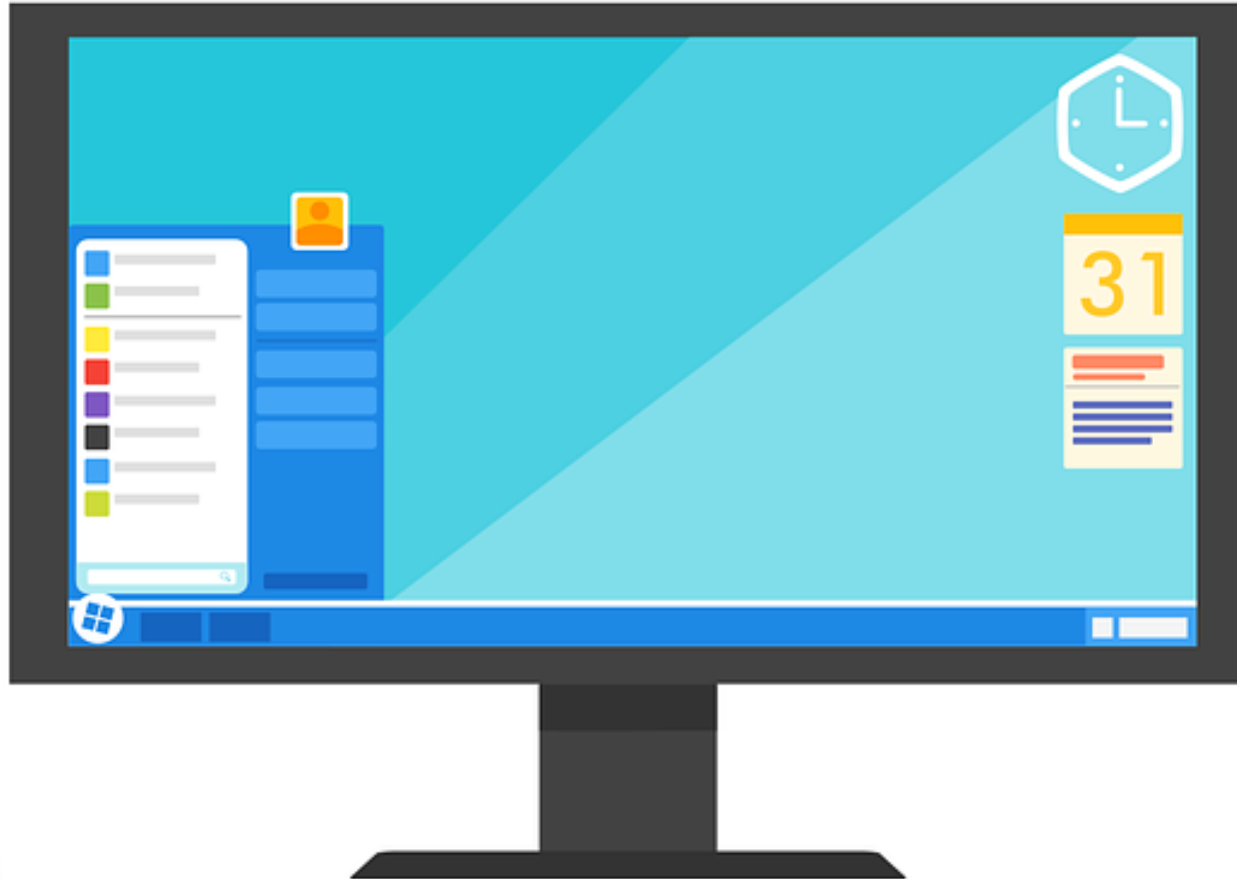


بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

برنامه نویسی کامپیوتر با محوریت آموزش نرم افزار MATLAB



مدرس: مجتبیٰ قربانے | کارشناسے ارشد مهندسے برق

ویرایش پنجم - پاییز ۹۸



دانشگاه تربت حیدریه

آموزش نرم افزار MATLAB

فصل سوم ...

متغیرها در MATLAB

www.ieeei.blog.ir

اصل بزرگ در برنامه نویسی :

- در برنامه نویسی باید دقت نمود که تا وقتی یک متغیر را معرفی نکرده ایم حق استفاده کردن از آن را نداریم.
- در متلب نیز این اصل را باید همواره رعایت نمود.



