolicnik)]);

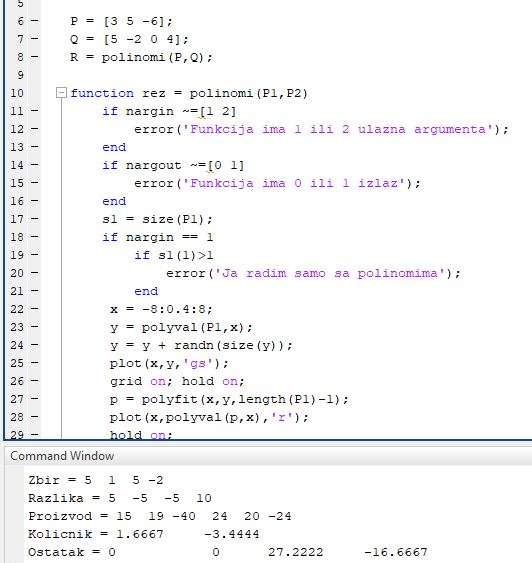
disp(['Ostatak = ' num2str(ostatak)]);

rez = NaN; %ovo sam stavio cisto da se kompajler ne bi zalio :)

end

end





%drugi zadatak

function math(znak)

if nargout~=0

error('Funkcija ne vraca nista');

end

if nargin~=1

error('Funkcija ima samo 1 ulazni argument');

end

if znak ~= ['a' 'b' 'c']

error('Pogresan unos');

end

if znak == 'a'

syms x;

brojnik = 2 + log(x);

nazivnik = 2 \* sqrt(x);

fun = brojnik/nazivnik;

f = int(fun,-4,4);

pretty(f);

y = dsolve('2\*D3y - 3\*Dy = sin(3\*x)','y(0)=1','Dy(0)=1','x');

pretty(y);

elseif znak == 'b'

syms x;

brojnik = log(x)+2;

nazivnik = sqrt(x);

f = brojnik/nazivnik;

fig = figure;

ezplot(f,[-1 10 -4 4],fig);

hold on;

nule = solve(brojnik);

plot(nule,0,'rs');

hold on;

ha = limit(f,inf);

va = solve(f==0);

limit(f,x,va,'right');

plot([-1 10],[ha ha],'g'); hold on;

plot([va va],[-4 4],'g'); hold on;

f1 = diff(f);

f1 = simplify(f1);

ef = solve(f1==0);

plot(double(ef),double(subs(f,ef)),'ko');

else

R = [110 2 0;

100 3 0;

90 1 2;

80 2 3;

70 3 4];

V = [12 1 0];

I = [2 0 4];

kolo.R = R;

kolo.V = V;

kolo.I = I;

kolo.simb = 0;

[G,B,C,D,A] = amatrica(kolo);

Z = zmatrica(kolo);

x = A\Z;

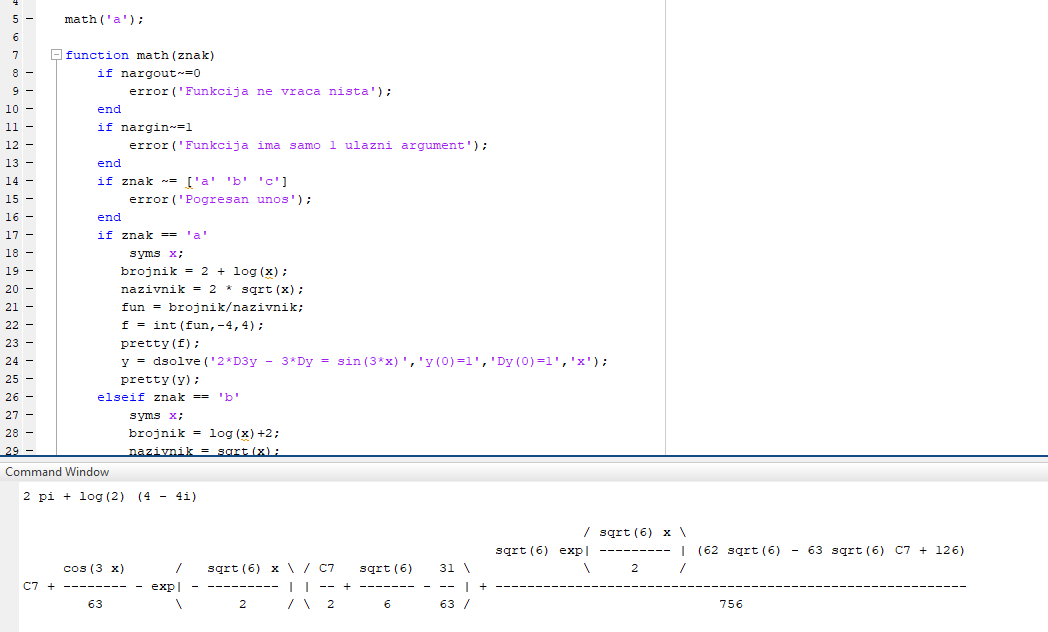
disp(['Vrijednost napona na krajevima R4 = '

