



آموزش برنامه نویسی متلب





به نام خدا

تقدیم به هموطنان عزیزم

متلب را با لذت یاد بگیر!



آموزش برنامه نویسی متلب

آموزش برنامه نویسی متلب

موضوع: ماتریس ها در متلب

جلسه: هفتم

مدرس : مدرسین جاواپرو

متلب را ساده، آسان و شیرین بنوشید!!!



این جلسه آموزشی رایگان است، فروش و ویرایش آن ممنوع و حرام می باشد. اما این کتاب را می توانید همین جور که هست در سایت و شبکه اجتماعی خود به اشتراک بگذارید.



آموزش برنامه نویسی متلب

ارتباط با ما:

سایت: www.javapro.ir

ایمیل: RAHMAN.ZARIE92@GMAIL.COM

کانال تلگرام:

[@javalike](https://t.me/javalike)

گروه پرسش و پاسخ برنامه نویسی :

[@javapro_ir](https://t.me/javapro_ir)



آموزش برنامه نویسی متلب

جلسه قبل کار با آرایه ها را شروع کردیم، و گفتیم که آرایه ها همان بردارها یا ماتریس های یک بعدی هستند. بنابراین دستورات گفته شده در جلسه قبل برای ماتریس ها نیز صدق می کند ولی یک سری تفاوت هایی نیز وجود داشته که در یک جلسه جداگانه قصد داریم به آن بپردازیم.

به دو شکل می توان یک ماتریس را مستقیما تعریف نمود:

```
>>m = [1 2 3;4 5 6;7 8 9]
```

```
m =
```

```
1     2     3
4     5     6
7     8     9
```

```
>>m = [1 2 3
```

```
4 5 6
```

```
7 8 9]
```

```
m =
```

```
1     2     3
4     5     6
7     8     9
```

حالت اول با استفاده از ";" و حالت دوم با فشردن Enter. در حالت دوم نیز می توان با ایجاد چند tab درایه های هر ستون را زیر هم قرار داد، این کار صرفا به خوانایی و زیبایی دستور کمک می کند و در هر حالت تفاوتی در نتیجه ایجاد نمی کند:

```
>>m = [1 2 3
```

```
4 5 6
```

```
7 8 9]
```

```
m =
```

```
1     2     3
```



آموزش برنامه نویسی متلب

```
4     5     6
7     8     9
```

بین درایه‌ها نیز می‌توان به جای فاصله از "," استفاده کرد.

ترانهاده‌ی ماتریس m نیز بدین شکل خواهد بود:

```
>>m'
ans =
     1     4     7
     2     5     8
     3     6     9
```

برای جمع، تفریق، ضرب، تقسیم و حتی توان درایه‌ای (نه ماتریسی که در نظریه ماتریس‌ها مطرح می‌شود) از عملگرهای +، -، *، / و ^ به ترتیب استفاده می‌شود.

مثال:

```
>>m = [1,2,3;1,2,3;1,2,3];
>>n = [1,0,0;0,1,0;0,0,1];
>>m,n
m =
     1     2     3
     1     2     3
     1     2     3
n =
     1     0     0
     0     1     0
     0     0     1
>>m .* n
```



آموزش برنامه نویسی متلب

```
ans =
     1     0     0
     0     2     0
     0     0     3
>>m .^ 2
ans =
     1     4     9
     1     4     9
     1     4     9
```

در این حالت باید ابعاد دو ماتریس دقیقاً برابر باشند. اما برای ضرب و تقسیم و توان ماتریسی نیازی به "قبل از عملگر نیست. به شرط درست بودن ابعاد ماتریس‌ها این عملیات قابل انجام می‌باشد.

مثال:

```
>>m = [1 2 3;4 5 6]
m =
     1     2     3
     4     5     6
>>n = [1 1;2 2;3 3]
n =
     1     1
     2     2
     3     3
>>m * n
ans =
    14    14
```



آموزش برنامه نویسی متلب

```
32    32
>>n * m
ans =
     5     7     9
    10    14    18
    15    21    27
>>m = [1,2,3;1,2,3;1,2,3];
>>m * m
ans =
     6    12    18
     6    12    18
     6    12    18
>>m ^ 2
ans =
     6    12    18
     6    12    18
     6    12    18
```

ماتریس وارون و دترمینان یک ماتریس مربعی از طریق $\text{inv}()$ و $\text{det}()$ به دست می‌آیند.

مثال: اگر A و B به صورت زیر تعریف شده باشد، معادله $Ax = B$ را حل نمایید.

```
>>A =
     4     1     4
     1     2     1
     1     4     2
```




آموزش برنامه نویسی متلب

```
>>B =
```

```

2     1     1
1     1     4
3     3     4
```

جواب:

برای حل معادله باید دو طرف را در A^{-1} ضرب نماییم. پس $x = A^{-1}B$. بنابراین ابتدا معکوس A را به دست آورده سپس در B ضرب می‌کنیم. بهتر است قبل از انجام اینکار دترمینان ماتریس A را حساب کرده تا مطمئن شویم که مخالف صفر بوده و در نتیجه A معکوس پذیر است:

```
>>det(A) == 0
```

```
ans =
```

```
0
```

پس داریم:

```
>>inv(A) * B
```

```
ans =
```

```

-1.0000    -1.0000     4.0000
 0.2857     0.4286     2.1429
 1.4286     1.1429    -4.2857
```

