**Izbor iz zadataka iz Trodimenzione grafike**

1. Zadatak iz knjige Gilata 10.6.1: Čestica se kraće po zakonu:

Nacrtati položaj čestice za 0≤t≤30.

t = 0:0.01:30;

x = (4-0.1\*t).\*sin(0.8\*t);

y = (4-0.1\*t).\*cos(0.8\*t);

z = 0.4\*t.^(3/2);

plot3(x,y,z,'y','LineWidth',2)

grid on % opciono

2. Heliks (zavojnica) je kriva određena jednačinama: . Nacrtati heliks zelene boje, debljine linije 5pt, sa crta – tačka linijom i oznaciti ose.

t = linspace(0,12\*pi,200); %200 tacaka od 0 do 12\*pi

plot3(sin(t),cos(t),t,'g-.','linewidth',5)

xlabel('x=sin(t)'), ylabel('y=cos(t)'),

zlabel('z=t')

3. Na prethodno nacrtanom heliksu, koristeći naredbu subplot isprobati komandu view tako što ćete grafički prozor podeliti na 4 dela i u prvom prozoru crtati heliks, u drugom prozoru primeniti view(0,90), u trećem view(0,0) i u četvrtom view(90,0).

clf %clear figure window

t = linspace(0,12\*pi,200); %200 tacaka od 0 do 12\*pi

subplot(2,2,1)

plot3(sin(t),cos(t),t)

xlabel('x=sin(t)'), ylabel('y=cos(t)'),

zlabel('z=t')

grid on

title('Heliks i view komanda')

subplot(2,2,2)

plot3(sin(t),cos(t),t,'m'),view(0,90) % gledamo pod uglom (0,90) - vidimo xy

% ravan ili "pogled sa vrha"

xlabel('x=sin(t)'), ylabel('y=cos(t)'), zlabel('z=t')

title('Vidimo samo xy ravan')

subplot(2,2,3)

plot3(sin(t),cos(t),t,'y'),view(0,0)

xlabel('x=sin(t)'), ylabel('y=cos(t)'),

zlabel('z=t')

title('Vidimo samo xz ravan')

subplot(2,2,4)

plot3(sin(t),cos(t),t,'r'),view(90,0),grid

xlabel('x=sin(t)'), ylabel('y=cos(t)'),

zlabel('z=t')

title('Vidimo samo yz ravan')

4. Data je funkcija . Najpre izračunati vrednost funkcije z u svakoj tački rešetke, a zatim nacrtati grafik funkcije z pozivom mesh i surf naredbe.

x = -3:3;

y = -3:3;

z = @(x,y) x.^2/3+2\*sin(3\*y) % ako definisemo funkciju na ovaj nacin

z(x,y) % obratite paznju da ovako dobijamo samo vrednosti funkcije u tackama (-3,-3), (-2,-2),...(3,3)

[X,Y] = meshgrid(x,y)

Z = X.^2/3+2\*sin(3\*Y) % ovako dobijamo vrednost funkcije na celoj mrezi

% crtamo na osnovu postojece mreze

figure

mesh(X,Y,Z)

figure

surf(X,Y,Z)

% obratite paznju da nam je za grafike potrebna „finija“ mreza

x = -3:0.1:3;

y = -3:0.1:3;

[X,Y] = meshgrid(x,y);

Z = X.^2/3+2\*sin(3\*Y) ;

figure

mesh(X,Y,Z)

figure

surf(X,Y,Z)

5. Na grafiku funkcije isprobati sledeće komande: meshz(X,Y,Z), meshc(X,Y,Z), surfc(X,Y,Z), surfl(X,Y,Z), waterfall(X,Y,Z), shading flat, shading interp,...,contour3(X,Y,Z), contour(X,Y,Z,n).

x = -10:0.3:10;

y = -10:0.3:10;

[X,Y] = meshgrid(x,y);

R = sqrt(X.^2+Y.^2);

Z = sin(R)./R;

