بسمه تعالی

دانشکده مهندسی برق
دانشگاه شهید بهشتی

آموزش MatLab مقدماتی

امیرحسین حاجی‌حسینی
AH.Hajihoseini@sbu.ac.ir

زمستان 1395

نرم‌افزار MatLab:

زبان برنامه‌نویسی سطح بالا با تمرکز بر روی تکنیک‌های محاسباتی

Matrix + Laboratory → MatLab

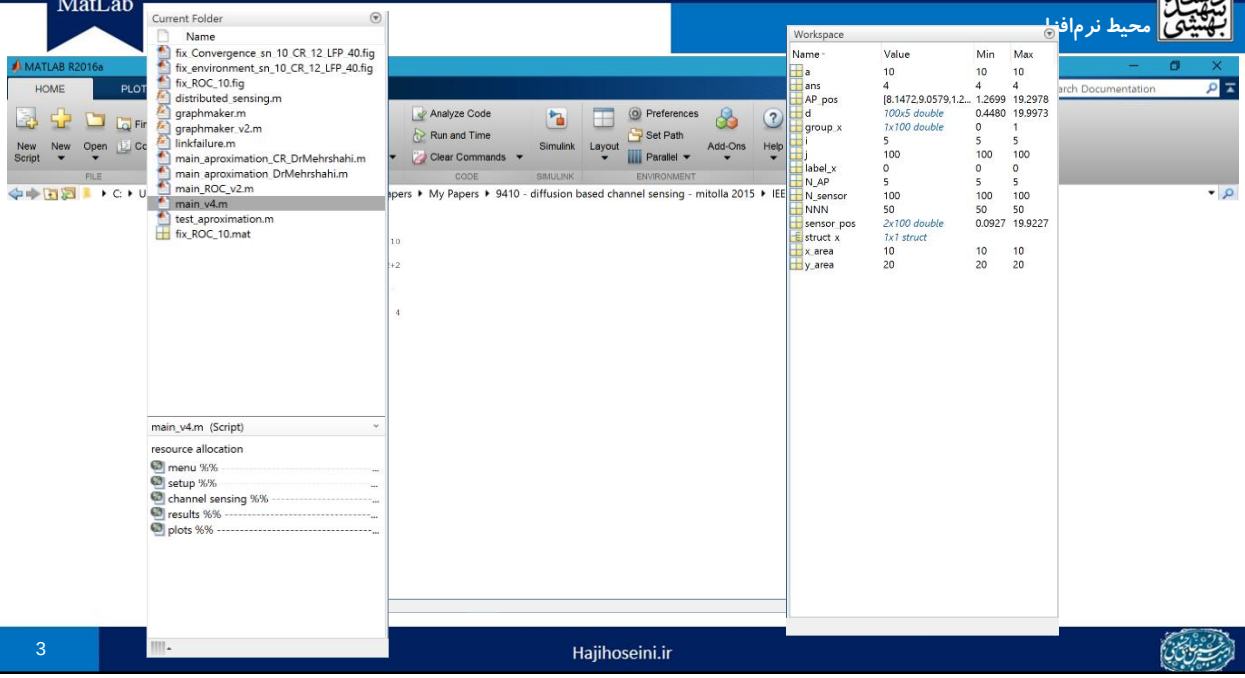
داده‌ها به صورت آرایه هستند.

در این نرم‌افزار جعبه ابزارهای مختلفی برای کاربردهای گوناگون نوشته شده است:

- مخابرات
- سیستم‌های کنترل
- منطق فازی
- پردازش تصویر
- شبکه عصبی
- ...

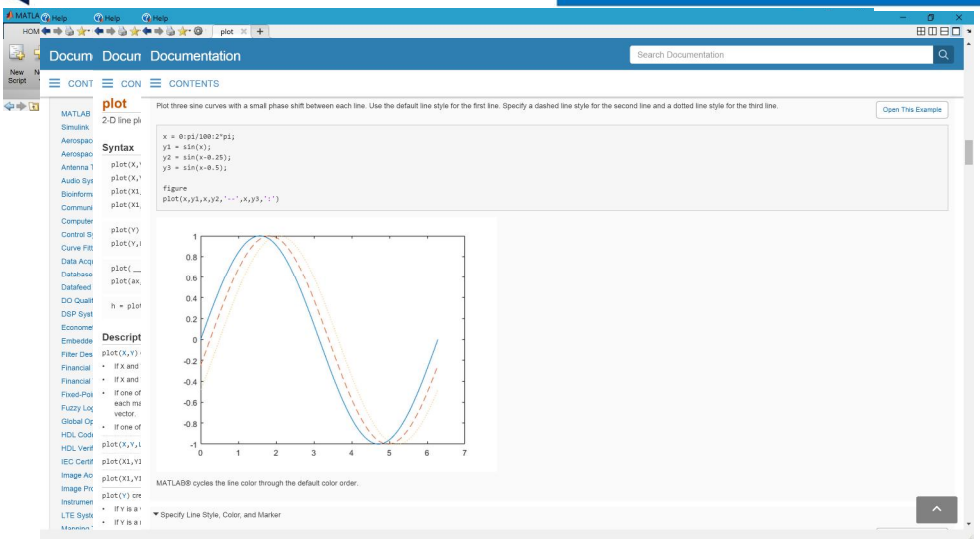
MatLab

محیط نرم‌افزار



MatLab

محیط نرم‌افزار



قوانین نام‌گذاری متغیرها

نسبت به حروف کوچک و بزرگ حساس است.

اسامی متغیرها حداکثر می‌تواند 31 عدد باشد.

اسامی متغیرها باید با حرف شروع شود.

1ab = 5

_c = 2

از نام دستوارت MatLab نمی‌توان برای نام‌گذاری متغیرها استفاده کرد.

for, end, while, function, return, elseif, switch, case, otherwise, continue, else, try, catch, global, break, …

متغیرهای ویژه

نام متغیر	توضیح
ans	
pi	
eps	
inf	
i یا j	
nargin	
nargout	
realmin	
realmax	
flintmax	

MatLab

نقطه‌گذاری

قراردادن ; در آخر یک دستور از چاپ نتایج آن جلوگیری می‌کند.

این متغیر در workspace ذخیره می‌شود اما در command window نمایش داده نمی‌شود.

```
Command Window
New to MATLAB? See resources for Getting Started.
>> a = 1;
>> b = 1

b =

     1

fx >>
```

از % برای کامنت‌گذاری استفاده می‌شود.

با استفاده از , و ; می‌توان چند دستور را در یک خط نوشت.

اگر یک عبارت طولانی باشد، ... قرار می‌دهیم و ادامه دستور را در خط بعد می‌نویسیم.

MatLab

توابع پرکاربرد

توابع گردکردن و باقی‌مانده

fix, floor, ceil, round, mod

توابع مختلط

abs, angle, phase, conj, imag, real

توابع مثلثاتی

sin, cos, cosh, sinh, asin, acos

توابع مربوط به متغیرها

whos a, clc, clear, clear c, clear c*

MatLab

قالب‌های نمایش اعداد

اگر نتیجه محاسبه یک عدد صحیح باشد، MatLab آن را به عنوان یک متغیر صحیح نشان می‌دهد و اگر به صورت یک عدد حقیقی باشد، آن را به صورت پیش‌فرض با چهار رقم اعشار نمایش می‌دهد.