قواعد نام گذاری متغیرها در متلب :

- در این نرم افزار نام گذاری متغیرها باید مطابق قوانین زیر باشد :

نام گذاری باید یک تکه باشد.

حروف بزرگ و کوچک با هم تفاوت دارد.

نام حتما باید با یک حرف آغاز شود.

در نام گذاری فقط می توان از (A-Z) و (a-z) و (0-9) و (-) استفاده نمود.

نمی توان از کلمات کلیدی و از پیش تعریف شده مثل (if) استفاده نمود.



متغیرهای خاص:

- متغیرهایی در متلب به صورت از پیش تعریف شده هستند و لذا می توان از آنها بدون اینکه ابتدا تعریف کنیم، استفاده نمود. در جدول زیر برخی از این متغیرها نشان داده شده است.

نام متغیر پیش فرض می باشد.	ans
عدد پی (۳,۱۴...)	pi
عدد مختلط	i j
کوچکترین عدد که از اختلاف دو عدد حاصل شده است. (2.22e-16)	eps
نشان دهنده بی نهایت (infinity)	Inf inf
عددی وجود ندارد.	Nan nan
تاریخ فعلی کامپیوتر را به صورت رشته نمایش می دهد.	date



مثال:



```
>> 2+3
```

```
ans =
```

```
5
```

1

```
>> a=2+3
```

```
a =
```

```
5
```

2

```
>> b=1/0
```

```
b =
```

```
Inf
```

3

```
>> a
```

```
a =
```

```
5
```

4

```
>> a=a+5;
```

```
>>
```

5

```
>> a
```

```
a =
```

```
10
```

6

```
>> c
```

```
Undefined  
function or  
variable 'c'.
```

7

```
>> c=a+2
```

```
c =
```

```
12
```

8

```
>> pi
```

```
ans =
```

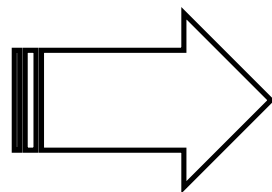
```
3.1416
```

9

آرایه ها در متلب:

- متلب تمام متغیرها را به صورت یک ماتریس در نظر میگیرد.
- برای مثال یک عدد اسکالر از دید متلب یعنی یک ماتریس یک در یک.
- چون همه ی متغیرها را به دید ماتریس می نگرد لذا باید قواعد محاسبات ماتریسی را به خوبی فراگرفت.

$$A = 3$$



$$A = [3]$$



مثال :

```
>> x=[1234]
```

```
x =
```

```
1234
```

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

```
y =
```

```
1 2 3 4
```

```
>> z=[1,2,3,4]
```

```
z =
```

```
1 2 3 4
```

```
>> p=[1;2;3;4]
```

```
p =
```

```
1
```

```
2
```

```
3
```

```
4
```

```
>> q=[1,2;3,4]
```

```
q =
```

```
1 2
```

```
3 4
```

```
>> w=[1,2,3;4]
```

Error using **vertcat**

Dimensions of matrices being concatenated are not consistent.



آرایه ها در متلب:

- ساده ترین روش برای معرفی یک ماتریس به متلب، تایپ آن متغیر می باشد؛ حال اگر ابعاد این ماتریس بسیار بزرگ باشد تایپ مستقیم تمام اعداد کاری عاقلانه ای نخواهد بود.
- برای معرفی آرایه های بزرگ در متلب دستورات مختلفی وجود دارد.

```
>> a=ones(4)
```

```
a =
```

```

1     1     1     1
1     1     1     1
1     1     1     1
1     1     1     1
```

```
>> b=ones(3,5)
```

```
b =
```

```

1     1     1     1     1
1     1     1     1     1
1     1     1     1     1
```



آرایه ها در متلب:

```
>> a=linspace(0,2,5)
a =
    0    0.5000    1.0000    1.5000    2.0000
>> b=linspace(0,2,7)
b =
    0    0.3333    0.6667    1.0000    1.3333    1.6667    2.0000
```

linspace

Generate linearly spaced vectors

Syntax

```
y = linspace(a,b)
y = linspace(a,b,n)
```

Description

The `linspace` function generates linearly spaced vectors. It is similar to the colon operator `:`, but gives direct control over the number of points.

`y = linspace(a,b)` generates a row vector `y` of 100 points linearly spaced between and including `a` and `b`.

