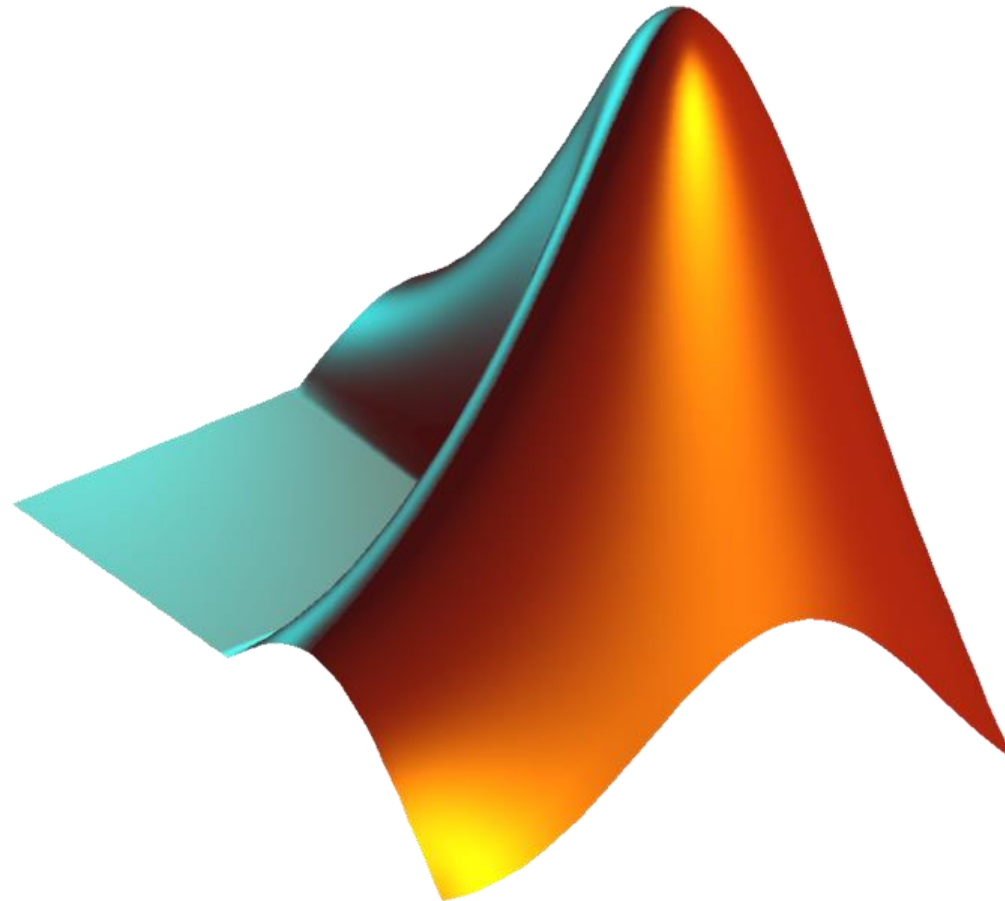




MATLAB COURSE



مدرس دوره : احمد خیراندیش

Session: **3**

آرایه های استاندارد در متلب

ones(a)	ایجاد یک ماتریس a در a که همه ی درایه های این ماتریس عدد یک هستند .
ones(a,b)	ایجاد یک ماتریس a در b که همه ی درایه های این ماتریس عدد یک هستند .
zeros(a)	ایجاد یک ماتریس a در a که همه ی درایه های این ماتریس عدد صفر هستند .
eye(a)	ایجاد یک ماتریس همانی a در a
eye(a,b)	ماتریسی با ویژگی همانی که مربعی نیست .
rand	تولید یک عدد تصادفی بین صفر و یک .
rand(a)	ایجاد یک ماتریس a در a بین صفر و یک .
rand(a,b)	ایجاد یک ماتریس a در b با عناصر تصادفی بین صفر و یک
randperm(a)	این دستور یک بردار سطری n عضوی به صورت جایگشت اعداد بین 1 تا n تولید می کند .
magic(a)	ایجاد ماتریسی a در a که جمع عناصر روی هر ستون و نیز قطرهای اصلی با هم برابر است .

```
a=7;  
ones(a)
```



