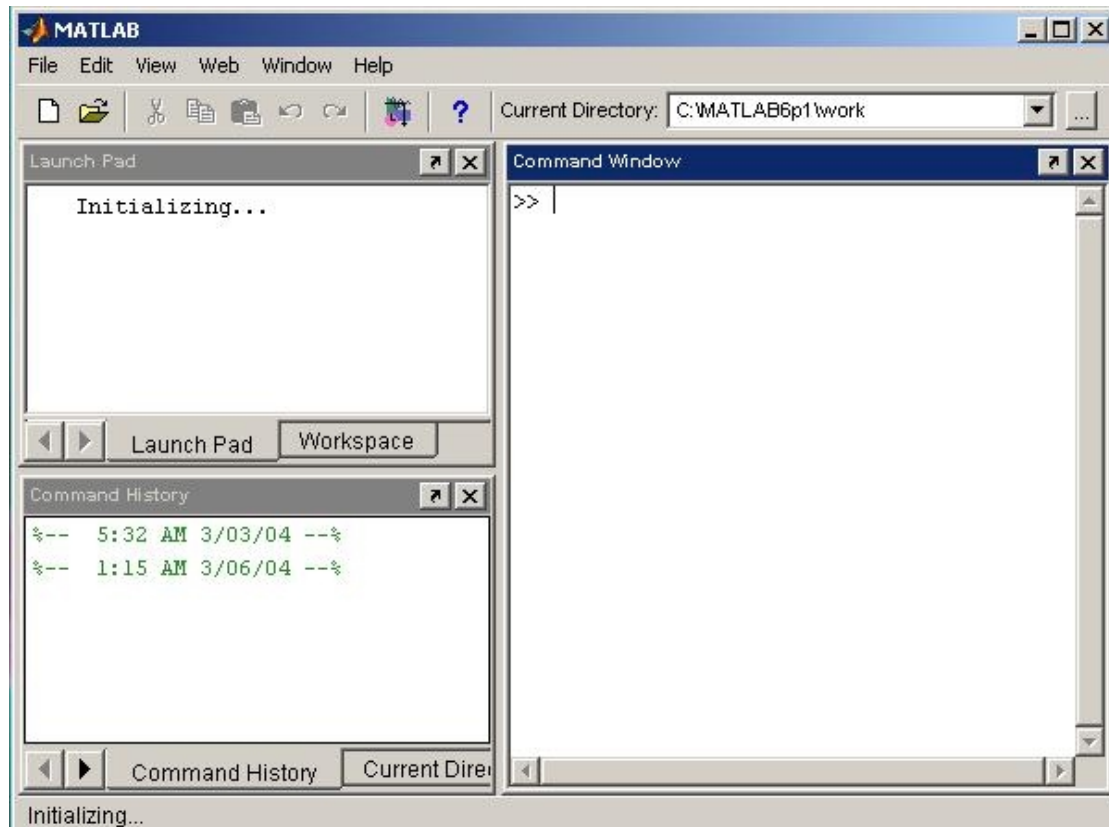


## بسمه تعالی برنامه نویسی MATLAB

اساس کار این نرم افزار بر اساس ماتریسها میباشد.  
ماتریس : آرایه ای از اعداد است به صورت ردیفهای منظم سطری و ستونی .  
به تعداد سطرها بعد اول و به تعداد ستونها بعد دوم گفته میشود .  
اعمال ریاضی بر روی ماتریسها  
جمع و تفریق ماتریسها :  
برای جمع و تفریق ماتریسها باید هر دو ماتریس از یک بعد باشند . در جمع دو  
ماتریس هم بعد اعداد متناظر نظیر به نظیر با هم جمع میشود .  
ضرب ماتریسها :  
برای ضرب دو ماتریس باید تعداد سطرها و تعداد ستونهای ماتریس بعدی با هم  
برابر باشند . برای ضرب دو ماتریس باید به نحوه زیر عمل کرد .  
سطر اول ماتریس اول را در ستون اول ماتریس دوم ضرب کرده سپس سطرهای  
بعدی در ستونهای بعدی به نوبت ضرب میشوند .  
تقسیم دو ماتریس :  
تقسیم یک ماتریس بر یک ماتریس دیگر به گونه ای که در ریاضی با آن آشنا هستیم  
امکان پذیر نیست .  
ضرب یک ماتریس در یک عدد :  
عدد مورد نظر را به طور نظیر به نظیر در کلیه اعداد ضرب میکنیم .  
نکته : باید توجه داشت که در ضرب ماتریسها خاصیت جابجایی امکان پذیر نیست  
یعنی با جابجا شدن دو ماتریس جواب ضرب تغییر خواهد کرد .  
$$A * B \neq B * A$$
  