`y = linspace(a,b,n)` generates a row vector `y` of `n` points linearly spaced between and including `a` and `b`. For `n = 1`, `linspace` returns `b`.

Examples

Create a vector of 100 linearly spaced numbers from 1 to 500:

```
A = linspace(1,500);
```

Create a vector of 12 linearly spaced numbers from 1 to 36:

```
A = linspace(1,36,12);
```

See Also

[colon operator](#) | [logspace](#)



آرایه ها در متلب:

- باتوجه به کاربرد وسیع ماتریس های با ویژگی طول یکسان، متلب یک دستور ساده تر برای ایجاد این آرایه ها در نظر گرفته است.

```
>> A=1:1:5
```

```
A =
```

```
1      2      3      4      5
```

```
>> B=1:1.1:5
```

```
B =
```

```
1.0000    2.1000    3.2000    4.3000
```

```
>> C=21:-3:2
```

```
C =
```

```
21    18    15    12     9     6     3
```

```
>> D=2:2*pi
```

```
D =
```

```
2      3      4      5      6
```



آرایه ها در متلب:

- برای دسترسی به درایه های ماتریس کافی است تا اندیس آن درایه را بنویسیم.

```
>> A=[13 17 14;11 6 73;23 53 91]
A =
    13    17    14
    11     6    73
    23    53    91

>> A(2,3)
ans =
    73

>> A(3,2)
ans =
    53

>> B=0:pi/2:2*pi
B =
         0    1.5708    3.1416    4.7124    6.2832

>> B(3)
ans =
    3.1416

>> B(1,3)
ans =
    3.1416
```



آرایه ها در متلب:

- می توان به صورت مستقیم نیز مقدار یک درایه را تغییر داد.

```
>> A
A =
    13    17    14
    11     6    73
    23    53    91

>> B
B =
         0    1.5708    3.1416    4.7124    6.2832

>> B(3)=0
B =
         0    1.5708         0    4.7124    6.2832

>> A(2,2)=4
A =
    13    17    14
    11     4    73
    23    53    91
```



آرایه ها در متلب:

- با تهی قرار دادن مقدار یک درایه می توان آن را حذف نمود.

```
>> A
A =
    13    17    14
    11     4    73
    23    53    91

>> B
B =
         0    1.5708         0    4.7124    6.2832

>> B(4)=[]
B =
         0    1.5708         0    6.2832

>> B(4)=[]
B =
         0    1.5708         0

>> A(2,3)=[]
Subscripted assignment dimension mismatch.
```



مثال:

۴- اگر ماتریس A به صورت ذیل باشد با تایپ دستورات زیر متلب چه چیزی را نمایش خواهد داد ؟ (2)

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \\ 16 & 17 & 18 & 19 & 20 \\ 21 & 22 & 23 & 24 & 25 \end{bmatrix}$$

1: A(end, 3) = ?

2: A(2:5, 2) = ?

3: A(:, :) = ?

4: A(1:2:5, 1:5) = ?

نمونه سوال امتحانی

```
>> A=[1:5;6:10;11:15;16:20;21:25]
```

A =

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25



مثال :

```
>> A(1,4)
ans =
     4
>> A(1:4)
ans =
     1     6    11    16
>> A([1 7 16 24])
ans =
     1     7     4    20
>> A(1:3,2:4)
ans =
     2     3     4
     7     8     9
    12    13    14
>> A(1:3:15)
ans =
     1    16     7    22    13
>> A(end,end)
ans =
    25
```

1

```
>> A(2,:)
ans =
     6     7     8     9    10
>> A(end,:)
ans =
    21    22    23    24    25
>> A(1:2:5,1:2:5)
ans =
     1     3     5
    11    13    15
    21    23    25
>> A(:, :)
ans =
     1     2     3     4     5
     6     7     8     9    10
    11    12    13    14    15
    16    17    18    19    20
    21    22    23    24    25
>> A(5:1)
ans =
Empty matrix: 1-by-0
```

2



برخی دستورات مفید :

- در متلب دستورات مختلفی برای ایجاد ماتریس های دلخواه وجود دارد. برخی از پرکاربردترین این دستورات در جدول زیر معرفی شده است.

ones(n,m)	ماتریس n در m با درایه های یک
ones(n)	ماتریس n در n با درایه های یک
zeros(n,m)	ماتریس n در m با درایه های صفر
zeros(n)	ماتریس n در n با درایه های صفر
eye(n)	ماتریس همانی با بعد n
rand(n,m)	ماتریس n در m با درایه های تصادفی
rand(n)	ماتریس n در n با درایه های تصادفی
randperm(n)	اعداد 1 تا n به صورت تصادفی در یک بردار سطری
randi([a,b],n,m)	اعداد صحیح بین a و b در یک ماتریس n در m

در فصل بعد کامل معرفی خواهد شد.



