

GRAFIKA

2-D crteži

Elementarni X-Y grafovi

plot	- linearni crtež
loglog	- logaritamsko-logaritamski crtež
semilogx	- linearno-logaritamski crtež
semilogy	- logaritamsko-linearni crtež
polar	- crtež u polarnim koordinatama
plotyy	- graf sa y oznakama na lijevo i na desno

15.1

```
x=0:pi/100:2*pi;
y1=sin(x);
y2=cos(x);
plot(x,y1,'b-',x,y2,'r:');
```

15.2

```
theta=0:pi/100:2*pi;
rho=sin(theta*3);
polar(theta,rho,'go');
```

15.3

```
x=1:0.1:7;
y1=exp(x);
y2=exp(-x.^2);
plotyy(x,y1,x,y2,'semilogy');
grid on
```

15.4

```
h=input('korak h=');
x=0:h:pi;
y=cos(x).*(x<=pi/2)+(-
x+pi/2).*(x>pi/2);
plot(x,y)
```

Koordinate osi

axis	- kontrolira umjeravanje osi i njihovo pojavljivanje
zoom	- zoom-ira (uvećavanje i smanjivanje) 2-D crteža
grid	- linije mreže
box	- okvir na osima
hold	- drži (zamrzne/otpusti on/off) trenutni graf
axes	- tvori osi u bilo kojim pozicijama
subplot	- tvori osi u (n x m) podgrafovima

15.5

```
x=0:pi/100:2*pi;
y=sin(x);

subplot(2,1,1);
plot(x,y,'r');
axis([0 2*pi -0.7 0.7])

subplot(2,1,2)
plot(x,y,'b');
axis off

axes('Position',[0.6 0.2 0.3 0.2])
plot(x,y,'g')
grid on
```

Označivanje grafova

legend - legenda grafa
 title - naslov grafa
 xlabel - oznaka (opis) X-osi
 ylabel - oznaka (opis) Y-axis
 text - oznaka teksta
 gtext - postavljanje teksta mišem

15.6

```

clf;
x=0:pi/100:2*pi;
y1=sin(x);
y2=cos(x);
plot(x,y1,'b',x,y2,'r');
legend('Sinus','Kosinus',0);
xlabel('X-axis');
ylabel('Y-axis');
title('Trigonometrija je OK');
text(4, 0.5,'Tekst na (4,0.5)');

```

Ispis grafike.

print - ispisuje graf ili sprema u m-datoteku.
 printopt - printer-ske opcije
 orient - postavljanje smjera papira

print naredba može ispisivati grafiku u različitim Postscript formatima, uključujući EPS. Da bi se stvorila slika *figure.eps*, treba otipkati **print -deps figure**.

Rasterske slike (npr. za Web) se još djelotvornije spremaju koristeći **imwrite** naredbu.

3-D crteži*Elementarni 3-D crteži*

plot3 - crta linije i točke u 3-D prostoru
 mesh - 3-D mesh (rešetkasta) površina
 surf - 3-D obojena površina
 fill3 - 3-D površina prekrivena poligonima

15.7nice.m

```

function z=nice(x,y)
z=x .* exp(-x.^2 - y.^2);

% --- poziv funkcije 'nice.m'
x=[-2:0.5:2];
y=[-2:2];
[X,Y]=meshgrid(x,y)
Z=nice(X,Y)

```

```

plot3(X,Y,Z,'*'), grid
mesh(X,Y,Z)
meshc(X,Y,Z)
meshz(X,Y,Z)
waterfall(X,Y,Z), view([10 35])

```

15.8

```

t = 0:pi/50:5*pi;
plot3(sin(t),cos(t),t);
grid on

```

15.9

```
[X,Y] = meshgrid(-2:.4:2, -
2:.4:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.^2 - Y.^2);
mesh(X,Y,Z)
```

15.10

```
[X,Y] = meshgrid(-2:.4:2, -
2:.4:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.^2 - Y.^2);
surf(X,Y,Z)
```