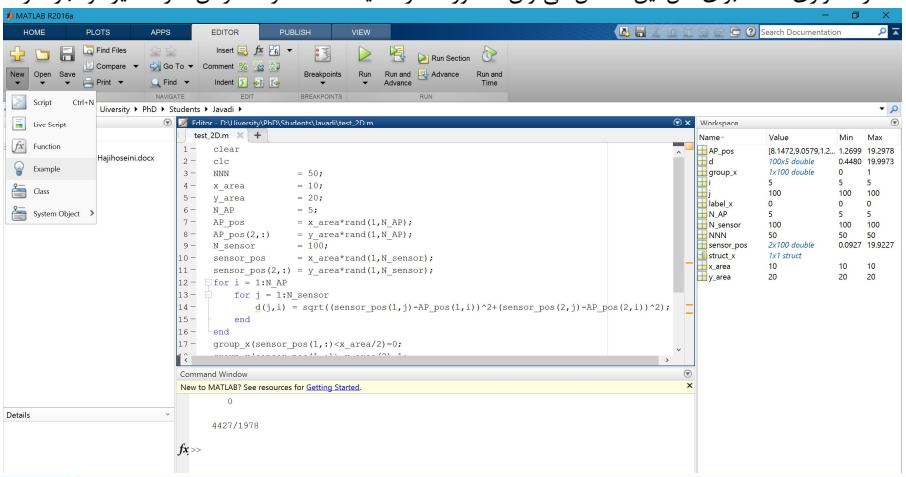
format long

حالت	توضیح
short	
long	
short e	
long e	
hex	
bank	
+	
rational	

MatLab

تابع یا mfile

برای حل مسائل ساده می‌توان دستورات خود را در command window بنویسیم، ولی وقتی تعداد دستورها و استفاده از توابع زیاد باشد، نوشتن و اجرای آن‌ها در command window کار دشواری است. برای حل این مشکل می‌توان دستورات در محیط mfile نوشته و آن‌ها را ذخیره و اجرا کرد.



disp(a)
echo on
echo off
a = input('input number')
pause
Pause(n)

MatLab

مثال 1

دو مقدار عددی را از ورودی بگیرید، سپس آن‌ها را در یکدیگر ضرب کنید و نتیجه را در command window نشان دهید.

MatLab

مثال 2

برنامه‌ای بنویسید که 4 عدد را از ورودی بگیرد، به این صورت که وقتی هر عدد گرفته می‌شود، صفحه پاک شود و سپس عدد بعدی گرفته شود. سپس معدل این اعداد را به دست آورد. در command window فقط معدل را نشان دهید.

MatLab

مثال 3

برنامه‌ای بنویسید که نام سه دانشجو را از ورودی بگیرد، سپس نمرات دروس این دانشجوها را نیز از ورودی دریافت کند و در آخر تنها نام دانشجو به همراه معدل آن نشان داده شود. ضرایب چهار درس به ترتیب 4، 3، 2 و 4 است.

MatLab

مثال 4

اگر a=3-4j و b=1+2j و c=6+8j باشند، برنامه‌ای بنویسید که اندازه z، زاویه z و z* را محاسبه کند.

$$z = \left(\frac{a}{b}\right)^2 c$$

MatLab

انواع داده

- آرایه
  - یک‌بُعدی
  - دو بُعدی
  - چند بُعدی
- رشته‌های کاراکتری
- سلول
- ساختار

MatLab

آرایه‌های یک‌بُعدی

آرایه یک‌بُعدی

آرایه‌هایی هستند که تعداد سطر یا ستون آن‌ها برابر 1 است.

برای تعریف آرایه از کروشه [ ] استفاده می‌کنیم.

```
a = [1,2]
b = [1 4 5 8 1 2 3]
```

هرگاه بخواهیم به هر کدام از درایه‌ها دسترسی داشته باشیم، مقدار اندیس آن درایه را درون یک پرانتز قرار می‌دهیم و در نتیجه عدد مربوط به آن اندیس را برمی‌گرداند.

```
c = b(3)
c =

     5
```

آرایه‌های یک‌بُعدی

اگر بخواهیم در یک زمان یک بلوک از درایه‌های یک ماتریس را نشان دهیم، داخل پرانتز اندیس عدد اول، سپس : و در نهایت اندیس عدد آخر را قرار می‌دهیم.

```
b = [1 4 5 8 1 2 3]
d = b(3:6)
d =

     5     8     1     2
```

اگر بخواهیم از یک اندیس تا انتهای آرایه را انتخاب کنیم، از end استفاده می‌کنیم.

```
e = b(3:end)
e =

     5     8     1     2     3
```


آرایه‌های یک‌بُعدی

اگر بخواهیم اعداد با یک ترتیب خاص را از یک آرایه انتخاب کنیم، مقدار وسط، مقدار نِمُو اندیس را قرار می‌دهیم.

```
a = [1 2 3 4 5 6 7 8 9]
b = a(1:2:end)
c = a(2:2:end)
d = a(1:3:9)
e = a([3 2 8 5])

b =
     1     3     5     7     9

c =
     2     4     6     8

d =
     1     4     7

e =
     3     2     8     5
```

MatLab

مثال 5

یک آرایه یک‌بُعدی با 9 عضو ایجاد کنید. سپس آرایه‌ی جدید را به گونه‌ای بسازید که درایه‌های آن، معادل درایه‌های آرایه اول ولی از آخر به اول باشد.

MatLab

آرایه‌های منظم

برای ساخت آرایه‌های منظم می‌توان از روش‌های زیر استفاده نمود:

```
a = [1:3:100];
b = (0:5:200);
c = 0:pi/10:2*pi;
d = linspace(0,1,100);
e = logspace(1,3,10);
```

MatLab

آرایه ستونی

اگر بخواهیم آرایه را از حالت سطری به ستونی تغییر دهیم، از ترانهاده ماتریس استفاده می‌کنیم:

```
a = 1:10
b = a'
a =
     1     2     3     4     5     6     7     8     9    10

b =
     1
     2
     3
     4
     5
     6
     7
     8
     9
    10
```


MatLab

آرایه ستونی

اگر بخواهیم آرایه‌ی ستونی به صورت مستقل ایجاد کنیم، بین داریه‌ها از ; استفاده می‌کنیم.

```
a = [1; 2; 3; 4; 5]
a =
     1
     2
     3
     4
     5
```

MatLab

آرایه‌های دوبُعدی

آرایه‌هایی که سطر و ستون آن‌ها بزرگ‌تر از یک باشد.

```
a = [1 2; 3 4]
a =
     1     2
     3     4
```

باید تعداد ستون‌های موجود در تمام سطرها با یکدیگر برابر باشد، در غیراینصورت پیغام خطا در صفحه ظاهر می‌شود.

MatLab

محاسبات عددی آرایه‌ها

تمام عملیات‌های ریاضی بین اعداد و آرایه‌ها برقرار است. برای ضرب ماتریسی، باید ابعاد ماتریس‌ها مناسب باشند، همچنین برای عملیات جمع، تفریق، ضرب نظیر به نظیر و تقسیم نظیر به نظیر، باید ابعاد دو ماتریس با هم برابر باشد.

```
a*b
a^2
a+b
a-b
a.*b
a./b
a.^2
```

MatLab

محاسبات عددی آرایه‌ها

برخی از توابع مربوط به عملیات‌های ریاضی عبارتند از:

نام تابع	توضیح
mean	
std	
sum	
prod	
cumsum	
cumprod	
dot	
cross	
norm	

اگر ورودی این توابع آرایه‌ی چندبُعدی باشد، عملیات روی هر ستون ماتریس انجام می‌شود ولی اگر ورودی آرایه‌ی یک‌بُعدی باشد، عملیات روی کل آرایه انجام می‌شود.

