

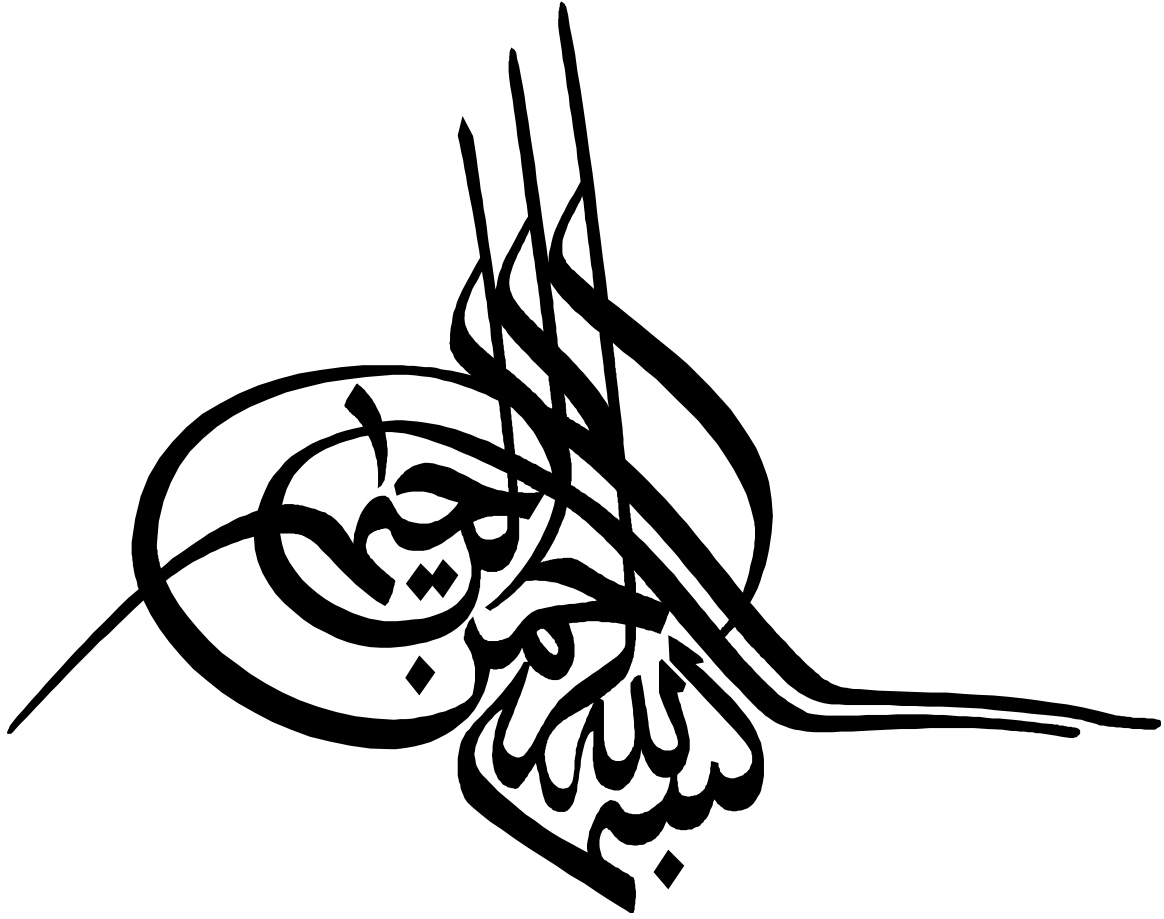


واحد استهبان

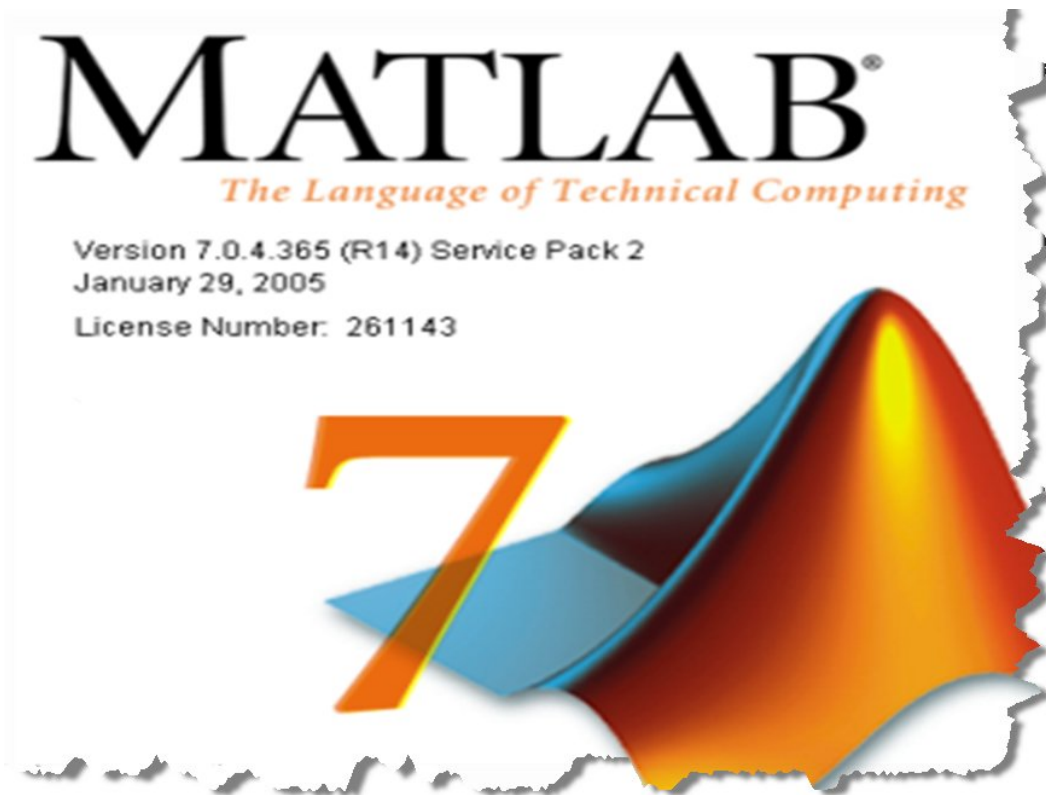
مجتمع فنی مهندسی نقشه برداری

جزوه درس برنامه نویسی کاردانی نقشه برداری (MATLAB)

مهندس ابراهیم راستگو



Matlab Programing



فهرست

- (۱) زبان برنامه نویسی و انواع آن از نظر قابلیت درک و خوانایی
- (۲) اصول کل برنامه نویسی
- (۳) الگوریتم و فلوچارت
- (۴) اولویت ها در محاسبات
- (۵) تعاریف اولیه (ماتریس، مرتبه ماتریس، ماتریس هم مرتبه، ضرب، جمع و تفریق ماتریس ها)
- (۶) تعاریف اولیه (ماتریس مربع، ماتریس واحد (I)، قطر اصلی ماتریس، ماتریس قطری)
- (۷) تعاریف اولیه (ماتریس بالا مثلثی، ماتریس پایین مثلثی، ترانهاده ماتریس، دترمینان ماتریس)
- (۸) تعاریف اولیه (معکوس یک ماتریس به سه روش)
- (۹) انواع پنجره های برنامه matlab و کار هر یک از آنها و نحوه پاک کردن هر کدام
- (۱۰) نکاتی در مورد متغیرها، نحوه آوردن پنجره های برنامه matlab
- (۱۱) نحوه ایجاد یک ماتریس، علامت (;)، علامت (:)
- (۱۲) فرمانهای version، license، date
- (۱۳) انتخاب درآیه های ماتریس و تغیر آنها
- (۱۴) فرمتهای نمایش اعداد و ایجاد آرایه با گامهای مشخص و دلخواه
- (۱۵) حذف سطر یا ستون یک ماتریس
- (۱۶) فرمانهای (ceil، floor، round، linspace)
- (۱۷) فرمانهای (zeros، rand، ans، pi، fix)
- (۱۸) فرمانهای (real max، real min، diag، eye، ones)
- (۱۹) تلفیق ماتریس ها، آرایه های سلولی
- (۲۰) عملگرهای ریاضی (جمع، تفریق، ضرب، توان، تقسیم)
- (۲۱) توابع مثلثاتی
- (۲۲) فرمانهای (min، max، mod، rem، log10، log، sqrt)
- (۲۳) فرمانهای (whos، who، load، save، triu، tril، abs، sign، mean، magic، sum)
- (۲۴) ترانهاده و معکوس و شبه معکوس یک ماتریس و دترمینان ماتریس
- (۲۵) فرمان های (disp، input، help، size، sort و ...)
- (۲۶) فرمان های (cov، var، sdt، median، factorial، beep)
- (۲۷) نحوه برنامه نویسی و ذخیره و بازیابی آن (m-file)
- (۲۸) فرمان های (pause، clear، clc)
- (۲۹) عملگرهای رابطه ای (>=، >، <=، <، ~ =، =)
- (۳۰) شرط با ساختار تک انتخابی (if-end)

- (۳۱) عملگر های منطقی (&, |, ~)
- (۳۲) شرط با ساختار دو انتخابی (if-else-end)
- (۳۳) شرط با ساختار چند انتخابی (if-elseif...-end)
- (۳۴) شرط با ساختار چند انتخابی با استفاده از فرمان switch-case
- (۳۵) حلقه for
- (۳۶) فرمان های (what, delete, cd, type, ls, dir, echo) (فایل)
- (۳۷) دستور world map
- (۳۸) شرط swich - case
- (۳۹) دستور tic toc, sr2num, num2sr
- (۴۰) حلقه while
- (۴۱) دستور prod (II)
- (۴۲) دستور xlsread و xlswrite
- (۴۳) ورود اطلاعات به برنامه matlab
- (۴۴) نمودارهای دو بعدی)
- (۴۵)
- (۴۶)
- (۴۷)
- (۴۸)
- (۴۹)
- (۵۰)
- (۵۱)
- (۵۲)
- (۵۳)
- (۵۴)
- (۵۵)
- (۵۶)
- (۵۷)
- (۵۸)
- (۵۹)

برنامه نویسی

زبان برنامه نویسی

مجموعه‌ای از دستور العمل‌ها و ساختارها و قواعد که برای ارتباط بین برنامه‌های کاربردی و سیستم کامپیوتر از آن استفاده می‌شود

تقسیم بندی زبان برنامه نویسی بر اساس درک و قابلیت خوانایی

۱) زبان سطح پایین

۲) زبان سطح میانی

۳) زبان سطح بالا

زبان سطح پایین

برنامه‌ای که با این زبان نوشته می‌شود درکش برای انسان خیلی مشکل است و در این زبانها برای نوشتن برنامه از یک سری اعداد یا یک سری علائم استفاده می‌شود، مانند زبان ماشینی و زبان اسمبلی

زبان سطح میانی

این زبان درکش نسبت به زبان سطح پایین بالاتر است ولی مانند زبان محاوره‌ای نیست؛ مانند زبان برنامه نویسی C

زبان سطح بالا

این زبان درکش برای انسان ساده بوده و عین زبان محاوره‌ای می‌باشد مانند زبان برنامه نویسی پاسکال، C#، ++C و ...

اگر برنامه malab را بخواهیم جزء زبانهای برنامه نویسی قرار دهیم. برنامه malab یک زبان برنامه نویسی سطح بالا است

به دلایل زیر برنامه نویسی matlab برای نقشه برداری مفید است

۱) یک زبان سطح بالا بوده و به راحتی می‌توان با آن کار کرد

۲) کار با ماتریس‌ها در آن بسیار ساده بوده

۳) وجود توابع مفید از قبل نوشته شده در آن

۴) امکانات بررسی مشکل در برنامه به صورت خط به خط و کاراکتر به کاراکتر

۵) محیط گرافیکی خوبی که دارد

۶) قابلیت انجام محاسبات سنگین با سرعت بالا و ...

به عنوان مثال اگر ما بخواهیم برنامه‌ای با زبان Qbasic بنویسیم که معکوس یک ماتریس را برای ما حساب کند باید در حدود ۲۰ خط برنامه نویسی کنیم اما در برنامه matlab با تایپ دستور $inv()$ و نام ماتریس، معکوس آن حساب می‌شود.

اصول برنامه نویسی

- (۱) نوشتن الگوریتم
- (۲) نوشتن فلوچارت
- (۳) نوشتن برنامه
- (۴) تست برنامه (program testing)
- (۵) تعمیم آن

الگوریتم

مجموعه‌ای از دستور العمل‌ها برای حل مسئله با زبانی ساده و بدون ابهام و دقیق و با ترتیب مراحل.

مراحل نوشتن یک الگوریتم

- (۱) خواندن ورودی‌ها
 - (۲) پردازش ورودی‌ها یا همان محاسبات
 - (۳) نمایش داده‌های خروجی
- مثال: الگوریتمی بنویسید که دو عدد را گرفته و مجموع این دو عدد را در خروجی چاپ کند.

(۰) شروع

(۱) مقدار عدد a و b را بگیر

(۲) مقدار عدد a را با مقدار عدد b جمع کن و در C قرار بده

(۳) مقدار عدد C را چاپ کن

(۵) پایان

مثال: الگوریتمی بنویسید که طول و عرض یک مستطیل را گرفته و محیط و مساحت مستطیل را در خروجی چاپ کند؟

(۰) شروع

(۱) مقدار X را به عنوان طول و مقدار Y را به عنوان عرض مستطیل بگیرد

(۲) طول را در عرض ضرب کن و به عنوان مساحت در نظر بگیر. ($X*Y$ = مساحت)

(۳) طول را با عرض جمع کن و بر ۲ تقسیم کن و به عنوان محیط در نظر بگیر. ($(X+Y)/2$ = محیط)

(۴) مقادیر محیط و مساحت را چاپ کن

(۵) پایان

مثال: الگوریتمی بنویسید که عددی را از ورودی گرفته و علامت آن را در خروجی چاپ کند؟

(۰) شروع

(۱) مقدار یک عدد را بگیر و به عنوان n قرار بده

(۲) اگر مقدار $n > 0$ بود برو به مرحله ۳ در غیر این صورت برو به مرحله ۵

(۳) چاپ کن مثبت

(۴) برو به مرحله ۶

(۵) چاپ کن منفی

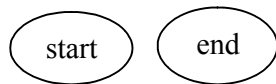
(۶) پایان

فلوچارت

مجموعه‌ای از علائم و نمادها و شکل‌های ساده است که می‌توانیم توسط آنها الگوریتم را به صورت نماد بصری نشان دهیم.

نماد شروع و پایان

برای نمایش شروع و پایان از بیضی استفاده می‌کنیم

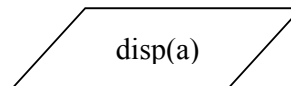
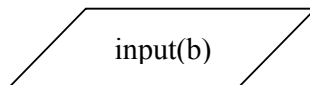


ترتیب مراحل

برای نشان دادن به ترتیب مراحل از مسیر جریان (پیکان) استفاده میشود $\rightarrow \leftarrow \uparrow \downarrow$

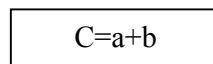
ورودی ها و خروجی ها

از متوازی‌الضلاع برای نمایش ورودی ها و خروجی ها استفاده می‌شود



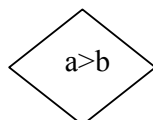
قسمت محاسبات

برای نشان دادن قسمت محاسبات از مستطیل استفاده می‌شود



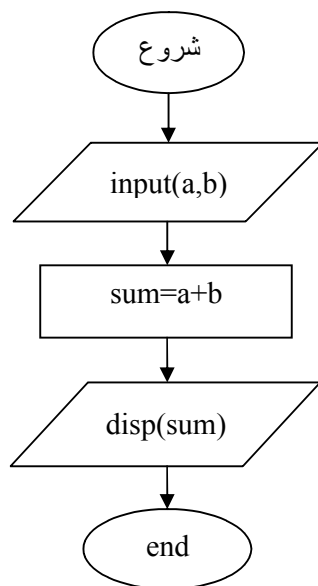
شرط ها

برای نشان دادن شرط ها از لوزی استفاده می‌شود

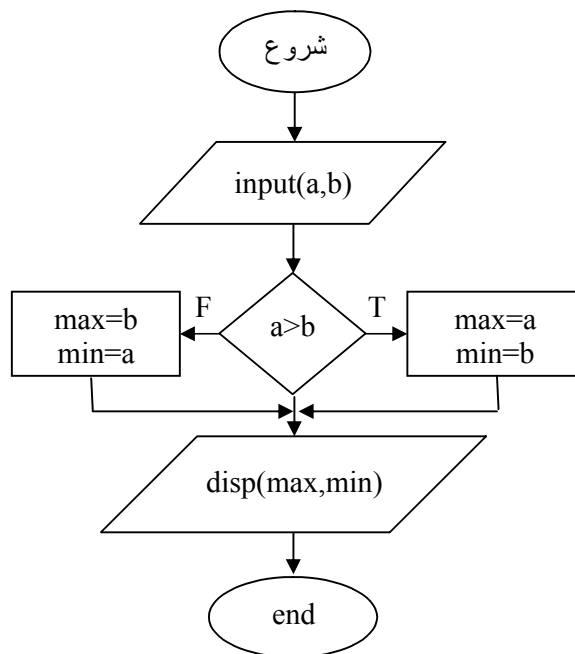


نکته: هر فلو چارت حتما باید خروجی داشته باشد. شرط تنها دو خروجی دارد ولی می تواند n ورودی داشته باشد.

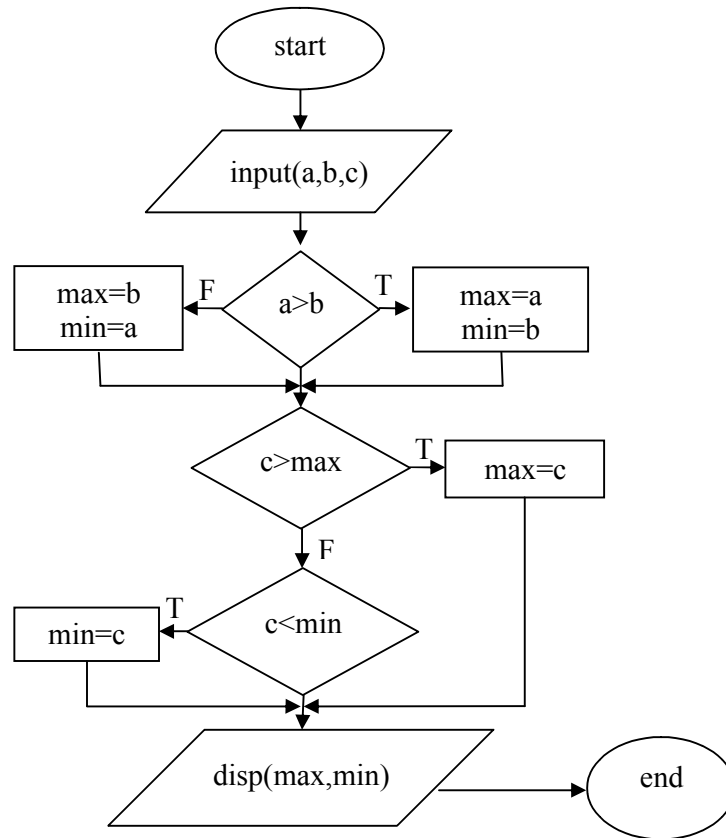
مثال: فلو چارتری رسم کنید که دو عدد از ورودی گرفته و مجموع این دو عدد را چاپ کند؟



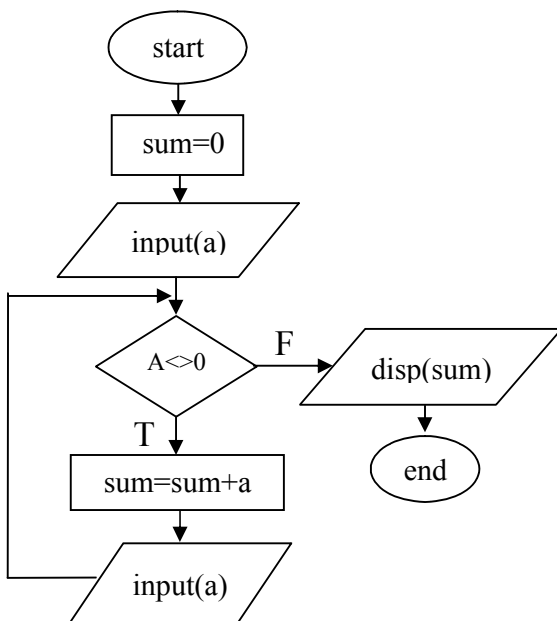
مثال: دو عدد را گرفته و ماکزیمم و مینیمم آن را چاپ کند؟



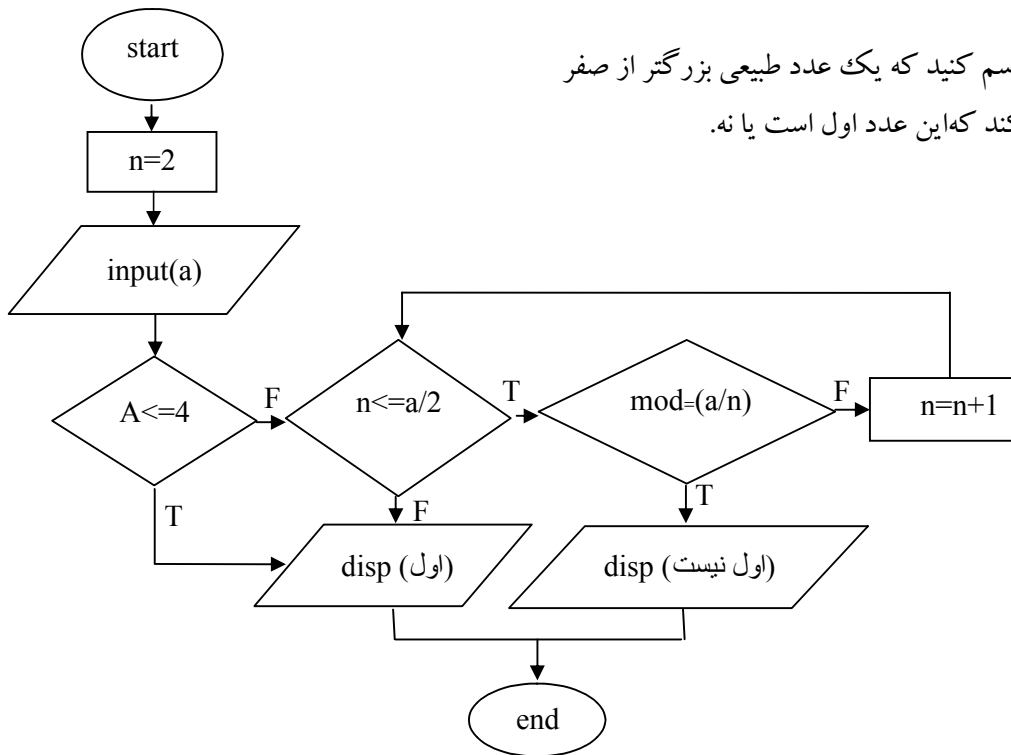
مثال: فلو چارتی رسم کنید که ۳ عدد را گرفته و ماکزیمم و مینیمم این سه عدد را چاپ کند؟



مثال: فلو چارتی رسم کنید که چندین عدد صحیح را با شرط پایان صفر از ورودی گرفته و مجموع این اعداد را در خروجی چاپ کند.