Boje

colormap - Tablica boja
 caxis - Pseudocolor umjeravanje (scaling) osi
 shading - mod osjenčavanja bojom
 hidden - brisanje skrivenih linija u rešetki
 brighten - osvjetljavanje ili zatamnjenje tablice boja

Mape boja

hsv - Hue-saturation-value color map.
 hot - Black-red-yellow-white color map.
 gray - Linear gray-scale color map.
 bone - Gray-scale with tinge of blue color map.
 copper - Linear copper-tone color map.
 pink - Pastel shades of pink color map.
 white - All white color map.
 flag - Alternating red, white, blue, and black color map.
 lines - Color map with the line colors.
 colorcube - Enhanced color-cube color map.
 jet - Variant of HSV.
 prism - Prism color map.
 cool - Shades of cyan and magenta color map.
 autumn - Shades of red and yellow color map.
 spring - Shades of magenta and yellow color map.
 winter - Shades of blue and green color map.
 summer - Shades of green and yellow color map.

15.11

```
[X,Y] = meshgrid(-2:.2:2, -
2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.^2 - Y.^2);
surf(X,Y,Z)
colormap(hot(15))
shading interp
```

15.12

```
[X,Y] = meshgrid(-2:.2:2, -
2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.^2 - Y.^2);
surf(X,Y,Z)
colormap(hot(15))
shading interp
caxis([0.05 0.2])
```

Osvjetljenje

surfl - 3-D osjenčana površina s osvjetljenjem
 lighting - modovi osvjetljavanja
 material - mod odbijajućih materijala
 specular - Specular odbijanje svjetla
 diffuse - difuzno odbijanje svjetla
 surfnorm - normale površine

15.13

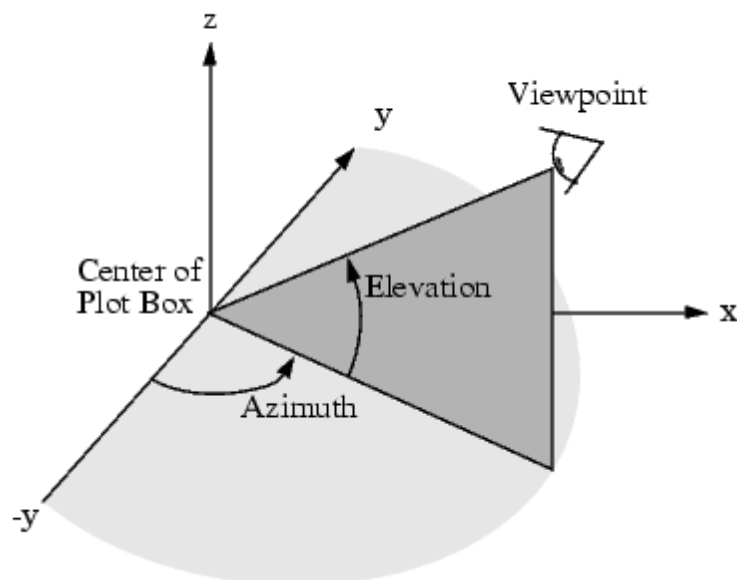
```
[X,Y] = meshgrid(-2:.2:2, -
2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.^2 - Y.^2);
surf(X,Y,Z)
colormap(jet)
```

15.14

```
[X,Y] = meshgrid(-2:.2:2, -
2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.^2 - Y.^2);
surf(X,Y,Z)
colormap(jet)
light
```

15.15

```
[X,Y] = meshgrid(-2:.2:2, -
2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.^2 - Y.^2);
surf(X,Y,Z)
colormap(jet)
f = findobj('Type','surface');
set(f,'FaceLighting','phong');
material shiny
shading interp
light
```

**Upravljanje pomaka točke gledišta**

- view - 3-D specifikacija točke gledišta
- viewmtx - Transformacijska matrica točke gledišta
- rotate3d - Interaktivno rotiranje točke gledišta u 3-D crtežu