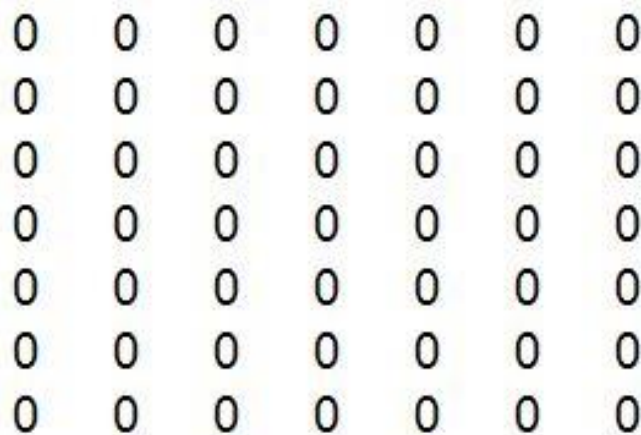
A 7x7 grid of ones, representing a 7x7 matrix of ones.

```
a=7;  
b=4;  
ones(a,b)
```



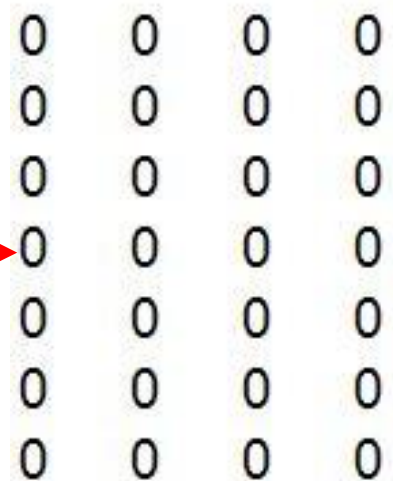
A 7x4 grid of ones, representing a 7x4 matrix of ones.

```
a=7;  
zeros(a)
```




A 7x7 grid of zeros, representing a 7x7 matrix of zeros.

```
a=7;  
b=4;  
zeros(a,b)
```




A 7x4 grid of zeros, representing a 7x4 matrix of zeros.

a=7;
eye(a)




1	0	0	0	0	0	0
0	1	0	0	0	0	0
0	0	1	0	0	0	0
0	0	0	1	0	0	0
0	0	0	0	1	0	0
0	0	0	0	0	1	0
0	0	0	0	0	0	1

a=7;
b=4;
eye(a,b)




1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1
0	0	0	0
0	0	0	0
0	0	0	0

a=7;
magic(a)



30	39	48	1	10	19	28
38	47	7	9	18	27	29
46	6	8	17	26	35	37
5	14	16	25	34	36	45
13	15	24	33	42	44	4
21	23	32	41	43	3	12
22	31	40	49	2	11	20

a=3;
zeros(a,b)



8	1	6
3	5	7
4	9	2

توابع تولید آرایه ها در متلب

length(A)	طول آرایه ی A را برمی گرداند .
size(A)	یک بردار شامل دو عنصر $[m,n]$ را برمی گرداند که m و n ابعاد آرایه ی A می باشند .
reshape(A,m,n)	این دستور برای چینش مجدد عناصر یک آرایه استفاده می شود . این دستور A ، را به m سطر در n ستون تغییر می دهد .
diag(V)	V یک بردار است ، این دستور یک ماتریس قطری ایجاد می کند که المان های روی قطر اصلی آن عناصر بردار V است.
diag(A)	این دستور یک بردار ستونی از عناصر شامل عناصر روی قطر ماتریس A تولید می کند .

```
A=[1 2 3 54 11 33 66 99]
```

```
length(A)
```

8

```
A=[1 2 3;  
54 11 33;  
66 99 25;  
1 -8 -9 ]
```

```
size(A)
```

4 * 3

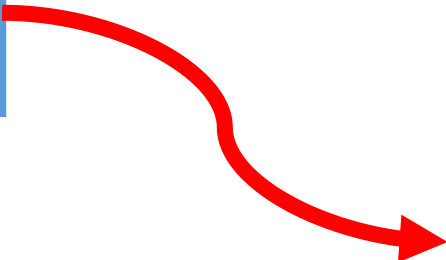
```
A=[1 2 3;  
54 11 33]
```

```
reshape(A,3,2)
```

1	11
54	3
2	33

`V=[1 2 3 54 11 33 66 99]`

`diag(V)`



1	0	0	0	0	0	0	0
0	2	0	0	0	0	0	0
0	0	3	0	0	0	0	0
0	0	0	54	0	0	0	0
0	0	0	0	11	0	0	0
0	0	0	0	0	33	0	0
0	0	0	0	0	0	66	0
0	0	0	0	0	0	0	99


`A=[1 2 3;
54 11 33;
66 99 25;
]`

`diag(A)`



`A=`

1	2	3
54	11	33
66	99	25



1
11
25

در این حالت باید آرایه ها دارای ابعاد مشابهی باشند (دارای سطر و ستون مشابه باشند).