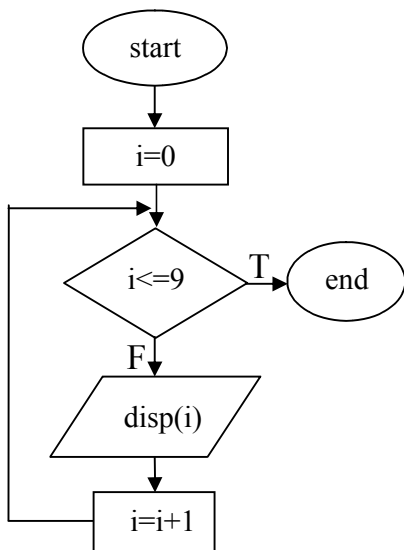
مثال: فلوچارتی رسم کنید که یک عدد طبیعی بزرگتر از صفر را گرفته و تعیین کند که این عدد اول است یا نه.



تست الگوریتم:

a	n	a/2
7	2	3.5
اول	3	
	4	

تمرین: فلوچارتی رسم کنید که اعداد یک رقمی را به صورت صعودی در خروجی چاپ کند



اولویت در محاسبات

اولویت محاسباتی در همه ی برنامه‌های کامپیوتری به ترتیب به صورت زیر می‌باشد

(۱) محاسبات از سمت چپ به راست می‌باشد

$$2*3*4 \Rightarrow 6*4 \Rightarrow 24$$

(۲) در عبارت اولویت اول با عبارت داخل پرانتز می‌باشد و اگر پرانتز تو در تو باشد اولویت با داخلی ترین پرانتز و

سمت چپی می‌باشد

(۳) توابع مثلثاتی

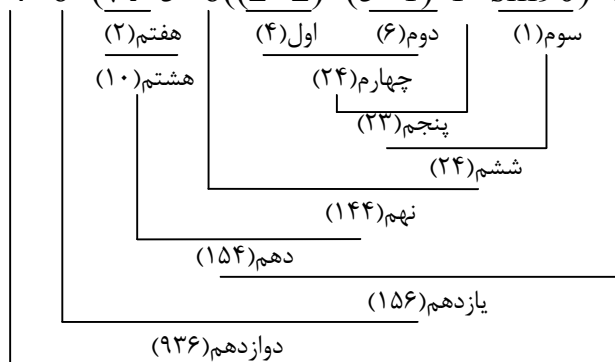
(۴) جزر و توان

(۵) ضرب و تقسیم

(۶) جمع و تفریق

مثال: حاصل عبارت زیر را با ذکر اولویت محاسبات آن بدست آورید

$$4+6*(\sqrt{4}*5+6((2+2)*(5+1)-1+\sin 90)+2)=1000$$



(۱۰۰۰) سیزدهم

در برنامه matlab نمادی که برای توان در نظر گرفته شده نماد ((^)) می‌باشد همانطور که می‌دانید هر جزر

را می‌توان به صورت یک توان به صورت یک توان کسری نوشت

$$5^{0.5} \quad \sqrt[3]{5} = 5^{\left(\frac{1}{3}\right)}$$

$$2^{0.5}$$

نکته: عبارت روبرو را در نظر بگیرید

جواب این دو عبارت طبق قوانین اولویت با هم برابر نخواهد بود، جواب $2^{0.5} = 1.4142$ ولی در مورد عبارت $2^{\frac{1}{2}}$

$\frac{1}{2}$ ، اول ۲ به توان یک رسیده و بعد بر ۲ تقسیم شده است.

پس حاصل عبارت $2^{\frac{1}{2}}$ برابر یک می‌شود برای دستیابی به حاصل درست باید به قوانین اولویت ها

توجه کرد.

$$2^{0.5} = 2^{(1/2)}$$

تعاریف اولیه

ماتریس

هر گاه دسته‌ای از اعداد یا اشیاء با آرایش سطری و ستونی درون گروه باز و گروه بسته قرار داده شوند تشکیل یک ماتریس یا آرایه می‌دهند.

$$\begin{matrix} a & b & & 1 & 2 \\ c & d & \text{یا} & 3 & 4 \end{matrix}$$

مرتبه یک ماتریس

به تعداد سطر در ستون ماتریس مرتبه ماتریس گفته می‌شود.

مثال: ماتریس a یک ماتریس مرتبه 2*3 می‌باشد

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

ماتریس هم مرتبه

دو ماتریس که تعداد سطر و ستون آنها با هم برابر باشد را ماتریس هم مرتبه گویند.

مثال: a, b هم مرتبه هستند

$$= [1 \quad 6 \quad 1] \quad = [-2 \quad 0 \quad 2]$$

ضرب ماتریس‌ها

نکته اول در ضرب ماتریس‌ها باید رعایت شود این است که در ضرب ماتریس باید تعداد ستون در ماتریس اول با تعداد سطر در ماتریس دوم برابر باشد.

$$A_{2 \times 3} \times B_{3 \times 3} = C_{2 \times 3}$$

به عنوان مثال:

در ضرب دو ماتریس درآیه‌های سطر اول ماتریس اول در درآیه‌های ستون اول ماتریس دوم به صورت نظیر به نظیر ضرب شده و حاصلشان با هم جمع و درآیه‌های ستون اول ماتریس حاصل را تشکیل می‌دهند

مثال: حاصل ضرب ماتریس‌های زیر را بدست آورید

$$\begin{pmatrix} 4 & 1 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 2 \\ 5 & 3 \end{pmatrix} = \begin{pmatrix} (4 * 1) + (1 * 5) & (4 * 2) + (1 * 3) \\ (-1 * 1) + (0 * 5) & (-1 * 2) + (0 * 3) \end{pmatrix}$$

$$= \begin{pmatrix} 9 & 11 \\ -1 & -2 \end{pmatrix}$$

$$= [2 \quad 6] \begin{pmatrix} -1 & 2 \\ 3 & 4 \end{pmatrix} = [(2 * -1) + (6 * 3) \quad (2 * 2) + (6 * 4)] = [16 \quad 28]$$

$$\begin{pmatrix} 3 & 5 \end{pmatrix} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = (3 * 1) + (5 * 1) = -2$$

$$\begin{pmatrix} 5 \\ 3 \end{pmatrix} \begin{pmatrix} 5 & 0 \\ 3 & 1 \end{pmatrix} = \begin{pmatrix} 5 * 5 & 5 * 0 \\ 5 * 3 & 5 * 1 \end{pmatrix} = \begin{pmatrix} 25 & 0 \\ 15 & 5 \end{pmatrix}$$

ماتریس مربع

ماتریسی که تعداد سطر و ستون آن با هم برابر باشد

$$\begin{pmatrix} 1 & 2 \\ 1 & -1 \end{pmatrix}$$

ماتریس واحد (ماتریس همانی) I

ماتریس مربعی که در آرایه‌های روی قطر اصلی آن یک و دیگر در آیه‌های آن صفر باشد را ماتریس واحد گویند.

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

نکته: حاصل ضرب هر ماتریس در ماتریس واحد، یا برعکس، خود ماتریس است.

$$A * I = I * A = A$$

نکته: در ضرب ماتریسها خواصیت جابجایی وجود ندارد

$$A * B \neq B * A$$

جمع و تفریق ماتریسها

در جمع و تفریق ماتریسها باید حتما دو ماتریس هم مرتبه باشند.

$$\begin{pmatrix} 1 & 3 & 0 \\ -1 & 6 & 2 \end{pmatrix} + \begin{pmatrix} -2 & 1 & 7 \\ 3 & 5 & 1 \end{pmatrix} = \begin{pmatrix} -1 & 4 & 7 \\ 2 & 11 & 3 \end{pmatrix}$$

قطر اصلی

قطر اصلی یک ماتریس، از در آیه‌هایی با سطر و ستون برابر آن ماتریس تشکیل شده.

$$\begin{pmatrix} 5 & 2 & 4 \\ -4 & 0 & 0 \\ 3 & 1 & -1 \end{pmatrix}$$

ماتریس قطری

ماتریسی که درآیه‌های روی قطر اصلی آن غیر صفر و دیگر درآیه‌های آن صفر باشد را ماتریس قطری گویند.

$$\begin{pmatrix} .5 & 0 & 0 \\ 0 & 1.3 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

ماتریس بالا مثلثی

ماتریسی که درآیه‌های زیر قطر اصلی آن صفر باشد را ماتریس بالا مثلثی گویند.

$$\begin{pmatrix} 1 & -1 & 0 \\ 0 & 5 & 4 \\ 0 & 0 & 7 \end{pmatrix}$$

ماتریس پایین مثلثی

ماتریسی که درآیه‌های بالای قطر اصلی آن صفر باشد را ماتریس پایین مثلثی گویند.

$$\begin{pmatrix} 4 & 0 & 0 \\ 5 & 2 & 0 \\ 6 & -2 & 3 \end{pmatrix}$$

ترانهاده یک ماتریس

هر گاه جای سطر و ستون ماتریس را عوض کنیم به ترانهاده یا همان ترانسپوز ماتریس می‌رسیم و علامت ترانهاده

$$A' = A^t \quad \text{یا } ((t)) \text{ است یا } ((')) \text{ مثلاً}$$

مثال: ترانهاده ماتریس A را بدست آورید

$$A = \begin{pmatrix} 2 & 5 & 6 \\ 3 & -1 & 1 \end{pmatrix} \implies A' = \begin{pmatrix} 2 & 3 \\ 5 & -1 \\ 6 & 1 \end{pmatrix}$$

دترمینان یک ماتریس:

مثال: دترمینان یک ماتریس ۲*۲

$$= \begin{matrix} * & * \\ * & * \end{matrix} \implies | | = (* \times *) - (* \times *)$$

$$= \begin{matrix} 3 & 4 \\ 1 & 6 \end{matrix} \implies | | = (3 \times 6) - (4 \times 1) = 14$$

نکته: زمانی دترمینان یک ماتریس غیر صفر خواهد بود که هیچ سطری با سطر دیگری و هیچ ستونی با ستون دیگر رابطه نداشته باشد

$$A = \begin{pmatrix} 2 & 4 & 5 \\ 1 & -1 & 0 \\ 4 & 8 & 9 \end{pmatrix} \rightarrow |A| = 0$$

دترمینان ماتریسهای بزرگتر از 2×2

روش اول

از راه بسط دادن (حول هر سطر یا ستونی که خواستیم می شود)

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \\ 3 & 1 & 2 \end{pmatrix} \Rightarrow |A| = 1 \cdot (+1) \begin{vmatrix} 1 & 3 \\ 1 & 2 \end{vmatrix} + (-1) \cdot 2 \begin{vmatrix} 2 & 3 \\ 3 & 2 \end{vmatrix} + (+1) \cdot 3 \begin{vmatrix} 2 & 1 \\ 3 & 1 \end{vmatrix} =$$

$$|A| = 1 \cdot (-1) + (-2) \cdot (-5) + 3 \cdot (-1) = -1 + 10 - 3 = 6$$

روش دوم

روش ساروس

$$|A| = \begin{vmatrix} 1 & 2 & 3 & 1 & 2 \\ 2 & 1 & 3 & 2 & 1 \\ 3 & 1 & 2 & 3 & 1 \end{vmatrix} \rightarrow$$

مجموع ضرب درآیه‌های قطر اصلی و فرعی

$$= ((1 \cdot 1 \cdot 2 + 2 \cdot 3 \cdot 3 + 3 \cdot 2 \cdot 1) - (2 \cdot 2 \cdot 2 + 1 \cdot 3 \cdot 1 + 3 \cdot 1 \cdot 3)) = 6$$

نکته: اگر تمامی درآیه‌های یک سطر یا ستون ماتریسی صفر باشد دترمینان آن ماتریس نیز صفر خواهد بود.

نکته: زمانی یک ماتریس دترمینان دارد که، یک ماتریس مربع باشد.

$$A = \begin{pmatrix} 0 & 5 & -1 \\ 0 & 3 & 2 \\ 0 & 1 & 6 \end{pmatrix} \Rightarrow |A| = 0, \quad A = \begin{pmatrix} 2 & 3 & 5 \\ 1 & -1 & 0 \end{pmatrix} \Rightarrow |A| = \text{وجود ندارد}$$

معکوس یک ماتریس

نکته: جهت گرفتن معکوس یک ماتریس، باید دترمینان ماتریس وجود داشته باشد و مقدار آن مخالف صفر

باشد. (یعنی ماتریس باید تمام شرایط دترمینان پذیری را داشته باشد)

به سه روش می توان معکوس یک ماتریس را حساب کرد.

روش اول

از راه تعریف معکوس ماتریس

$$A \cdot A^{-1} = I \quad A = \begin{bmatrix} 3 & 4 \\ 5 & 7 \end{bmatrix} \quad |A| = 21 - 20 = 1 \neq 0$$

مثال:

اول دترمینان آن را بررسی می کنیم

$$\begin{bmatrix} 3 & 4 \\ 5 & 7 \end{bmatrix} * = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \Rightarrow \begin{array}{l} 3 + 4 = 1 \\ 5 + 7 = 0 \end{array} \quad \begin{array}{l} 3 + 4 = 0 \\ 5 + 7 = 1 \end{array} \quad \begin{array}{l} 1 = 7 \\ 0 = -5 \end{array}$$

$$\begin{cases} 3 + 4 = 1 & \rightarrow & = 7 \\ 5 + 7 = 0 & \rightarrow & = -5 \end{cases} \quad \Rightarrow \quad A^{-1} = \begin{bmatrix} 7 & -4 \\ -5 & 3 \end{bmatrix}$$
$$\begin{cases} 3 + 4 = 0 & \rightarrow & = -4 \\ 5 + 7 = 1 & \rightarrow & = 3 \end{cases}$$

نکته: معکوس ماتریس 2×2 بدست می آید بصورت زیر بدست می آید:

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \rightarrow A^{-1} = \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$$

برای امتحان

$$-1 \quad \begin{bmatrix} 3 & 4 \\ 5 & 7 \end{bmatrix} \begin{bmatrix} 7 & -4 \\ -5 & 3 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$A \cdot A = I \rightarrow \begin{matrix} 5 & 7 & -5 & 3 \\ & & & & 0 & 1 \end{matrix} =$$

روش دوم

روش گوس

$$\left[A \mid I \right] \xrightarrow[\text{سطری}]{\text{عملیات مقدماتی}} \left[I \mid A^{-1} \right]$$

مثال: معکوس ماتریس A را بدست آورید

$$A = \begin{pmatrix} 1 & 3 & 3 \\ 1 & 4 & 3 \\ 1 & 3 & 4 \end{pmatrix}$$

$$\rightarrow |A| = \begin{vmatrix} 1 & 3 & 3 & | & 1 & 3 \\ 1 & 4 & 3 & | & 1 & 4 \\ 1 & 3 & 4 & | & 1 & 3 \end{vmatrix} \rightarrow |A| = 16 + 9 + 9 - (12 + 9 + 12) = 1$$

$$1 \rightarrow A \neq 0$$

$$\begin{pmatrix} 1 & 3 & 3 \\ 1 & 4 & 3 \\ 1 & 3 & 4 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \xrightarrow[\text{سطر سوم} + (-) \text{سطر اول}]{\text{سطر دوم} + (-) \text{سطر اول}} \begin{pmatrix} 1 & 3 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \xrightarrow[\text{سطر اول} + (-3) \text{سطر دوم}]{\text{سطر اول} + (-3) \text{سطر سوم}}$$

$$\begin{pmatrix} 1 & 0 & 3 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & -3 & 0 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix} \xrightarrow[-3]{+} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 4 & -3 & 0 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}$$

روش سوم

استفاده از ماتریس الحاقی

$$(\text{adj}(A))'$$

$$A^{-1} = \frac{\quad}{|A|}$$

مثال: مثال قبل را به این روش حل کنید

$$\text{Adj}(A) = \begin{pmatrix} + & - & + \\ (4*4-3*3) & (1*4-3*1) & (1*3-4*1) \\ - & + & - \\ (3*4-3*3) & (1*4-3*1) & (1*3-3*1) \\ + & - & + \\ (3*3-3*4) & (1*3-3*1) & (1*4-3*1) \end{pmatrix} =$$

$$\begin{pmatrix} 7 & -1 & -1 \\ -3 & 1 & 0 \\ -3 & 0 & 1 \end{pmatrix}$$

$$(\text{Adj}(A))' = \begin{pmatrix} 7 & -3 & -3 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}, \quad |A| = 1$$

$$A^{-1} = 1/1 \begin{pmatrix} 7 & -3 & -3 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{pmatrix}$$

تمرین: معکوس ماتریسهای زیر را بدست آورید.

$$\begin{pmatrix} -1 & 5 & 16 \\ 0 & 2 & -2 \end{pmatrix}$$

18

$$\begin{pmatrix} 5 & 1 & 0 & 2 \\ 3 & 1 & 0 & 3 \end{pmatrix}$$

1 3 6

2 8 -4 3

7 6 -3 1

$$\begin{pmatrix} 4 & 2 & 3 \\ 3 & 5 & 1 \\ 2 & 1 & -2 \end{pmatrix}$$

برنامه MALAB

MATLAB®

The Language of Technical Computing

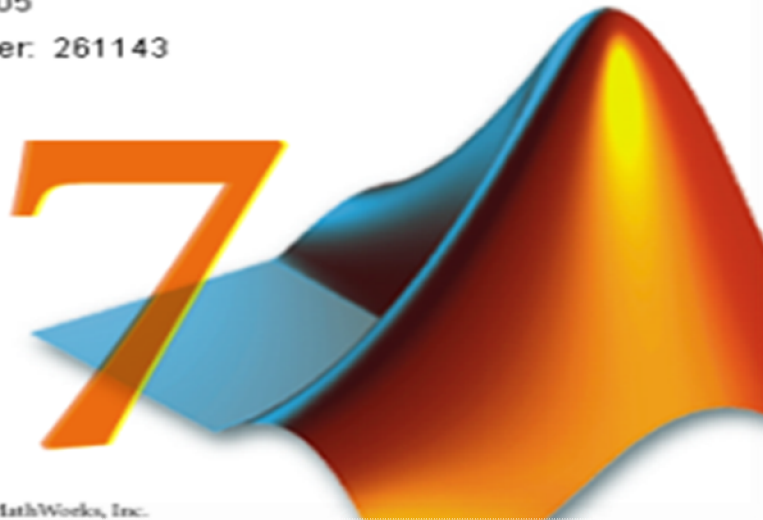
Version 7.0.4.365 (R14) Service Pack 2

January 29, 2005

License Number: 261143

a

a



Copyright 1984–2005, The MathWorks, Inc.

برنامه matlab دارای ۵ پنجره اصلی می باشد.

۱. پنجره Command window

۲. پنجره **Command history**

۳. پنجره **Current folder**

۴. پنجره **Work space**

۵. پنجره **Editor window**

پنجره Command window: در این پنجره می توان متغیر ها را وارد کرد و دستوراتی اجرا نمود. می توان برنامه نویسی کرد ولی نمی توان آن را ذخیره کرد. در حقیقت فقط جواب را به ما نشان می دهد. جهت جاری کردن این پنجره از $ctrl+0$ می توان استفاده کرد.

پنجره Command history: کلید دستورات و متغیرهای وارد و اجرا شده را به ما نشان می دهد. جهت جاری کردن این پنجره از $ctrl+1$ می توان استفاده کرد.

پنجره Curren folder: در این پنجره کلید برنامه های اجرا شده نگهداری می شود و قابل دسترسی بوده. جهت جاری کردن این پنجره از $ctrl+2$ می توان استفاده نمود.

پنجره Work space: در این پنجره متغیرهای وارد شده را می توان دید و در صورت لزوم از آنها استفاده کرد. جهت جاری کردن این پنجره از $ctrl+3$ می توان استفاده کرد.

پنجره Editor window: در این پنجره می توان فایل های **Script** یا همان **m**فایل ها را ایجاد یا ویرایش نمود.

برنامه **matlab** دارای پنجره های دیگری نیز می باشد که کمتر با آنها سروکار داریم. مانند:

پنجره Figure window: از این پنجره زمانی استفاده می شود که بخواهیم یک گراف را به نمایش در آوریم یا به عبارتی زمانی که ما از دستورات گرافیکی استفاده می کنیم گراف در این پنجره ایجاد می شود.

پنجره Launch pad window: این پنجره جهت دسترسی به ابزارها، دموها و مستندات می باشد.

اگر بخواهیم محتوای هر یک از پنجره ها را پاک کنیم، به ترتیب برای پاک کردن محتوای پنجره **Command window** در همین پنجره فرمان **clc** را تایپ کرده و **enter** می کنیم. و برای پاک کردن محتویات پنجره های **Command window** و **Command history** و **Work space**، از منوی **Edit** می توان متغیرهای هر کدام از پنجره ها را پاک نمود. اگر فرمان **clear** را در پنجره **Command window** تایپ کنیم و سپس **Enter** کنیم. محتویات دو پنجره **Command window** و **Work space** پاک می شود.

نکاتی در مورد متغیرها:

۱. برنامه **matlab** به حروف کوچک و بزرگ حساس بوده پس می توان با یک کلمه دو متغیر داشت یکی با

حروف کوچک و یکی با حروف بزرگ. $a=-1$ $A=3$

۲. برای تعریف نام متغیر حتما باید کاراکتر اول آن از حروف باشد نمی تواند عدد یا علائم و ... باشد.