15.16

```
[X,Y] = meshgrid(-2:.2:2, -
2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.^2 - Y.^2);
colormap(hsv)
subplot(1,2,1)
```

```
surf(X,Y,Z)
az=30;el=10;
view([az,el]);

subplot(1,2,2)
surf(X,Y,Z)
az=120;el=60;
view([az,el]);
```

15.17

```
Z=peaks(20);
colormap(winter)
surf(Z)
colorbar
```

15.18

```
Z=peaks(20);
colormap(winter)
surf(Z);
h=colorbar('horiz');
set(h, 'PlotBoxAspectRatio',[10
0.4 1.0])
set(h, 'FontSize', 15);
```

Specializirani grafovi

Specijalizirani 2-D grafovi.

area	- crtež ispunjene površine
bar	- Bar graf
barh	- horizontalni bar graf
bar3	- 3-D bar graf
bar3h	- horizontal 3-D bar graf
comet	- kometu slična trajektorija
errorbar	- 'error bar' crtež
ezplot	- jednostavno funkcijsko crtalo
feather	- feather crtež
fill	- ispunjeni 2-D poligoni
fplot	- function plot
hist	- histogram.
pareto	- pareto crtež
pie	- kriška (Pie) crtež
pie3	- 3-D kriška crtež
plotmatrix	- Scatter plot matrix.
ribbon	- crta 2-D linije kao vrpce u 3-D.
stem	- disketni niz ili "stem" crtež
stairs	- stepeničasti crtež

15.19

```
X=93:1:98;
Y=[1 2 3
   4 5 4
   2 3 4
   3 6 5
   1 2 4
   2 3 2 ];
bar3(X,Y)
```

15.20

```
yn=randn(10000,1);
x=min(yn):0.2:max(yn);
hist(yn,x);
```

15.21

```
x = 1:10;
y = 100*rand(10,1);
e = sqrt(y);
errorbar(x,y,e)
```

15.22

```
ezplot('besselj(0,x)')
```

15.23

```
fplot('erf(x)', [-pi pi])
```

Konturni i 2-1/2 D grafovi

contour	- konturni crtež
contourf	- ispunjeni konturni crtež
contour3	- 3-D konturni crtež.

clabel - konturni crtež s oznakama visina
 pcolor - Pseudocolor (šahovnica, checkerboard) crtež
 quiver - quiver crtež
 voronoi - Voronoi dijagrami

15.24

```
[X,Y] = meshgrid(-2:.2:2, -
2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.*X - Y.*Y);
[fx,fy] = gradient(Z);
contour(X,Y,Z,10)
hold on
quiver(X,Y,fx,fy);
hold off
```

15.25

```
[X,Y] = meshgrid(-2:.2:2, -
2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.*X - Y.*Y);
[fx,fy] = gradient(Z);
[cs,h]=contourf(X,Y,Z,10);
clabel(cs,h);
```

15.26

```
[X,Y] = meshgrid(-2:.2:2, -
2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.*X - Y.*Y);
contour3(X,Y,Z,20);
```