MatLab

مثال 6

برنامه‌ای بنویسید که دو بردار را به عنوان ورودی دریافت کند، سپس زاویه بین آن‌ها را در خروجی نمایش دهد.

MatLab

آرایه‌های استاندارد

این نرم‌افزار دارای توابعی است که می‌توانند آرایه‌های استاندارد را با ابعاد مختلف تولید کنند. برخی از این توابع عبارتند از:

نام تابع	توضیح
ones(m,n)	
ones(m)	
zeros(m,n)	
eye(m,n)	
rand(m,n)	
randn(m,n)	

MatLab

مثال 7

یک آرایه 4 در 5 بسازید که درایه‌های آن اعداد صحیح رندوم بین دو تا دوازده باشد.

MatLab

تابع diag

اگر ورودی این تابع یک ماتریس باشد، عناصر روی قطر اصلی را برمی‌گرداند و اگر ورودی این تابع یک آرایه‌ی یک‌بُعدی باشد، خروجی آن یک ماتریس است که عناصر قطر اصلی آن، ماتریس ورودی تابع است.

```
a = rand(4)
b = diag(a)
c = rand(1,4)
d = diag(c)
```

Command Window

```
a =

    0.7803    0.0965    0.5752    0.8212
    0.3897    0.1320    0.0598    0.0154
    0.2417    0.9421    0.2348    0.0430
    0.4039    0.9561    0.3532    0.1690


b =

    0.7803
    0.1320
    0.2348
    0.1690


c =

    0.6491    0.7317    0.6477    0.4509


d =

    0.6491         0         0         0
         0    0.7317         0         0
         0         0    0.6477         0
         0         0         0    0.4509

fx >>
```


MatLab

تابع repmat

اولین ورودی این تابع آرایه‌ای است که می‌خواهیم آن را تکرار کنیم. ورودی‌های بعدی نیز تعداد سطر و ستون را مشخص می‌کند. با یک مثال این دستور را بررسی می‌کنیم.

```
a = [1 2 3; 4 5 6]
b = repmat(a,4,5)
```

Command Window

```
a =

     1     2     3
     4     5     6


b =

     1     2     3     1     2     3     1     2     3     1     2     3     1     2     3
     4     5     6     4     5     6     4     5     6     4     5     6     4     5     6
     1     2     3     1     2     3     1     2     3     1     2     3     1     2     3
     4     5     6     4     5     6     4     5     6     4     5     6     4     5     6
     1     2     3     1     2     3     1     2     3     1     2     3     1     2     3
     4     5     6     4     5     6     4     5     6     4     5     6     4     5     6
     1     2     3     1     2     3     1     2     3     1     2     3     1     2     3
     4     5     6     4     5     6     4     5     6     4     5     6     4     5     6

fx >>
```

MatLab

تابع reshape

با استفاده از این دستور می‌توان چینش سطری و ستونی یک آرایه را تغییر داد. باید این نکته را در نظر داشت که تعداد درایه‌های آرایه‌ی اصلی و تغییریافته یکسان باشد. ورودی‌های این تابع آرایه‌ی اصلی و سایز آرایه‌ی جدید هستند.

```
a = [1 2 3 4; 5 6 7 8; 9 10 11 12]
b = reshape(a,2,6)
c = reshape(a,4,3)
```

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

b =
     1     9     6     3    11     8
     5     2    10     7     4    12

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


MatLab

الحاق آرایه‌ها

در این بخش می‌خواهیم با استفاده از تعدادی از آرایه‌ها، یک آرایه‌ی جدید بسازیم.

```
a = [1 2]
b = [3 4]
c = [5;6]
d = [a;b]
e = [d c]
f = [e e;a b c']
```

```
a =
     1     2

b =
     3     4

c =
     5
     6

d =
     1     2
     3     4

e =
     1     2     5
     3     4     6

f =
     1     2     5     1     2     5
     3     4     6     3     4     6
     1     2     3     4     5     6
```

MatLab

آدرس‌دهی در آرایه‌ها و تغییر آن‌ها

برای ایجاد تغییر در درایه‌های یک آرایه باید ابتدا اندیس درایه‌ی مورد نظر را به دست آورد. سپس درایه‌ی مورد نظر را مقداردهی می‌کنیم. در آرایه‌ها، آدرس‌دهی مقادیر آرایه به دو صورت انجام می‌شود:

1) اندیس

2) سطر و ستون

```
a =
   1  1    5  2    9  3   13  4
   2  5    6  6   10  7   14  8
   3  9    7 10   11 11   15 12
   4 13    8 14   12 15   16 16

b = a(2,3)
c = a(10)
```


MatLab

آدرس‌دهی در آرایه‌ها و تغییر آن‌ها

برای تبدیل این دو فرمت به یکدیگر می‌توان از دو تابع زیر استفاده کرد:

Index = sub2ind(size(a),l,m)

[m,n] = ind2sub(size(a),ind)

[i,j] = size(a)

```
a =
     1     2     3     4
     5     6     7     8
     9    10    11    12
    13    14    15    16
```

```
Index = sub2ind(size(a),3,2)
[m,n] = ind2sub(size(a),7)

    Index =
         7
    m =
         3
    n =
         2
```

MatLab

آدرس‌دهی در آرایه‌ها و تغییر آن‌ها

برای تغییر در آرایه‌ها، ابتدا اندیس درایه‌هایی که باید تغییر کنند را مشخص می‌کنیم، سپس آن‌ها را مقداردهی می‌کنیم.

```
a =
     1     2     3     4
     5     6     7     8
     9    10    11    12
    13    14    15    16
```

```
a(1,2)= 99
a(12) = 88
a(2:3,2:3) = 77
a(:,4) = 66
```

```
a =
     1    99     3    66
     5    77    77    66
     9    77    77    66
    13    14    88    66
```


MatLab

مثال 8

یک ماتریس چهار در چهار به صورت رندوم تولید کنید که درایه‌های آن از اعداد صحیح بین صفر تا صد باشد. سپس جای ستون اول را با سطر اول عوض کنید.

MatLab

مثال 9

یک ماتریس چهار در چهار به صورت رندوم تولید کنید که درایه‌های آن از اعداد صحیح بین صفر تا صد باشد. سپس جای ستون‌ها را از آخر به اول عوض کنید.