غ $2A=5$ ص $A2=5$

۳. تعداد کاراکتر یک متغیر نباید بیش از ۳۱ کاراکتر در برنامه matlab ورژن ۶ و بیش از ۶۳ کاراکتر در برنامه matlab ورژن ۷ گردد. در غیر این صورت نرم افزار کاراکترهای بعدی را در نظر نمی گیرد.
۴. نامی که برای متغیر مان استفاده می کنیم نباید جزء توابع برنامه matlab باشد. البته می توان آن را با حروف بزرگ نوشت و استفاده کرد ولی توصیه نمی شود.
 $SUM=35$ $sum=35$
۵. نامی که برای متغیر مان استفاده می کنیم باید به صورت یکپارچه باشد و فاصله دار نباشد.
۶. حتماً بعد از نام متغیر علامت انتصاب که همان علامت مساوی است باید بیاید.
- ص Δx غ Δx

آوردن پنجره‌های برنامه matlab:

اگر هر یک از پنجره‌های برنامه matlab بسته شده بود و می خواستیم آن را بیاوریم به منو desktop رفته و در قسمت desktop layout می توان گزینه default را انتخاب کرد تا تنظیمات پنجره‌های برنامه matlab مثل روز اول شود (حالت پیش فرض). یا به منو desktop رفته و هر پنجره‌ای که می خواهیم وجود داشته باشد را تیک دارد و هر پنجره‌ای را که نمی خواهیم باشد بدون تیک می کنیم

نحوه ایجاد آرایه یا همان ماتریس در برنامه matlab:

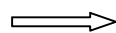
ابتدا در خط فرمان، حرف یا حروفی که مشخص کنند نام آرایه هست را (با توجه به نکاتی که در مورد انتخاب نام آرایه گفته شد) انتخاب می کنیم. سپس علامت مساوی را که نشان دهنده انتساب بوده تایپ می کنیم سپس علامت کروشه باز را تایپ کرده و بعد در آرایه اول را وارد کرده برای وارد کردن در آرایه‌های بعدی که در همان سطر قرار دارند برای مشخص کردن جدایی آنها از علامت کاما (,) یا علامت فاصله space () استفاده کرده و برای مشخص کردن سطر بعد از نقطه ویرگول (;) یا علامت enter استفاده می کنیم و در پایان از علامت کروشه بسته استفاده می کنیم .

مثال: ماتریس A به ابعاد 2*3 تعریف کنید به دو روش

>> A=[1 5 6;-1 3 5]

>> A=[1,5,6

-1,3,5]



A= 1 5 6
 -1 3 5

علامت سمیکلون (;):

این علامت در پایان هر خط که بیاید نتیجه آن خط نمایش داده نمی شود.

>> A=B;

علامت کالن (:):

علامت دو نقطه (:) یا کالن بیشتر به معنای تا می باشد مثلا اگر بخواهیم یک آرایه ایجاد کنیم که از عدد ۲ شروع شده و تا عدد ۷ باشد به این صورت عمل می کنیم.

A=[2:7] a=[2 3 4 5 6 7]

a=[2:5;10:13] A= 2 3 4 5
 10 11 12 13

فرمان date :

اگر بخواهیم به تاریخ دست پیدا کنیم کلمه date را تایپ کرده و enter می کنیم تاریخ را به ما نشان می دهد .

version

اگر بخواهیم که ورژن برنامه matlab را مشاهده کنیم این دستور را تایپ کرده و اینتر می کنیم.

>> version ← 7.00.294

license

اگر بخواهیم شماره license برنامه matlab را بیایم فرمان license را تایپ کرده و اینتر می کنیم.

>> license ← 207352

انتخاب در آیه ای دلخواه از ماتریس:

جهت این کار ابتدا نام ماتریس و سپس درون پرانتز، اول شماره سطر و بعد علامت و بعد شماره ستون را وارد می کنیم. ماتریس مفروض A را در نظر بگیرید.

مثال: سطر سوم ماتریس A را انتخاب کنید

 2 5 7 -1
A= 6 -2 4 3
 1 7 3 2
 6 -3 0 8

B=A(3,:) B=[1 7 3 2]

کل ستون ها

مثال: ستون چهارم ماتریس A را انتخاب کنید

$$B=A(:,4) \longrightarrow B = \begin{matrix} -1 \\ 3 \\ 2 \\ 8 \end{matrix}$$

کل سطر ها

مثال: درآیه ۳*۳ ماتریس A را بدست آورید

$$B=A(3,3) \quad B=3$$

مثال: درآیه ۲*۴ ماتریس A را بدست آورید

$$B=A(4,2) \quad B=-3$$

مثال: سطر دوم و سوم و ستون اول تا سوم ماتریس A را انتخاب کنید

$$B=A(2:3,1:3) \Rightarrow D = \begin{matrix} 6 & -2 & 4 \\ 1 & 7 & 3 \end{matrix}$$

تغییر دادن درآیه های یک ماتریس

جهت این کار به صورت زیر می توان عمل کرد

مثال: درآیه ۲*۳ ماتریس A را به عدد ۲۰ تغییر دهید

$$A(3,2)=20$$

$$\Rightarrow A = \begin{matrix} 2 & 5 & 7 & -1 \\ 6 & -2 & 4 & 3 \\ 1 & 20 & 3 & 2 \\ 6 & -3 & 0 & 8 \end{matrix}$$

مثال: سطر سوم ماتریس A را صفر قرار دهید

$$A(3,1:4)=0 \quad \text{یا} \quad A(3,:)=0$$

$$\Rightarrow A = \begin{matrix} 2 & 5 & 7 & -1 \\ 6 & -2 & 4 & 3 \\ 0 & 0 & 0 & 0 \\ 6 & -3 & 0 & 8 \end{matrix}$$

مثال: یک سطر و یک ستون به ماتریس A اضافه کنید

$$b = a[3; 5; 4; 6] \Rightarrow b = \begin{matrix} 2 & 5 & 7 & -1 & 3 \\ 6 & -2 & 4 & 3 & 5 \\ 0 & 0 & 0 & 0 & 4 \\ 6 & -3 & 0 & 8 & 6 \end{matrix}$$

$$c = b; [3 \ 1 \ 2 \ -7 \ 8] \Rightarrow c = \begin{bmatrix} 2 & 5 & 7 & -1 & 3 \\ 6 & -2 & 4 & 3 & 5 \\ 0 & 0 & 0 & 0 & 4 \\ 6 & -3 & 0 & 8 & 6 \\ 3 & 1 & 2 & -7 & 8 \end{bmatrix}$$

فرمتهای نمایش اعداد:

در هنگام استفاده از برنامه matlab جهت نمایش بهینه تعداد ارقام اعشاری می توان از این فرمتها استفاده نمود.

format bank

اگر این فرمان را تایپ کنیم و اینتر کنیم تعداد ارقام اعشاری که پس از محاسبات به ما نشان می دهد ۲ رقم خواهد بود.

مثال:

forma bank ←
A=33.046781 ← → A= 33.05

format short

اگر این دستور را تایپ کرده و اینتر کنیم. تا ۴ رقم پس از ممیز را به ما نشان می دهد.

مثال:

forma bank ←
A=33.046781 ← → A= 33.0468

format short g

این دستور همان دستور format short است با این تفاوت که دیگر اگر عدد صفر بعد از ممیز وجود داشته باشد آنرا نشان نمی دهد.

format short g
3.5300 => 3.53

format short e

این دستور همان دستور format short است با این تفاوت عدد را بصورت علمی نمایش می دهد.

format short e
3.5300 => 3.5300e+000

format

اگر این دستور را تایپ کرده و اینتر کنیم. به صورت پیش فرض همانند دستور format short عمل کرده و تا ۴ رقم پس از ممیز را به ما نشان می دهد.

مثال:

forma bank ←
A=33.046781 ← → A= 33.0468

format long

اگر این دستور را تایپ کرده و اینتر کنیم تا ۱۵ رقم پس از ممیز را به ما نشان می‌دهد.

forma long ←

مثال:

A=33.046781 ← → A=33.04678100000000

format long g

این دستور همان دستور format long است با این تفاوت که دیگر اگر عدد صفر بعد از ممیز وجود داشته باشد آنرا نشان نمی‌دهد (عدد غیر ضروری را نشان نمی‌دهد).

format long g

3.35000 => 3.35

format long e

این دستور همان دستور format long است با این تفاوت عدد را بصورت علمی نمایش می‌دهد.

format long e

3.5300 => 3.5300000000000000e+000

format +

اگر این دستور را تایپ کرده و اینتر کنیم. اگر بازگشتی عبارت عدد مثبت باشد، تنها علامت مثبت (+) را برمی‌گرداند و اگر بازگشتی عبارت عدد منفی باشد، تنها علامت منفی (-) را برمی‌گرداند.

مثال:

forma +

A=33.046781 ← → A= +

B=-33.046781 ← → B= -

نکته: زمانی که ما فرمت نمایش اعداد را تغییر می‌دهیم در دقت محاسبات هیچ گونه اثری ندارد و نرم افزار با بالاترین دقت خود محاسبات را انجام می‌دهد.

ایجاد آرایه با فواصل (گامهای) دلخواه و مشخص:

اگر ما بخواهیم آرایه‌ای سطری با فواصل و شروع و پایان مشخص ایجاد کنیم به نحو زیر عمل می‌کنیم.

مثال:

A=[3:2:20]

A=3 5 7 9 11 13 15 17 19

در مثال بالا عدد ۳ شروع آرایه و عدد ۲ گام حرکتی آرایه و عدد ۲۰ پایان آرایه می‌باشد. اگر گام حرکتی یک باشد می‌توان از نوشتن آن صرفه نظر کرد.

مثال:

$$A=[2:5] \longrightarrow A=[2 \ 3 \ 4 \ 5]$$

نکته: اگر گام حرکتی را مشخص نکنیم گام را یک واحد در نظر می گیرند.

نکته: گام حرکتی می تواند منفی نیز باشد اما باید عضو آخر آرایه باشد

$$A=[10:-1:5] \longrightarrow A=[10 \ 9 \ 8 \ 7 \ 6 \ 5]$$

حذف کردن سطر یا ستون یک ماتریس

ماتریس A را در نظر بگیرید

$$A = \begin{bmatrix} -3 & 4 & 8 \\ 1 & 0 & 2 \\ 5 & 6 & 5 \end{bmatrix}$$

مثال: ستون سوم ماتریس A را حذف کنید

$$A(:,3) = [] \longrightarrow A = \begin{bmatrix} -3 & 4 \\ 1 & 0 \\ 5 & 6 \end{bmatrix}$$

مثال: ستون دوم و سطر دوم ماتریس A را حذف کنید

$$\begin{array}{l} A(2,:) = [] \longleftarrow \\ A(:,2) = [] \longleftarrow \end{array} \longrightarrow \begin{bmatrix} -3 & 8 \\ 5 & 5 \end{bmatrix}$$

تبدیل یک بازه به تعداد فواصل مشخص:

جهت تقسیم یک بازه به تعداد مشخص از دستور linspace می توان استفاده نمود.

به این صورت است که دستور linspace (بدون فاصله) را تایپ کرده و پرانتز باز نموده و مقدار شروع و سپس

کاما و بعد مقدار پایانی و سپس کاما و در آخر تعداد فواصل درآیه را نوشته و پرانتز را می بندیم.

$$A=\text{linspace}(2,10,5) \longrightarrow A=2 \ 4 \ 6 \ 8 \ 10$$

$$A=\text{linspace}(2,10,4) \longrightarrow A=2 \ 4.66 \ 7.33 \ 10$$

$$A=[5 \ 3 \ 4;\text{linspace}(10,20,3);1 \ 0 \ -1]$$

دستور round

اگر این دستور را وارد کنیم اعداد را به صورت رند شده به ما نشان می دهد و در محاسبات به کار می برد

$$A=\text{round}(35.3) \rightarrow A=35 \quad A= \text{round} \quad (35.6) \rightarrow A=36$$

$$A=\text{round}(-35.3) \rightarrow A=-35 \quad A= \text{round} \quad (-35.6) \rightarrow A= -36$$

دستور floor

این دستور جهت گرد کردن به سمت پایین است این دستور باعث می شود که جزء صحیح عدد تنها نمایش داده شود.

$$A=\text{floor}(35.3) \rightarrow A=35 \quad A=\text{floor}(35.99) \rightarrow A=35$$

$$A=\text{floor}(-35.3) \rightarrow A=-36 \quad A=\text{floor} (-35.99) \rightarrow A=-36$$

دستور ceil

این دستور جهت گرد کردن به سمت بالا است

مثال:

$$A=\text{ceil}(35.3) \rightarrow a=36 \quad A=\text{ceil} (35.9) \rightarrow A=36$$

$$A=\text{ceil}(-35.3) \rightarrow a=-35 \quad A=\text{ceil}(-35.9) \rightarrow A=-35$$

دستور fix

اگر این دستور استفاده شود. تنها قسمت صحیح عدد را نمایش می دهد

مثال:

$$A=\text{fix}(35.3) \rightarrow a=35 \quad A=\text{fix}(35.9) \rightarrow A=35$$

$$A=\text{fix}(-35.3) \rightarrow a=-35 \quad A=\text{fix}(-35.9) \rightarrow A=-35$$

a	ceil(a)	fix(a)	floor(a)
-2.5	-2	-2	-3
-1.75	-1	-1	-2
-1.25	-1	-1	-2
-0.5	0	0	-1
0.5	1	0	0
1.25	2	1	1
1.75	2	1	1
2.5	3	2	2

نکته: در ریاضیات اگر عددی را بر صفر تقسیم کنیم به ما مقدار بی نهایت را می‌دهد و در برنامه matlab هم نیز چنین است و عبارت inf را می‌دهد.

مثال:

$$A=5/0 \longrightarrow A=Inf = \infty \quad \text{بی نهایت}$$

نکته: برنامه matlab اعداد مبهم را نیز با عبارت nan نشان می‌دهد. که نشان دهنده مقدار غیر عددی است.

$$A=0/0 \longrightarrow A=nan$$

مثال:

(π) pi

در برنامه matlab به جای علامت π که مقدار آن ۳.۱۴۱۵ رادیان است از عبارت pi استفاده می‌شود.

$$A=pi*2 \longrightarrow A=6.2830$$

ans

نام متغیر پیش فرض است که برای نشان دادن نتایج از آن استفاده می‌شود.

مثال:

>> 3 ←

$$A=ans*2 \longrightarrow A=6$$

دستور rand

اگر بخواهیم یک ماتریس ایجاد کنیم که درآیه‌های آن به صورت تصادفی از بازه ۰ تا ۱ انتخاب شده باشد از این دستور استفاده می‌کنیم.

مثال: یک ماتریس ۳*۲ به صورت تصادفی ایجاد کنید

$$A = \text{rand}(3,2) \Rightarrow A = \begin{bmatrix} 0.20 & 0.20 \\ 0.02 & 0.47 \\ 0.45 & 0.85 \end{bmatrix}$$

مثال: یک ماتریس 4×4 به صورت تصادفی ایجاد کنید

$$A = 4 * \text{rand}(2) \Rightarrow A = \begin{bmatrix} 4.30 & 0.89 & 1.48 & 4.27 \\ 4.27 & 1.71 & 2.48 & 0.67 \\ 4.09 & 0.45 & 3.22 & 4.19 \\ 2.97 & 2.48 & 4.19 & 3.30 \end{bmatrix}$$

دستور zeros

اگر بخواهیم ماتریس با آرآی‌های صفر تولید کنیم از این دستور می‌توان استفاده نمود.

مثال: یک ماتریس 3×1 ایجاد کنید که تمام درآی‌های آن صفر باشد

$$A = \text{zeros}(3,1) \Rightarrow A = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

دستور ones

اگر بخواهیم یک ماتریس با درآی‌های یک تولید کنیم از این دستور می‌توان استفاده کرد.

مثال:

$$A = \text{ones}(3,2) \Rightarrow A = \begin{bmatrix} 1 & 1 \\ 1 & 1 \\ 1 & 1 \end{bmatrix}$$

دستور eye

با این دستور می‌توان ماتریسی ایجاد کرد که درآی‌های قطر اصلی آن یک باشد و دیگر درآی‌های آن صفر (ماتریس واحد یا ماتریس همانی).

مثال: ماتریس واحدی با ابعاد 3×3 ایجاد کنید

$$A = \text{eye}(3) \Rightarrow A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

مثال: ماتریس واحدی با ابعاد 2×3 ایجاد کنید

$$A = \text{eye}(2,3) \Rightarrow A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

دستور diag

اگر بخواهیم از یک ماتریس درآیه‌های قطر اصلی آن را استخراج کنیم از این دستور استفاده می‌کنیم.

مثال: درآیه‌های قطر اصلی ماتریس A را بدست آورید

$$\begin{bmatrix} 4.30 & 0.890 & 1.48 \\ 4.27 & 1.71 & 2.48 \\ 4.09 & 0.45 & 3.22 \end{bmatrix} \Rightarrow \begin{bmatrix} 4.30 & & \\ & 1.71 & \\ & & 3.22 \end{bmatrix}$$

$$\begin{bmatrix} 4.30 & 0 & 0 \\ 0 & 1.71 & 0 \\ 0 & 0 & 3.22 \end{bmatrix}$$

دستور realmin

این دستور برابر است با کوچکترین عدد حقیقی مثبتی که می‌توان از آن استفاده کرد

یعنی

$$A = \text{realmin} \longrightarrow A = 2.2251e-308$$

دستور realmax

این دستور برابر است با بزرگترین عدد حقیقی مثبتی که می‌توان از آن استفاده کرد

یعنی

$$A = \text{realmax} \longrightarrow A = 1.7977e+308$$

تلفیق دو ماتریس

دو ماتریس A, B را در نظر می‌گیریم

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

$$B = [3 \ 2; 6 \ 7]$$

جهت تلفیق این دو ماتریس به صورت سطری، به گونه زیر عمل می‌کنیم

$$C = [A, B] \Rightarrow C = \begin{bmatrix} 1 & 2 & 3 & 2 \\ 4 & 5 & 6 & 7 \end{bmatrix}$$

*بین نام دو ماتریس از استفاده می کنیم.

جهت تلفیق این دو ماتریس به صورت ستونی، به گونه زیر عمل می کنیم

$$C=[A;B] \Rightarrow C = \begin{bmatrix} 1 & 2 \\ 4 & 5 \\ 3 & 2 \\ 6 & 7 \end{bmatrix}$$

*بین نام دو ماتریس از استفاده می کنیم.

مثال: همه درآیه های ماتریس A را در یک ستون قرار دهید

$$B=A(:) \Rightarrow C = \begin{bmatrix} 1 \\ 4 \\ 3 \\ 6 \end{bmatrix}$$

آرایه های سلولی:

تفاوت این نوع آرایه ها با دیگر آرایه ها در این است که آرایه های سلولی می توانند در هر خانه خود یک آرایه یا یک رشته را جای دهد.

مثال: یک آرایه سلولی ۲*۲ ایجاد کنید

$$A = \{ [1 \ 2; 3 \ 5], 3, 'SALAM', 2i \}$$

$$\longrightarrow A = \begin{bmatrix} [1*4 \text{ double}] & [3] \\ \text{'salam' ' } & [0+2i] \end{bmatrix}$$

برای مشاهده هر آرایه از آرایه سلولی مانند آرایه های دو بعدی عمل می کنیم

$$B=A(1,1) \longrightarrow B = \begin{bmatrix} 1 & 2 \\ & \end{bmatrix}$$

31

همان طور که مشاهده می کنید ممکن است هر آرایه از آرایه سلولی خود چندین آرایه باشد یا می تواند حتی رشته باشد.

$$C=A(2,1) \longrightarrow C= \text{SALAM}$$

عملگرهای ریاضی

(+,-) جمع و تفریق

عمل جمع و تفریق اعداد و جمع و تفریق ماتریسها را انجام می دهد

نکته: در جمع و تفریق ماتریسها باید ابعاد دو ماتریس یک اندازه باشد

$$A=3+4 \rightarrow A=7 \quad \text{و} \quad A=4-2 \rightarrow A=2 \quad \text{مثال:}$$

$$A=\begin{pmatrix} 2 & 5 \\ -1 & 3 \end{pmatrix}, \quad B=\begin{pmatrix} -1 & 0 \\ 2 & 1 \end{pmatrix} \quad C=A+B \longrightarrow C=\begin{pmatrix} 1 & 5 \\ 1 & 4 \end{pmatrix}$$

$$D=A-B \longrightarrow D=\begin{pmatrix} 3 & 5 \\ -3 & 2 \end{pmatrix}$$

(*) ضرب

عمل ضرب اعداد و آرایهها با این علامتصورت می گیرد

مثال:

$$A=3*4 \longrightarrow A=12$$

$$C=a*b \longrightarrow C=\begin{pmatrix} 8 & 5 \\ 7 & 3 \end{pmatrix}$$

نکته: هر گاه قبل از عملگر ریاضی علامت دات (.) قرار گیرد عملگر به صورت جزء به جزء عمل می کند

(.*)

عمل ضرب در آیه به در آیه در ماتریس‌ها را انجام می‌دهد یا همان ضرب متناظر

مثال:

$$C=a.*b \longrightarrow C=\begin{pmatrix} -2 & 0 \\ -2 & 3 \end{pmatrix}$$

(^)

عمل به توان رساندن اعداد و آرایه‌ها را انجام می‌دهد

مثال:

$$A=2^2 \longrightarrow A=4 \quad A=25^{0.5} \quad A=5$$

$$A=\begin{pmatrix} 2 & 4 \\ 6 & -1 \end{pmatrix} \longleftarrow A^2=A*A$$

$$B=A^2 \longrightarrow B=\begin{pmatrix} 28 & 4 \\ 6 & 25 \end{pmatrix}$$

(\)

تقسیم بر عکس (Back slash)

این دستور اصلش عدد سمت راست را بر عدد سمت چپ تقسیم می‌کند و به همین دلیل به آن تقسیم معکوس گویند.

مثال:

$$A=5\backslash 15 \longrightarrow A=3$$

(/)

تقسیم (slash)

این دستور عمل تقسیم را انجام می‌دهد عدد سمت چپ را بر عدد سمت راست تقسیم می‌کند

مثال:

$$A=5/15 \longrightarrow A=0.333$$

(./)

تقسیم آرایه‌ها

این دستور عمل تقسیم تک تک آرایه‌ها را به صورت نظیر به نظیر انجام می‌دهد

مثال:

دو ماتریس A, B را در نظر بگیرید

$$A = \begin{pmatrix} 2 & 6 \\ 8 & 4 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}$$

$$C = A ./ B \longrightarrow C = \begin{bmatrix} 2 & 2 \\ 4 & 1 \end{bmatrix}$$

(۱) تقسیم بر عکس آرایه‌ها

این دستور عمل تقسیم بر عکس تک تک آرایه‌ها را به صورت نظیر به نظیر انجام می‌دهد

مثال:

$$C = A . \backslash B \longrightarrow C = \begin{bmatrix} 0.5 & 0.5 \\ 0.25 & 1 \end{bmatrix}$$

توابع

برنامه‌هایی هستند که برای سهولت در کار قبلاً نوشته شده‌اند و دیگر نیازی به نوشتن نیست مانند توابع مثلثاتی و حسابی و ...

توابع مثلثاتی

نکته: نکته قابل توجه در هنگام استفاده از توابع مثلثاتی این است که در برنامه matlab باید مقادیر زاویه را بر حسب رادیان وارد کرد و ورودی و خروجی برنامه بر حسب رادیان است.