Specializirani 3-D crteži

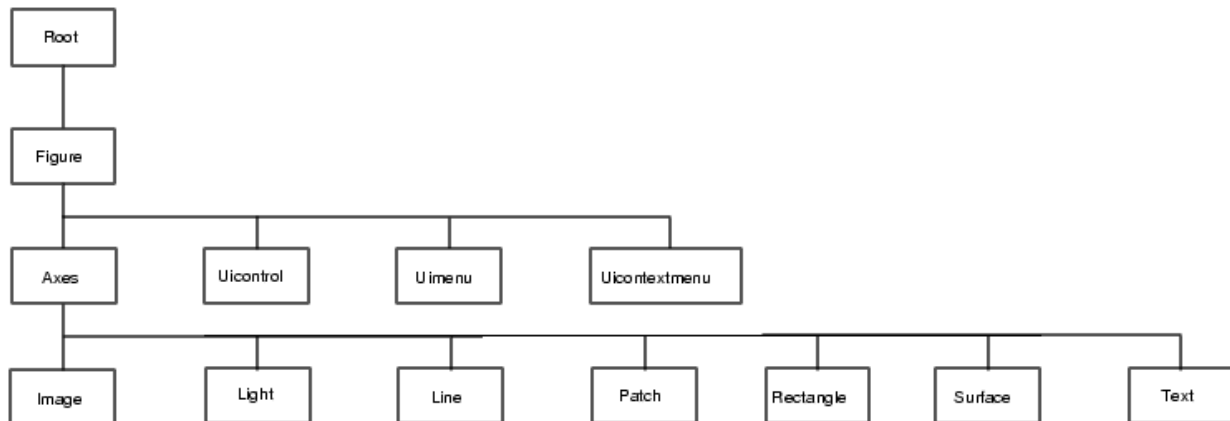
comet3 - 3-D comet-like trajectories.
 meshc - Combination mesh/contour plot.
 meshz - 3-D mesh with curtain.
 stem3 - 3-D stem plot.
 quiver3 - 3-D quiver plot.
 slice - Volumetric slice plot.
 surfc - Combination surf/contour plot.
 trisurf - Triangular surface plot.
 trimesh - Triangular mesh plot.
 waterfall - Waterfall plot.

15.27

```
[x,y,z] = meshgrid(-2:.2:2, -2:.25:2, -2:.16:2);
v = x .* exp(-x.^2 - y.^2 - z.^2);
slice(x,y,z,v,1.5, 1, 0)
```

15.28

```
[X,Y] = meshgrid(-2:.2:2, -2:.2:2);
Z = sin(X).* sin(Y).* ...
    exp(-X.*X - Y.*Y);
waterfall(X,Y,Z);
```

HANDLE GRAPHICS (grafika najniže razine)

OBJEKTI = atributi + vrijednosti
GET - dohvaća vrijednost(i) atributa
SET - postavlja vrijednosti atributa

```
plot(1:10,'o-')
```

```
h=findobj
```

```
h =
     0
    1.0000
  100.0011
    3.0012
```

```
get(h,'type')
```

```
ans =
    'root'
    'figure'
    'axes'
    'line'
```

```
set(h(3))
```

```
ALim
ALimMode: [ {auto} | manual ]
AmbientLightColor
Box: [ on | {off} ]
CameraPosition
CameraPositionMode: [ {auto} | manual ]
.....
```

```
UserData
Visible: [ {on} | off ]
```

```
set(h(4))
```

```
Color
EraseMode: [ {normal} | background | xor | none ]
LineStyle: [ {-} | -- | : | -. | none ]
LineWidth
```

```

    Marker: [ + | o | * | . | x | square | diamond | v | ^ | > | < |
pentagram | hexagram | {none} ]
    MarkerSize
    ....
    UserData
    Visible: [ {on} | off ]

```

```

set(h(4), 'Marker')
[ + | o | * | . | x | square | diamond | v | ^ | > | < | pentagram |
hexagram | {none} ]

```

```

set(h(4), 'Marker', 's', 'MarkerSize', 16)
set(h(3), 'XScale')
[ {linear} | log ]
set(h(3), 'XScale', 'log')

```

```

x=linspace(0,2*pi,35);
a1=subplot(2,1,1); % Axes object
l1=plot(x,sin(x),'x'); %Line object
a2=subplot(2,1,2); % Axes object
l2=plot(x,cos(x).*sin(x),'x'); %Line object
tx2=xlabel('x');ty2=ylabel('y'); %Text objects