MatLab

مرتب کردن آرایه‌ها

برای مرتب کردن آرایه‌ها از کوچک به بزرگ، از تابع sort استفاده می‌کنیم. با استفاده از sort دو مقدار برگردانده می‌شود که مقدار اولی آرایه‌ی مرتب‌شده می‌باشد و مقدار دومی اندیس آرایه‌ی sortشده می‌باشد.

```
a = [2 4 5 6 1 3 4 2 1 7 8]
[x,ind] = sort(a)
```

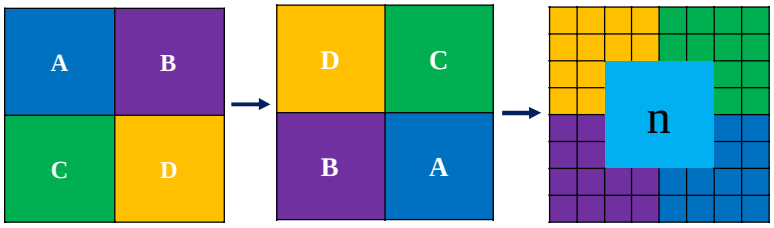
```
a =
     2     4     5     6     1     3     4     2     1     7     8
x =
     1     1     2     2     3     4     4     5     6     7     8
ind =
     5     9     1     8     6     2     7     3     4    10    11
```

برای آرایه‌های دوبُعدی sort تنها روی ستون انجام می‌شود، نه کل آرایه‌ها.

MatLab

مثال 10

یک ماتریس 8 در 8 به صورت رندوم انتخاب کنید که مقادیر آن بین صفر تا 20 باشد. سپس این آرایه را به 4 زیرآرایه‌ی A، B، C و D مطابق شکل تبدیل کنید. سپس با استفاده از این آرایه‌ها، آرایه‌ای جدید مطابق شکل دوم بسازید. آرایه n را از ماتریس به دست آمده از مرحله دوم استخراج کنید و آن را به صورت ستونی از بزرگ به کوچک قرار دهید.



MatLab

جستجو در زیرآرایه‌ها

با استفاده از تابع find می‌توان اندیس یک سری از درایه‌های خاص را که مدنظر می‌باشد، از آرایه‌ی اصلی جدا کنیم. برای مثال می‌خواهیم اعداد بزرگتر از 4 را در آرایه‌ی a مشخص کنیم. این کار را می‌توان به دو صورت انجام داد:

```
a      = 10*rand(5)
b      = find(a>4);
[m,n] = find(a>4);
```

```
a =
    8.0101    5.7853    5.2114    3.9552    9.1329
    0.2922    2.3728    2.3159    3.6744    7.9618
    9.2885    4.5885    4.8890    9.8798    0.9871
    7.3033    9.6309    6.2406    0.3774    2.6187
    4.8861    5.4681    6.7914    8.8517    3.3536
b  =
    1    3    4    5    6    8    9   10   11   13   14   15   18   20   21   22
m  =
    1    3    4    5    1    3    4    5    1    3    4    5    3    5    1    2
n  =
    1    1    1    1    2    2    2    2    3    3    3    3    4    4    5    5
```

MatLab

مثال 11

یک آرایه هشت در هشت متشکل از اعداد صحیح بین صفر تا 10 تولید کنید. سپس اندیس آرایه‌هایی که دقیقا برابر شش هستند را به دست آورید.

MatLab

جستجو در زیرآرایه‌ها

با استفاده از تابع max(min) می‌توان بزرگترین(کوچکترین) درایه‌ی هر ستون را همراه اندیس آن به دست آورد. اگر آرایه یک‌بُعدی باشد، بزرگترین(کوچکترین) درایه‌ی کل آرایه را می‌دهد.

```
a       = 10*rand(5)
[m1,n1] = max(a);
[m2,n2] = min(a);
```

```
a =
    1.4761    6.9667    0.0052    4.7952    5.7466
    0.5497    5.8279    8.6544    8.0135    8.4518
    8.5071    8.1540    6.1257    2.2784    7.3864
    5.6056    8.7901    9.8995    4.9809    5.8599
    9.2961    9.8891    5.2768    9.0085    2.4673


m1 =
    9.2961    9.8891    9.8995    9.0085    8.4518

n1 =
     5     5     4     5     2
```

MatLab

مثال 12

یک آرایه‌ی سه در سه که درایه‌های آن بین صفر تا 5 باشد را ایجاد نمایید. سپس بزرگترین درایه را به همراه سطر و ستون آن به دست آورید.

MatLab

توابع مربوط به تغییر در آرایه

تعدادی از توابع برای تغییر در سطر و ستون آرایه‌ها یا مقادیر آن‌ها استفاده می‌شوند که برخی از آن‌ها را مرور می‌نماییم.

نام تابع	توضیح
flipud(a)	
fliplr(a)	
rot90(a)	
rot90(a,2)	
triu(a)	
tril(a)	
trace(a)	
inv(a)	
det(a)	

MatLab

مثال 13

یک آرایه‌ی سه در سه که درایه‌های آن بین صفر تا 20 باشد را ایجاد نمایید. سپس با یک دستور قسمت پایین مثلثی ماتریس را بدون قطر اصلی محاسبه کنید.

MatLab

توابع مربوط به سایز آرایه‌ها

توابع مربوط به سایز آرایه‌ها عبارتند از:

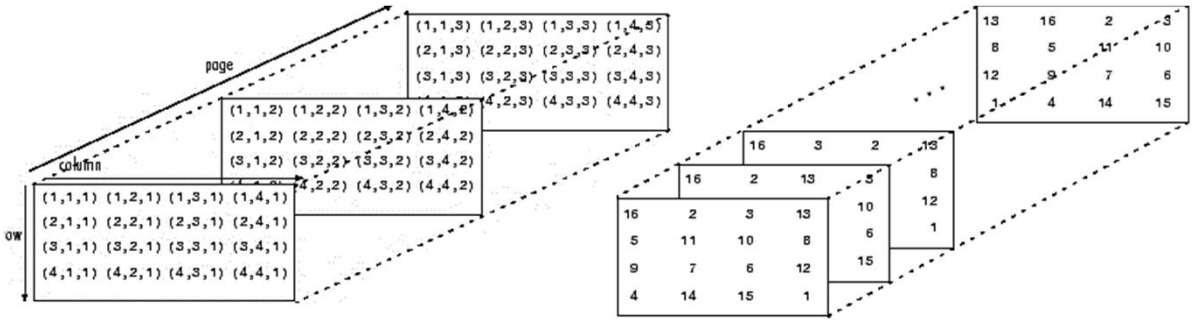
نام تابع	توضیح
[m,l,…] = size(a)	
k= numel(a)	
p = length(a)	

MatLab

آرایه‌های چندبُعدی (سه‌بُعدی)

این آرایه‌ها بجز سطر و ستون دارای بُعد سومی به نام صفحه هستند که هر صفحه دارای سطر و ستون می‌باشد.

```
a(i,j,k)
```



MatLab

مثال 14

یک آرایه‌ی سه‌بُعدی شامل دو سطر، دو ستون و سه صفحه بسازید به طوریکه تمام درایه‌های صفحه اول، صفر، صفحه دوم، یک و صفحه سوم، چهار باشند.

MatLab

تبدیل آرایه‌های چندبُعدی به دوبُعدی و برعکس

چگونه می‌توان یک ماتریس سه‌بُعدی را به یک ماتریس دوبُعدی تبدیل کرد؟

```
c = [a(:,:,1) a(:,:,2) ... ]
c = reshape(a,m,n)
```

حاصل‌ضرب تعداد سطر و ستون ماتریس جدید باید برابر کل درایه‌های ماتریس قبلی باشد.