- سه ماتریس $ones(m,n)$ ، $zeros(m,n)$ و به ترتیب ماتریس‌های تهی، یک‌ه با ابعاد $m*n$ و صفر با ابعاد $m*n$ هستند که به شکل زیر تعریف می‌گردند:

```
>>ones(2,3)
```

```
ans =
```

```

1     1     1
1     1     1
```

```
>>zeros(3,3)
```



آموزش برنامه نویسی متلب

```
ans =
```

```
0     0     0
0     0     0
0     0     0
```

مطالب گفته شده پیرامون استخراج آرایه در مورد ماتریس‌ها نیز صادق است. ماتریس زیر را در نظر بگیرید:

```
m =
```

```
1     2     3
4     5     6
7     8     9
```

اگر شمارش درایه‌ها را از ستون اول به صورت ستونی انجام دهیم درایه پنجم برابر خواهد بود با:

```
m(4)
```

```
ans =
```

```
2
```

برای دسترسی به درایه‌ها می‌توان اندیس مؤلفه‌های آنرا نیز وارد کرد:

```
m(2,2)
```

```
ans =
```

```
5
```

```
m(end)
```

```
ans =
```

```
9
```

مثال‌های زیر را با دقت بررسی کنید. دستورات گفته شده در جلسه قبل این بار با ماتریس فوق بررسی شده‌اند.

```
>>m(:)
```



آموزش برنامه نویسی متلب

```
ans =  
    1  
    4  
    7  
    2  
    5  
    8  
    3  
    6  
    9  
  
>>m(1:3)  
ans =  
    1    4    7  
  
>>m(1,2:3)  
ans =  
    2    3  
  
>>m(2:3,2:3)  
ans =  
    5    6  
    8    9  
  
>>m([1 3],[1 3])  
ans =  
    1    3  
    7    9
```



آموزش برنامه نویسی متلب

تمرین: در مثال فوق جای سطر دوم و اول را در ماتریس m تعویض کنید.

جواب:

```
>>m([2 1 3],[1 2 3])
```

```
ans =
```

```
4     5     6
1     2     3
7     8     9
```

آرایه اول، مربوط به اندیس سطرها و آرایه دوم اندیس ستون‌هاست.

برای استخراج ستون دوم علاوه بر روش‌های گفته شده بدین ترتیب نیز می‌توان عمل کرد:

```
>>m(:,2)
```

```
ans =
```

```
2
5
8
```

نکته: علامت ":" معادل end:1 می‌باشد.

دستور find در ماتریس‌ها به دوشکل قابل استفاده است. حالت اول مشابه قبل:

```
>>find(m>=5)'
```

```
ans =
```

```
3     5     6     8     9
```

```
>>m(ans)
```

```
ans =
```

```
7     5     8     6     9
```

توجه: از "" برای تبدیل به بردار سطری استفاده شده است.



آموزش برنامه نویسی متلب

ابتدا اندیس‌ها و سپس درایه‌های مربوط به هر اندیس را استخراج کردیم. اما حالت دوم، چنانچه اندیس‌های سطر و ستون مؤلفه‌های مربوطه را بخواهیم بدین شکل عمل می‌کنیم:

```
>> [a b] = find(m>=5)
```

```
a =
```

```
3
```

```
2
```

```
3
```

```
2
```

```
3
```

```
b =
```

```
1
```

```
2
```

```
2
```

```
3
```

```
3
```

-دستور diag:

```
>>diag(m)
```

```
ans =
```

```
1
```

```
5
```

```
9
```

این دستور اگر یک ماتریس را در آرگومان بگیرد عناصر قطر اصلی را برمی‌گرداند. (برای ماتریس‌های مستطیلی نیز امتحان کنید.)



آموزش برنامه نویسی متلب

اما چنانچه یک بردار را در آرگومان بگیرد یک ماتریس قطری (ماتریسی که در آن همه درایه‌ها به جز قطر اصلی صفر هستند) با درایه‌های بردار داده شده برمی‌گرداند:

```
>>diag(ans)
ans =
     1     0     0
     0     5     0
     0     0     9
```

-دستور sum:

مجموع عناصر ستون‌های یک ماتریس را برمی‌گرداند. اگر آرگومان آن یک بردار باشد، مجموع مؤلفه‌های آن بردار را می‌دهد:

```
>>sum(m)
ans =
    12    15    18
>>d = diag(m);
>>sum(d)
ans =
    15
```

واضح است که برای به دست‌آوردن مجموع کل درایه‌های ماتریس می‌توان نوشت:

```
>>sum( sum(m) )
ans =
    45
```

-ماتریس جادویی:



آموزش برنامه نویسی متلب

ماتریسی مربعی است که مجموع سطرها یا ستون‌های آن یکسان خواهد بود. برای مثال برای حالت ۳ در ۳:

```
>>magic(3)
ans =
     8     1     6
     3     5     7
     4     9     2

>>sum(ans)
ans =
    15    15    15
```

مجموع عناصر قطر اصلی این ماتریس نیز برابر با ۱۵ است:

```
>>sum( diag( magic(3) ) )
ans =
    15
```

پیروز و موفق باشید



آموزش برنامه نویسی متلب

سایت آموزشی رایگان جاواپرو

www.JAVAPRO.ir

برنامه نویسی با تجربه شخصی و به زبان خودمونی یاد بگیرید!!!!

بازدید از کانال

بازدید از سایت

هر روز مفاهیم و مثال های جدید به سایت اضافه می شود برای اطلاع از مطالب جدید روی سایت عضو کانال شوید.

دخل و تصرف ، ویرایش و کپی زدن تمامی آموزش های جاواپرو به دور از اخلاق حرفه ای ست و حرام می باشد.