% isprobacemo razne opcije 3dim grafickih naredbi

% radi boljeg predleda crtacemo sa subplot

clf; % obrisemo graficki prozor

subplot(2,1,1)

meshz(X,Y,Z)

title('Naredba meshz')

xlabel('x'), ylabel('y'), zlabel('z')

subplot(2,1,2)

meshc(X,Y,Z)

title('Naredba meshc - dodaje konture')

xlabel('x'), ylabel('y'), zlabel('z')

figure % otvorimo novi graficki prozor

subplot(2,1,1)

surfc(X,Y,Z)

title('Naredba surfc - surf + contour')

xlabel('x'), ylabel('y'), zlabel('z')

subplot(2,1,2)

surfl(X,Y,Z)

title('Naredba surfl - surf + light')

xlabel('x'), ylabel('y'), zlabel('z')

% probati ovde i shading flat, zatim shading interp

figure % otvorimo jos jedan graficki prozor

subplot(2,1,1)

waterfall(X,Y,Z)

title('Naredba waterfall')

xlabel('x'), ylabel('y'), zlabel('z')

subplot(2,1,2)

contour3(X,Y,Z) % mozete probati i contour3(X,Y,Z,15)

title('Naredba contour')

xlabel('x'), ylabel('y'), zlabel('z')

6. zadatak 10.6.11 iz knjige Gilata

close all

clc

T = linspace(0,50,1000); % temperatura (F)

v = linspace(0,70,1000); % brzina vetra (mi/h)

[T1,v1] = meshgrid(T, v);

Twc = 35.74 + 0.6215\*T1-35.75\*v1.^(0.16)+0.4275\*T1.\*v1.^(0.16);

mesh(T1,v1,Twc)

xlabel('T'), ylabel('v');

7. % zadatak 10.6.20

K = 300; %ksi\*sqrt(in), dato u zadatku

theta = linspace(0,pi/2,1000);

r = linspace(0.02,0.2,1000);

[theta,r] = meshgrid(theta,r);

sigmaxx = K./sqrt(2\*pi\*r).\*cos(theta/2).\*(1-sin(theta/2).\*sin(3\*theta/2));

sigmayy = K./sqrt(2\*pi\*r).\*cos(theta/2).\*(1+sin(theta/2).\*sin(3\*theta/2));

tauxy = K./sqrt(2\*pi\*r).\*cos(theta/2).\*sin(theta/2).\*cos(3\*theta/2);

subplot(3,1,1)

% akcenat u ovom zadatku jeste da pre crtanja povrsi

% moramo polarne koordinate prebaciti u Dekartove

[x,y,z] = pol2cart(theta,r,sigmaxx);

mesh(x,y,z)

shading interp

xlabel('x (in)');

ylabel('y (in)');

zlabel('sigmaxx (ksi)');

axis([0 0.2 0 0.2 0 500])

subplot(3,1,2)

[x,y,z] = pol2cart(theta,r,sigmayy);

mesh(x,y,z)

shading interp

xlabel('x (in)');

ylabel('y (in)');

zlabel('sigmayy (ksi)');

axis([0 0.2 0 0.2 0 500])

subplot(3,1,3)

[x,y,z] = pol2cart(theta,r,tauxy);

mesh(x,y,z)

shading interp

xlabel('x (in)');

ylabel('y (in)');

zlabel('tauxy (ksi)');

axis([0 0.2 0 0.2 -400 100])

8. Kriva odredjena grafikom funkcije f(x)=x^3-6\*x^2+8\*x za x izmedju 1 i 4 rotira oko x ose. Nacrtati površ koja nastaje tom rotacijom i izračunati zapreminu tako dobijenog tela.

clear all

clc; clf;

F = @(x)x.^3-6\*x.^2+3\*x; % funkcija koju rotiramo

t = linspace(0,2\*pi,60);

x = linspace(1,4,60);

[T,X] = meshgrid(t,x);

Y = F(X).\*cos(T);

Z = F(X).\*sin(T);

surfl(X,Y,Z)

hold on

% docrtajmo grafik funkcije cijom rotacijom smo

% dobili povrs

y = F(x);

z = 0\*x;

plot3(x,y,z,'g', 'LineWidth',4)

% oznacimo i ose radi preglednosti

xlabel('x')

ylabel('y')

zlabel('z')

% docrtajmo i xy ravan zbog bolje preglednosti

x1 = linspace(1,4,30);

y1 = linspace(-40,40,30); % opseg vidimo sa grafika

[X1,Y1] = meshgrid(x1,y1);