```

```

set(a1, 'Box', 'off') % box off
set(a1, 'XTick', [])
set(a1, 'YAxisLocation', 'right')
set(a1, 'TickDir', 'out')
set(l1, 'Marker', '<')
set(a2, 'Position', [0.2 0.15 0.65 0.35])
set(a2, 'XLim', [0 2*pi])
set(a2, 'FontSize', 14)
set(a2, 'XTick', [0 pi/2 pi 2*pi])
set(a2, 'XTickLabel', '0 |pi/2| pi| 2*pi')
set(a2, 'XGrid', 'on')
set(a2, 'XScale', 'log')
set(a2, 'XScale', 'log')

```

```

set(l2, 'LineWidth', 6)
set(tx2, 'FontAngle', 'italic', 'FontSize', 20)
set(ty2, 'Rotation', 0, 'FontAngle', 'italic', 'FontSize', 20)
get(l1)
    Color = [0 0 1]
    EraseMode = normal
    LineStyle = none
    LineWidth = [0.5]
    Marker = <
    MarkerSize = [6]
    ....
    Type = line
    UIContextMenu = []
    UserData = []
    Visible = on

```

```

get(a2,'XTick')
ans =
    0    1.5708    3.1416    6.2832

get(gca,'FontWeight')
ans =
normal

set(gca,'FontWeight')
[ light | {normal} | demi | bold ]
set(gca,'FontWeight','bold')
set(gco,'Marker','*')

```

Animacija

Ugrađena funkcija - comet()

```

x=linspace(-2,2,500);
y=exp(x) .* sin(1./x);
comet(x,y)

```

drugi način animacije - slikom na sliku

```

clear % Remove existing variables
Z=peaks; surf(Z)
axis tight
set(gca,'nextplot','replacechildren')
disp('Creating the movie...')
for j=1:11
    surf(cos(2*pi*(j-1)/10) .* Z,Z)
    F(j)=getframe;
end
disp('Playing the movie...')
movie(F)

```

treći način - crtanje točke po točku

```

x=linspace(-pi,pi,2000);
y=cos(tan(x)) - tan(sin(x));
p=plot(x(1),y(1),'.','EraseMode','none','MarkerSize',5);
axis([min(x) max(x) min(y) max(y)])
hold on
for i=2:length(x)
    set(p,'Xdata',x(i),'Ydata',y(i))
    drawnow
end
hold off

```

GUI - Graphical User Interface (korisničko grafičko sučelje)

```
>> h=waitbar(0,'Computing...')
```

najčešće u programima:

```
>> for j=1:n
    %Some computation ...
    waitbar(j/n)
end
>> close(h)
```

Primjer:

```
set(gcf,'DefaultUIControlUnits','Normalized')

frame1_=uicontrol(gcf,'Style','Frame','Position',[0.1 0.1 0.8 0.3]);

frame2_=uicontrol(gcf,'Style','Frame','Position',[0.1 0.6 0.8 0.3]);
set(frame1_,'BackgroundColor',[0.5 0.5 0.5]);

set(frame2_,'BackgroundColor',[0.5 0.5 0.5]);

text_f_=uicontrol(gcf,'Style','Text','String',
'Fahrenheit:','Position',[0.3 0.7 0.2 0.05],...
'HorizontalAlignment','Left');

edit_f_=uicontrol(gcf,'Style','Edit','String','68.0','Position',[0.6 0.7
0.1 0.05],...
'HorizontalAlignment','right','Callback','fc_calc');

text_c1_=uicontrol(gcf,'Style','Text','String',
'Celsius:','Position',[0.3 0.3 0.2 0.05],...
'HorizontalAlignment','Left');

text_c2_=uicontrol(gcf,'Style','Text','String','20.0','Position',[0.6
0.3 0.1 0.05],...
'HorizontalAlignment','Right');

slider_f=uicontrol(gcf,'Style','Slider','Min',32.0,'Max',212.0,'Value',68
.0,'Position',...
[0.6 0.8 0.2 0.05], 'Callback','fc_slider; fc_calc');

>> guide % program za modeliranje grafičkog okruženja Matlab programa
```

D O D A T A K

ULAZ/IZLAZ

a) spremanje varijabli

SAVE - sprema varijable iz radnog prostora (workspace) na disk.