تعاریف اولیه :  
ماتریس صفر : ماتریسی که کلیه اعضاء آن صفر باشد .  
ماتریس قطبی : ماتریسی که اعداد روی قطر اصلی آن یکسان بوده و بقیه ماتریس  
صفر باشد .  
ماتریس مربع : ماتریسی که تعداد سطرها و ستونهای آن برابر باشد را ماتریس  
مربع گویند .  
ماتریس یکه : ماتریسی که اعضاء روی قطر اصلی آن یک باشند را ماتریس یکه  
گویند .  
ماتریس بالا مثلثی : ماتریسی که اعضاء زیر قطر اصلی آن صفر باشد .  
ماتریس بال مثلثی : ماتریسی که اعضاء بالای قطر اصلی آن صفر باشد .  
قطر اصلی : اعضایی را تشکیل میدهد که تعداد سطر و ستون مساوی داشته باشند  
ماتریس واحد : حاصل ضرب یک ماتریس در معکوس همان ماتریس ماتریس واحد  
را ایجاد میکند .

## آشنایی با منوی MATLAB

در صفحه بعد يك نمونه از منوي مطلب را مشاهده مينماييد كه به اختصار درباره بعضي از گزينه هاي آن توضيح ميدهيم .  
NEW: براي باز كردن يك پوشه جديد و فراخواني پوشه هاي قبلي و نيز جهت ذخيره برنامه ها از اين گزينه ميتوانيم استفاده كنيم .



View: باز زدن اين ايكون و انتخاب گزينه Desktop layout و سپس زير شاخه Default ميتوانيم صفحه اصلي مطلب را ايجاد كنيم .  
CurrentDirectory: در اين قسمت ميتوانيم دايركتوري پيش فرض مطلب را وارد كنيم .  
Workspase: با زدن اين گزينه ميتوانيم متغيرهاي معرفي شده به محيط را ديده و در صورت نياز از آنها استفاده كنيم .  
Command history: براي مشاهده دستورات اعمال شده ميتوان از اين گزينه استفاده كرد .

نحوه ايجاد ماتريس در مطلب :

در خط فرمان حرف و يا حروفي را به عنوان نام ماتريس قرار ميدهيم سپس علامت = را قرار داده كه نشانه انتصاب ماتريس سمت راست به نام مقابل آن ميباشد .

برای وارد کردن سطر میتوانیم رقم اول سطر را وارد کرده سپس یک فاصله داده و رقم دوم همان سطر را وارد میکنیم و به همین نحو کلیه ارقام یک سطر را وارد میکنیم. برای وارد کردن سطر بعدی علامت ( ; ) را قرار میدهیم و سپس سطر بعدی را مانند قبل وارد میکنیم.

مثال:  $A=[1\ 5\ 7; 4\ 7\ 6; 7\ 5\ 6]$

نکته: در محیط مطلب حروف کوچک از حروف بزرگ متمایز بوده و در صورت نامگذاری یک برنامه با حرف بزرگ، آن برنامه با حرف کوچک اجرا و یا فراخوانی نخواهد شد.

ترانزیپوز (ترانهاده) یک ماتریس:

بعد از ترانهاده کردن یک ماتریس جای سطر و ستون عوض میشود و فقط اعضای روی قطر اصلی تغییر نمیکنند و بقیه اعضا جابجا میشوند.

برای معرفی ترانهاده یک ماتریس به برنامه مطلب، بعد از تایپ نام ماتریس علامت (') را قرار میدهیم.

$A=[1\ 2\ 4; 4\ 5\ 2]$

$A'=[1\ 4; 2\ 5; 4\ 2]$

علامت (:)

این علامت را در انتهای هر دستوری قرار دهیم در موقع اجرای برنامه آن دستور را ملاحظه نمیکنیم و فقط نتیجه آن دستور را خواهیم دید.

علامت (:)

اگر بخواهیم یک ماتریس تولید کنیم که از یک عدد شروع و به یک عدد دیگر ختم شود از این گزینه استفاده میکنیم.