Z1=0\*X1;

surf(X1,Y1,Z1,'FaceAlpha',0.2)

zapremina = pi\*quad(@(x)(x.^3-6\*x.^2+8\*x).^2,1,4)

9. Kriva x=1+cos(z) za 0<=y<=2\*pi rotira oko z ose. Nacrtati obrtnu površ koja nastaje na taj način.

clc;clf

F = @(z)1+cos(z);

t = linspace(0,2\*pi,60);

z = linspace(0,2\*pi,60);

[T,Z] = meshgrid(t,z);

X = F(Z).\*cos(T);

Y = F(Z).\*sin(T);

surf(X,Y,Z)

% docrtajmo krivu cijom rotacijom nastaje povrs

hold on

x = 1 + cos(t);

y = 0.\*t;

z = t;

plot3(x,y,z,'m','linewidth',3)

axis equal square

10. Nacrtati elipticki paraboloid z = 100-3\*x.^2-4\*y.^2 za -6<=x<=6 i -6<=y<=6 na dva načina. Najpre kao standardni trodimenzioni grafikon u Matlabu, a zatim koristeći činjenicu da je reč o obrtnoj površi, rotirajući parabolu z = 100-3\*x^2 oko y ose.

close all

x = -6:0.5:6;

y = -6:0.5:6;

[X,Y] = meshgrid(x,y);

Z = 100-3\*X.^2-4\*Y.^2; % paraboloid

surf(x,Y,Z,'FaceAlpha',0.5)

% elipticki paraboloid je rotaciona povrs, sto znaci da nastaje

% rotacijom recimo parabole z=100-3x^2 u xz ravni (y=0)

% iskoristimo to da ga nacrtamo na drugi nacin

u = linspace(0,6,70);

v = linspace(0,2\*pi,70);

[U,V] = meshgrid(u,v);

X2 = U.\*cos(V);

Y2 = U.\*sin(V);

Z2 = 100-3\*X2.^2-4\*Y2.^2;

figure

surf(X2,Y2,Z2,'FaceAlpha',0.4)

% primetite da smo ovako dobili povrs koja vise lici na

% paraboloid, zbog toga sto je ovde nacrtan kao

% rotaciona povrs, sto sustinski jeste nacin kako nastaje

pause

% docrtajmo generatrisu - parabolu z=100-3x^2

% cijom rotacijom nastaje paraboloid

hold on

t=linspace(-6,6,70);

X3 = t;

Y3 = 0\*t; % da bi vektori X3 i Y3 bili iste duzine

% ako samo otkucate Y3=0, nece raditi

% alternativa je Y3=zeros(70)

Z3 = 100-3\*t.^2;

plot3(X3,Y3,Z3,'r','LineWidth',4)

% oznacimo ose

xlabel('x osa')

ylabel('y osa')

zlabel('z osa')

% da bismo naglasili da je presek sa xy ravni, dodajmo

% na crtez i tu ravan

x4 = linspace(-6,6,10);

z4= linspace(-50,100,10);

% pokusajte sa 40 i sa 100 umesto 10 tacaka

[X4,Z4] = meshgrid(x4,z4);

Y4 = 0\*X4+0\*Z4; % ravan y=0

surf(X4,Y4,Z4,'FaceAlpha',0.2)

11. Nacrtati paraboloid z = 144 –x^2-y^2 polazeci od parabole koja je presek paraboloida sa xz ravni (y=0) i rotirajuci je oko z ose u for petlji.

clear all

clc

t = linspace(-12,12,60);

v =0;

X = t\*cosd(v);

Y = t\*sind(v);

% ovako crtamo parametarske jednacine kruga koji je presek paraboloida i ravni z=const

Z = 144-t.^2;

plot3(X,Y,Z,'r','LineWidth',4)

xlabel('x')

ylabel('y')

zlabel('z')

axis([-12 12 -12 12 0 144])

grid on

pause

hold on

for v = 1:180

plot3(t\*cosd(v),t\*sind(v),144-t.^2);

axis([-12 12 -12 12 0 144])

pause(1/50)

end

hold off