مثال :

```
>> a1=ones(2)
a1 =
     1     1
     1     1
>> a2=ones(2,3)
a2 =
     1     1     1
     1     1     1
>> a3=zeros(3)
a3 =
     0     0     0
     0     0     0
     0     0     0
>> a4=zeros(1,4)
a4 =
     0     0     0     0
>> a5=rand(2)
a5 =
    0.4218    0.7922
    0.9157    0.9595
```

1

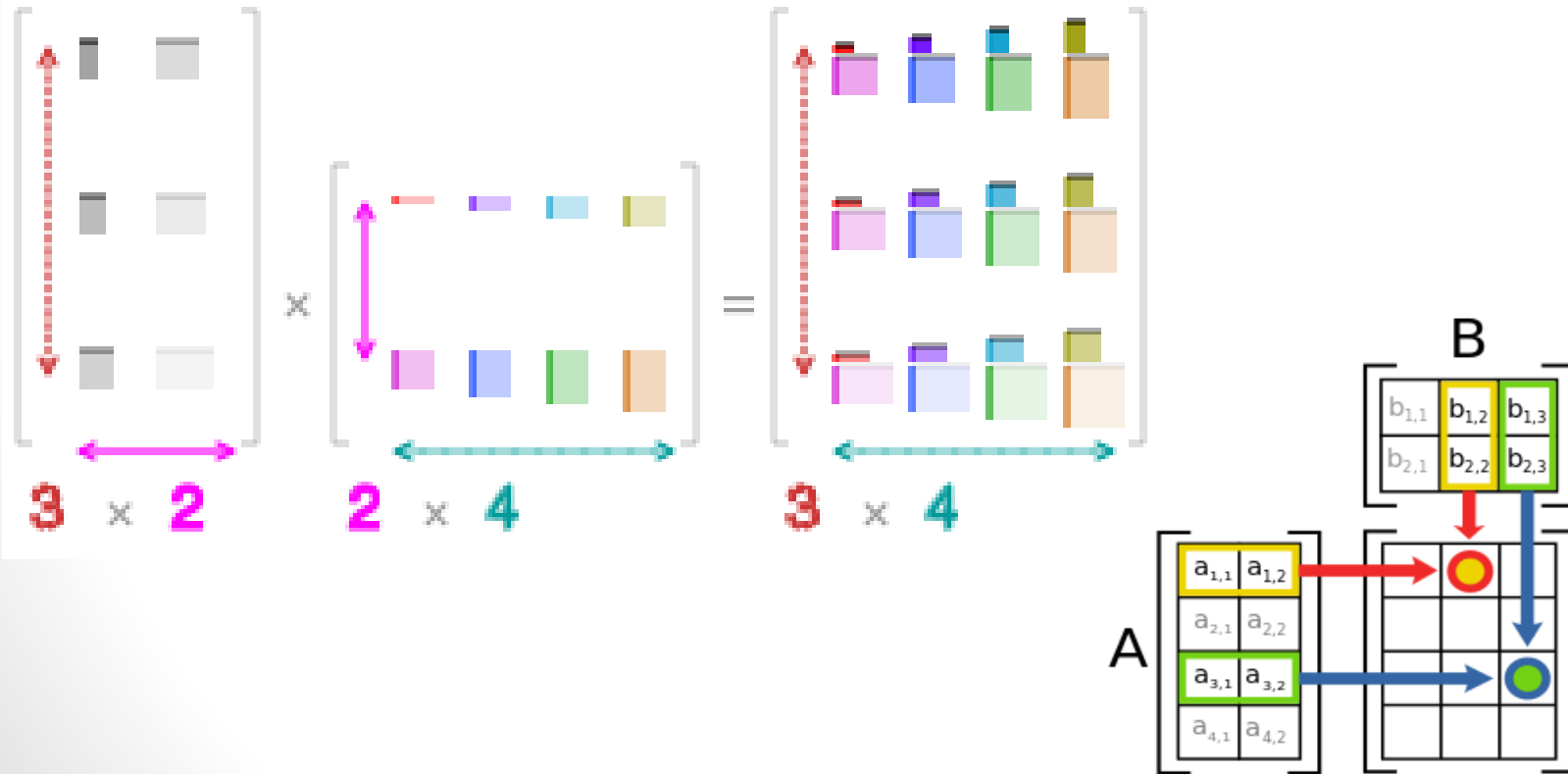
```
>> a6=rand(2,3)
a6 =
    0.1190    0.9597    0.5853
    0.4984    0.3404    0.2238
>> a7=randperm(4)
a7 =
     2     3     4     1
>> a8=randi([-5,5],3,2)
a8 =
     4    -4
     5    -4
     1    -3
>> a9=eye(3)
a9 =
     1     0     0
     0     1     0
     0     0     1
```