	0	90	180	270	360	30	60	45
sin	0	1	0	-1	0	1/2	$\sqrt{3}/2$	$\sqrt{2}/2$
cos	1	0	-1	0	1	$\sqrt{3}/2$	1/2	$\sqrt{2}/2$
tan	0		0	$-\infty$	0	$\sqrt{3}/3$	$\sqrt{3}$	1
cotg	∞	0	$-\infty$	0	∞	$\sqrt{3}$	$\sqrt{3}/3$	1

sin (زاویه)

بوسیله این دستور می‌توان Sin یک زاویه را محاسبه نمود.

مثال:

$$A = \sin(\pi/2) \longrightarrow A = 1$$

asin (مقدار)

بوسیله این دستور می‌توان arc sin یک مقدار را حساب نمود.

مثال:

$$A = \text{asin}(-1) \longrightarrow A = -1.57 \longrightarrow A = 270$$

cos (زاویه)

بوسیله این دستور می توان COS یک زاویه را حساب نمود.

مثال:

$$A = \text{cos}(\pi/2) \longrightarrow A = 0$$

acos (مقدار)

بوسیله این دستور می توان arc COS یک مقدار را حساب نمود.

مثال:

$$A = \text{acos}(-1) \longrightarrow A = 3.1415 \longrightarrow A = 180$$

tan (زاویه)

بوسیله این دستور می توان tan یک زاویه را حساب نمود.

$$A = \text{tan}(\pi/4) \longrightarrow A = 1$$

atan (مقدار)

بوسیله این دستور می توان arc tan یک مقدار را حساب کرد.

$$\sqrt{3} = 1.732$$

$$A = \text{atan}(1.732) \longrightarrow A = 1.05 \longrightarrow A = 30$$

به همین ترتیب بقیه توابع مثلثاتی

(مقدار) acosh: آرک کسینوس هایپربولیک را محاسبه می کند.

(مقدار) acot: آرک کوتانژانت را محاسبه می کند.

(مقدار) acoth: آرک کوتانژانت هایپربولیک را محاسبه می کند.

(مقدار) acsc: آرک کسکانت را محاسبه می کند.

(مقدار) acsch: آرک کسکانت هایپربولیک را محاسبه می کند.

(مقدار) asec: آرک سکانت را محاسبه می کند.

(مقدار) asech: آرک سکانت هایپربولیک را محاسبه می کند.

(مقدار) asinh: آرک سینوس هایپربولیک را محاسبه می کند.

(مقدار) atanh: آرک تانژانت هایپربولیک را محاسبه می کند.

(زاویه) cosh: کسینوس هایپربولیک را محاسبه می کند.

- (زاویه) **cot** : کوتانژانت را محاسبه می کند.
- (زاویه) **coth**: کوتانژانت هایپربولیک را محاسبه می کند.
- (زاویه) **csc** : کسکانت را محاسبه می کند.
- (زاویه) **csch** : کسکانت هایپربولیک را محاسبه می کند.
- (زاویه) **sec** : سکانت را محاسبه می کند.
- (زاویه) **sech** : سکانت هایپربولیک را محاسبه می کند.

دستور sqrt

برای گرفتن جذر یک عدد یا درآیه های یک ماتریس از این دستور می توان استفاده نمود.

مثال: جذراعداد و ماتریس زیر را بدست آورید؟

$$A=\text{sqrt}(36) \longrightarrow A=6$$

$$A=[25 \ 36; 49 \ 9]$$

$$B=\text{sqrt}(A) \Rightarrow B = \begin{bmatrix} 5 & 6 \\ 7 & 3 \end{bmatrix}$$

دستور exp

برای محاسبه تابع نمایی بکار می رود.

مثال:

$$A = \text{exp}(1) \Rightarrow = 2.71828$$

log

برای محاسبه لگاریتم طبیعی یک عددی می توان از این دستور استفاده نمود. که همان Ln عدد مورد نظر است.

مثال:

$$e = x \longrightarrow \text{Ln}(x) = a$$

$$a = \text{log}(2.71828) \longrightarrow a=1 \quad \text{یعنی} \quad e = 2.71828$$

log10 (مقدار)

برای محاسبه لگاریتم بر مبنا ۱۰ می توان از این دستور استفاده نمود.

مثال:

$$\text{log} \frac{1000}{10} = 3 \quad \text{log}_{10} = x \quad a = 10$$

$$A = \log_{10}(10000) \Rightarrow = 4 \quad 10000 = 10^4$$

دستور rem

برای بدست آوردن باقیمانده تقسیم دو عدد بر هم می توان از این دستور استفاده کرد.

مثال:

$$\begin{array}{r|l} 20 & 3 \\ 18 & 6 \\ \hline & 2 \text{ rem} \end{array}$$

$$A = \text{rem}(20,3) \rightarrow A = 2$$

$$A = \text{rem}(-15.5,3) \rightarrow A = -0.5$$

دستور mod

$\text{mod}(x,y)$: (باقیمانده) تقسیم x به y را به ما نشان می دهد

$$\text{mod}(20,7) = 6$$

$$\text{mod}(-20,7) = 1$$

دستور max

برای یافتن بزرگترین عدد در هر ستون یا سطر از یک ماتریس، می توان از این دستور استفاده کرد

مثال: بزرگترین عدد در سطر های ماتریس A را بیابید

$$A = \begin{bmatrix} 4.30 & 0.89 & 1.48 & 4.27 \\ 4.27 & 1.71 & 2.48 & 0.67 \\ 4.09 & 0.45 & 3.22 & 4.19 \\ 2.97 & 2.48 & 4.19 & 3.30 \end{bmatrix}$$

$$B = \max(A) \quad \text{یا} \quad B = \max(A,1) \Rightarrow B = [4.30 \quad 2.48 \quad 4.19 \quad 4.27]$$

مثال: بزرگترین عدد در ستونهای ماتریس A را بیابید

$$B = \max(A,2) \Rightarrow B = \begin{bmatrix} 4.30 \\ 4.27 \\ 4.19 \\ 3.30 \end{bmatrix}$$

دستور min

برای یافتن کوچکترین عدد در هر ستون یا سطر از یک ماتریس، می توان از این دستور استفاده کرد

مثال: کوچکترین عدد در هر ستون از ماتریس A را بیابید

$$A = \begin{bmatrix} 4.30 & 0.89 & 1.48 & 4.27 \\ 4.27 & 1.71 & 2.48 & 0.67 \\ 4.09 & 0.45 & 3.22 & 4.19 \\ 2.97 & 2.48 & 4.19 & 3.30 \end{bmatrix}$$

$$B = \min(A) \text{ یا } B = \min(A,1)$$

$$B = [2.97 \quad 0.45 \quad 1.48 \quad 0.67]$$

مثال: کوچکترین عدد در هر سطر از ماتریس A را بیابید

$$B = \max(A,2) \Rightarrow B = \begin{bmatrix} 0.89 \\ 0.67 \\ 0.45 \\ 2.47 \end{bmatrix}$$

دستور sum

توسط این دستور می توان مجموع درآیه های هر ستون یا سطر را یافت

مثال: مجموع درآیه های هر ستون از ماتریس A را بیابید

$$B = \text{sum}(A) \text{ یا } B = \text{sum}(A,1)$$

$$\Rightarrow B = [15.63 \quad 5.53 \quad 11.37 \quad 12.43]$$

مثال: مجموع درآیه های هر سطر از ماتریس A را بیابید

$$B = \text{sum}(A,2) \Rightarrow B = \begin{bmatrix} 10.94 \\ 9.13 \\ 11.95 \\ 12.94 \end{bmatrix}$$

دستور magic

توسط این دستور می توان یک ماتریس مربع ایجاد نمود که مجموع درآیه های همه ستونهای و مجموع درآیه های همه سطرها و مجموع درآیه های قطر اصلی آن با هم برابر بوده.

مثال:

$$A = \text{magic}(4) \longrightarrow A = \begin{pmatrix} 16 & 2 & 3 & 13 \\ 5 & 11 & 10 & 8 \\ 9 & 7 & 6 & 12 \\ & & & 38 \end{pmatrix}$$

4 14 15 1

$$B=\text{sum}(A) \Rightarrow B=[34 \ 34 \ 34 \ 34]$$

$$C=\text{sum}(A') \Rightarrow C=[34 \ 34 \ 34 \ 34]$$

$$D=\text{sum}(\text{diag}(A)) \Rightarrow D=34$$

نکته: نکته قابل توجه دیگر در مورد دستور magic آن است که بزرگترین درآیه موجود در ماتریس آن برابر است با ابعاد ماتریس به توان ۲

مثال:

$$A=\text{magic}(4) \longrightarrow 4^2 = 16$$

بزرگترین درآیه در این ماتریس $4^2 = 16$ می باشد .

دستور mean

به کمک این دستور می توان میانگین درآیه های هر ستون یا سطر از یک ماتریس را بدست آورد.

مثال: میانگین درآیه های هر یک از ستونهای ماتریس A را بست آورید

$$A = \begin{pmatrix} 0 & -1 & 5 \\ 6 & 2 & 4 \\ 3 & 1 & 6 \end{pmatrix} \quad B = \text{mean}(A)$$
$$B = \text{Mean}(A,1) \longrightarrow B = [3 \ 0.66 \ 5]$$

مثال: میانگین درآیه های هر یک از سطرهای ماتریس A را بست آورید

$$B = \text{mean}(A,2) \longrightarrow B = \begin{pmatrix} 2/3 \\ 4 \\ 10/3 \end{pmatrix}$$

دستور sign

این دستور اعداد بزرگتر از صفر در یک ماتریس را به عدد ۱ و اعداد کوچکتر از صفر را به عدد -۱ تبدیل کرده و عدد صفر را تغییر نمی دهد

مثال:

$$A = \begin{pmatrix} 0 & -.0001 \\ 000.1 & 1000 \end{pmatrix}$$

$$B=\text{sign}(A) \longrightarrow B = \begin{bmatrix} 0 & -1 \\ 1 & 1 \end{bmatrix}$$

دستور prod

با استفاده از این دستور می توان درآیه های هر سطر یا ستون را در هم ضرب کرد
 مثال: حاصل ضرب درآیه های هر سطر یا ستون از ماتریس A را بیابید.

$$= \begin{bmatrix} 1 & 2 & 3 \\ 5 & 7 & 6 \end{bmatrix}$$

$$= (, 1) \Rightarrow = \begin{matrix} 6 \\ 210 \end{matrix}$$

$$= (, 2) \Rightarrow = [5 \quad 14 \quad 18]$$

دستور real(x)

با استفاده از این دستور می توان بخش حقیقی عدد مختلط X را بدست آورد.

مثال:

$$A=2+\text{sqrt}(-66)$$

$$B=\text{real}(A) \Rightarrow B=2$$

دستور imag(x)

با استفاده از این دستور می توان بخش موهومی عدد مختلط X را بدست آورد.

مثال:

$$A=2+\text{sqrt}(-66)$$

$$B=\text{imag}(A) \Rightarrow B=8.1240$$

دستور isreal(x)

با استفاده از این دستور می توان مشخص کرد که عدد X یک عدد حقیقی است یا نه. اگر بازگشتی این دستور عدد 1 باشد عدد X یک عدد حقیقی است و اگر بازگشتی این دستور عدد 0 باشد عدد X یک عدد غیر حقیقی است.

مثال:

$$A=2$$

$$B=\text{isreal}(A) \Rightarrow b=1$$

$$A=\text{sqrt}(-66)$$

$$B=\text{isreal}(A) \Rightarrow b=0$$

دستور rat(x)

این دستور دو عدد صحیح حقیقی را برمی گرداند که حاصل تقسیم آنها تقریباً برابر X می شود.

مثال:

$$A=\text{rat}(.6) \Rightarrow A=1+1/(-3+1/(2))$$

دستور abs

با استفاده از این دستور می توان عمل قدر مطلق را بر روی اعداد و درآیه های ماتریس ها انجام داد

مثال:

$$A=-354 \quad \longleftarrow \quad B= \text{abs}(A) \quad \longrightarrow \quad b=0.354$$

$$a= \begin{pmatrix} 1 & -3 \\ -2 & 0 \end{pmatrix} \quad \longleftarrow$$

$$B=\text{abs}(A) \quad \longrightarrow \quad B= \begin{pmatrix} 1 & 3 \\ 2 & 0 \end{pmatrix}$$

دستور tril (A)

به کمک این دستور می توان یک ماتریس رابه ماتریس پایین مثلثی تبدیل کرد.

$$A= \begin{pmatrix} 2 & 3 & 5 & 5 \\ 6 & 1 & 0 & 1 \\ -2 & 5 & 8 & 3 \\ 2 & -3 & 5 & 4 \end{pmatrix} \quad \Longrightarrow \quad \text{tril}(A)= \begin{pmatrix} 2 & 0 & 0 & 0 \\ 6 & 1 & 0 & 0 \\ -2 & 5 & 8 & 0 \\ 2 & -3 & 5 & 4 \end{pmatrix}$$

دستور triu (A)

به کمک این دستور می توان یک ماتریس رابه ماتریس بالا مثلثی تبدیل کرد.

$$A= \begin{pmatrix} 2 & 3 & 5 & 5 \\ 6 & 1 & 0 & 1 \\ -2 & 5 & 8 & 3 \\ 2 & -3 & 5 & 4 \end{pmatrix} \quad \Longrightarrow \quad \text{triu}(A)= \begin{pmatrix} 2 & 3 & 5 & 5 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 8 & 3 \\ 0 & 0 & 0 & 4 \end{pmatrix}$$

فرمان save data

تمام متغیرهای جاری در work space در فایل daa.ma ذخیره می شوند

```
>>save data
```

فقط متغیرهای a ,b ,c در daa.ma ذخیره می شود

>>save data a,b,c

فرمان load

با استفاده از این دستور می توان در هر زمانی کلیه متغیرهای ذخیره شده در فراخوانی نمود.

>> load data

این دستور حتی اگر از فرمان استفاده کرده باشیم مقادیر را می تواند فراخوانی کند.
مقدار a را فراخوانی می کند

>> load data a

فرمان who

لیست متغیرهای ذخیره شده در work space را نشان می دهد.

فرمان whos

لیست متغیرهای ذخیره شده در work space را با ابعاد و مقدار حافظه ای که اشغال می کند و همچنین حقیقی یا مختلط بودن آنها... را نشان می دهد.

ترانواده یک ماتریس:

جهت بدست آوردن ترانواده یک ماتریس، از علامت کتیشن (') یا از دستور transpose استفاده می کنیم.

مثال: ترانواده ماتریس A را بدست آورید؟

$$A = \begin{pmatrix} 3 & -2 & 3 & 1 \\ 0 & 2 & 1 & 5 \\ 1 & -1 & 3 & 4 \end{pmatrix}$$

$$B = A'$$
$$B = \text{transpose}(A)$$

$$B = \begin{pmatrix} 3 & 0 & 1 \\ -2 & 2 & -1 \\ 3 & 1 & 3 \\ 1 & 5 & 4 \end{pmatrix}$$

دترمینال یک ماتریس det

برای بدست آوردن دترمینان یک ماتریس می توان از دستور det استفاده نمود

نکته: زمانی یک ماتریس دارای دترمینان است که مربع باشد و زمانی دترمینان آن غیر صفر است که هیچ سطر یا ستونی باهم رابطه نداشته باشند.

مثال: دترمینان ماتریسهای زیر را بدست آورید؟

$$A = \begin{pmatrix} 1 & 3 & 0 \\ 2 & 1 & 5 \\ -1 & 3 & 4 \end{pmatrix}$$

$$A = \begin{pmatrix} 1 & 5 & 11 \\ 3 & 15 & -1 \\ 2 & 10 & 0 \end{pmatrix}$$

42

$$B=\det(A) \rightarrow B=-50$$

$$B=\det(A) \rightarrow B=0$$

معکوس یک ماتریس: inv

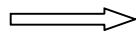
جهت بدست آوردن معکوس یک ماتریس، آنرا به توان (-1) می رسانیم یا اینکه از دستور inv استفاده می کنیم.
نکته: جهت بدست آوردن معکوس یک ماتریس اول باید توجه کرد که ماتریس مربع باشد و نیز دترمینان آن ماتریس مخالف صفر باشد.

مثال: معکوس ماتریس زیر را بدست آورید؟

$$A = \begin{pmatrix} 3 & 0 & 1 \\ -1 & 2 & 5 \\ 4 & 1 & 3 \end{pmatrix}$$

$$B=A^{-1}$$

$$B=\text{inv}(A)$$



$$B = \begin{pmatrix} -0.16 & -0.16 & 0.33 \\ -3.8 & -0.8 & 2.6 \\ 1.5 & 0.5 & -1 \end{pmatrix}$$

شبه معکوس یک ماتریس: pinv

ماتریسی که مربع نباشد یا دارای دترمینان صفر باشد معکوس پذیر نبوده ولی شبه معکوس دارد.

مثال: شبه معکوس ماتریسهای زیر را بدست آورید؟

$$A = \begin{pmatrix} 2 & 3 & 5 \\ 4 & 6 & 10 \\ 3 & -1 & 2 \end{pmatrix} \Rightarrow B=\text{pinv}(A) \Rightarrow B = \begin{pmatrix} -0.006 & -0.012 & 0.24 \\ 0.03 & 0.06 & -0.21 \\ 0.02 & 0.05 & 0.03 \end{pmatrix}$$

$$A = \begin{pmatrix} 3 & 2 & 1 \\ 5 & -1 & 0 \end{pmatrix} \Rightarrow B=\text{pinv}(A) \Rightarrow B = \begin{pmatrix} 0.67 & 0.16 \\ 0.33 & -0.21 \\ 0.13 & -0.07 \end{pmatrix}$$

دستور trace

برای بدست آوردن اثر یک ماتریس می توان از این دستور استفاد کرد. اثر یک ماتریس مجموع درآیه های قطر اصلی یک ماتریس است.

مثال: اثر ماتریس A را بدست آورید.

$$\begin{aligned} & 5 \quad 10 \quad 1 \\ = & 3 \quad 6 \quad 2 \\ & 4 \quad 5 \quad 2 \\ = & (\quad) \Rightarrow = 13 \end{aligned}$$

دستور rank

برای بدست آوردن رتبه یک ماتریس می توان از این دستور استفاده کرد. رتبه یک ماتریس تعداد سطرها یا ستون های (اگر تعداد سطر بیشتر از تعداد ستون ها باشد در مورد سطرها بررسی می شود و در غیر این صورت مورد ستون ها بررسی می شود) مستقل یک ماتریس می باشد.
مثال: رتبه ماتریس های زیر را بدست آورید.

$$\begin{aligned} & \begin{pmatrix} 5 & 10 & 1 \\ 3 & 6 & 2 \\ 4 & 5 & 2 \end{pmatrix} \\ = & \begin{pmatrix} 5 & 10 & 1 \\ 3 & 6 & 2 \\ 4 & 5 & 2 \end{pmatrix} \Rightarrow = 3 \\ & \begin{pmatrix} 5 & 10 & 1 \\ 3 & 6 & 1 \\ 4 & 8 & 1 \end{pmatrix} \\ = & \begin{pmatrix} 5 & 10 & 1 \\ 3 & 6 & 1 \\ 4 & 8 & 1 \end{pmatrix} \Rightarrow = 2 \end{aligned}$$

دستور eig

برای بدست آوردن مقادیر ویژه (λ) یک ماتریس می توان از این دستور استفاده کرد. مقادیر ویژه یک ماتریس از فرمول زیر قابل محاسبه است.

$$\left([A] - \begin{bmatrix} \lambda & 0 & \dots & 0 \\ 0 & \lambda & \dots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 0 & \dots & 0 & \lambda \end{bmatrix} \right) = 0$$

مثال: مقادیر ویژه ماتریس A زیر را بدست آورید.

$$\begin{aligned} & \begin{pmatrix} 1 & 3 \\ 2 & 5 \end{pmatrix} \Rightarrow \begin{pmatrix} 1 & 3 & -\lambda & 0 \\ 2 & 5 & 0 & \lambda \end{pmatrix} = 0 \\ & \begin{pmatrix} 1-\lambda & 3 \\ 2 & 5-\lambda \end{pmatrix} = 0 \Rightarrow (1-\lambda) * (5-\lambda) - 6 = 0 \end{aligned}$$

$$\Rightarrow 5 - 5 -$$

نکته: مجموع مقادیر ویژه یک ماتریس برابر trace آن ماتریس است.

$$\square = (\quad)$$

$$\text{trace}(A)=6, \quad \sum \square = 3 + \sqrt{40} + 3 - \sqrt{40} = 6$$

مثال: مقادیر ویژه ماتریس A زیر را بدست آورید.

$$\begin{aligned} & \begin{pmatrix} 5 & 3 & 1 \\ 8 & 3 & 5 \\ 7 & 1 & 2 \end{pmatrix} \\ = & \begin{pmatrix} 5 & 3 & 1 \\ 8 & 3 & 5 \\ 7 & 1 & 2 \end{pmatrix} \end{aligned}$$

10.8742

$$B=\text{eig}(A) \Rightarrow B = \begin{bmatrix} -0.4371 & + & 2.0773 \\ -0.4371 & - & 2.0773 \end{bmatrix}$$

دستور sort

با استفاده از این دستور در یک ماتریس می توان آرایه ها را براساس سطر یا ستون از کوچک به بزرگ مرتب کرد.
مثال: درآیه های هر ستون از ماتریس A را از کوچک به بزرگ مرتب کنید؟

$$A = \begin{pmatrix} -1 & 0 & 2 \\ 5 & 2 & -2 \\ 3 & 6 & 3 \end{pmatrix} \quad B = \text{sort}(A,1) , B = \text{sort}(A) \Rightarrow \begin{pmatrix} -1 & 0 & -2 \\ 3 & 2 & 2 \\ 5 & 6 & 3 \end{pmatrix}$$

مثال: درآیه های هر سطر از ماتریس A را از کوچک به بزرگ مرتب کنید؟

$$B = \text{sort}(A,1) \Rightarrow B = \begin{pmatrix} -1 & 0 & 2 \\ -2 & 2 & 5 \\ 3 & 3 & 6 \end{pmatrix}$$

دستور size

اگر بخواهیم به ابعاد یک ماتریس پی ببریم از این دستور استفاده می کنیم.
مثال: ابعاد ماتریس A را بدست آورید؟

$$A = \begin{pmatrix} 3 & 5 & 6 & 4 \\ 1 & 3 & -1 & 0 \\ 0 & 2 & 5 & 4 \end{pmatrix}$$

$$B = \text{size}(A) \Rightarrow B = [3 \ 4]$$

مثال: تعداد سطر ماتریس A را بدست آورید.

$$B = \text{size}(A,1) \Rightarrow B = 3$$

مثال: تعداد ستون ماتریس A را بدست آورید.

$$B = \text{size}(A,2) \Rightarrow B = 4$$

مثال: تعداد ستون و سطر ماتریس A را بدست آورید.

$$[a,c] = \text{size}(A) \Rightarrow a = 3 , c = 4$$

دستور numel

به کمک این دستور می توان تعداد کل درآیه های یک ماتریس را بدست آورد
مثال: تعداد کل درآیه های ماتریس A را بدست آورید؟

$$B = \text{numl}(A) \Rightarrow B = 9$$

help

اگر در مورد یکی از دستورات برنامه matlab بخواهیم از help کمک بگیریم، راه ساده آن این است که ابتدا کلمه را تایپ کرده سپس داخل کنیشن دستور مورد نظر را تایپ کرده .

```
>> help 'sqrt'
```

مثال: help برنامه matlab را در مورد sqrt بیاورید؟

دستور input

جهت دریافت یک مقدار عددی یا رشته‌ای اختصاص آن با یک نام از این دستور می‌توان استفاده نمود.