جمع و تفریق آرایه ها

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}$$

$$B = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \end{bmatrix}$$

$$A + B = \begin{bmatrix} a_{11} + b_{11} & a_{12} + b_{12} & a_{13} + b_{13} \\ a_{21} + b_{21} & a_{22} + b_{22} & a_{23} + b_{23} \end{bmatrix}$$

اگر A و B ماتریس باشند ، در این حالت خروجی شامل یک بردار
سطری بوده که حاصل ضرب نظیر به نظیر عناصر ستون های
ماتریس A و B و سپس جمع عناصر ستون های ماتریس حاصل
است .
برای اینکار از دستور **dot** استفاده می کنیم .

حاصل ضرب نقطه ای

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

$$\text{dot}(A,B) = (a_{11} * b_{11}) + (a_{21} * b_{21}) \quad (a_{12} * b_{12}) + (a_{22} * b_{22})$$

ماتریس همانی

ماتریس مربعی است که همه ی عناصر روی قطر اصلی آن یک بوده و سایر عناصر ماتریس، صفر می باشند . خصوصیت این ماتریس بصورت زیر است :

$$IA = AI = A$$

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$I = \begin{bmatrix} 1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & 1 \end{bmatrix}$$

$$AI = IA = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

ماتریس جدید از ماتریس قبلی بوده که جای سطر ها و ستون های آن عوض شده است و در متلب برای ماتریس A ، آن را به صورت A' نشان می دهند .

ترانپوز ماتریس

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$A' = \begin{bmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \end{bmatrix}$$

mean(A)	این دستور میانگین مقادیر عناصر بردار A را بر می گرداند .
C=max(A)	اگر A یک بردار باشد ، C بزرگترین عنصر ماتریس A است و اگر A یک ماتریس است ، C یک بردار سطری شامل بزرگترین عناصر هر ستون است .
[d,n]=max(A)	اگر A یک بردار باشد ، d بزرگترین عنصر A و n شماره ی جایگاه مقدار ماکزیمم است .
C=min(A)	مشابه دستور max ولی برای کوچکترین
[d,n]=min(A)	اگر A یک بردار باشد ، d کوچکترین عنصر A و n شماره ی جایگاه مقدار کوچکترین است .
sum(A)	اگر A یک بردار باشد ، مجموع عناصر آن بردار را برمی گرداند .
sort(A)	اگر A یک بردار باشد ، المان های یک بردار را به صورت صعودی از پایین به بالا مرتب می کند .
median(A)	اگر A یک بردار باشد ، این دستور مقدار میانه را برمی گرداند .
std(A)	اگر A یک بردار باشد ، این دستور مقدار انحراف معیار عناصر بردار را برمی گرداند .

```
A=[1 2 3 54 11 33 66 99]
```

```
mean(A)
```

33.6250

```
A=[1 2 3 54 11 33 66 99]
```

```
max(A)
```

99

```
A=[1 2 3;  
54 11 33;  
66 99 25;  
]
```

```
max(A)
```

A=

	1	2	3
1	54	11	33
2	66	99	25

66 99 33

A=[1 2 3 54 11 33 66 99]

min(A)

1

A=[1 2 3;
54 11 33;
66 99 25;
]

min(A)

A=

1	2	3
54	11	33
66	99	25

1 2 3


```
A=[1 2 3 54 11 33 66 99]
```

```
sum(A)
```

269

```
A=[1 2 3;  
54 11 33;  
66 99 25;  
]
```

```
sum(A)
```

A=

1	2	3
54	11	33
66	99	25

121 112 61

```
A=[1 2 3 54 11 33 66 99]  
sort(A)
```

1 2 3 11 33 54 66 99

```
A=[1 2 3;  
54 11 33;  
66 99 25;  
]  
sort(A)
```

1 2 3
A= 54 11 33
66 99 25

1 2 3
54 11 25
66 99 33

مفاهیم اولیه کار با آرایه ها :