مثال: نمونه مقابل ارقام از 1 تا 10 را برای ما به نمایش در می آورد.  $A=1:10$   
حال اگر نمونه بالا را به صورت زیر وارد کنیم اعداد را یکی در میان برای ما نمایش میدهد.  $A=1:2:10$

اگر بخواهیم سطر یا ستون معینی از یک ماتریس را انتخاب کنیم می توانیم از این علامت استفاده کنیم به طور مثال در دستور زیر مقصود برداشتن ارقام از ستون دوم تا چهارم، در سطر دوم می باشد.

$\gg F=A(2,2:3)$

دستور `format long`:

توسط این دستور می توانیم تا 16 رقم اعشار اعداد را مشاهده نماییم.

$\gg \text{format long}$

$\gg A=2.34$

A =

2.3400000000000000

دستور `format short`:

توسط این دستور می توانیم تا 4 رقم اعشار اعداد را مشاهده نماییم.

```
>>format short
```

```
>> A=2.34
```

```
A =
```

```
2.3400
```

دستور `:clc`

اگر صفحه اجرا شلوغ شده باشد و بخواهیم آنرا خلوت کنیم میتوانیم از این دستور استفاده کنیم. به این ترتیب که با وارد کردن این دستور و زدن اینتر کل صفحه پاک می شود.

دستور `:clear`

توسط این دستور می توان یک متغیر مورد نظر و یا کلیه دستورات را از حافظه پاک کرد.

دستور `:rand`

اگر بخواهیم یک ماتریس به طور تصادفی ایجاد کنیم می توان از این دستور استفاده کرد. با قرار دادن یک عدد داخل پرانتز در جلوی این دستور می توان تعداد سطر و ستون آن را تعیین کرد.

```
<< rand(3)
```

```
= ans
```

```
0.4565 0.4860 0.9501
```

```
0.0185 0.8913 0.2311
```

```
0.8214 0.7621 0.6068
```

دستور `:zeros`

اگر بخواهیم یک ماتریس با اعضای صفر ایجاد کنیم از این دستور استفاده می کنیم.

دستور `:ones`

برای ایجاد ماتریسی با اعضای یک، از این دستور استفاده میشود.

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

```
a =
```

```
1 1 1
```

```
1 1 1
```

دستور `:eye`

این دستور یک ماتریس واحد ایجاد می کند که اعضای قطر اصلی یک بوده و بقیه صفر می باشد. می توانیم تعداد سطر و ستون آن را تعیین کنیم.

```
<<eye(3)
```

```
= ans
```

```
0 0 1
```

```
0 1 0
```

```
1 0 0
```

دستور `diag` :

با استفاده از این دستور میتوانیم از یک ماتریس قطر اصلی آن را بیرون کشید .

`D=diag(A)`

در مثال بالا قطر اصلی ماتریس `A` را بیرون کشیده و به صورت یک بردار در `D` ذخیره میکنیم .

```
D=diag(A)
```

```
D=
```

```
1
```

```
5
```

```
>> b=[1 2 3];
```

```
>> a=diag(b)
```

```
a =
```

```
1 0 0
```

```
0 2 0
```

```
0 0 3
```

دستورات ریاضی :

`Sin-cos-tan-sqr-max-min-sum-std`

دستور `sin` :

برای گرفتن سینوس از این دستور استفاده میشود . باید توجه داشت که ورودی این دستور رادیان میباشد . پس باید برای تبدیل به درجه باید در دستور (`pi/180`) ضرب نمود . در این دستور `pi` علامت (`3.14`) میباشد . برای وارد کردن یک زاویه به حالت درجه به مطلب باید به شکل زیر عمل کرد .

`(D M/60 S/3600)*pi/180`

عمل نمود .

```
sin(90)
```

```
= ans
```

```
0.8940
```

دستور cos:

این دستور عیناً مانند دستور بالا بوده با این تفاوت که کسینوس اعداد مورد نظر را به ما می دهد .

دستور max:

توسط این دستور می توانیم بزرگترین رقم يك بردار و یا بزرگترین ارقام هر ستون در يك ماتریس را مشاهده نماییم.

```
A =  
4 2 1  
2 5 4
```

```
>>max(A)  
= ans  
4 5 4
```

دستور min:

عینه مانند بالا است با این تفاوت که کوچکترین ارقام را به ما نشان می دهد .