SAVE IME sprema sve varijable u binarnu "MAT-datoteku" nazvanu IME.mat.

Samo SAVE stvara datoteku 'matlab.mat'.

SAVE IME X - sprema samo X.

SAVE IME X Y Z - sprema X, Y, i Z.

Oznaka za sve '*' može se koristiti za sva imena koja zadovoljavaju uvjet.

LOAD - puni (load) varijable radnog prostora s podacima spremljenim naredbom SAVE.

LOAD IME vraća sve varijable koje su spremljene u datoteci IME.mat.

U slučaju da IME nije specificirano otvara se datoteka s imenom 'matlab.mat'.

b) formatirano čitanje i spremanje podataka

FSCANF - čita formatirane podatke iz datoteke.

[A,COUNT] = FSCANF(FID,FORMAT,SIZE) čita podatke iz datoteke specificirane identifikatorom datoteke FID, pretvara te podatke s obzirom na specificirani string FORMAT, i vraća tako obrađene podatke u matricu A. COUNT jedozvoljeni (ali ne nužni) argument koji vraća broj elemenata koji su uspješno pročitani. FID je cijeli broj dobiven iz naredbe FOPEN koja otvara datoteku. SIZE je također dozvoljen, ali ne nužan argument, koji stavlja ograničenje na broj elemenata koji se čitaju iz datoteke. Ako nije specificiran onda se čita cijela datoteka, a ako se specificira, onda je moguće:

N čita najviše N elementa u stupčasti vektor

inf čita najviše do konca datoteke

[M,N] čita najviše M * N elemenata koji popunjavaju MxN matricu u

stupčastom redoslijedu (stupac po stupac). N može biti inf, ali ne može M.

Primjeri:

S = fscanf(fid,'%s') čita (i vraća) niz znakova (string)

A = fscanf(fid,'%5d') čita 5-znamenasti decimalni cijeli broj

FPRINTF - upisuje formatirane podatke u datoteku.

COUNT = FPRINTF(FID,FORMAT,A,...) formatira podatke u realnom dijelu matrice A (i bilo kojeg dodatnog matričnog argumenta), pod kontrolom zadanog Format stringa, i upisuje podatke u datoteku specificiranu identifikatorom FID. COUNT je broj okteta (byte-ova) koji su uspješno upisani u datoteku.

FID je cijeli broj dobiven kao identifikator u FOPEN naredbi. Može također biti 1 za standardni izlaz (zaslon) ili 2 za standardnu pogrešku. Ako je FID izostavljen, onda izlaz ide na zaslon.

FORMAT je niz znakova (string) koji sadrži konverzijske specifikacije iz C programskog jezika. One uključuju znak %, dozvoljene zastavice (flags), bilo kakvu širinu (width) i polja preciznosti, proizvoljni podtip i konverzijske znakove d, i, o, u, x, X, f, e, E, g, G, c, kao i znak s.

Specijalni formati \n, \r, \t, \b, \f mogu se iskoristiti za tvorbu linefeed, carriage return, tab, backspace, i formfeed znakova.

Koristite \ for tvorbu znaka \ i %% za tvorbu znaka postotka %.

Primjer:

```
x = 0:.1:1; y = [x; exp(x)];
fid = fopen('exp.txt','w');
fprintf(fid,'%6.2f %12.8f\n',y);
fclose(fid);
```

načinit će tekst datoteku koja će sadržavati skraćenu tablicu eksponencijalne funkcije:

```
0.00    1.000000000
0.10    1.10517092
...
1.00    2.71828183
```