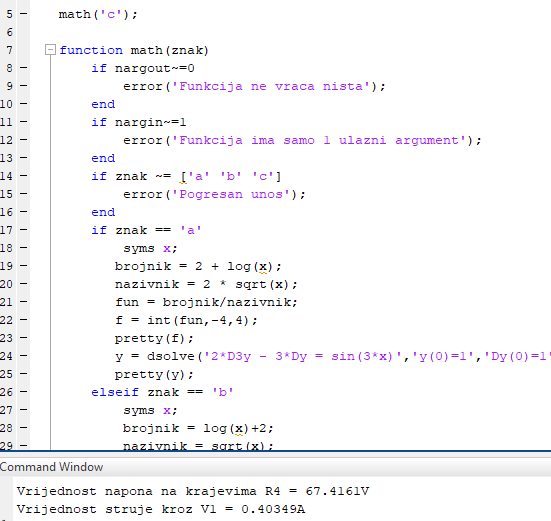
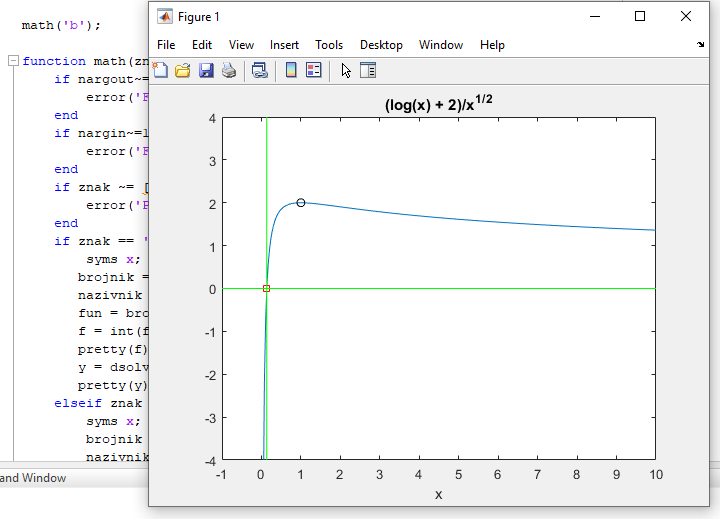
num2str(x(3)-x(2)) 'V']);

disp(['Vrijednost struje kroz V1 = ' num2str(x(5)) 'A']);

end

end





%treci zadatak

function rez = struktura(S)

if nargin ~= 1

error('Funkcija ima samo 1 ulazni argument');

end

if nargout ~= 1

error('Funkcija ima samo 1 izlazni argument');

end

if length(S)<5

error('Radim sa minimalno 5 tacaka');

end

minx = S(1).x;

maxx = S(1).x;

miny = S(1).y;

maxy = S(1).y;

for i = 1 : length(S)

if S(i).x < minx

minx = S(i).x;

elseif S(i).x > maxx

maxx = S(i).x;

end

if S(i).y < miny

miny = S(i).y;

elseif S(i).y > maxy

maxy = S(i).y;

end

end

fprintf("Gornja lijeva tacka: (%d,%d)\n",minx,maxy);

fprintf("Donja desna tacka: (%d,%d)\n",maxx,miny);

fprintf("Tacke u pravougaoniku: ");

for i = 1 : length(S)

if S(i).x < maxx && S(i).x > minx && S(i).y < maxy && S(i).y > miny

fprintf("%c",S(i).Oznaka);

end

end

fprintf("\nTacke na rubu pravougaonika: ");

for i = 1 : length(S)

if (S(i).x == minx && S(i).y ~= miny && S(i).y ~= maxy)...

|| (S(i).x == maxx && S(i).y ~= miny && S(i).y ~= maxy )...

|| (S(i).y == miny && S(i).x ~= minx && S(i).x ~= maxx)...

|| (S(i).y == maxy && S(i).x ~= minx && S(i).x ~= maxx)...

fprintf("%c",S(i).Oznaka);

end

end

fprintf("\nTacke tjemena pravougaonika: ");

for i = 1 : length(S)

if S(i).x == minx && (S(i).y == miny || S(i).y == maxy)...

|| S(i).x == maxx && (S(i).y == miny || S(i).y == maxy)

fprintf("%c",S(i).Oznaka);

end

end

fprintf("\n");

X = [minx maxx maxx minx minx];

Y = [miny miny maxy maxy miny];

plot(X,Y,'r--','LineWidth',2);

hold on; grid on;

axis([0 6 0 6]);

for i = 1:length(S)

plot(S(i).x,S(i).y,'bo');

hold on;

end

a = maxx - minx;

b = maxy - miny;

povrsina = a\*b;