دستور sum:

توسط این دستور می توان ارقام يك بردار را جمع نمود .  
در ماتریس ها نیز این دستور مجموع ستونها را به صورت بردار برای ما به نمایش در می آورد

```
sum(A)  
= ans  
6 7 5
```

دستور mean:

این دستور به ما میانگین يك بردار و یا يك ماتریس را نشان می دهد.

```
mean(A)  
= ans  
3.0000 3.5000 2.5000
```

دستور std:

این دستور برای ما انحراف معیار را به نمایش در می آورد . انحراف معیار يك بردار ، يك عدد و انحراف معیار يك ماتریس ، انحراف معیار ستونها است .

```
std(A)  
= ans  
1.4142 2.1213 2.1213
```

دستور sqrt(تابع جذر):

این دستور جذر يك عدد را به صورت يك عدد ، جذر يك بردار را به صورت يك بردار و جذر يك ماتریس را به صورت يك ماتریس محاسبه می کند.

```
sqrt(A)  
= ans
```

```
2.0000  1.4142  1.0000
1.4142  2.2361  2.0000
```

دستور (^) توان :

این دستور ماتریس را به توان يك عدد مي رساند. باید توجه داشت که ماتریس مربع باشد.

```
B=
```

```
5  2
```

```
3  4
```

```
D=B.^(2)
```

```
= D
```

```
25  4
```

```
9  16
```

دستور size:

برای به دست آوردن ابعاد يك ماتریس و مشاهده نمودن تعداد سطر ها و ستون ها می توان از این دستور استفاده کرد.

```
size(A)
```

```
= ans
```

```
3  2
```

دستور sign:

این دستور اعداد بزرگتر از صفر يك ماتریس را با عدد يك، و کوچکتر از صفر را با عدد -1 و عدد صفر را با صفر نشان می دهد.

```
P=
```

```
-5  0  1
```

```
0  -8  2
```

```
>>sign(P)
```

```
= ans
```

```
-1  0  1
```

```
0  -1  1
```

دستور floor:

این دستور ارقام اعشاری را به سمت منهای بینهایت رند می کند.

```
= D
```

```
3.5468  -0.5479  1.0245
```

```
0.2580  2.3550  -2.1548
```

```
>>floor(D)
```

```
= ans
```

```
3  -1  1
```

```
0  2  -3
```

دستور ceil:

این دستور ارقام اعشاری را به سمت مثبت بینهایت رند می نماید.

دستور fix:

این دستور ارقام اعشاری را به سمت صفر رند می نماید.

دستور round:

این دستور ارقام اعشاری را به سمت نزدیکترین عدد صحیح رند می نماید.

```
round(D)
```

```
Ans=
```

```
4 -1 1
```

```
0 2 -2
```

دستور abs:

این دستور قدر مطلق اعداد، بردار و ماتریس را حساب می نماید.

```
F=
```

```
4 -1 1
```

```
0 2 -2
```

```
>>abs(F)
```

```
= ans
```

```
4 1 1
```

```
0 2 2
```

دستور for:

دستوری برای تکرار یک حلقه شرطی است .

مثال : حلقه زیر ابتدا مختصات نقطه اول را می گیرد و با استفاده از فرمولهای معرفی شده به آن مختصات نقطه دوم را حساب کرده و درحافظه خود نگاه می دارد ، سپس دوباره مختصات نقطه سوم را همانند مختصات اول محاسبه کرده و این کار را تاجایی که در دستوره عنوان n معرفی شده است ، انجام میدهد .

```
for i=1:n
```

```
dx(i)=l(i)*sin(az(i));
```

```
dy(i)=l(i)*cos(az(i));
```

```
a(i+1,1)=a(i,1)+dx(i);
```

```
a(i+1,2)=a(i,2)+dy(i);
```

```
end
```

دستور if:

دستور شرط است.مثلا در مثال زیر اگر تابع d(2) برابر 3 باشد تابع h اجرا میشود .

```
if d(2)==3
```

```
h=(a(:,1)+a(:,2)/60.0+a(:,3)/3600.0)*pi/180.0 ;
```

```
end
```

دستور else:

(در غیر اینصورت) به این معنی که پس از دادن یک دستور شرطی حالات دیگر را نیز بررسی نماید. مثلاً در دستور زیر اگر تابع  $d(2)$  برابر 3 نباشد عبارت داخل پرانتز نمایان میگردد.

```
if d(2)==3
h=(a(:,1)+a(:,2)/60.0+a(:,3)/3600.0)*pi/180.0 ;
else
disp('dimention is not true ')
end
```

دستور ifelse:

این دستور در درون دستور if نوشته میشود و به این معنی است که میتوانیم شرطهایی را در تناقض با شرط اصلی بنویسیم. به طور مثال در برنامه زیر شرطهای زیادی در تناقض با شرط اصلی موجود است که در صورت درست بودن هر شرط آن شرط اجرا میشود.

```
if dx>0 &dy>0
az=alfa ;
elseif dx>0&dy<0
az=pi-alfa ;
elseif dx<0&dy<0
az=pi+alfa ;
elseif dx<0&dy>0
az=2*pi-alfa ;
elseif dx==0&dy>0
az=0 ;
elseif dx==0&dy<0
az=pi ;
end
```

شرطهای تو در تو :

با توجه به اینکه اگر بخواهیم در یک شرط، شرط دیگری بگذاریم دو مرتبه از دستور if استفاده میکنیم که نحوه اجرای آن در برنامه زیر مشاهده میشود.

```
if dy~=0
(alfa=atan(abs(dx)/abs(dy) ) ;
if dx>0 &dy>0
az=alfa ;
elseif dx>0&dy<0
az=pi-alfa ;
elseif dx<0&dy<0
az=pi+alfa ;
elseif dx<0&dy>0
```

```

az=2*pi-alfa ;
elseif dx==0&dy>0
az=0 ;
elseif dx==0&dy<0
az=p ;
end
end
end

```

انواع روشهای برنامه نویسی MATLAB:

روش اول به این ترتیب است که در خط فرمان دستورات را وارد و بعد اجرا مینماییم البته باید توجه داشت که این دستورات قابل بازیافت نبوده و به این منظور داخل فایلهایی با پسوند M وارد کرده و سپس نام فایل را وارد میکنیم . روش دوم به این ترتیب است که میتوان دستورات را داخل m file تایپ کرده و سپس با يك نام ذخیره مینماییم . حال میتوانیم در پنجره اصلي مطلب برنامه مورد نظر را فراخانی کرده و اجرا نماییم .

نحوه نامیدن متغیرها و ماتریسها :

میتوان برای نامگذاری از حروف بزرگ و نیز از حروف كوچك و یا از اعداد استفاده نمود . در بین علائم تنها میتوان از گزینه منفي استفاده نمود . در ضمن میتوان از مجموع اعداد و حروف استفاده نمود با این شرط که اولین کاراکتر يك حرف باشد .

در زیر چند نمونه از برنامه هایی که در مطلب نوشته و اجرا میشوند را ذکر مینماییم .

1- برنامه ای که توسط آن معادله درجه دو را میتوان محاسبه نمود .

```

clc;
clear all;
%this program solve equation
a=input('Enter a(a*x*x+b*x+c;):');
b=input('Enter b(a*x*x+b*x+c;):');
c=input('Enter c(a*x*x+b*x+c;):');
if a==0&b~=0
    x=-c/b
elseif a~=0
delta=b^2-4*a*c;
x1=(-b+sqrt(delta))/(2*a)
x2=(-b-sqrt(delta))/(2*a)
elseif a==0& b==0
    disp('this is not equation;')
end
end

```

2- برنامه ای که با دادن یک زاویه آزیموت آنرا محاسبه مینماید .

```
function az=azimut(A,B,c)
dx=B(1)-A(1) ;
dy=B(2)-A(2) ;
if dy~=0
    alfa=atan(abs(dx)/abs(dy));
    if dx>0 &dy>0
        az=alfa;
    elseif dx>0&dy<0
        az=pi-alfa;
    elseif dx<0&dy<0
        az=pi+alfa;
    elseif dx<0&dy>0
        az=2*pi-alfa;
    elseif dx==0&dy>0
        az=0;
    elseif dx==0&dy<0
        az=pi;
    end
else
    if dx>0
        az=pi/2;
    elseif dx<0
        az=3*pi/2.0;
    elseif dx==0
        disp('two point are same') ;
        az=nan;
    end
end
c=lower(c);
if c=='d'&az~=nan
    az=R2DMS(az) ;
elseif c=='g'& az~=nan
```

```
az=az*200/pi;  
end
```

3- برنامه ای که درجه را به رادیان تبدیل مینماید .

```
function H=DMS2R(a)  
d=size(a;)  
if d(2)==3  
h=(a(:,1)+a(:,2)/60.0+a(:,3)/3600.0)*pi/180  
.0;  
else  
disp('dimention is not true;(')  
end
```

پایان