A=[1 2 3 54 11 33 66 99]
median(A)

22

A=[1 2 3 54 11 33 66]
median(A)

11

A=[1 2 3;
54 11 33;
66 99 25;
]
median(A)

پس از مرتب کردن هر ستون به صورت صعودی

1 2 3
A= 54 11 33
66 99 25

54 11 25

```
A=[1 2 3 54 11 33 66 99]  
std(A)
```

36.3551

```
A=[1 2 3;  
54 11 33;  
66 99 25;  
]  
std(A)
```

پس از مرتب کردن هر ستون به صورت صعودی

	1	2	3	
A=	54	11	33	→ 34.5881
	66	99	25	53.5942
				15.5349

det(A)	دترمینان ماتریس مربعی A را محاسبه می کند .
cross(a,b)	این دستور ، حاصل ضرب متقاطع دو بردار a و b را محاسبه می کند ، دو بردار فقط باید شامل ۳ عنصر باشند .
inv(A)	این دستور معکوس یک ماتریس مربعی را محاسبه می کند .
dot(a,b)	حاصل جمع ضرب نقطه ای دو بردار a و b را محاسبه می کند .

```
A=[1 2 3;  
54 11 33;  
66 99 25;  
]  
det(A)
```

A=

1	2	3
54	11	33
66	99	25

12524

```
A=[ -5 3;  
54 -11]  
det(A)
```

A=

-5	3
54	-11

-107

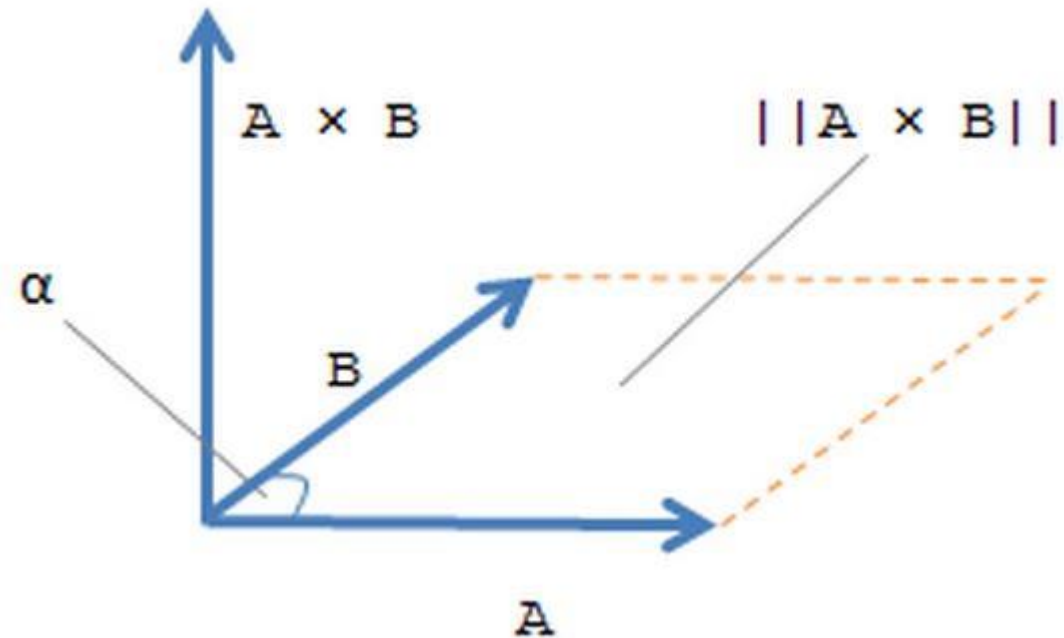
$$A = a_1 \hat{i} + a_2 \hat{j} + a_3 \hat{k}$$

$$B = b_1 \hat{i} + b_2 \hat{j} + b_3 \hat{k}$$

$$C = A \times B = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{vmatrix}$$

$$= (a_2 b_3 - a_3 b_2) \hat{i} + (a_3 b_1 - a_1 b_3) \hat{j} + (a_1 b_2 - a_2 b_1) \hat{k}$$