TEXTREAD - čita formatirane podatke iz tekst datoteke

```
A = TEXTREAD('IME')
A = TEXTREAD('IME',' ',N)
A = TEXTREAD('IME',' ',param,value, ...)
A = TEXTREAD('IME',' ',N,param,value, ...) čita numeričke podatke iz
```

datoteke IME u jednu varijablu.

```
[A,B,C, ...] = TEXTREAD('IME','FORMAT')
[A,B,C, ...] = TEXTREAD('IME','FORMAT',N)
[A,B,C, ...] = TEXTREAD('IME','FORMAT',param,value, ...)
[A,B,C, ...] = TEXTREAD('IME','FORMAT',N,param,value, ...) čita
```

podatke iz datoteke IME u varijable A,B,C, itd. Tip svakog argumenta je specificiran FORMAT stringom. Broj ozlaznih argumenata mora odgovarati broju konverzijskih članova iz FORMAT stringa.

Ako je N specificiran, onda se format string ponavlja N puta. Ako je N is -1 ili nije specificiran, onda TEXTREAD čita cijelu datoteku.

Ako su specificirani (param,value) parovi, onda korisnik posebno ugađa ponašanje naredbe TEXTREAD.

TEXTREAD traži podudarnost i čini konverziju (pretvorbu) u skupinama znakova works Postoje znakovi koji definiraju ograde ili međe (razdjelnike, engl. delimiters) među znakovima.

Podržane su sljedeće specifikacije:

```
%n - čita broj - float ili integer (vraća double array)
      %5n čita do 5 znamenki ili do idućeg razdjelnika
%d - čita integer vrijednost s predznakom (vraća double array)
      %5d čita do 5 znamenki ili do idućeg razdjelnika
%u - čita cjelobrojnu vrijednost (vraća double array)
      %5u čita do 5 znamenki ili do idućeg razdjelnika
%f - čita realni broj (vraća double array)
      %5f čita do 5 znamenki ili do idućeg razdjelnika
%s - čita string omeđen prazninom (vraća cellstr)
      %5s čita do 5 znakova ili do idućeg praznine
%q - čita (moguće s dvostrukim navodnicima omećeni) string
      (i vraća cellstr)
      %5q čita do 5 non-quote znakova ili do iduće praznine
%c - čita znak ili prazninu (vraća char array)
      %5c čita do 5 characters uključujući praznine
%[...] - čita znakove koji se podudaraju sa znakovima unutar zagrada
        sve do prvog znaka koji se ne podudara ili praznine
        (vraća cellstr)
        koristite %[...] da se uključi ]
      %5[...] čita do 5 znakova
%[^...] - čita znakaove koji se ne podudaraju sa znakovima između
        zagrada sve do prvog znaka koji se podudara (vraća cellstr)
        koristite %[^...] da se isključi ]
      %5[^...] čita do 5 characters
```

Primjedba: Formatirani stringovi se interpretiraju kao sa naredbom 'sprintf' prije parsiranja (prepoznavanja i izvođenja).
 Na primjer, `textread('mydata.dat','%s\t')` tražit će znak tab ('\\t'), a ne znak '\\' iza kojeg slijedi znak 't'.
 Upotreba `/*` umjesto `%` u pretvorbi uzrokuje `TEXTREAD` da preskoči podudarne znakove u ulazu (i pritom ne nastaje izlaz za tu konverziju).

Primjeri:

Pretpostavimo da datoteka 'mydata.dat' sadrži podatke u sljedećem obliku:

```
Sally    Type1 12.34 45 Yes
Joe      Type2 23.54 60 No
Bill     Type1 34.90 12 No
```

Čita svaki stupac u pojedinu varijablu:

```
[names,types,x,y,answer] = textread('mydata.dat','%s%s%f%d%s');
```

Čita prvi stupac u polje ćelija (preskačući ostatak linije)

```
[names]=textread('mydata.dat','%s%*[^\\n]')
```

Čita prvi znak u polje znakova (preskačući ostatak linije)

```
[initials]=textread('mydata.dat','%c%*[^\\n]')
```

Čita datoteku kao datoteku fiksnog formata prekačući uz to double podatke

```
[names,types,y,answer] = textread('mydata.dat','%9c%5s%f%2d%3s');
```

Čita datoteku i traži literal 'Type'

```
[names,typenum,x,y,answer]=textread('mydata.dat','%sType%d%f%d%s');
```

Čita m-datoteku u ćeliju polja stringova

```
file = textread('fft.m','%s','delimiter','\\n','whitespace','');
```

Pročitati sve podatke iz tekst datoteke s razdjelnicima, koristite jedan izlazni argument, prazni format string i prikladni razdjelnik.