مثال: برنامه‌ای بنویسید که مقدار یک عدد را گرفته و آن را در خروجی نشان دهد.

```
a= input ('a=') => a=5 => ans =5
```

مثال: برنامه‌ای بنویسید که از شما نامتان را بگیرد و آن را چاپ کند.

```
a= input ('NAME : ','s') => NAME : ali => ans = ali
```

نکته: اگر در پایان هر خط از برنامه (;) را قرار دهیم نتیجه آن خط در خروجی نمایش داده نمی‌شود و همچنین عمل جدایی بین دو خط از برنامه را نیز نشان می‌دهد.

نکته: اگر به جای (;) از (,) برای جدایی دو خط از برنامه استفاده کنیم، نتیجه هر خط در خروجی نمایش داده می‌شود.

مثال: برنامه‌ای بنویسید که دو عدد را گرفته و مجموع آن دو عدد را نمایش دهد؟

```
a=input('a=');b=input('b=');c=a+b  
a=5  
b=6  
c=11
```

دستور disp

در این دستور می‌توان هر چیزی که درون پرانتز جلوی آن است را نمایش داد.

مثال: مقدار متغیر a را بدون چاپ نام خود متغیر نشان دهید؟

```
disp(a) =>5
```

مثال: برنامه‌ای بنویسید که پیغام نام خود را چاپ کند؟

```
disp('ali') =>ali
```

مثال: برنامه‌ای بنویسید که دو عدد را گرفته سپس ما بین سه خط چین جمع و حاصل تفریق این دو عدد را چاپ کند؟

```
a=input('a=');b=input('b=');disp('-----'); c=a+b;disp('-----');d=a-b;disp('-----');
```

نکته: اگر طول خط برنامه زیاد شد می‌توان با تایپ (...) سه نقطه به سطر بعد رفت و ادامه برنامه را در سطر بعد نوشت.

مثال: مثال قبل را نوشته و طوری باشد که دیگر نام نتایج c,d در میان خط چین ها چاپ شود. این کار را می توان تا ۴۰۹۶ کاراکتر ادامه داد.

```
>> a=input('a='); b=input('b='); c=a+b; ...
d=a-b; disp('----+----'); disp(c); disp('----_----'); ...
disp(d); disp('-----')=> a=3 , b=6
---+-----
  9
----
-3
-----
```


نکته: اگر بخواهیم از دستورات و متغیرهایی که در command history وجود دارد استفاده کنیم. آن متغیر را پیدا کرده و بر روی آن دابل کلیک می کنیم تا در پنجره command window ظاهر شود یا آن متغیر را از پنجره command history با کلیک چپ کردن روی آن و نگه داشتن گرفته و به پنجره می آوریم.

نکته: اگر بخواهیم دستوراتی را که قبلا وارد کردیم دوباره وارد کنیم با دکمه جهت روبه بالای کیبورد می توان تمام دستورات نوشته شده را آورد.

دستور beep

اگر از این دستور در برنامه استفاده به هنگام اجراء برنامه وقتی به این دستور می رسد صدای بیپ ایجاد می کند

مثال: برنامه ای بنویسید که دو عدد a,b را گرفته و در هم ضرب کند و در پایان با نمایش جواب، یک صدای بیپ نیز ایجاد کند؟

```
>> a=input('a='); b=input('b='); ...
c=a*b,beep => a=2 , b=4 , c=8 
```

مثال: برنامه ای بنویسید که یک عدد را گرفته و جذر آن را در خروجی چاپ کند؟

```
>> a=input('a='); jazra=sqrt(a); =>
a=36
jazra=6
```

مثال: برنامه ای بنویسید که ماتریس A را گرفته و ترانهاد آن را در خروجی چاپ کند؟

```
'>> a=input('mat A='); b=a'
a=[3 2 1]
b=  $\begin{pmatrix} 3 \\ 2 \\ 1 \end{pmatrix}$ 
```


مثال: برنامه‌ای بنویسید که ماتریس A, B را گرفته و مجموع آن را در ماتریس C بعد از یک سطر خالی نمایش دهد؟

```
>> A=input('mat A='); B= input('mat B='); disp(' '); C=A+B
```

مثال: برنامه‌ای بنویسید که یک ماتریس را گرفته و دترمینان آن ماتریس را در خود ماتریس ضرب کرده و جواب را در ماتریس C نمایش دهد؟

```
>> a=input('mat A='); c=det(a)*a
```

مثال: برنامه‌ای بنویسید که دو عدد را گرفته و اولی را به توان دومی برساند؟

```
>> a=input('a='); b= input('b='); c=a^b
```

مثال: برنامه‌ای بنویسید که سه عدد را گرفته و میانگین آن را چاپ کند؟

```
>> a=input('a='); b= input('b='); c= input('c='); Miyangin = (a+b+c)/3
```

مثال: برنامه‌ای بنویسید که نمرات یک دانشجو را در یک ماتریس سطری گرفته و میانگین آن را چاپ کند؟

```
>> a=input('number='); b=sum(a,2); ...
```

```
c=size(a,2); Miyangin=b/c
```

یا

```
>> a=input('number='); Miyangin=Mean(a,2)
```

مثال: برنامه‌ای بنویسید که معادله درجه یک را حل کند؟

$ax+b=0$

```
>>a=input('a=');b=input('b=');x=-b/a
```

مثال: برنامه‌ای بنویسید که با گرفتن ضرایب و ثوابت یک معادله درجه دوم ریشه‌های آن را پیدا کند؟

$ax^2+bx+c=0$

```
>>a=input('a=');b=input('b=');c=input('c=');delta=b^2-4*a*c;x1=(-b+sqrt(delta))/(2*a)...
```

```
x2=(-b-sqrt(delta))/(2*a);disp('x1=');disp(x1);disp('x2=');disp(x2);
```

مثال: برنامه‌ای بنویسید که نمرات یک دانشجو را بصورت یک ماتریس ستونی گرفته و ماکزیمم و مینیمم و میانگین نمرات او را چاپ کند؟

```
>>a=input('number=');...
```

Max=max(a),min=max(a),miyangin=mean(a)

مثال: برنامه‌ای بنویسید که دو ماتریس مربع را گرفته و درآیه‌های نظیر به نظیر آنها را در هم ضرب کرده و ماتریس حاصل را چاپ کند؟

```
>>a=input('mat a=');  
b=input('mat b=');  
c=a.*b
```

مثال: برنامه‌ای بنویسید که SIN, COS, TAN, COT زاویه 90° را حساب کند؟

```
>>alfa=90*pi/180;  
Tan=tan(alfa),Cot=cot(alfa),Cos=cos(alfa),Sin=sin(alfa), ...
```

مثال: برنامه‌ای بنویسید که ماتریس A را گرفته و مقدار $(A^t * A)^{-1}$ را حساب کند؟

```
>>a=input('a=');  
X=inv(a'*a)
```

دستور factorial

بوسیله این دستور می‌توان فاکتوریل یک عدد را حساب کرد.

مثال: فاکتوریل عدد 4 را حساب کنید؟

```
a=factorial(4) => a=24
```

دستور median

بوسیله این دستور می‌توان مقدار میانی یک آرایه سطری یا ستونی را به تعداد آرایه‌ها زوج باشد از دو مقدار میانی میانگین می‌گیرد و نمایش می‌دهد.

نکته‌ای که در هنگام استفاده از دستور median باید در نظر گرفت این است که طبق تعریف میانه، برای بدست آوردن میانه ابتدا تماس اعضای یک نمونه از کوچک به بزرگ مرتب می‌شوند و سپس مقدار میانه مشخص می‌گردد. حال این دستور نیز همین عمل را انجام می‌دهد.

مثال: مقدار میانه مجموعه‌های زیر را بدست آورید؟

```
a=[3 5 6 4 1 2 7 ,8 -1 -5]  
b=median(a)=>b=4  
a=[-1 3 -5 2 0 4 3 1]  
b=median(a)=>b=1.5
```

دستور std

بوسیله این دستور می‌توان انحراف معیار یک مجموعه را حساب نمود. فرمول انحراف معیار بصورت زیر می‌باشد .

$$\delta = \frac{\sum \quad^2}{-1}$$

مثال: انحراف معیار اعضای ماتریس A را بدست آورید؟

$$a=[3.5, 5, 4, 5.5, 4]$$

$$b=\text{std}(a) \Rightarrow b=0.8216$$

مثال: برنامه‌ای بنویسید که مشاهدات یک طول را که چندین بار اندازه گیری شده در یک ماتریس بگیرد و بهترین طول و انحراف معیار آن را نیز به ما نشان دهد؟

```
>>a=input('tol hay andazehgiri shodh=');distance=mean(a),...
Enheraf meyar=std(a)
```

دستور var

این دستور، واریانس یک مجموعه را برای ما حساب می‌کند. فرمول واریانس یک نمونه به شرح زیر است

$$\delta^2 = \frac{\sum \quad^2}{n}$$

مثال: واریانس مجموعه A را بدست آورید؟

$$A=[4, 4.4, 4.5, 5, 5.1]$$

$$B=\text{var}(A) \Rightarrow B=0.205$$

دستور cov

این دستور کوواریانس یک مجموعه را برای ما حساب می‌کند. زمانی واریانس و کوواریانس یک مجموعه باهم برابر نیست که ما دو نوع مشاهده داشته باشیم یا با دو وسیله با دقت‌های متفاوت برداشت کرده باشیم. و فرمول آن برابر است با:

$$C = -\sum (V - M) (V - M)$$

مثال: کوواریانس مجموعه‌های زیر را بدست آورید؟

$$a=[4, 4.4, 4.5, 5, 5.1]$$

$$b=[6, 5, 5.5, 5, 5.3]$$

$$c=\text{cov}(a,b) \Rightarrow c = \begin{bmatrix} .205 & -.125 \\ -.125 & .173 \end{bmatrix}$$

نکته: کوواریانس a_{12} باید برابر کوواریانس a_{21} باشد.

$$\text{Cov} = \text{Cova}$$

برنامه نویسی قابل دستیابی در برنامه matlab (Script فایل ها):

تاکنون هر برنامه‌ای که نوشته‌ایم در پنجره command window بود و تنها می‌توانستیم آن را اجرا کنیم و حاصل را مشاهده کنیم و برنامه نوشته شده قابل ذخیره سازی نبود، به عبارتی برای استفاده مجدد از برنامه باید آنرا مجدداً نوشت. جهت رفع این نوع مشکل می‌توان از فایل های اسکریپت یا m-file استفاده نمود که پسوند این نوع فایل ها (.m) می‌باشد. این نوع فایل ها قابل ذخیره سازی هستند.

یک فایل script یک مجموعه دستورهای برنامه matlab است که برنامه (program) نیز نامیده می‌شوند. وقتی یک فایل اسکریپت اجرا شد برنامه matlab دستورها را به ترتیبی که نوشته شده اند اجرا می‌کند. خروجی های این فایلها در پنجره command ظاهر می‌گردد.

فایلهای اسکریپت را می‌توان در هر نرم افزار متنی نوشت و بعد در بخش editor آن را paste نمود. برای اجرای این فایلها می‌توان نام آن را در پنجره command تایپ کرد و کلید Enter را فشار داد و یا در پنجره Editor آیکن Run (F5) را کلیک کرد. بدین منظور فایل باید در current folder یا در مسیر جستجو باشد.

جهت ساختن یک m-file به مسیر زیر می‌رویم:

File → New → m-file

سپس پنجره‌ای باز می‌شود که در آن می‌توان برنامه‌ها را بصورت خط به خط نوشت و ذخیره و اجرا نمود و در هر زمانی آنها را فراخوانی کرد و اجرا نمود.

مثال: برنامه‌ای بنویسید که دو عدد را از ورودی بگیرد و مجموع و حاصل تفریق و حاصل ضرب و حاصل تقسیم این دو عدد را نشان دهد؟

```
a=input('a=');
b=input('b=');
C=a+b
D=a-b
E=a*b
F=a/b
```

دستور `clear` (clear command)

با تایپ این دستور می‌توان پنجره command window را کاملاً پاک نمود.

دستور `clear`

با تایپ این دستور می‌توان حافظه پنجره command window را پاک نمود. یعنی اگر یک بار شماره در پنجره command window یک متغیر را وارد کنیم می‌توانید با تایپ نام آن متغیر، از آن متغیر استفاده کنید ولی اگر این دستور را اجرا کنید، کلیه متغیرها از حافظه command window پاک خواهد شد.

`clear a,b,c`: تنها منغیر های a,b,c را از حافظه پاک می‌کند.

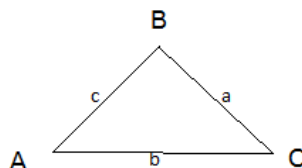
نکته: بهتر است در هنگام برنامه نویسی در سطر اول و دوم از دستورات `clear,clc` استفاده نمود و در سطر سوم فرمت نمایش اعداد را مشخص کرد.

مثال: برنامه‌ای بنویسید که مختصات دو نقطه را بصورت ماتریس سطری گرفته و فاصله بین این دو نقطه را نمایش دهد؟

```
clc
clear
format short g
a=input('a[x,y]= ');
b=input('b[x,y]= ');
delta=b-a;
distance=sqrt((delta(:,1)^2)+(delta(:,2)^2))
```

مثال: برنامه‌ای بنویسید که با گرفتن طول دو ضلع از یک مثلث و زاویه بین آن، طول ضلع سوم مثلث را حساب کند؟

```
clc
clear
format short g
c=input('zele aval= ');
b=input('zele dovom= ');
A=input('zaviyeh beyn= ');
a=sqrt(c^2+b^2-2*c*b*cos(a*pi/180))
% a=  $\sqrt{c^2 + b^2 - 2cb \cdot \cos \hat{A}}$ 
```



نکته: هرگاه بخواهیم در مورد یک خط از برنامه توضیحاتی بدهیم که در برنامه نیز تاثیری نداشته باشد می‌توان از علامت `%` استفاده نمود. به عبارتی هرگاه در نوشتن برنامه از علامت `%` استفاده کنیم، فرمان‌هایی بعد از این علامت نوشته شود به رنگ سبز در می‌آید و به عنوان یک توضیحات در نظر گرفته می‌شود و در روند اجرای برنامه تاثیری نخواهد داشت.

مثال: برنامه‌ای بنویسید که با گرفتن طول سه ضلع از یک مثلث محیط و مساحت آن مثلث را چاپ کند؟

```
clc
clear
format short g%   nemayesh adad ta 4 ragham ashar
a=input(' tole azla be sorat matris satri=')
Mohit=sum(a)
p=Mohit/2
masahat=sqrt(p*(p-a(1,1))*(p-a(1,2))*(p-a(1,3)))
```

دستور pause

هرگاه در برنامه به این دستور برسیم اجرا برنامه تا زمانی که کار بر کلیدی را بفشارد متوقف می‌شود.


می‌توان این دستور را با مدت زمان مکث استفاده کرد مثلاً به این خط از برنامه که رسید n ثانیه مکث کند.

```
>> pause (n)
```

مثال: برنامه‌ای بنویسید که با گرفتن دو زاویه و ضلع بین از یک مثلث، دیگر اضلاع و زاویه سوم را حساب کند. و در

هر مرحله از نمایش جوابها ۳ ثانیه مکث کند؟

```
clc
clear
format short g
a=input('zaviyh(a)=');
b=input('zaviyh(b)=');
C=input('tole(c)=');
pause(3)
c=180-(a+b)
pause(3)
A=C*sin(a*pi/180)/sin(c*pi/180)
pause(3)
B=C*sin(b*pi/180)/sin(c*pi/180)
```

نکته: جهت اجرای برنامه می‌توان از کلید F5 یا رفتن به منو بار Debug و زدن Run یا فشردن آی‌کون  استفاده نمود.

مثال: برنامه‌ای بنویسید که زوایای یک چهار ضلعی را بگیرد و تصحیح روی آنها انجام دهد؟

```
clc
clear
format long g
A=input('zaviyh(A)=');%zaviyh bar hasb daragh
B=input('zaviyh(B)=');% zaviyh bar hasb daragh
C=input('zaviyh(C)=');% zaviyh bar hasb daragh
D=input('zaviyh(D)=');% zaviyh bar hasb daragh
S=A+B+C+D
e=360-S
c=e/4
A=A+c
B=B+c
C=C+c
D=D+c
یا
clc
clear
format long g
a=input('zavaya[A,B,C,D]=')
s=sum(a,2)
e=360-s
c=e/4
a=a+c
```

مثال: برنامه‌ای بنویسید که با داشتن اضلاع مثلث زوایای آن مثلث را حساب کند؟

$$a^2 = b^2 + c^2 - 2bc \cos A \Rightarrow A = \cos^{-1} \left(\frac{b^2 + c^2 - a^2}{2bc} \right)$$

```
clc
clear
format long g
a=input('a=');
b=input('b=');
c=input('c=');
A= acos((b^2+c^2-a^2)/(2*b*c))*180/pi
B= acos((a^2+c^2-b^2)/(2*a*c))*180/pi
C= acos((a^2+b^2-c^2)/(2*a*b))*180/pi
```

نکته: بعد از هر اجرا برنامه به صورت خودکار ذخیره می‌شود.

مثال: برنامه‌ای بنویسید که با گرفتن شعاع قوس و زاویه انحراف مسیر کلیه مولفه‌های قوس ساده را حساب کند؟

```
clc
clear
format short g
R=input('radius=');%
delta=input('zaviyh enheraf=');% زاویه بر حسب درجه
del=delta*pi/180;% زاویه بر حسب رادیان
T=R*tan(del/2)% T=R.tan(Δ/2)
L=R*del% L=R.Δ
C=R*2*sin(del/2)% C=2Rsin(Δ/2)
Bis=T*tan(del/4)% Bis=T.tan(Δ/4)
M=R*(1-cos(del/2))% M=R(1-cos(Δ/2))
```

مثال: برنامه‌ای بنویسید که مختصات منحنی الخط را به مختصات کارتیزین تبدیل کند.

$$\begin{aligned} \varphi &= 66^\circ 34' 59'' \\ &= \lambda = 20^\circ 0' 16'' \quad , \quad a=6378137 \quad , \quad f=1/298.2572221 \\ h &= 171.472568199 \end{aligned}$$

```
clc
clear
format long
a=input('a= ');%a=6378137
f=input('f= ');%f=1/298.2572221;
b=a*(1-f);
```

```

%b=input('b= ');
phi=input('phi mat= ');%phi=[66 34 59];
phi=phi(1,1)+(phi(1,2)/60)+(phi(1,3)/3600)
landa=input('landa mat= ');% landa =[20 0 16];
landa = landa (1,1)+( landa (1,2)/60)+( landa (1,3)/3600)
h=input('h= ');%h=171.47256819997
phi=phi*pi/180; landa = landa *pi/180;
e=sqrt(1-b^2/a^2);
N=a/(1-e^2*(sin(phi))^2)^.5;
M=(a*(1-e^2))/(1-e^2*(sin(phi))^2)^1.5;
X=(N+h)*cos(phi)*cos(landa)
Y=(N+h)*cos(phi)*sin(landa)
Z=(N*b^2/a^2+h)*sin(phi)
%X =2.388666469347600e+006
%Y =8.696133357440095e+005
%Z = 5.830247232893203e+006

```

مثال: برنامه‌ای بنویسید که طول مشاهده شده بر روی زمین را به طول بر روی سطح مبنا (سطح بیضوی) تبدیل کند.

```

clc
clear
format long g
a=input('a= ');%a=6378137
f=input('f= ');%f=1/298.2572221;
b=a*(1-f);
%b=input('b= ');
e=sqrt(1-b^2/a^2);
phi1=input('phi(1) mat= ');%phi1=[66 34 59];
phi1=phi1(1,1)+(phi1(1,2)/60)+(phi1(1,3)/3600)
phi2=input('phi(2) mat= ');%phi2=[66 35 24];
phi2=phi2(1,1)+(phi2(1,2)/60)+(phi2(1,3)/3600)
phi=(phi1+phi2)/2;
N=a/(1-e^2*(sin(phi))^2)^.5;
M=(a*(1-e^2))/(1-e^2*(sin(phi))^2)^1.5;
R1=input('R1= ');%R1=6375701
R2=input('R2= ');%R2=6373636
R=(R1+R2)/2;
h1=input('h1= ');%h1=163.231
h2=input('h2= ');%h2=56.385
D=input('D= ');%D=7562.534
L0=sqrt((D^2-(h1-h2)^2)/((1+h1/R)*(1+h2/R)))
S=2*R*asin(L0/(2*R))
%L0 =7561.64893010818
%S =7561.64937343412

```

عملگرهای رابطه‌ای:

این عملگرها به بررسی روابط بین دو عبارت می پردازد.

== (مساوی): تساوی دو عبارت را بررسی می کند.

~= (مخالف): مخالف بودن دو عبارت را بررسی می کند.

< (کوچکتر): کوچکتر بودن دو عبارت را بررسی می کند.

> (بزرگتر): بزرگتر بودن دو عبارت را بررسی می کند.

<= (کوچکتر مساوی): کوچکتر یا مساوی بودن دو عبارت را بررسی می کند.

>= (بزرگتر مساوی): بزرگتر یا مساوی بودن دو عبارت را بررسی می کند.

خروجی عبارتهای رابطه‌ای و منطقی، یک عبارت منطقی می‌باشد. که رابطه یا درست است که به آن True می‌گوییم و مقدار یک دارد. و یا رابطه درست نیست که به آن False می‌گوییم و مقدار صفر دارد.

مثال: عملگرهای رابطه‌ای را بر روی دو عدد a,b بررسی کنید؟

a=5	b= -1	
a= b	→	ans=0
a~=b	→	ans=1
a<b	→	ans= 0
a<=b	→	ans=0
a>b	→	ans=1
a>=b	→	ans=1

- نکته:** در هنگام ذخیره کردن یک m-file باید توجه داشت که نامی را که برای ذخیره انتخاب می‌کنیم
- ۱- اعداد نباشد.
 - ۲- از دستورات برنامه matlab نباشد.
 - ۳- نام با فاصله نوشته نشود.