چگونه می‌توان ماتریس سه‌بُعدی ساخت؟

```
c = repmat(a,m,n,k)

d = cat(3,a,b,c)

d(:,:,1) = a;
d(:,:,2) = b;
d(:,:,3) = c;
```


MatLab

ترانهاده کردن ماتریس سه‌بُعدی

برای جابجا کردن سطر و ستون در هر صفحه یا ترانهاده کردن هر صفحه، می‌توان از تابع زیر استفاده کرد:

```
a = ones(2,4,3)
b = permute(a,[2 1 3])
```

برای برگرداندن به حالت قبل می‌توان از دستور زیر استفاده کرد:

```
c = ipermute(b,[2 1 3])
```

MatLab

عملگرهای منطقی

از عملگرهای منطقی برای پاسخ‌های درست نادرست (true, false) استفاده می‌شود.

```
~=    ==    >=    <=    <    >
```

```
a = [1 2; 3 4];
b = [4 2; -1 5];
c = a>=b
```

```
a =
     1     2
     3     4

b =
     4     2
    -1     5

c =

     0     1
     1     0
```


MatLab

عملگرهای منطقی

برخی از توابع دارای خروجی درست نادرست هستند که معمولا با is شروع می‌شوند. در ادامه برخی از این توابع را معرفی می‌نماییم.

نام تابع	توضیح
isinf	
iscell	
isfield	
isstruct	
ischar	
isreal	
isletter	
isspace	

MatLab

مثال 15

دو آرایه‌ی یک در 9 تعریف کنید که درایه‌های آن اعداد صحیح بین صفر تا 10 باشد. آرایه‌ی سوم را به نحوی تعریف کنید که هرگاه یک درایه از ماتریس اول بزرگتر از سه بود، درایه‌ی نظیر آن در ماتریس دو یک واحد کم شود.

MatLab

عملگرهای رابطه‌ای

and &

or |

not ~

آرایه‌ی A = [4 5 2 1 6 7] را در نظر بگیرید. با توجه به این آرایه، آرایه‌ی جدیدی بسازید به این صورت که اعداد بزرگتر از 2 و کوچکتر از 7 را معکوس کنید و بقیه اعداد را در جای خود حفظ کنید.

```
format rational
a = [4 5 2 1 6 7]
b = (a>2 & a<7)./a + ~(a>2 & a<7).*a

a =
        4          5          2          1          6          7

b =
        1/4        1/5        2          1          1/6        7
```

MatLab

رشته‌های کاراکتری

رشته‌های کاراکتری مانند آرایه‌ها عمل می‌کنند و هنگامی که یک رشته کاراکتری را به یک متغیر نسبت می‌دهیم، مانند یک ماتریس عمل می‌کند که درایه‌های آن از حروف تشکیل شده‌اند.

```
b = 'ALI';
[m,n] = size(b)
c = b(2)

     m =
          1

     n =
          3

     c =
          L
```


MatLab

رشته‌های کاراکتری

با استفاده از تابع double یا تابع abs می‌توان کد اسکی رشته‌های کاراکتری را به دست آورد.

```
a = 'ALI';
b = abs(a)
c = double(a)

                    b =
                        65    76    73

                    c =
                        65    76    73
```

با استفاده از تابع char می‌توان کاراکتر معادل کد اسکی را به دست آورد.

```
d = char(c)

                    d =

                        ALI
```

MatLab

رشته‌های کاراکتری

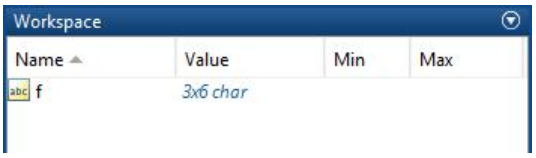
برای ایجاد توابع تمام کاراکتری می‌توان از دستورات زیر استفاده کرد:

```
f = char('ali', 'reza', 'behnam')

    f =
       ali
       reza
       behnam

f1 = strcat('ali', 'reza', 'behnam')
    f1 =
       alirezabehnam

f2 = strvcat('ali', 'reza', 'behnam')
    f2 =
       ali
       reza
       behnam
```



MatLab

رشته‌های کاراکتری

برخی از توابع مورد نیاز برای کار با رشته‌های کاراکتری در جدول زیر نشان داده شده‌اند:

نام تابع	توضیح
upper(a)	
lower(a)	
findstr(a,'is')	
strrep(s1,'a','b')	

MatLab

رشته‌های کاراکتری

با استفاده از تابع‌های fprintf و sprintf می‌توان خروجی‌های برنامه را به صورت توضیحی در صفحه نمایش داد.

```
a = 15;
b = 30;
f = fprintf('The 1st Number is %0.5g and the 2nd Number is %0.5g \n',a,b);
s = sprintf('The 1st Number is %0.5g and the 2nd Number is %0.5g \n',a,b);
```

Command Window

```
The 1st Number is 15 and the 2nd Number is 30
>>
```

تابع fprintf تعداد کاراکترها را در متغیر ذخیره می‌کند و نتیجه را در خروجی نمایش می‌دهد، اما تابع sprintf کاراکترهای خروجی را در خود ذخیره می‌کند و نتیجه را نمایش نمی‌دهد. برای نمایش نتیجه باید از دستور disp استفاده کرد.

MatLab

استفاده از syms

با استفاده از این دستور می‌توان متغیر سمبلیک تعریف کرد.

```
syms x y
f(x,y) = x^2+y;
f(2,1)
```

```
ans =

     5
```

```
syms t
a = laplace(t)
```

```
a =

     1/s^2
```

```
syms t
a = fourier(cos(t))
```

```
a =

     pi*(dirac(w - 1) + dirac(w + 1))
```

MatLab

روند کنترل

بعضی از دستورات کنترلی می‌توانند روند اجرای دستورات را کنترل کنند. این دستورات عبارتند از:

- حلقه‌های for
- حلقه‌های while
- شرط if else end
- شرط switch case
- بلوک‌های try catch

MatLab

حلقه for

حلقه for این امکان را به ما می‌دهد که گروهی از محاسبات را برای مقادیر مختلف یک متغیر تکرار کنیم.

```
for i=array
        command
end
```

با استفاده از حلقه for جواب‌های تابع sin در بازه [1 pi] و طول نمو pi/10 را به دست آورید.

```
clear; clc;
for i=1:10
    x(i) = sin(i*pi/10);
end
```

```
clear; clc;
n = 1;
for i=1:pi/10:10
    x(n) = sin(i);
    n    = n + 1;
end
```

MatLab

مثال 16

برنامه‌ای بنویسید که عدد n را از کاربر دریافت کند و !n را در خروجی نمایش دهد.

MatLab

حلقه while

برخلاف حلقه for که گروهی از دستورات به دفعات ثابتی اجرا می‌شوند، در حلقه while می‌توان گروهی از دستورات را به تعداد دفعات نامحدود اجرا کرد. تعداد دفعات حلقه while توسط شرط این حلقه به دست می‌آید. تا وقتی که شرط برقرار باشد، حلقه تکرار می‌شود و وقتی شرط برقرار نباشد، حلقه به اتمام می‌رسد.

```
while (شرط)
        commands
end
```

برای خروج آنی و لحظه‌ای از حلقه‌ها می‌توان از دستور break استفاده کرد.

MatLab

مثال 17

با استفاده از حلقه while کوچکترین مقدار عددی را به دست آورید که وقتی با یک عدد جمع می‌شود، از آن عدد بزرگتر شود.