Na primjer, pretpostavimo da 'data.csv' sadrži:

```
1,2,3,4
5,6,7,8
9,10,11,12
```

Pročitajte čitavu matricu u jednoj varijabli:

```
[data] = textread('data.csv','', 'delimiter','','');
```

Pročitajte prva dva stupca u dvije varijable:

```
[col1, col2] = textread('data.csv','%n%n%*[^\\n]','delimiter','','');
```

Za datoteke s praznim mjestima koristite parametar 'emptyvalue'

Pretpostavimo da datoteka 'data.csv' sadrži:

```
1,2,3,4,,6
7,8,9,,11,12
```

Pročitajte ovu datoteku koristeći NaN na praznim mjestima:

```
[data] = textread('data.csv','', 'delimiter','','','emptyvalue',NaN);
```

STRREAD čita formatirane podatke iz stringa, na sličan način kao i s naredbom `TEXTREAD`.

Primjer:

```
s = sprintf('a,1,2\\nb,3,4\\n');
[a,b,c] = strread(s,'%s%d%d','delimiter','','')
```

`DLMREAD` čita ASCII dataoteku s razdjelnicima.

RESULT= DLMREAD(IME,DELIMITER) čita numeričke podatke iz ASCII datoteke IME koristeći razdjelnik DELIMITER. Resultat se sprema u RESULT.

DLMWRITE piše ASCII datotetu s razdjelnicima između podataka.

DLMWRITE(IME,M,DLM) piše matricu M u datoteku IME koristeći znak DLM kao razdjelnik.

Na sličan način WK1WRITE i WK1READ spremaju i čitaju podatke u tabličnom obliku u Lotus (ili Excell) formatu.

Formati datoteka

Formati podataka	naredba	vraća
MAT - MATLAB workspace	load	Variables in file.
CSV - Comma separated numbers	csvread	Double array.
DAT - Formatted text	importdata	Double array.
DLM - Delimited text	dlmread	Double array.
TAB - Tab separated text	dlmread	Double array.
Spreadsheet (tablični kalkulatori) formati		
XLS - Excel worksheet	xlsread	Double array and cell array.
WK1 - Lotus 123 worksheet	wk1read	Double array and cell array.
Znanstveni (Scientific data) podatčani formati		
CDF - Common Data Format	cdfread	Cell array of CDF records
FITS, HDF - Hierarchical Data		
Filmski (movie) format		
AVI - Movie	aviread	MATLAB movie.
Slikovni (image) formati		
TIFF - TIFF image	imread	Truecolor, grayscale or indexed
PNG - PNG image	imread	Truecolor, grayscale or indexed
HDF - HDF image	imread	Truecolor or indexed image(s).
BMP - BMP image	imread	Truecolor or indexed image.
JPEG - JPEG image	imread	Truecolor or grayscale image.
GIF - GIF image	imread	Indexed image.
PCX - PCX image	imread	Indexed image.
XWD - XWD image	imread	Indexed image.
CUR - Cursor image	imread	Indexed image.
ICO - Icon image	imread	Indexed image.
RAS - Sun raster image	imread	Truecolor or indexed.
PBM - PBM image	imread	Grayscale image.
PGM - PGM image	imread	Grayscale image.
PPM - PPM image	imread	Truecolor image.
Audio formati		
AU - NeXT/Sun sound	auread	Sound data and sample rate.
SND - NeXT/Sun sound	auread	Sound data and sample rate.
WAV - Microsoft Wave sound	wavread	Sound data and sample rate.