دستورات شرطی

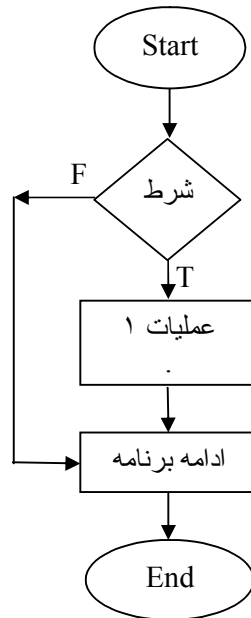
دستور if

در برخی از برنامه‌ها ما نیاز به تصمیم‌گیری در سر دو راهی‌ها یا سه راهی‌ها و یا ... را داریم. جهت انجام تصمیم‌گیری می‌توان از فرمان if استفاده نمود که دارای ساختارهای متفاوتی است که برحسب نیاز از آنها استفاده می‌شود.

(A) ساختار تک انتخابی

نکته: هر دستور if باید با یک دستور end بسته شود.

if جمله شرطی
 دستورات
 .
 .
 end



مثال: برنامه‌ای بنویسید که دو عدد را گرفته و به کمک دستور if ماکزیمم این دو عدد را چاپ کند؟

```

clc
clear
format short g
a=input('a=');
b=input('b=');
if a>b
m=a;
end
if a<b
m=b;
end
disp(' **max**')
disp(m)
  
```

مثال: برنامه‌ای بنویسید که یک عدد را گرفته و با یک پیغام مشخص کند که عدد زوج است یا فرد؟

```

clc
clear
format short g
a=input('a=');
if mod(a,2)==0
disp('zoz')
end
if mod(a,2)~=0
disp('fard')
end
  
```

مثال: برنامه‌ای بنویسید که ریشه‌های یک معادله درجه دو را بدست آورد و در صورتی که دلتا (b^2-4ac) منفی شود پیغامی مبنی بر عدم وجود ریشه چاپ کند؟

```

clc
clear
format short g
a=input('a=');
b=input('b=');
c=input('c=');
d=b^2-4*a*c;
if d<0
disp('rishh nadarad')
end
if d>0
D=d^0.5;
x1=(-1*b+D)/(2*a);
x2=(-1*b-D)/(2*a);
x=[x1 x2];
disp(' **** x1 ***** x2 ****')
disp(x)
end

```

مثال: برنامه‌ای بنویسید که با گرفتن سه زاویه با یک پیغام مشخص کند که آیا می‌توان با این سه زاویه یک مثلث تشکیل داد یا خیر؟

```

clc
clear
format short g
a=input('a=');
b=input('b=');
c=input('c=');
s=a+b+c;
if s==180
disp(' mosalas')
end
if s~=180
disp('no mosalas')
end

```

مثال: برنامه‌ای بنویسید که یک عدد را گرفته و با یک پیغام مثبت یا منفی بودن آن عدد را چاپ کند؟

```

clc
clear
format short g
a=input('a=');
if a>0
disp('+')


```


```
end
if a<0
disp('-')
end
```

مثال: برنامه‌ای بنویسید که برای ورود به برنامه یک کد عبور از کاربر درخواست کند در صورت درست بودن کد وارد برنامه شده و سه زاویه از یک مثلث

عملگرهای منطقی

عملگرهای منطقی روشی را برای ترکیب یا قرینه کردن عبارتهای رابطه‌ای ارائه می‌کنند.
& (AND): به معنای ((و)) می‌باشد .

| (OR): به معنای ((یا)) می‌باشد، با ترکیب کلید  Shift+ حاصل می‌شود.

~ (NOT): به معنای نه‌این و نه آن می‌باشد. با ترکیب کلید  Shift+ حاصل می‌شود.

اگر عبارت A و B داشته باشیم. عملگر NOT این گونه بر روی آنها عمل می‌کند.

A	B	NOT
T	T	F
F	T	T
T	F	T
F	F	F

مثال: برنامه‌ای بنویسید که عدد صحیحی را از ورودی گرفته و در صورت زوج و یک رقمی بودن آن، یک پیام ok دهد؟

```
clc
clear
a=input('a=');
if mod(a,2)==0 & (a<10 & a>-10)
disp('ok')
end
if mod(a,2)~=0 | (a>=10 | a<=-10)
disp('no')
end
```

مثال: برنامه‌ای بنویسید که یک عدد را از ورودی گرفته و با یک پیام مشخص کند که این عدد بر ۲ و ۵ بخش پذیر است یا خیر؟ (پیغام YES یا NO)

```
clc
clear
```

```

a=input('number=');
m1=mod(a,5);
m2=mod(a,2);
if m1==0&m2==0
disp('yes')
end
if m1~=0 | m2~=0
disp('no')
end

```

دستور if-else-end

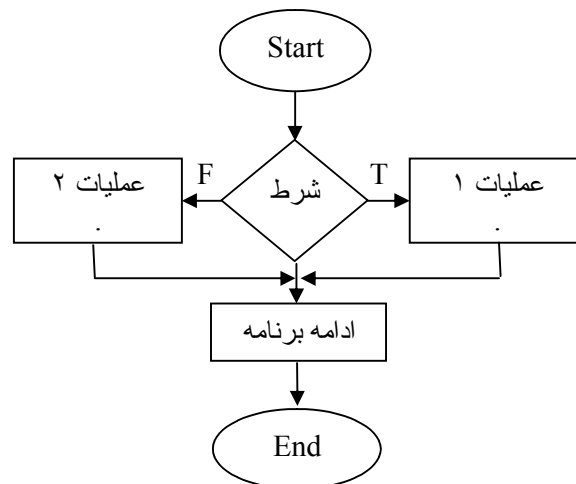
زمانی است که دو انتخاب داریم، یعنی می خواهیم در صورت برقراری شرط یک عملیات را انجام دهد و در صورت برقرار نبودن شرط عملیات را انجام دهد، از این دستور استفاده می کنیم که به آن ساختار دو انتخابی گفته می شود.

(B) ساختار دو انتخابی

```

if شرط
    عملیات ۱
    :
    :
else
    عملیات ۲
    :
    :
end

```



مثال: برنامه ای بنویسید که عددی را گرفته و در صورتی که این عدد مثبت بود یکی به آن اضافه کند و در غیر این صورت یکی از آن کم کند؟

```

clc
clear
format long g
a=input('number=');
if a>0
mosbat=a+1
else
manfi=a-1
end

```

مثال: برنامه ای بنویسید که عدد صحیحی را گرفته و در صورت زوج بودن، دوبرابر آن عدد و در صورت فرد بودن سه برابر آن عدد را در خروجی چاپ کند؟

```

clc

```

```

clear
format short g
a=input('number=');
if mod(a,2)==0
zoy=a*2
else
zoy=a*3
end

```

مثال: برنامه‌ای بنویسید که سه عدد صحیح را از ورودی گرفته و ماکزیمم این سه عدد را در خروجی چاپ کند؟ (بدون استفاده از فرمان max)

```

clc
clear
format short g
a=input('number A=');
b=input('number B=');
c=input('number C=');
max=a;
if b>max
max=b;
end
if c>max
max=c;
end
disp('*** max ***')
disp(max)

```

مثال: برنامه‌ای بنویسید که سه صحیح را از ورودی گرفته و ماکزیمم و مینیمم این سه عدد را در خروجی چاپ کند؟ (بدون استفاده از فرمان max و min)

```

clc
clear
format short g
a=input('A=');
b=input('B=');
c=input('C=');
if a>b
max=a;
min=b;
else
max=b;
min=a;
end
if c>max
max=c;

```

```

end
if c<min
min=c;
end
disp('*** max ****')
disp(max)
disp('*** min ****')
disp(min)

```

دستور if-elseif---end

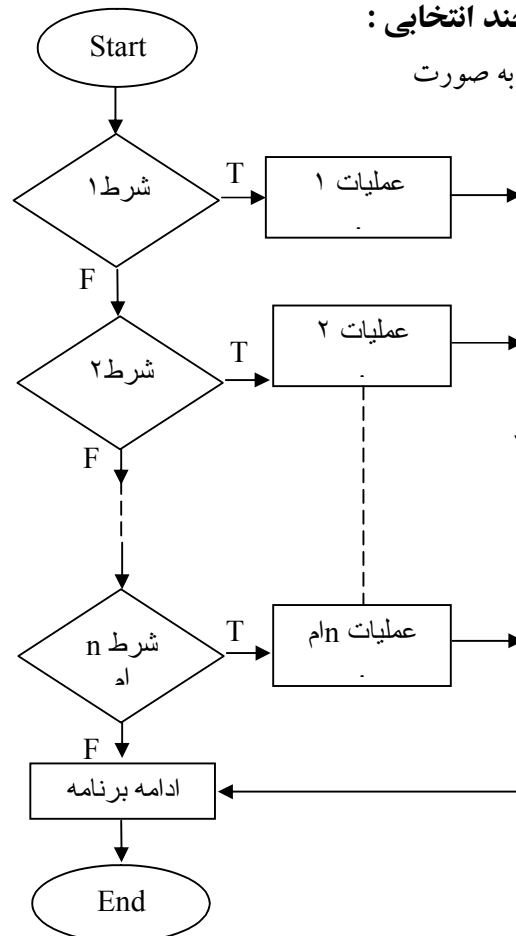
زمانی که ما بخواهیم شرطی بگذاریم که از دو انتخاب بیشتر باشد از این دستور استفاده می‌کنیم که حالت ساختار چند انتخابی دارد یا همان شرط های تو در تو .

(C) ساختار چند انتخابی :
حالت کلی آن به صورت

```

if           جمله شرطی اول
              عملیات اول
elseif      جمله شرطی دوم
              عملیات دوم
elseif      جمله شرطی سوم
              .
              .
              .
              عملیات n ام
end

```



نکته: اگر بین دستور elseif فاصله بگذاریم برای هر if یک end باید گذاشت .

مثال: برنامه‌ای بنویسید که معادله درجه دو را حل کند؟

```

clc
clear
format long
a=input('enter a (a*x^2+b*x+c):');
b=input('enter b (a*x^2+b*x+c):');
c=input('enter c (a*x^2+b*x+c):');
delta=b^2-4*a*c;
if a==0 & b~=0
x=-c/b;
elseif delta<0
disp('not equation')
elseif a~=0
x1=(-b+sqrt(delta))/(2*a)
x2=(-b-sqrt(delta))/(2*a)
elseif a==0 & b==0
disp('not equation')
end

```

مثال: برنامه‌ای بنویسید که مختصات دو سر یک امتداد را گرفته و طول و ژیزمان این امتداد را حساب کند؟ (بر حسب گراد)

```

clc
clear
format long g
x1=input('x1='); %vorod noghat aval
y1=input('y1='); %#ok<NASGU>
x2= input('x2=');% vorod noghat dovom
y2=input('y2=');
dx=x2-x1
dy=y2-y1
tol=sqrt(dx^2+dy^2);
disp('***tol***');
disp(tol);
v=atan(abs(dx/dy))*200/pi;
if dx>=0 & dy>=0
az=v;
elseif dx>=0 & dy<=0
az=200-v;
elseif dx<=0 & dy<=0
az=200+v;
elseif dx<=0 & dy>=0
az=400-v;
end
disp('azimuth=')
disp(az)

```


دستور شرطی switch-case

زمانی که می خواهیم از بین چندین عملیات یکی انجام شود که همگی آنها به یک شرط بستگی دارد بهتر است به جای استفاده از if-elseif از switch-case استفاده نمود.

صورت کلی شرط switch-case است.

switch

یک متغیر سنجش که می تواند عدد یا رشته باشد

case

متغیر سنجش دوم که با متغیر اول سنجش می شود
و در صورت درستی عملیات ۱ انجام می شود

عملیات ۱

case

متغیر سنجش سوم که با متغیر اول سنجش می شود
و در صورت درستی عملیات ۲ انجام می شود

عملیات ۲

case

.

.

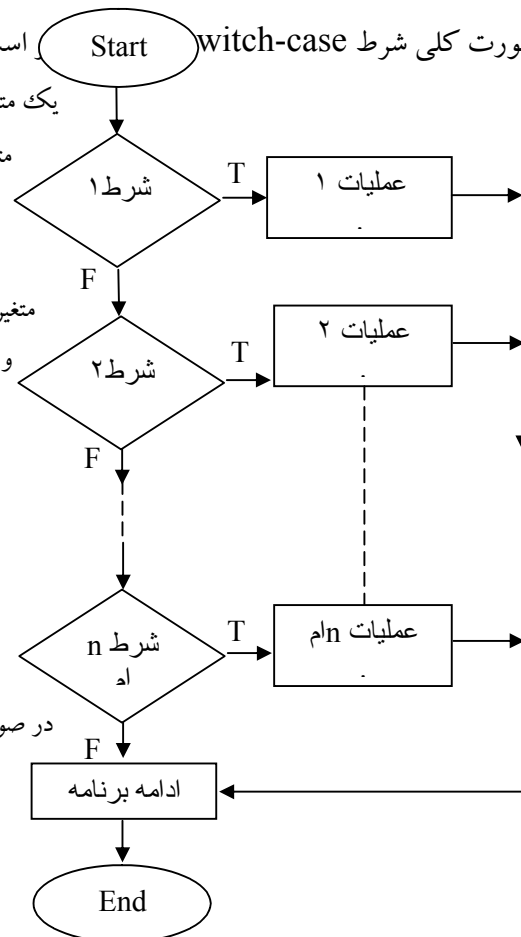
.

Otherwise

در صورتی که هیچ کدام از شروط برقرار نبود عملیات n ام را انجام بده

عملیات n ام

end



مثال: برنامه ای بنویسید که یک کاراکتر را از کاربر گرفته و تشخیص دهد که آیا کاراکتر Y یا X یا S را وارد کرده یا هیچ کدام را وارد نکرده؟

```
clc
clear
a=input('enter string=','s');
switch a
case 'y'
disp('yyy')
case 'n'
disp('nnn')
case 's'
disp('sss')
otherwise
disp('none')
```

```
end
```

مثال: برنامه‌ای بنویسید که دو عدد را از کاربر گرفته و با انتخاب کاربر عملیات جمع، تفریق، تقسیم یا ضرب را روی آنها انجام دهد و در صورت انتخاب صفر از برنامه خارج شود

```
clc
clear
format short g
a=input('number 1=> ');
b= input('number 2 => ');
disp('1:sum (+)')
disp('2:subtract (-)')
disp('3:divide (/)')
disp('4:multiply (*)')
disp('0:exit')
c=input('select number for(0-4): ');
switch c
case 1
disp(a+b)
case 2
disp(a-b)
case 3
disp(a/b)
case 4
disp(a*b)
case 0
exit
end
```

مثال: برنامه‌ای بنویسید که یک طول را برحسب سانتیمتر گرفته و آن را به دستور کاربر به یکی از واحد های متر، میلیمتر، سانتیمتر، فوت، اینچ تبدیل کند.

```
clc
clear
format short
disp('centimeter units distance')
x=input('distance =');
a=input('units=','s');
disp('input convert')
switch a
case {'inch','in'}
distance=x*2.54
case {'feet','ft'}
distance = x*2.54/12
```

```

case {'meter' , 'm'}
distance = x/100
case{'millimeter' , 'mm'}
distance = x*10
case {'centemeter' , 'cm'}
distance=x
otherwise
disp ('unknown units')
distance='non'
end

```

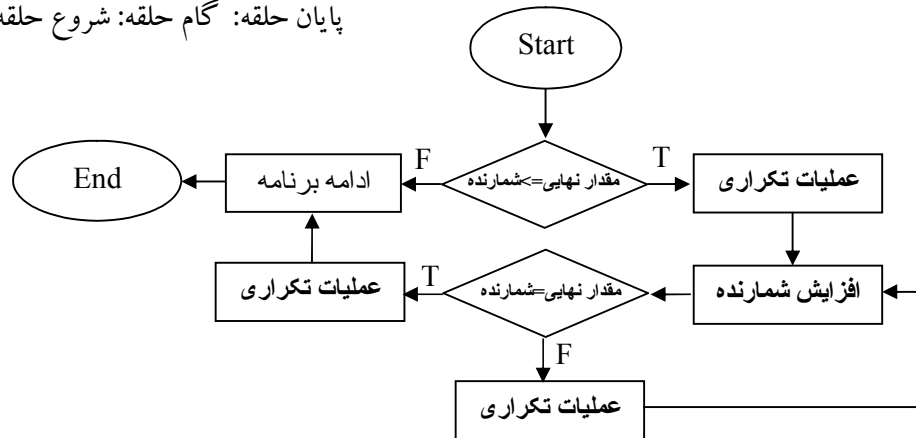
حلقه for

حلقه for این امکان را به گروهی از دستورات می دهد تا به صورت ثابت و به تعداد دفعات از پیش شده ای تکرار شوند. که ساختار کلی آن به صورت زیر می باشد.

```

for      پایان حلقه: گام حلقه: شروع حلقه = اسم متغیر
        عملیات تکراری
end

```



دستوراتی که بین for و end قرار دارند یک بار برای هر گاه اجرا می شود تا پایان حلقه.

مثال:

```

for n=1:10
x(n)=sin(n*pi/10);
end
x= 0.3090 0.5878 0.8090 0.9511 1.0000 0.9511 0.8090 0.5878 0.3090
0.0000

```

مثال: برنامه ای بنویسید که پنج عدد را گرفته و میانگین و ماکزیمم و مینیمم آنها را در خروجی نمایش دهد؟

```

clc
clear
format long g
for i=1:5
شمارنده می باشد و از یک شروع می شود و یکی یکی اضافه می شود تا به پنج برسد
a(i)=input('ai=')
end

```

نکته: اگر گام برنامه مشخص نشود برنامه یکی یکی جلوی رود، یعنی گام آن یک در نظر گرفته می شود.

```
for i=1:2:5
```

مثال: حلقه روبرو چند بار تکرار می شود

۳ بار

مثال: برنامه ای بنویسید که میانگین نمرات یک درس مربوط به دانشجویان یک کلاس ۳۰ نفری را گرفته و میانگین

نمرات دانشجویان کلاس، را مشخص کند؟

```
clc
clear
format long g
for i=1:30
a(i)=input('number= ');
end
m=mean(a)
```

مثال: برنامه ای بنویسید که تعداد دانشجویان کلاس را گرفته و سپس با گرفتن نمرات یک درس آنها میانگین کلاس

و مجموع نمراتشان را نشان دهد؟

```
clc
clear
format short g
n=input('tedad daneshjoyan=');
for i=1:n
a(i)=input('number=');
end
miyangin=mean(a)
majmoe=sum(a)
```

مثال: برنامه ای بنویسید که مجموع اعداد فرد از ۱ تا ۲۰ را چاپ کند؟

```
clc
clear
format short g
s=0;
for i=1:2:20
s=s+i;
end
s
```

مثال: برنامه ای بنویسید که مجموع اعداد زوج را از دو تا یک عدد دلخواه بدست آورد؟

مثال: برنامه‌ای بنویسید که اعداد زوج از ۲۰ تا ۲ را به صورت نزولی در خروجی چاپ کند؟

```
for i=20:-2:2
i
end
```

مثال: برنامه‌ای بنویسید که نمرات ۷ دانشجو را در یک ماتریس ستونی قرار دهد و چاپ کند؟

```
for i=1:7
a(i,:)=input('ai=');
end
a
یا
for i=1:7
a(i)=input('ai=');
end
a'
```

مثال: برنامه‌ای بنویسید اعداد یک تا پنج را در دو ضرب کرده و با ۲ جمع کند و نمایش دهد؟

```
for i=1:5
i=i*2+2
end
```

مثال: برنامه‌ای بنویسید که جدول ترازیبی تدریجی را گرفته و حل کند؟

```
clc
clear
format long g
n=input('number reading= ');
for i=1:n
a(i,1)=input('backside reading= ');
a(i,2)=input('forside reading= ');
end
ertefa=input('bm=');
b=.001*a(:,1);
c=.001*a(:,2);
deltaH=b-c
ertefanoghat=ertefa+deltaH
```

مثال: برنامه‌ای بنویسید که نمرات درس یک دانشجو را به همراه تعداد واحد گرفته و معدل آن را حساب کند و در صورت مشروط شدن با یک پیغام اعلام کند؟

```
clc
clear
format short g
n=input('tedad dars: ');
s=0;
majmoevahed=0;
disp('*****');
for i=1:n
nomreh=input('nomreh= ');
vahed=input('vahed= ');
disp('*****');
s=s+nomreh*vahed;
majmoevahed=majmoevahed+vahed;
end
moadel=s/majmoevahed
if moadel<12
disp('***<< mashrot >>***')
end
```

مثال: برنامه‌ای بنویسید که فاکتوریل یک عدد را محاسبه کند (بدون استفاده از فرمان factorial)

```
clc
clear
format short g
a=input('a=');
fact=1;
for i=a:-1:1
fact=fact*i;
end
fact
```

مثال: برنامه‌ای بنویسید که جذر اعداد ۱ تا ۱۰۰۰ را گرفته و اعدادی را که جذر کامل دارند در یک ماتریس قرار دهد.