MatLab

شرط if-else-end

در بسیاری از مواقع گروهی از دستورات را می‌توان با اعمال یک شرط اجرا کرد. برای این کار از دستور if استفاده می‌کنیم.

```
if (شرط)
        commands
else
        commands
end
```

اگر شرط برقرار باشد، دستورات زیر if اجرا می‌شوند و اگر شرط برقرار نباشد، دستورات زیر else اجرا می‌شوند.

اگر تعداد شرط‌ها بیش از سه باشد، می‌توان از روابط زیر استفاده کرد:

```
if (شرط اول)
        commands
elseif (شرط دوم)
        commands
elseif (شرط سوم)
        commands
end
```

MatLab

مثال 18

فرض کنید رمز عبور یک دانشجو برای دیدن معدل خود عدد 313 است. برنامه‌ای بنویسید که رمز عبور را از کاربر بگیرد. اگر رمز عبور درست بود، یک عدد بین 14 تا 18 به صورت رندوم به او نمایش دهد و اگر رمز غلط بود، عبارت 'Your password is incorrect' را نمایش دهد.

MatLab

شرط switch case

هنگامی که یک متغیر می‌تواند مقادیر مختلفی را اختیار کند و بر اساس هر کدام از این مقادیر، دستورات متفاوتی باید اجرا شوند، از این دستور شرطی استفاده می‌کنیم.

```
switch (متغیر مورد نظر)
        case (حالت اول)
                commands
        case (حالت دوم)
                commands
        .
        .
        .
        otherwise
                commands
end
```

شرط switch می‌تواند یک مقدار عددی یا کاراکتر باشد که توسط هر دستور case شرط switch و شرط case بررسی می‌شود، اگر این دو شرط با یکدیگر برابر باشند، دستورات case مورد نظر اجرا می‌شوند و از حلقه خارج می‌شود، در غیر اینصورت case بعدی بررسی می‌شود.

MatLab

مثال 19

برنامه‌ای بنویسید که یک عدد برحسب سانتی متر و واحد مورد نظر برای خروجی را از کاربر می‌گیرد. سپس عدد را در واحد جدید نمایش دهد (تبدیل سانتی‌متر به میلی‌متر، متر و اینچ).

MatLab

بلوک‌های try catch

برای تست کردن خطای یک دستور می‌توان از try catch استفاده کرد. نحوه استفاده از این دستور به صورت زیر است:

```
try
        command 1
catch
        command 2
end
```

اگر دستور زیرمجموعه‌ی try صحیح باشد، اجرا می‌شود و در صورتیکه این دستور نادرست باشد، دستور زیرمجموعه‌ی catch اجرا می‌شود.

MatLab

مثال 20

برنامه‌ای بنویسید که 2ماتریس از کاربر بگیرد و به صورت ماتریسی در هم ضرب کند و اگر ضرب خطا داشت، پیغام ‘X and Y are not conform’ را نمایش دهد.

MatLab

استفاده از function

تعریف تابع در نرم‌افزار MatLab به صورت زیر است:

```
function [Output1,Output2,...] = FunctionName(Input1,Input2,...)
        commands

        Output1 = output*1;
        Output2 = output*2+5;
        .
        .
        .
end
```

MatLab

مثال 21

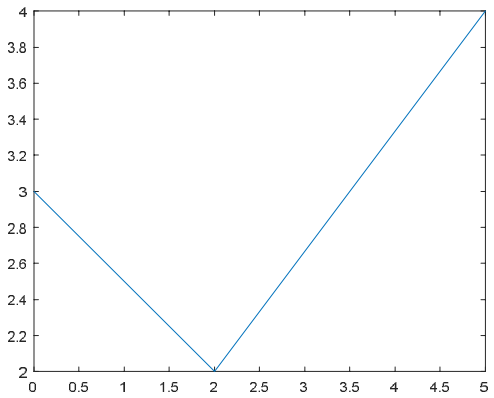
برنامه‌ای بنویسید و در آن دو ماتریس a و b را تعریف کنید. سپس تابعی به نام g بنویسید که در این تابع دو ماتریس تعریف شده را به عنوان ورودی بگیرد، سپس این دو ماتریس را نظیر به نظیر در هم ضرب کرده و همه درایه‌ها را معکوس کند.

MatLab

رسم نمودار

با استفاده از تابع plot می‌توان نمودارهای دوبُعدی رسم کرد. ورودی‌های این تابع دو آرایه هستند که ورودی اول آرایه، محور افقی و ورودی دوم محور عمودی می‌باشد. درایه‌های این دو آرایه به صورت نظیر به نظیر روی نمودار رسم می‌شوند.

```
x = [0 2 5];
y = [3 2 4];
plot(x,y)
```



MatLab

مثال 22

تابع sin(x) را در بازه‌ی صفر تا 2*pi رسم کنید.

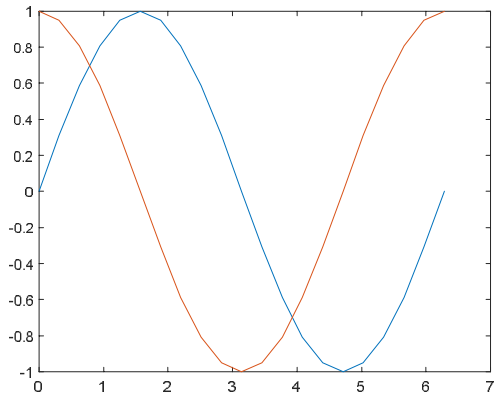
MatLab

رسم نمودار

برای رسم همزمان دو شکل در یک نمودار می‌توان از روش‌های زیر استفاده کرد:

```
x = 0:pi/10:2*pi;
y = sin(x);
z = cos(x);
plot(x,y,x,z)
```

```
x = 0:pi/10:2*pi;
y = sin(x);
z = cos(x);
w = [y;z];
plot(x,w)
```



MatLab

رسم نمودار

اگر تابع plot دارای فقط یک ورودی باشد، در محور افقی، مقدار حقیقی را قرار می‌دهد و در محور عمودی مقدار موهومی را.

```
plot(x)
plot(real(x),imag(x))
```

اگر تابع plot دارای فقط یک ورودی با یک بُعد باشد، در محور افقی، مقدار از عدد یک تا طول آرایه را قرار می‌دهد.

```
x = [5 4 8 4 1];
plot(x)
plot([1 2 3 4 5],x)
```


MatLab

رسم نمودار

نوع خطوط، علائم و رنگ‌ها در جداول زیر نشان داده شده‌اند:

نماد	رنگ
b	آبی
g	سبز
r	قرمز
c	فیروزه‌ای
m	ارغوانی
y	زرد
k	سیاه
w	سفید

نماد	علامت
.	نقطه
o	دایره
+	علامت جمع
*	ستاره
s	مربع
×	علامت ضربدر
d	لوزی
v	مثلث رو به پایین
^	مثلث رو به بالا
>	زاویه رو به چپ
<	زاویه رو به راست
p	ستاره پنج پر
h	ستاره شش پر

```
plot(x,y,'bo-')
```

نماد	نوع خط
-	توپر
:	نقطه چین
-.	خط نقطه
--	خط چین

MatLab

مثال 23

تابع sin(x)، cos(x) و 2*cos(x) را با استفاده از نماد و رنگ‌های مختلف در یک نمودار رسم کنید.