$$\|A \times B\| = \|A\| \|B\| \sin \alpha$$



$A=[1\ 2\ 3]$

$B=[5\ 6\ 9]$

$\text{cross}(A,B)$



0

6

-4

$A=[1\ 2\ 3\ 5]$

$B=[5\ 6\ 9\ -6]$

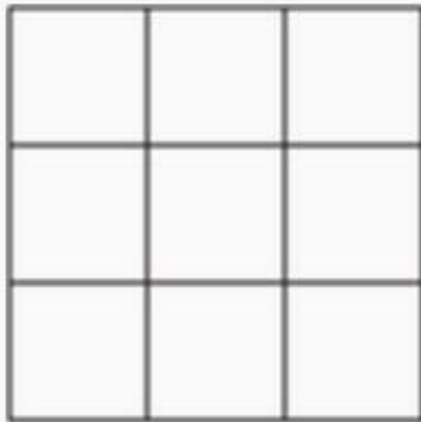
$\text{cross}(A,B)$



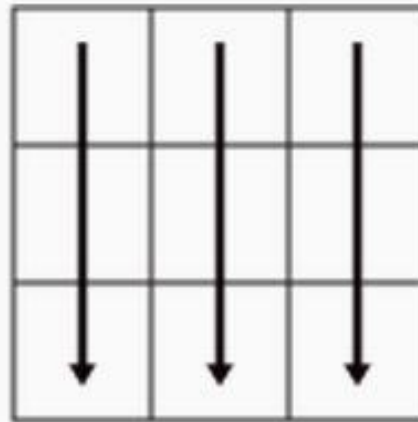
?

! ERROR !

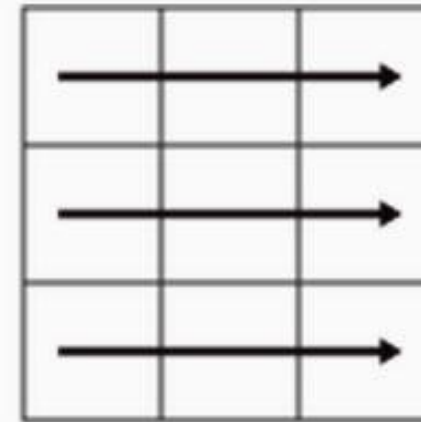
- `cross(A,B,1)` treats the columns of A and B as vectors and returns the cross products of corresponding columns.
- `cross(A,B,2)` treats the rows of A and B as vectors and returns the cross products of corresponding rows.



A



`cross(A,B,1)`



`cross(A,B,2)`

$A = \begin{bmatrix} 1 & 2 & 3 \\ 6 & 5 & -1 \\ 1 & -6 & -22 \end{bmatrix}$

$B = \begin{bmatrix} -1 & -12 & 30 \\ -3 & 5 & 5 \\ 1 & -1 & 0 \end{bmatrix}$

$\begin{bmatrix} 9 & 25 & 110 \\ -2 & 74 & -660 \\ 3 & 70 & 45 \end{bmatrix}$

$\begin{bmatrix} 96 & -33 & -10 \\ 30 & -27 & 45 \\ -22 & -22 & 5 \end{bmatrix}$

```
A=[1 2 3;  
2 -9 -8;  
1 22 15 ]  
inv(A)
```



```
0.3306    0.2903    0.0887  
-0.3065    0.0968    0.1129  
0.4274   -0.1613   -0.1048
```

```
A=[1 2 3;  
2 -9 -8;  
1 22 15;  
1 2 9 ]  
inv(A)
```



?

! ERROR !

تولید اعداد تصادفی



در بسیاری از کاربردهای مهندسی ، تولید اعداد تصادفی نقش فراوانی دارند در نرم افزار متلب برای این کار از دو دستور **rand** و **randn** جهت تخصیص اعداد تصادفی به متغیرها استفاده می شود. با استفاده از این دستورات می توان یک اسکالر ، یک بردار یا یک ماتریس تصادفی تولید کرد .