مثال: برنامه‌ای بنویسید که با گرفتن مقادیر X و Y مقدار S را محاسبه کند. ($S = \sum X$)

```
clc
clear
format short g
n=input('n=');
x=input('x=');
s=0
for i=1:n
s=s+x^i;
end
s
```

مثال: برنامه‌ای بنویسید که با گرفتن مقادیر نمونه میانگین هندسی آنها را حساب کند. ($s = \sqrt[n]{\prod 1}$)

```
clc
clear
format short g
n=input('n=');
s=1
for i=1:n
li=input('li=');
s=s*li;
end
s=s^(1/n)
```

مثال: برنامه‌ای بنویسید که یک عدد صحیح بزرگتر از صفر را گرفته و مشخص کند این عدد اول هست یا نه؟

عدد اول عددی است که تنها بر خودش و یک بخشپذیر است

```
clc
clear
format short g
n=input('n=');
if n<4
a='aval';
end
for i=2:(n/2)
b=mod(n,i);
```

```

if b==0
a='aval nist';
end
if b~=0
a='aval';
end
end
disp(a)

```

مثال: برنامه‌ای بنویسید که یک عدد را گرفته و مقسوم علیه‌های آن را چاپ کند؟ (هر عدد به ۱ و خودش بخش پذیر است ← $j=2$)

```

clc
clear
format short g
a=input('a=');
c=(a/2);
j=2;
for i=2:c
if mod(a,i)==0
j=j+1;
end
end
j

```

مثال: برنامه‌ای بنویسید که یک عدد صحیح بزرگتر از صفر را گرفته و مشخص کند این عدد اول هست یا نه و در صورت اول نبودن مقسوم علیه‌های آن را در خروجی چاپ کند؟

```

clc
clear
format short g
n=input('x=');
j=2;
if n<4
a='aval';
disp(a);
end
for i=2:(n/2)
b=mod(n,i);
if b==0
j=j+1;
end
end
if j==2
disp('aval')

```



```

else
    disp('aval nist')
j
end

```

مثال: برنامه‌ای بنویسید که زوایای یک n ضلعی را گرفته و زوایا را سرشکن کند؟

```

clc
clear
format long g
disp('***n=tedad zele***')
n=input('n=');
disp('***alfa be grad*8*')
for i=1:n
    alfa(i)=input('alfa=');
end
e=(2*n-4)*100-sum(alfa);
c=e/n%correction;
for i=1:n
    delta(i)=alfa(i)+c;
end
delta

```

مثال: برنامه‌ای بنویسید که مقدار زوایای یک پلی گون n ضلعی را گرفته و در صورت مجاز بودن خطا، زوایا را سرشکن کند؟

```

clc
clear
format long g
disp('***n=tedad zele polygon***')
n=input('n=');
d.alfa=input('d.alfa=');
disp('***alfa be grad***')
for i=1:n
    alfa(i)=input('alfa=');
end
disp('***m=tedad defeat gharaate zavey***')
m=input('m=');
emax=2.5*d.alfa*sqrt(n/m);% برحسب ثانیه گراد می‌باشد
e=(2*n-4)*100-sum(alfa);% مقدار خطای بست زوایه‌ای
c=e/n;
if abs(emax)>abs(e)
for i=1:n
    delta(i)=alfa(i)+c;
end
delta

```

```
else
disp('out of error');
end
```

مثال: برنامه‌ای بنویسید که چندین طول را بگیرد و میانگین و انحراف معیار آن را چاپ کند؟

$$\sigma = \frac{\sum}{n}$$

```
clc
clear
format long g
n=input('n=')% تعداد طول اندازه گیری شده
for i=1:n
l(i)=input('li=');% طول اندازه گیری شده
end
L=sum(l)/n
s=0;
for i=1:n
s=(l(i)-L)^2+s;
end
sigma=s^(1/2)/(n^.5)
```

یا

```
clc
clear
format long g
n=input('n=')% تعداد طول اندازه گیری شده
for i=1:n
l(i)=input('l=');% طولهای اندازه گیری شده
end
L=mean(l);
sigma=std(l);
L
sigma
```

فرمان echo

هر گاه بخواهیم هر دستوری که اجرا می‌شود خود فرمان روی صفحه نمایش داده شود از فرمان `echo on` استفاده می‌کنیم و برای غیر فعال کردن آن از فرمان `echo off` استفاده می‌کنیم.

dir: کلید فایل‌های فهرست یا مسیر جاری را لیست می‌کند.

ls: همان کار فرمان `dir` را انجام می‌دهد.

type test: محتوای فایل test را نمایش می دهد. (اگر برنامه باشد کل برنامه را نشان می دهد)
cd: فهرست مسیرهای جاری را نشان می دهد.
delete test: فایل test.m را پاک می کند.
what: لیست m فایل ها و فایل هایی با پسوند mat را نمایش می دهد.

اجرای یک برنامه استفاده از فرمان cd :

برای اجرای یک برنامه ابتدا cd و سپس مسیر و نام برنامه را می نویسیم و اجرا می کنیم
مثال: اجرای برنامه test که در درایو f و فولدر works قرار دارد

```
cd f:\works\ali
```

طریقه آوردن نقشه کشورها و قاره ها در سیستم utm:

دو روش دارد:

(۱) با استفاده از نام کشور یا قاره:

```
>>world map('iran')
>>world map('asia')=world map('a')
South America=sa
North pole=np
South pole=sp
World=wo
Asia= a
Africa=af
Pacific=pa
Europe=e
North america=na
```

(۲) با استفاده از مشخصات محدوده جغرافیایی منطقه: λ, ϕ

```
>> worldmap ([ $\phi_1$  ,  $\phi_2$ ] , [ $\lambda_1$  ,  $\lambda_2$ ])
```

دستور num2str

با این دستور می توان یک عدد را به رشته (string) تبدیل نمود

```
a=num2str (12) => a= '12'
```

دستور str2num

با این دستور می توان یک را به عدد تبدیل کرد

```
a= '12' => c= str2num (a) => c=12
```

دستور tic toc

این دستور از دو قسمت تشکیل می‌شود: قسمت tic که بعد از فرمان clear و قسمت toc که در آخرین خط برنامه قرار می‌گیرند.

این دستور زمان انجام عملیات اجرای یک برنامه را حساب کرده و به ما نشان می‌دهد

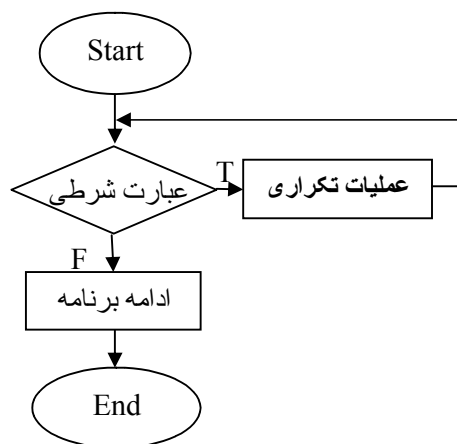
حلقه while

این حلقه بر خلاف حلقه for که گروهی از دستورات را به دفعات ثابتی اجرا می‌کند، می‌تواند گروهی از دستورات را به تعداد دفعات نامحدودی اجرا می‌کند.

صورت کلی این حلقه به صورت زیر می‌باشد.

```

while      شرط پایان حلقه
    عملیات ۱
    عملیات ۲
    .
    .
    .
    عملیات n
end
    
```



مثال: برنامه‌ای بنویسید که یک عدد را از کاربر گرفته و آن عدد را به توان دو برساند و تا زمانی که کاربر کاراکتر n به معنی انصراف را وارد نکرده این عمل را انجام دهد.

```

clc
clear
format long g
b='y'
a=input('number=')
while b ~= 'n'
    p=a^2;
    a=p
    disp('edameh => y');
    disp('enseraf => n')
    b=input('vorod: ', 's');
end
    
```

مثال: با استفاده از فرمان while برنامه‌ای بنویسید که یک عدد را بگیرد و فاکتوریل آن را در خروجی چاپ کند

```

clc
clear
format long g;
    
```

```

i=1;
f=1;
a=input('a=');
while i<a;
i=i+1;
f=f * i;
end;
disp(f)

```

مثال: برنامه‌ای بنویسید که یک سری مشاهدات را گرفته و مقدار ماکزیمم، مینیمم، مقدار میانه median، مقدار میانگین، انحراف معیار، واریانس این مشاهدات را در خروجی چاپ کند.

```

clc
clear
format long g
i=0;
b='y';
while b~='x'
a=input('number=');
i=i+1;
disp('repetition => y exit => x');
c(i,1)=a;
b=input('wat...?', 's');
end
MAX=max(c)
MIN =min(c)
MEDIAN= median(c)
VAR=var(c)
STD=std(c)
MEAN=mean(c)

```

مثال: برنامه‌ای بنویسید که مختصات کارتیزین را به مختصات منحنی الخط تبدیل کند.

```

X = 2381914.2501
= Y = 867157.75143 , a=6378137 , f=1/298.2572221
Z = 5852715.2350

```

```

clc
clear
format long g
a=6378137; %or a=input('a= ')
f=1/298.2572221; %or f=input('f= ')
b=a*(1-f);
e=2*f-f^2;
X=2.381914250138316e+006; %or X=input('X= ')

```

```

Y=8.671577514392249e+005; %or Y=input('Y= ')
Z=5.852715235032766e+006; %or Z=input('Z= ')
%for phi=66 34 59;lan=20 0 16;171.47256819997;
P=sqrt(X^2+Y^2);
lan=2*atan((-X+sqrt(X^2+Y^2))/Y);
N0=a;h0=(X^2+Y^2+Z^2)^.5-(a*b)^.5;phi0=atan(Z/P*(1-(e^2*N0)/(N0+h0))^-1);
N=1;h=1;phi=1;
while (abs(h-h0)>.00001)&(abs(phi-phi0)>.0000001)
N=a/(1-e^2*(sin(phi0))^2)^.5;
h=P/cos(phi0)-N0;
phi=atan(Z/P*(1-e^2*N0/(N0+h0))^-1);
N0=N;h0=h;phi0=phi;
end;
phi=phi*180/pi;
phi=abs([fix(phi) fix((fix(phi)-phi)*60) round((( fix(( fix(phi)phi)*60))-((fix(phi)-
phi)*60))*60)])
lan=lan*180/pi;
lan=abs([fix(lan) fix((fix(lan)-lan)*60) round((( fix(( fix(lan)lan)*60))-((fix(lan)-
lan)*60))*60)])
h

```

دستور prod

همان حاصلضرب می باشد \prod حاصلضرب، کلیه درآیه‌ها را در هم ضرب می کند

مثال:

$$a = \begin{pmatrix} 2 & 3 & -1 \\ 5 & 6 & 0 \\ 1 & 4 & 8 \end{pmatrix} \Rightarrow \text{prod}(a,1) = [10 \quad 72 \quad 0]$$

$$\text{prod}(a,2) = \begin{pmatrix} -6 \\ 0 \\ 32 \end{pmatrix}$$

مثال: برنامه‌ای بنویسید که چندین مشاهده یک طول را گرفته و در صورت وجود اشتباه در مشاهدات، مقدار آن را

نمایش داده و مقدار صحیح میانگین و انحراف معیار را نمایش دهد

```

clc
clear
format long g
i=0;
b='y';
z=input('accuracy instrument = ');
while b~= 'x'
a=input('distance =');
i=i+1;

```

```

disp('repetition =>y  exit =>x');
c(i,1)=a;
b=input('wate...?' , 's')
end
L=mean(c) ;
for e=1:i
v(e,1)=L
emax =2.5*z
if abs(v(e,1)) >abs(emax) ;
c(e,:)=[]
end
end
L =mean(c)

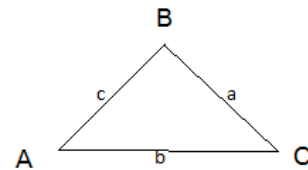
```

مثال: برنامه‌ای بنویسید که با گرفتن مختصات سه راس مثلث، طول هر ضلع و زوایای مثلث و مساحت و محیط مثلث را به ما بدهد

```

clc
clear
format long g
XA =input ('xA=');
YA= input ('yA=');
XB =input ('xB=');
YB= input ('yB=');
XC =input ('xC=');
YC= input ('yC=');
a= sqrt ((xC-Xb)^2 +(yC-Yb)^2)
b= sqrt ((xC-XA)^2 +(yC-YA)^2)
c= sqrt ((xB-XA)^2 +(yB-YA)^2)
A=a cos((c^2+b^2-a^2)/(2*c*b))
B=a cos((c^2+a^2-b^2)/(2*c*a))
C=a cos((a^2+b^2-c^2)/(2*a*b))
p= a+b+c ;
s=p/2 ;
Area=sqrt (s*(s-u)*(s-b)*s-c))

```



$$a = \sqrt{(x_C - x_B)^2 + (y_C - y_B)^2}$$

$$\hat{A} = \cos\left(\frac{b^2 + c^2 - a^2}{2bc}\right)$$

$$P = \frac{1}{2}bc \sin A$$

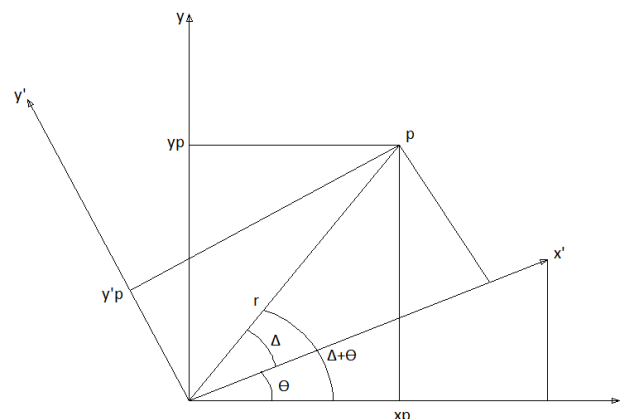
$$S = \frac{1}{2}(x_B - x_A)(y_C - y_A) - \frac{1}{2}(x_C - x_A)(y_B - y_A)$$

تبدیل دو سیستم مختصات دو بعدی به یکدیگر

$$\Rightarrow \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} ? \\ ? \end{bmatrix} * \begin{bmatrix} r \\ \theta \end{bmatrix}$$

$$Xp = r \cos(\Delta + \Theta) = r \cos \Delta \cdot \cos \Theta - r \sin \Delta \cdot \sin \Theta$$

$$*1 \Rightarrow x_p = x'_p \cos \Theta - y'_p \sin \Theta$$



$$Y_p = r \sin(\Delta + \theta) = r \sin \Delta \cdot \cos \theta - r \cos \Delta \cdot \sin \theta$$

$$*2 \Rightarrow y_p = y'_p \cos \theta - x'_p \sin \theta$$

$$\Rightarrow *1, *2 \Rightarrow \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} + \begin{pmatrix} y \\ z \end{pmatrix}$$

این ماتریس دوران برای زمانی است که θ در جهت عقربه ساعت است.

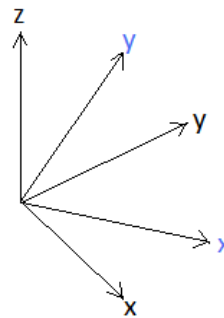
اگر θ در جهت عقربه ساعت باشد یک منفی در ماتریس دوران ضرب می‌شود و جای سینوسها عوض می‌شود.

اگر مبدا نیز تغییر مکان داده باشد داریم:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} + \begin{pmatrix} y \\ z \end{pmatrix}$$

اگر مقیاس سیستم مختصات دوم نیز تغییر کرده باشد، داریم:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \lambda \begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} + \begin{pmatrix} y \\ z \end{pmatrix}$$



تبدیل در حالت سه بعدی:

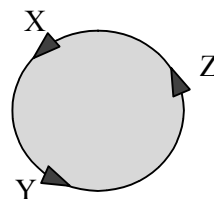
$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} + \begin{pmatrix} y \\ z \end{pmatrix}$$

پس داریم:

$$\omega : \text{حول } x \Rightarrow y \rightarrow z$$

$$\varphi : \text{حول } y \Rightarrow z \rightarrow x$$

$$\kappa : \text{حول } z \Rightarrow x \rightarrow y$$



ماتریس‌های دوران $(\omega, \varphi, \kappa)$:

خلاف عقربه‌های ساعت چرخش

$$m_{\omega} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \omega & \sin \omega \\ 0 & -\sin \omega & \cos \omega \end{pmatrix} \quad m_{\phi} = \begin{pmatrix} \cos & 0 & -\sin \\ 0 & 1 & 0 \\ \sin & 0 & \cos \end{pmatrix}$$

$$m_{\kappa} = \begin{pmatrix} \cos \kappa & \sin \kappa & 0 \\ -\sin \kappa & \cos \kappa & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \begin{pmatrix} y \end{pmatrix} = \begin{pmatrix} \cos & -\sin \\ \sin & \cos \end{pmatrix} \begin{pmatrix} y' \end{pmatrix}$$

$$\Rightarrow \begin{pmatrix} y' \\ y' \end{pmatrix} \begin{pmatrix} \cos & -\sin \\ \sin & \cos \end{pmatrix} \begin{pmatrix} y \end{pmatrix} \Rightarrow \begin{pmatrix} y' \\ y' \end{pmatrix} = \begin{pmatrix} \cos & \sin \\ -\sin & \cos \end{pmatrix} \begin{pmatrix} y \end{pmatrix}$$

مثال: برنامه‌ای بنویسید که مختصات یک نقطه را در سیستم X,Y بگیرد و سپس با گرفتن مقدار θ و مقدار انتقال سیستم X' و Y'، مختصات نقطه را در سیستم X' و Y' به ما بدهد

```

clc
clear
format long g
a=input('Xa= ');
b=input('Ya= ');
c=input('X0= ');
d=input('Y0= ');
disp('teta bar hasb darageh');
tata=input(' tata= ');
A=[a;b];
T=[c;d];
tata=tata*pi/180;
xy=[cos(tata),sin(tata);-sin(tata),cos(tata)]*A+T

```

نحوه ورود اطلاعات از نرم افزار Excel به Matlab:

دستور xlsread

جهت وارد کردن اطلاعات از نرم افزار اکسل به برنامه matlab از این فرمان می‌توان استفاده کرد. صورت کلی این دستور به شکل ذیل می‌باشد .

$A = \text{xlsread}(\text{'محدوده مورد نظر در شیت' , 'نام شیت مورد نظر' , 'نام فایل اکسل مورد نظر'})$

نکته: در هنگام استفاده از این دستور نباید فایل اکسل در حالت اجرا باشد. فایل اکسل مورد نظر باید در مسیر جاری برنامه matlab قرار داشته باشد و دارای فرمت xls باشد یعنی نباید فایل اکسل از 2003 بالاتر باشد.
مثال: اطلاعات موجود در محدوده E1 تا G25 در شیتی به نام number که در work book به نام Book1 وجود دارد را وارد برنامه matlab کنید.

```
A=xlsread('Book1','number','E1:G25')
```

نحوه خروج اطلاعات از نرم افزار Matlab به Excel:

دستور xlswrite

جهت خارج کردن اطلاعات از نرم افزار برنامه matlab به اکسل از این فرمان می توان استفاده کرد. صورت کلی این دستور به شکل ذیل می باشد.

```
A=xlswrite('محدوده مورد نظر در شیت','نام شیت مورد نظر','نام متغیر','نام فایل اکسل مورد نظر')
```

مثال: مقادیر ماتریس A را در محدوده E1 تا G25 در شیتی به نام number که در work book به نام Book1 خروجی بگیرد.

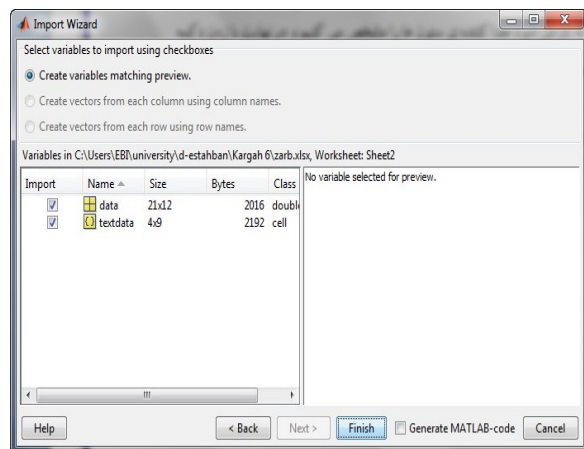
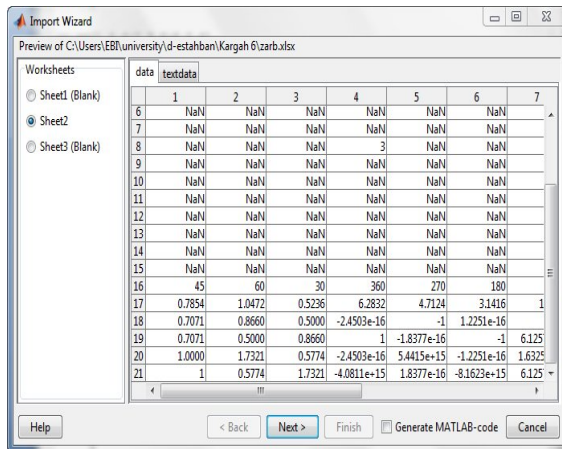
```
A=xlswrite('Book1','a','number','E1:G25')
```

نکته: اگر در آدرس دهی نام فایل خروجی اکسل یا نام شیت تعریف شده وجود نداشته باشد، به صورت خودکار آنها را ایجاد می کند.

ورود اطلاعات به برنامه matlab:

جهت این ورود اطلاعات می توان به مسیر File → Import Data رفت و فایل مورد نظر را انتخاب کرد. سپس در پنجره ای که باز می شود جدا کننده ی ستون ها را مشخص می کنیم و در نهایت با زدن دکمه Finish اطلاعات وارد می شوند.

برای ورود اطلاعات بجز روش بالا می توان از دستور uiimport نیز استفاده کرد. که با اجرای این دستور همانند روش قبل پنجره ای جهت ورود اطلاعات باز می شود که همانند روش قبل عمل می کنیم.



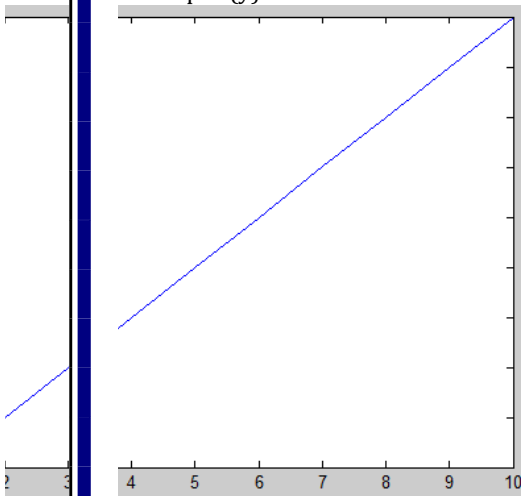
نمودارهای دو بعدی:

دستور plot

به کمک این دستور می توان نمودارهای منحنی دو بعدی را ترسیم کرد.
صورت پیش فرض این دستور به صورت $\text{plot}(x,y)$ می باشد .

مثال:

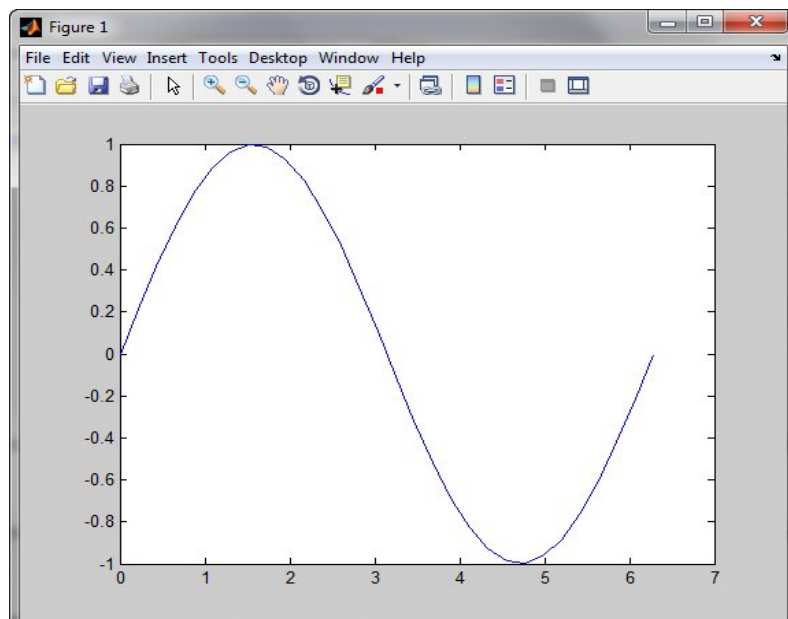
```
y=10:10:100;  
plot(y)
```



مثال:

```
x=linspace(0,2*pi,30);  
y=sin(x);  
plot(x,y)
```

e.rastguo@gmail.com



صورت کلی دستور plot به صورت ذیل می باشد .