MatLab

رسم نمودار

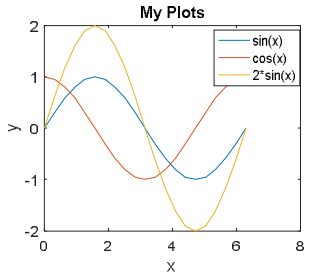
توابع پرکاربرد برای رسم نمودارها به شرح زیر هستند:

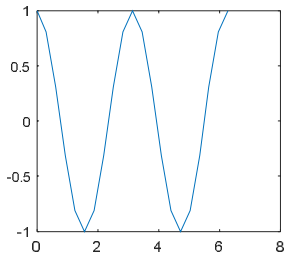
نام تابع	توضیح
figure	
hold on	
hold off	
grid on/grid off	
xlabel('x')	
ylabel('y')	
title('title')	
axis([x1 x2 y1 y2])	
legend	
gtext('text')	

MatLab

رسم نمودار

```
clear
clc
x  = 0:pi/10:2*pi;
y1 = sin(x);
y2 = cos(x);
y3 = 2*sin(x);
y4 = cos(2*x);
figure;
plot(x,y1);
hold on
plot(x,y2)
plot(x,y3)
xlabel('x'); ylabel('y'); title('My Plots')
legend('sin(x)','cos(x)','2*sin(x)')
hold off
figure
plot(x,y4)
```



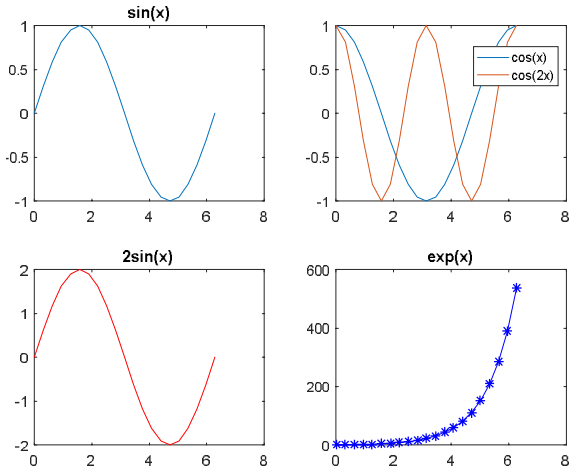


MatLab

رسم نمودار – تابع subplot

با استفاده از این دستور می‌توان یک پنجره را به چند قسمت تقسیم کرده و در هر قسمت یک نمودار را نمایش داد.

```
clear
clc
x  = 0:pi/10:2*pi;
y1 = sin(x);
y2 = cos(x);
y3 = cos(2*x);
y4 = 2*sin(x);
y5 = exp(x);
figure;
subplot(2,2,1); plot(x,y1); title('sin(x)')
subplot(2,2,2);
plot(x,y2,x,y3);legend('cos(x)','cos(2x)');
subplot(2,2,3); plot(x,y4,'r');
title('2sin(x)')
subplot(2,2,4); plot(x,y5,'b*-');
title('exp(x)')
```



MatLab

رسم نمودار

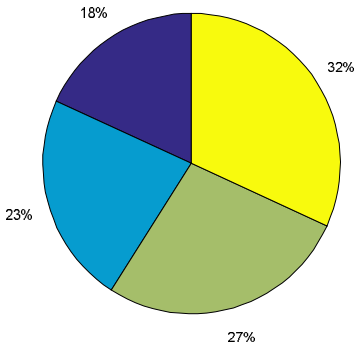
برای اینکه بتوانیم محور افقی را به صورت لگاریتمی درآوریم به جای تابع plot می‌توان از semilogx استفاده کرد. همچنین اگر بخواهیم محور عمودی به صورت لگاریتمی باشد، می‌توان از دستور semilogy استفاده کرد.

دستور loglog هر دو محور را به صورت لگاریتمی نمایش می‌دهد.

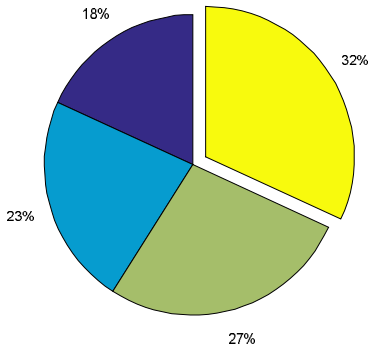
MatLab

رسم نمودار – تابع pie

```
x  = [4 5 6 7];
figure; pie(x)
```



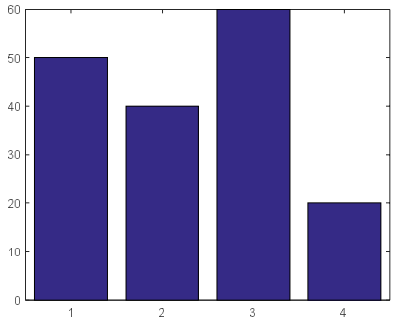

```
figure; pie(x,x==max(x))
```



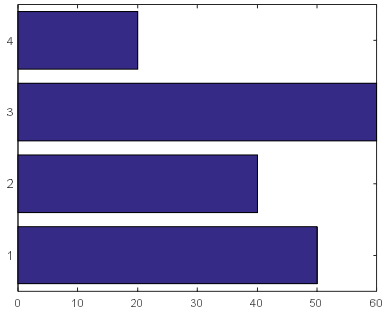
MatLab

رسم نمودار – تابع bar

```
x = [1 2 3 4];
y = [50 40 60 20];
figure; bar(x,y)
```



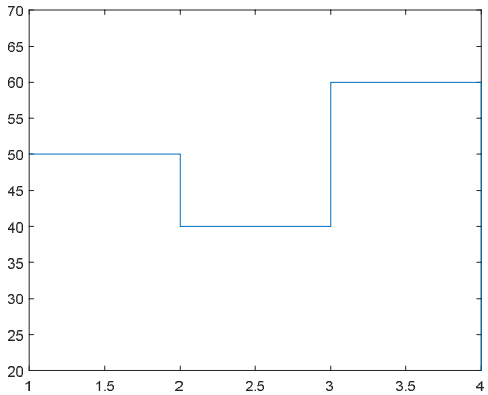

```
figure; barh(x,y)
```



MatLab

رسم نمودار – تابع stairs

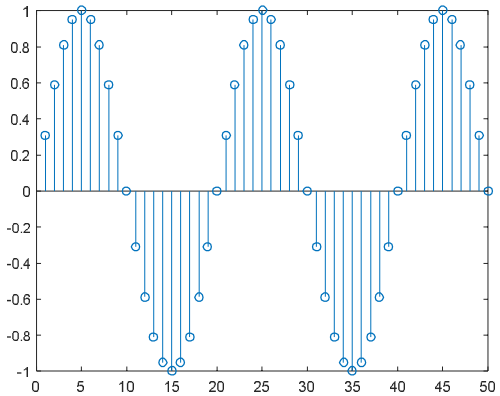
```
x = [1 2 3 4];
y = [50 40 60 20];
figure; stairs(x,y)
axis([1 4 20 70])
```