آرایه های استاندارد در متلب

rand	این دستور ، تنها یک عدد تصادفی بین صفر و یک تولید می کند .
rand(1,n)	این دستور یک بردار سطری n عضوی از اعداد تصادفی در بازه $[0,1]$ تولید می کند .
rand(n)	این دستور یک ماتریس $n*n$ با اعداد تصادفی در بازه $[0,1]$ ، تولید می کند .
rand(m,n)	این دستور یک ماتریس $m*n$ با اعداد تصادفی در بازه $[0,1]$ تولید می کند .
randperm(n)	این دستور یک بردار سطری n عضوی به صورت جایگشت اعداد بین 1 تا n تولید می کند .

rand



ans = 0.8147 ans = 0.9058 ...

rand(1,4)



ans =
0.1270 0.9134 0.6324 0.0975

rand(1,4)



0.2785	0.1576	0.8003	0.7922
0.5469	0.9706	0.1419	0.9595
0.9575	0.9572	0.4218	0.6557
0.9649	0.4854	0.9157	0.0357

ans =

Rand(2,4)

0.8491	0.6787	0.7431	0.6555
0.9340	0.7577	0.3922	0.1712

randperm(10)

2 10 4 5 3 8 7 1 6 9

5 2 1 7 6 8 9 10 3 4

دستور `randn`

با استفاده از این دستور مقادیر گوسی ، با میانگین صفر و واریانس ۱ در صورتی که توزیع نرمال تعیین شده باشد تولید می کند.

دستور `randn(n)`

این دستور یک ماتریس $n \times n$ شامل اعداد تصادفی گوسی (نرمال) ایجاد می کند . که میانگین آن صفر و انحراف معیار آن یک است .

دستور `randn(n,m)`

این دستور یک ماتریس $n \times m$ شامل اعداد تصادفی گوسی (نرمال) ایجاد می کند . که میانگین آن صفر و انحراف معیار آن یک است .

عملیات عنصر به عنصر

عملگرهای ماتریسی		عملگرهای آرایه ای	
+	جمع	+	جمع
-	تفریق	-	تفریق
.	ضرب	.*	ضرب آرایه ای نقطه ای
.	توان	.^	توان آرایه ای نقطه ای
/	تقسیم راست	./	تقسیم راست آرایه ای نقطه ای
\	تقسیم چپ	.\	تقسیم چپ آرایه ای نقطه ای

چند جمله ای ها

یک چند جمله ای ، تابعی از یک متغیر می باشد که به شکل زیر نشان داده می شود :

$$f(x) = a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \dots + a_{n-1}x^1 + a_n$$

در نرم افزار متلب از یک بردار ، برای نمایش چند جمله ای استفاده می شود ، برای اینکار کافی است ضرایب چند جمله ای را از بالاترین توان به پایین ترین توان در یک بردار وارد کنیم .

$$x=[a_0 \quad a_1 \quad a_2 \quad \dots \quad a_n]$$

مفاهیم اولیه کار با چندجمله ای ها:

دستور
 $\text{conv}(a,b)$

این دستور ضرب دو بردار ضرایب a و b را محاسبه می کند .
 a و b می تواند دو بردار با ابعاد غیر یکسان باشد .

دستور
 $[q,r]=\text{deconv}(a,b)$

از این دستور برای تقسیم دو چند جمله ای استفاده می کنیم . اولین بردار که به ما برمی گرداند ، بردار ضرایب خارج قسمت و دومین بردار ضرایب باقیمانده چند جمله ای را در بردارد .

$q \rightarrow$ بردار ضرایب خارج قسمت

$r \rightarrow$ بردار ضرایب باقیمانده

```
a=[1 5 7 6]  
B=[5 2 6]  
conv(a,b)
```

$$f(a) = 1x^3 + 5x^2 + 7x^1 + 6$$

$$f(b) = 5x^2 + 2x^1 + 6$$

```
>> a=[1 5 7 6];  
>> b=[5 2 6];  
>> conv(a,b)
```

ans =

5 27 51 74 54 36

```
a=[1 5 7 6]  
B=[5 2 6]  
[q,r]=deconv(a,b)
```

```
>> [q,r]=deconv(a,b)
```

q =

0.2000 0.9200

r =

0 0 3.9600 0.4800

برای تعیین ریشه ها از این دستور استفاده می کنیم ، این دستور ریشه های یک چند جمله ای را که ضرایب آن در بردار **a** ذخیره شده است را محاسبه می کند .
- این دستور ، بردار ستونی شامل ریشه های چندجمله ای را باز می گرداند .
-تعداد ریشه ها برابر درجه ی چند جمله ای است .

دستور
roots(a)

a=[1 5 7 6]
roots(a)

```
>> roots(a)
```

```
ans =
```

```
-3.4856 + 0.0000i
```

```
-0.7572 + 1.0715i
```

```
-0.7572 - 1.0715i
```