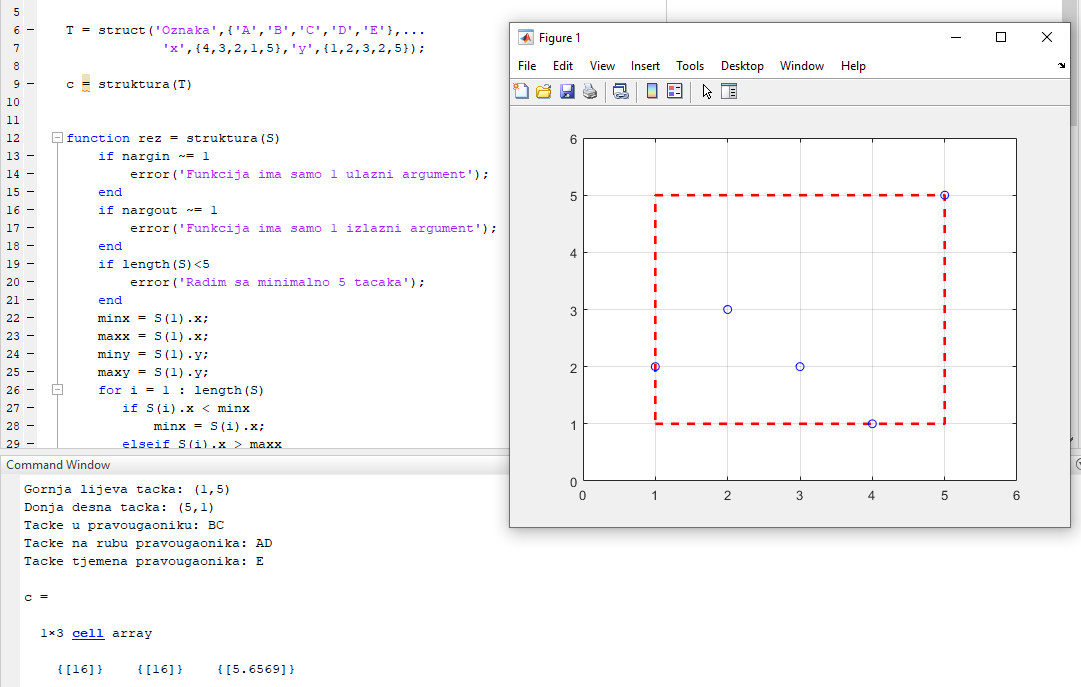
obim = 2\*a + 2\*b;

dijagonala = sqrt(a^2 + b^2);

rez = cell(1,3);

rez = {obim,povrsina,dijagonala};

end



clear all

close all

clc

%zadatak4

f1 = figure('Tag','figure1','menubar','none');

grafik = axes('Position',[0.1 0.1 0.8 0.7],'parent',f1);

izbor = uicontrol('Style','popup','Units','normalized',...

'Position',[0.74 0.82 0.17 0.05],'String',' |\*|+|o|x',...

'parent',f1,'Callback','plotting');

decor = uicontrol('Style','text','Units','normalized',...

'Position',[0.07 0.82 0.23 0.05],'String','Polinom i interval',...

'parent',f1,'FontSize',12);

naslov = uicontrol('Style','text','Units','normalized',...

'parent',f1,'FontSize',12,'String','APROKSIMACIJA PODATAKA SUMOM',...

'Position',[0.1 0.9 0.8 0.05]);

unos1 = uicontrol('Style','edit','Units','normalized',...

'Position',[0.3 0.82 0.2 0.05],'parent',f1);

unos2 = uicontrol('Style','edit','Units','normalized',...

'Position',[0.52 0.82 0.2 0.05],'parent',f1);

%plotting callback

try

P = eval(get(unos1,'string'));

x = eval(get(unos2,'string'));

catch

disp('Greska u unosu');

end

y = polyval(P,x);

y = y + randn(size(y));

mojIzbor = get(izbor,'value');

cla;

switch mojIzbor

case 1

cla;

case 2

plot(x,y,'b\*');

case 3

plot(x,y,'b+');

case 4

plot(x,y,'bo');

case 5

plot(x,y,'bx');

end

axis tight;

hold on; grid on;

P1 = polyfit(x,y,length(P)-1);

menus(x,polyval(P1,x));

%padajuci meni callback

function menus(x,y)

meni1 = uimenu('Label','Vrsta linije');

linija = plot(x,y,'Tag','linija');

set(linija,'Color',[1 0 0]);

cb1 = ['set(findobj(''Tag'',''linija''), ''LineStyle'', ''--'')'];

cb2 = ['set(findobj(''Tag'',''linija''), ''LineStyle'', '':'')'];

cb3 = ['set(findobj(''Tag'',''linija''), ''LineStyle'', ''-'')'];

opcija1 = uimenu(meni1,'Label','isprekidana','Callback',cb1);

opcija1 = uimenu(meni1,'Label','tackasta','Callback',cb2);

opcija1 = uimenu(meni1,'Label','puna','Callback',cb3,'separator','on');

end