MatLab

رسم نمودار – تابع stem

```
x = 1:50;
y = sin(2*pi/20*x);
figure; stem(x,y)
```

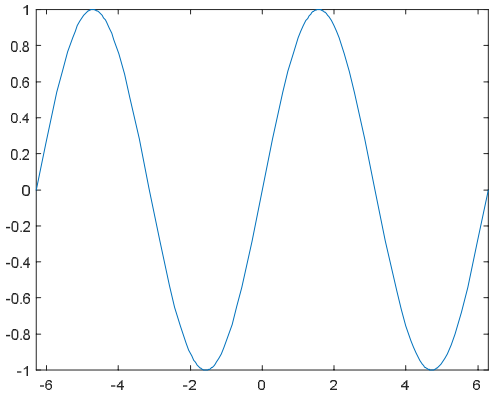


MatLab

رسم نمودار – تابع fplot

با استفاده از این دستور می‌توان یک تابع ریاضی و محدوده محور افقی را از ورودی گرفته و نمودار را در بازه خواسته شده رسم کرد.

```
fplot(@(x) sin(x),[-2*pi 2*pi])
```

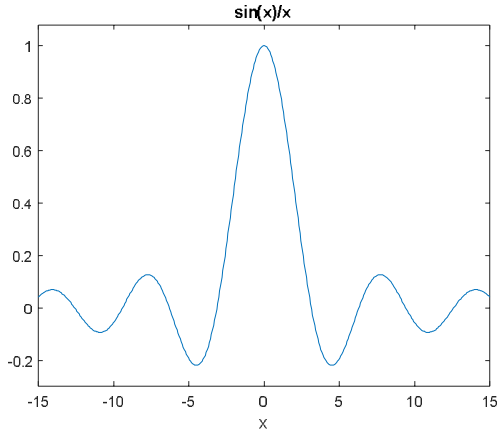


MatLab

رسم نمودار – تابع ezplot

با استفاده از این دستور می‌توان یک تابع ریاضی و محدوده محور افقی را از ورودی گرفته و نمودار را در بازه خواسته شده رسم کرد.

```
s = 'sin(x)/x';
ezplot(s,[-15 15])
```



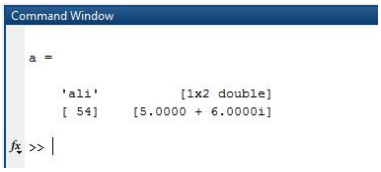
MatLab

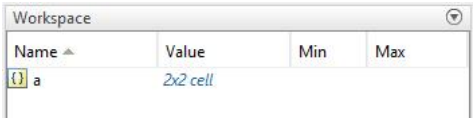
آرایه‌های سلولی

در آرایه‌های سلولی و ساختمانی می‌توان به جای درایه‌های آن انواع داده‌ها را مانند عدد، حرف، عدد مختلط و حتی ماتریس قرار داد.

برای ایجاد آرایه‌های سلولی از { } استفاده می‌شود.

```
a = {'ali',[1,2];54,5+6*1i}
```






```
b = a(1,2)
c = a{1,2}
```

در حالت اول که از پرانتز استفاده شده است، اطلاعات درون آن سلول نمایش داده نمی‌شود اما اگر از آکولاد استفاده کنیم، اطلاعات درون آن نیز نمایش داده می‌شود.

```
b =
    [1x2 double]
c =
     1     2
```

MatLab

آرایه‌های سلولی

برای نشان دادن اطلاعات درون سلول از تابع celldisp استفاده می‌شود.

برای ایجاد یک آرایه‌ی سلولی خالی، از تابع cell استفاده می‌شود.

```
a = cell(3,4)
```

دستور فوق یک آرایه‌ی سلولی سه در چهار خالی ایجاد می‌نماید و آن را در متغیر a ذخیره می‌کند.

در آرایه‌های سلولی برای مقداردهی از آکولاد استفاده می‌شود.

```
a{1,1}= [2 3 5;7 8 1];
a{1,2} = 'ali';
```

قوانین الحاق آرایه‌ها و دستور reshape در آرایه‌های سلولی نیز صادق است.

MatLab

مثال 24

یک سلول دو در دو ایجاد نمایید که درایه‌ی چهارم آن یک ماتریس چهار در چهار رندوم باشد. سپس ستون سوم این ماتریس را که درون درایه‌ی چهارم سلول قرار گرفته، داخل درایه‌ی اول آرایه‌ی سلولی قرار دهید.

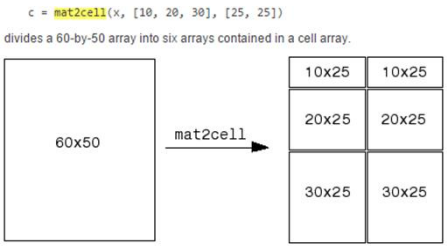
MatLab

تبدیل سلول به ماتریس و برعکس

برای تبدیل ماتریس به سلول از دستور mat2cell استفاده می‌شود.

```
x = [1 2;3 4];
y = mat2cell(x,[1 1],[1 1])
```

```
y =
    [1]    [2]
    [3]    [4]
```



برای تبدیل سلول به ماتریس از دستور cell2mat استفاده می‌شود.

```
C = {[1],[2 3 4]; [5; 9], [6 7 8; 10 11 12]}
A = cell2mat(C)
```

```
C =
    [          1]    [1x3 double]
    [2x1 double]    [2x3 double]
```

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


MatLab

ساختار

ساختارها شبیه آرایه‌های سلولی هستند با این تفاوت که هر درایه‌ی آن‌ها از چند field تشکیل شده است.

درایه 1	a	b	c
درایه 2	a	b	c

درون هر field می‌توان انواع داده به صورت ماتریس، کاراکتر، سلول و ... قرار داد. به طور مثال شکل بالا یک ساختار با دو درایه و سه field را نشان می‌دهد.

نام ساختار (شماره درایه) . نام فیلد =

```
a(2).c = 'ali';
```

```
1    2
3    4
```

درایه دوم از ساختار a و فیلد c را مقداردهی کرده‌ایم.

MatLab

مثال 25

یک ساختار با یک درایه و دو فیلد بسازید. در فیلد اول یک ماتریس رندوم قرار دهید و در فیلد دوم، سطر دوم ماتریس فیلد اول را قرار دهید.

MatLab

ساختار

با استفاده از دستور deal می‌توان یک ساختار ایجاد کرد.

```
[test(1:3).a] = deal(54,34,[]);
[test(1:3).b] = deal('ali',[1 2],58);
[test(1:3).c] = deal('amir',[],[]);
```

در اینجا یک ساختار یک در سه به نام test وجود دارد که دارای سه فیلد a، b و c است.

برای تشخیص فیلدهای یک ساختار از دستور b = fieldnames(a) استفاده می‌شود.

برای حذف یک فیلد خاص از ساختار از دستور d = rmfield(test,'a') استفاده می‌شود.

MatLab

مثال 26

تابعی بنویسید که مختصات دو نقطه را به عنوان ورودی دریافت کند و معادله خط را در خروجی نمایش دهد.

MatLab

مثال 27

برنامه‌ای بنویسید که یک ماتریس ده در ده به صورت رندوم ایجاد کند که درایه‌های آن اعداد صحیح بین صفر تا ده باشد. سپس یک نمودار میله‌ای رسم کند که روی محور افق آن اعداد صفر تا ده قرار گرفته باشد و محور عمودی آن تعداد هر عدد را نشان دهد.

MatLab

مثال 28

تابع sin(x)/x را در بازه [-4*pi,4*pi] رسم کنید.

MatLab

حضرت مهدی (عج)

برای امر فرج بسیار دعا کنید ؛

که فرج شما نیز به آن پیوسته است

بحارالانوار 90/52