2



عملیات ماتریسی :

- در ماتریس های هنگام استفاده از عملیات باید قواعد حاکم بر ماتریس ها را در نظر گرفت. یعنی برای جمع و تفریق باید ابعاد یکسان و برای ضرب و تقسیم باید ابعاد متناسب در نظر گرفته شود.



محاسبات در متلب :

- با توجه به اینکه در متلب تمام متغیرها به صورت ماتریس در نظر می گیرد، لذا همواره باید قواعد عملیات ماتریسی را در متلب در نظر گرفت.
- در صورتی که خواهان محاسبات غیرماتریسی باشیم، می توان از روش محاسبه درایه به درایه استفاده نمود.



مثال :

```
>> a=[1 2;3 4]
a =
     1     2
     3     4
>> b=[5 6;7 8]
b =
     5     6
     7     8
>> a+b
ans =
     6     8
    10    12
>> a-b
ans =
    -4    -4
    -4    -4
>> a*b
ans =
    19    22
    43    50
```

1

```
>> a.*b
ans =
     5    12
    21    32
>> a/b
ans =
    3.0000   -2.0000
    2.0000   -1.0000
>> a./b
ans =
    0.2000    0.3333
    0.4286    0.5000
>> a^3
ans =
    37    54
    81   118
>> a.^3
ans =
     1     8
    27    64
```

2

```
>> a^b
Error using ^
Inputs must be a scalar and a
square matrix.
To compute elementwise POWER, use
POWER (.^) instead.
>> a.^b
ans =
         1         64
    2187    65536
>> a.+b
    a.+b
    |
Error: Unexpected MATLAB
operator.
>> a.-b
    a.-b
    |
Error: Unexpected MATLAB
operator.
```

3



انواع متغیرها در متلب :

- در متلب متغیرها انواع مختلفی دارند، که هر نوع متغیر باید متناسب با ویژگی های آن مورد پردازش و محاسبات قرار گیرد.

نوع داده	توضیحات	نشان
Double	آرایه های عددی با دقت Double که مرسوم ترین نوع متغیر در MATLAB هستند.	
Single	آرایه های عددی با دقت Single که فضای کمتری نسبت به داده های Double دارند.	
Char	آرایه های کارکتری و رشته ای	
Logical	آرایه هایی که مقدار آن 0 یا 1 است. (درست یا غلط)	
Structure	آرایه های ساختمانی	
Cell	آرایه های سلولی که مولفه های آن آرایه های دیگر را می تواند شامل شود.	



مثال :

```
>> a=eye(4)
```

```
a =
```

```
    1    0    0    0
    0    1    0    0
    0    0    1    0
    0    0    0    1
```

double

```
>> b=single(a)
```

```
b =
```

```
    1    0    0    0
    0    1    0    0
    0    0    1    0
    0    0    0    1
```

single

```
>> c='matlab'
```

```
c =
```

```
matlab
```

```
>> d=1>3
```

```
d =
```

```
0
```

logical

```
>> e.family='ali';
```

```
>> e.phone='0987654321';
```

```
>> e.age=30
```

```
e =
```

```
family: 'ali'
```

```
phone: '0987654321'
```

```
age: 30
```

```
>> f={a,c;d,e}
```

```
f =
```

```
[4x4 double]
```

```
'matlab'
```

```
[          0]
```

```
[1x1 struct]
```

structure

cell



رشته ها :

- خروجی های رشته ای مجموعه ای از کارکترهای الفبایی یا عددی هستند که در بین یک جفت کوتیشن ('') قرار می گیرند. هر کارکتر رشته یک عضو در یک بردار را اشغال می کند. یک رشته در MATLAB از نوع char بوده و هر کارکتر در دو بایت از حافظه ذخیره می شود.

```
>> text='My hero is a martyr'
text =
My hero is a martyr
>> text2='قهرمان من یک شهید است'
text2 =

>> A=text(1:7)
A =
My hero
>> B=text(14:end)
B =
martyr
```



آرایه های رشته ای :

- آرایه های رشته ای، آرایه های هستند که مولفه های آنها رشته های کارکتری می باشند.

```
>> class=char('ali','ahmad','hassan')  
class =  
ali  
ahmad  
hassan  
  
>> class(2,:)   
ans =  
ahmad
```



آرایه های ساختمانی:

- می توان با استفاده از آرایه های ساختمانی انواع داده را فیلدهای مختلف ذخیره نمود.

```
>> student.name='ali alavi';  
>> student.id=987654321;  
>> student.marks=[12 15 18 20];  
>> student  
student =  
    name: 'ali alavi'  
    id: 987654321  
 marks: [12 15 18 20]
```



آرایه های ساختمانی:

- برای اضافه کردن المان دوم به ساختمان به صورت زیر می توان عمل کرد.

```
student(2).name='ahmad ahmadi';
student(2).id=123456789;
student(2).marks=[13 16 19 20];
```

1

```
>> student
```

```
student =
```

```
1x2 struct array with fields:
```

```
    name
```

```
    id
```

```
  marks
```

```
>> student.id
```

```
ans =
```

```
    987654321
```

```
ans =
```

```
    123456789
```

2

```
>> student.name
```

```
ans =
```

```
ali alavi
```

```
ans =
```

```
ahmad ahmadi
```

```
>> student(1).name(1:5)
```

```
ans =
```

```
ali a
```

3

4



آرایه های سلولی:

- آرایه های سلولی یک ظرف حاوی انواع داده ها می باشند. به طور مثال آرایه های عددی، رشته ای، ساختاری، خود سلول ها و ...
- ممکن است یک سلول از آن، حاوی آرایه اعداد حقیقی بوده، سلول دیگر حاوی آرایه ای متشکل از رشته ها و دیگری حاوی آرایه های از اعداد مختلط باشد.
- مشخصه یک سلول { } می باشد.

```
>> c{1,1}=rand(3);
>> c{1,2}=char ('ali','alavi');
>> c{2,1}=pi;
>> c{2,2}=student;
>> c

c =

    [3x3 double]    [2x5 char  ]
    [    3.1416]    [1x2 struct]
```



آرایه های سلولی:

```
>> d={rand(3),char('ahmad','ahmadi');eps,student}
```

```
d =
```

```
    [3x3 double]    [2x6 char  ]  
    [2.2204e-16]    [1x2 struct]
```

```
>> c{1,1}
```

```
ans =
```

```
    0.8147    0.9134    0.2785  
    0.9058    0.6324    0.5469  
    0.1270    0.0975    0.9575
```

```
>> d{1,2}
```

```
ans =
```

```
ahmad
```

```
ahmadi
```



مثال :

۴- اگر ماتریس A به صورت زیر باشد با تایپ دستورات زیر متلب چه چیزی را نمایش خواهد داد ؟ (2)

نمونه سوال امتحانی

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 6 & 7 & 8 & 9 & 10 \\ 11 & 12 & 13 & 14 & 15 \\ 16 & 17 & 18 & 19 & 20 \\ 21 & 22 & 23 & 24 & 25 \end{bmatrix}$$

1: $A(1:1:3, 3:5) = ?$

2: $A(1:A(2)) = ?$

3: $A(1:2:4, 4:-1:1) = ?$

4: $A(A(A(A(3)))) = ?$



منابع مفید :

- ۱- تمرینتمرینتمرین
- ۲- جزوه آموزشی متلب دانشگاه صنعت آب و برق ، سرافراز
- ۳- کاملترین مرجع آموزشی و کاربردی matlab ، مهندس علمداری ، انتشارات نگارنده دانش
- ۴- سایر منابع موجود در اینترنت
- ۵- ...