$\text{plot}(\text{مقدار } x, \text{مقدار } y, \text{مقدار } \text{آن}, \text{نوع خط و رنگ آن}, \text{مشخصات خط}, \text{شاخص ها و مشخصات خط}, \text{مشخصات خط})$

شاخص های نوع خط:

شاخص	نوع خط
-	خط پیوسته (Solid)
--	خط چین (Dashed)
:	خط بصورت دو نقطه
-.	خط و نقطه

شاخص های رنگ خط:

شاخص های نوع علامت:

شاخص	نوع علامت
+	بعلاوه
o	دایره
*	ضربدر
S	مربع
d	لوزی
p	ستاره پنج رأس
h	ستاره شش رأس

شاخص	نوع رنگ
b	آبی
r	قرمز
g	سبز
y	زرد
k	سیاه
w	سفید
m	ارغوانی
c	فیروزه‌ای
o	زیتونی

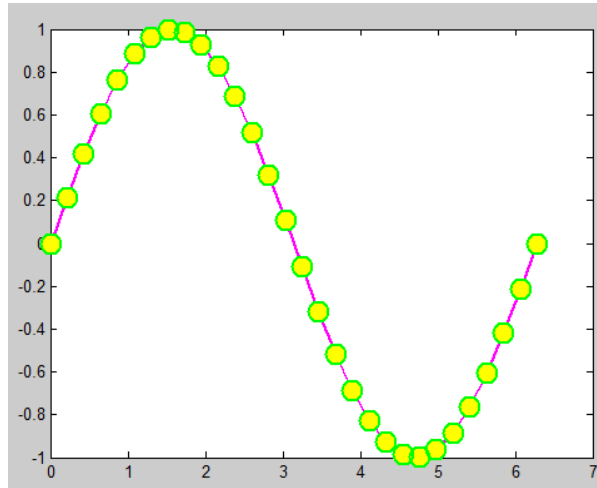
نقطه	
.	مثلاً رو به پایین
v	مثلاً رو به چپ
<	مثلاً رو به راست
>	مثلاً رو به بالا
^	

شاخص‌ها و مشخصات خط:

کاربرد دستور	دستور مورد استفاده
تعیین عرض خط	linewidth (مقدار ضخامت خط)
تعیین اندازه علامت	marksize (اندازه علامت)
تعیین رنگ لبه علامت‌های توپر	markeredgecolor (شاخص رنگ)
تعیین رنگ داخلی علامت‌های توپر	Markerfacecolor (شاخص رنگ)

مثال:

```
x=linspace(0,2*pi,30);
y=sin(x);
plot(x,y,'-mo','linewidth',2,'markersize',12,'markeredgecolor','g','markerfacecolor','y')
```



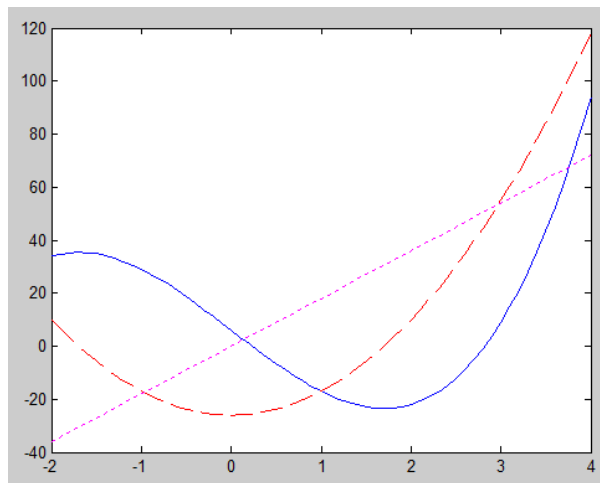
ترسیم چند نمودار روی هم در یک پنجره figure :

صورت کلی آن بصورت زیر می باشد .

plot(x₁ , y₁ , 'تنظیمات نمودار اول', x₂ , y₂ , 'تنظیمات نمودار دوم',')

مثال:

```
x=[-2:1:4];
y1=3*x.^3-26*x+6;
y2=9*x.^2-26;
y3=18*x;
plot(x,y1,'-b',x,y2,'-r',x,y3,':m')
```



دستور hold

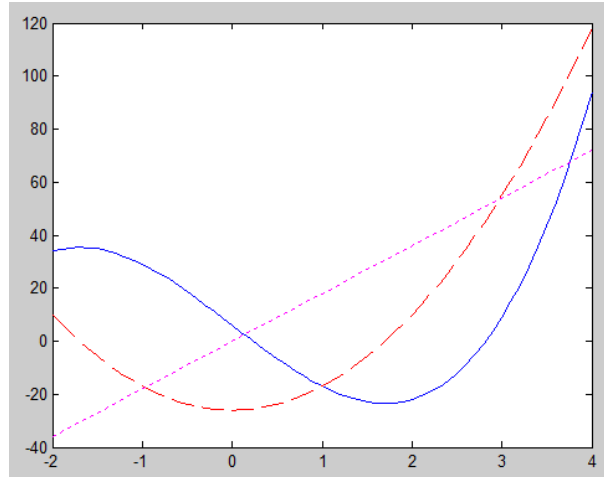
می توان بجای اینکه بصورت متوالی متغیرها را در یک فرمان plot اجرا کنیم، از فرمان hold برای ترکیب نمودارهای مختلف استفاده کنیم. به این صورت که نمودار اول را با دستور plot ترسیم کرده سپس برای ترسیم نمودارهای بعدی بر روی این نمودار ابتدا دستور hold on را اجرا کرده و بعد از دستور plot برای ترسیم نمودارهای بعدی استفاده می کنیم. زمانی که دیگر نمی خواهیم نمودارها بر روی هم ترسیم شوند از دستور hold off استفاده می کنیم.

مثال: مثال قبل را با استفاده از دستور hold اجرا کنید.

```

x=[-2:1:4];
y1=3*x.^3-26*x+6;
y2=9*x.^2-26;
y3=18*x;
plot(x,y1,'-b')
hold on
plot(x,y2,'--r')
plot(x,y3,':m')
hold off

```



دستور title

اگر بخواهیم برای گراف خود عنوانی بنویسیم از این دستور استفاده می‌کنیم. صورت کلی این دستور بصورت ذیل می‌باشد.

```
title('متن عنوان گراف')
```

دستور xlabel و ylabel

به کمک این دو دستور می‌توانیم در کنار محورهای X و Y برچسبی چاپ کرد. صورت کلی این دستور بصورت ذیل می‌باشد.

```
xlabel('متن کناری محور xها')
```

```
ylabel('متن کناری محور yها')
```

دستور text

اگر بخواهیم درون گراف متنی بنویسیم از این دستور می‌توان استفاده کرد. صورت کلی این دستور بصورت ذیل می‌باشد.

```
text('متن مورد نظر', مقدار Y محل قرار گیری متن , مقدار X محل قرار گیری متن)
```

دستور gtext

این دستور همان دستور text است با این تفاوت که محل قرار گیری متن با ماوس مشخص می‌شود. صورت کلی این دستور بصورت ذیل می‌باشد.

```
gtext('متن مورد نظر')
```

دستور legend

به کمک این دستور می توان برای هر یک از نمودارهای ترسیم شده در یک پنجره figure یک راهنما ایجاد کرد. صورت کلی این دستور بصورت ذیل می باشد .

('محل قرار گیری راهنما'....., 'متن راهنمای اولین نمودار', 'متن راهنمای اولین نمودار') legend

متن های راهنمایی که نوشته می شود به ترتیب در مقابل نوع خط نمودار اول، دوم،... تا آخرین نمودار قرار می گیرد. محل قرار گیری راهنمای نمودار با عدد ۱- تا ۴ مشخص می گردد.

عدد	محل قرار گیری راهنمای در نمودار
۱-	خارج از مرزهای محورها در سمت راست
۰	داخل مرزهای محورها که در یک محل با کمترین تداخل با نمودار ها
۱	بالا سمت راست (به صورت پیش فرض)
۲	گوشه بالا و سمت چپ
۳	پائین و سمت چپ
۴	گوشه پائین و سمت راست

تعیین نوع فونت دستورهای title و text و gtext و xlabel و ylabel و legend

زمانی که از دستورهای فوق برای نوشتن متن استفاده می کنیم می توان نوع فونت متن را به کمک دستورهای ذیل تغییر داد.

دستور	عمل کرد دستور
\bf	باعث می شود فونت بولد شود.
\it	باعث می شود فونت ایتالیک (کج) شود.
\rm	باعث می شود فونت نرمال شود.
\fontname{ نام فونت مورد نظر }	مشخص کردن فونت متن
\fontsize{ اندازه فونت مورد نظر }	مشخص کردن اندازه فونت متن

بکارگیری نمادهای یونانی در نوشته های گراف:

نمادهای مختلف در برنامه matlab با یک سری کاراکترها تعریف شده اند که در ذیل تعدادی از آنها آورده شده.

Character Sequence	Symbol	Character Sequence	Symbol	Character Sequence	Symbol
<code>\alpha</code>	α	<code>\upsilon</code>	υ	<code>\sim</code>	\sim
<code>\beta</code>	β	<code>\phi</code>	ϕ	<code>\leq</code>	\leq
<code>\gamma</code>	γ	<code>\chi</code>	χ	<code>\infty</code>	∞
<code>\delta</code>	δ	<code>\psi</code>	ψ	<code>\clubsuit</code>	\clubsuit
<code>\epsilon</code>	ϵ	<code>\omega</code>	ω	<code>\diamondsuit</code>	\diamondsuit
<code>\zeta</code>	ζ	<code>\Gamma</code>	Γ	<code>\heartsuit</code>	\heartsuit
<code>\eta</code>	η	<code>\Delta</code>	Δ	<code>\spadesuit</code>	\spadesuit
<code>\theta</code>	θ	<code>\Theta</code>	Θ	<code>\leftrightharpoonright</code>	\leftrightharpoonright
<code>\vartheta</code>	ϑ	<code>\Lambda</code>	Λ	<code>\leftarrow</code>	\leftarrow
<code>\iota</code>	ι	<code>\Xi</code>	Ξ	<code>\uparrow</code>	\uparrow
<code>\kappa</code>	κ	<code>\Pi</code>	Π	<code>\rightarrow</code>	\rightarrow
<code>\lambda</code>	λ	<code>\Sigma</code>	Σ	<code>\downarrow</code>	\downarrow
<code>\mu</code>	μ	<code>\Upsilon</code>	Υ	<code>\circ</code>	\circ
<code>\nu</code>	ν	<code>\Phi</code>	Φ	<code>\pm</code>	\pm
<code>\xi</code>	ξ	<code>\Psi</code>	Ψ	<code>\geq</code>	\geq
<code>\pi</code>	π	<code>\Omega</code>	Ω	<code>\propto</code>	\propto
<code>\rho</code>	ρ	<code>\forall</code>	\forall	<code>\partial</code>	∂
<code>\sigma</code>	σ	<code>\exists</code>	\exists	<code>\bullet</code>	\bullet
<code>\varpi</code>	ϖ	<code>\ni</code>	\ni	<code>\div</code>	\div
<code>\tau</code>	τ	<code>\cong</code>	\cong	<code>\neq</code>	\neq
<code>\equiv</code>	\equiv	<code>\approx</code>	\approx	<code>\aleph</code>	\aleph
<code>\Im</code>	\Im	<code>\Re</code>	\Re	<code>\wp</code>	\wp
<code>\otimes</code>	\otimes	<code>\oplus</code>	\oplus	<code>\oslash</code>	\oslash
<code>\cap</code>	\cap	<code>\cup</code>	\cup	<code>\supseteq</code>	\supseteq
<code>\supset</code>	\supset	<code>\subseteq</code>	\subseteq	<code>\subset</code>	\subset
<code>\int</code>	\int	<code>\in</code>	\in	<code>\emptyset</code>	\emptyset
<code>\rfloor</code>	\rfloor	<code>\lceil</code>	\lceil	<code>\nabla</code>	∇
<code>\lfloor</code>	\lfloor	<code>\cdot</code>	\cdot	<code>\ldots</code>	\dots
<code>\perp</code>	\perp	<code>\neg</code>	\neg	<code>\prime</code>	\prime
<code>\wedge</code>	\wedge	<code>\times</code>	\times	<code>\emptyset</code>	\emptyset
<code>\rceil</code>	\rceil	<code>\surd</code>	\surd	<code>\mid</code>	\mid
<code>\vee</code>	\vee	<code>\varpi</code>	ϖ	<code>\copyright</code>	\copyright
<code>\langle</code>	\langle	<code>\rangle</code>	\rangle		

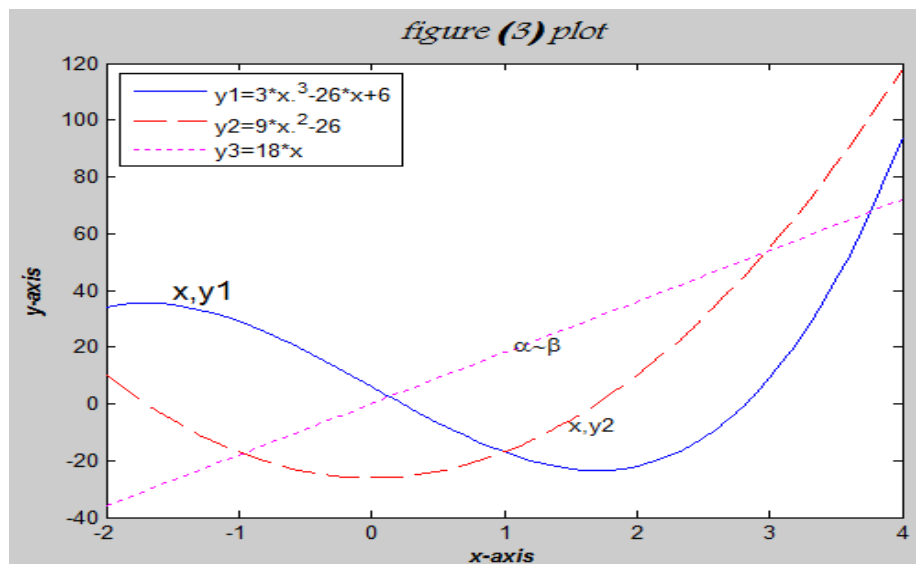
مثال:

`x=[-2:1:4];`

```

y1=3*x.^3-26*x+6;
y2=9*x.^2-26;
y3=18*x;
plot(x,y1,'-b')
xlabel('\bf\itx-axis')
ylabel('\bf\ity-axis')
title('\it\bf\fontname{titr}\fontsize{14} igure (3) plot')
text(-1.5,40,'\fontsize{14}x,y1')
hold on
plot(x,y2,'--r')
gtext('\rm x,y2')
plot(x,y3,':m')
gtext('\alpha\sim\beta')
legend('y1=3*x.^3-26*x+6','y2=9*x.^2-26','y3=18*x',2)
hold off

```



دستور box off و box on

دستور box on همه خطوط اطراف گراف را بجز خطوط محور مختصات را غیر فعال می کند.

دستور box off همه خطوط اطراف گراف را فعال می کند.

دستور grid off و grid on

دستور grid on شبکه مختصات بصورت خط چین بر روی نمودار فعال می کند.

دستور grid off شبکه مختصات بر روی نمودار را غیر فعال می کند.

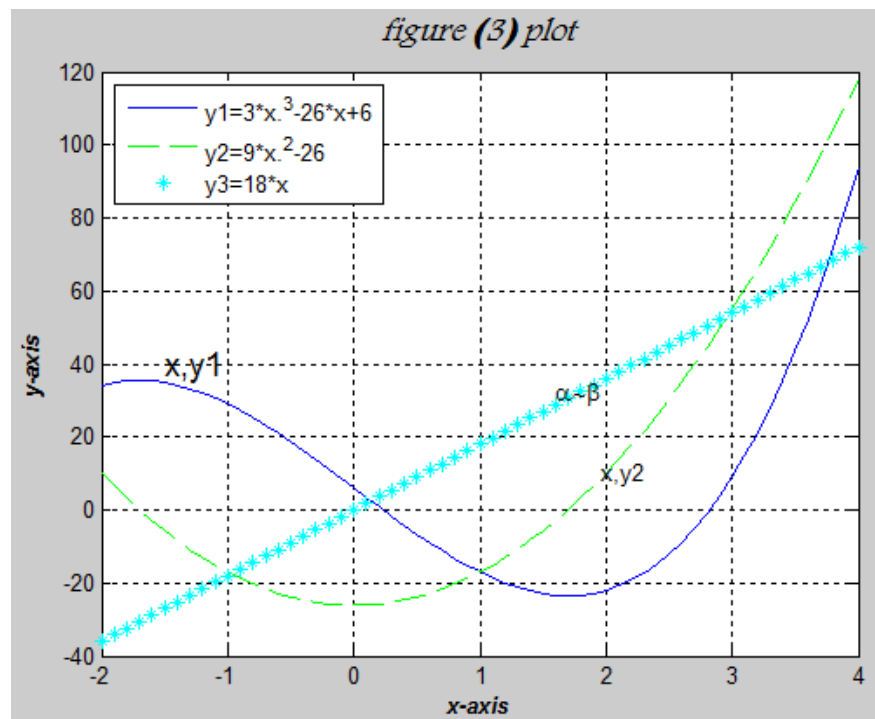
اگر دستور grid یا box را بدون پسوند on,off بکار ببریم اگر فعال باشد آن را غیر فعال و در غیر این صورت آن را فعال می کند.

مثال:

```

x=[-2:1:4];
y1=3*x.^3-26*x+6;
y2=9*x.^2-26;
y3=18*x;
plot(x,y1,'-b')
xlabel('\bf\itx-axis')
ylabel('\bf\ity-axis')
grid
title('\it\bf\fontname{titr}\fontsize{14} igure (3) plot')
text(-1.5,40,'\fontsize{14}x,y1')
hold on
pause(3)
plot(x,y2,'--g')
gtext('\rm x,y2')
grid
box
plot(x,y3,'*c')
pause(3)
box
grid
gtext('\alpha\sim\beta')
legend('y1=3*x.^3-26*x+6','y2=9*x.^2-26','y3=18*x',2)
hold off

```



دستورات محور مختصات

axis([xmin xmax ymin ymax]): محدوده نمایش محورهای مختصات را مشخص می کند.

axis auto: درجه بندی محور مختصات را به حالت پیش فرض برمی گرداند.

axis ij : محور مختصات را به حالت ماتریس تنظیم می کند. محور افقی از سمت چپ به راست و محور قائم از سمت بالا به پائین افزایش می یابند.

axis xy : محور مختصات را به حالت کارترین تنظیم می کند. محور افقی از چپ به راست و محور قائم از پائین به بالا افزایش می یابند.

axis equal : ضرایب تناسب را به گونه ای تنظیم می کند که فاصله بین دو علامت بر روی محورهای مختصات با هم به یک اندازه باشند.

axis image : برای نمایش یک تصویر محدوده محورهای مختصات را تنظیم می کند.

axis square : اندازه محورهای مختصات را به شکل مربع در می آورد.

axis normal : اندازه جعبه محورهای مختصات را به اندازه حداکثر تغییرات داده ها می رساند و هر نوع محدودیتی را از واحد اندازه گیری برمی دارد.

axis off : تمام حالت های مربوط به برچسب های محورهای مختصات، علائم و زمینه نمودار را غیر فعال می کند.

axis on : تمام حالت های مربوط به برچسب های محورهای مختصات، علائم و زمینه نمودار را فعال می کند.
نکته: می توان چند دستور تابع axis را یکباره انجام داد. مثلاً:

axis on auto xy

figure(number) : اگر بخواهیم پنجره گرافیکی جدیدی با شماره دلخواهی ایجاد کنیم از این دستور استفاده می کنیم.

clf : هر گاه بخواهیم تنها پنجره گرافیکی را پاک کنیم و نمودار را پاک نکنیم از دستور **clf (clear figure)** استفاده می کنیم.

clg : هر گاه بخواهیم کل پنجره گرافیکی را با نمودارهای آن پاک کنیم از دستور **clg (clear graph window)** استفاده می کنیم.

close (window figure number) : اگر بخواهیم پنجره گرافیکی را ببندیم از این دستور استفاده می کنیم.

close all : اگر بخواهیم تمام پنجره های گرافیکی را ببندیم از این دستور استفاده می کنیم.

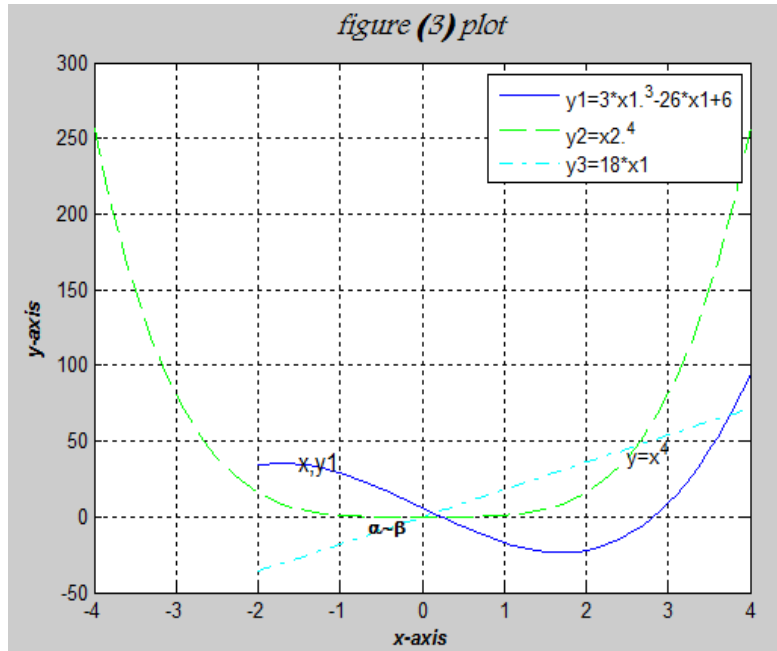
مثال:

```
x1=-2:.1:4;  
x2=-4:.1:4;  
y1=3*x1.^3-26*x1+6;  
y2=x2.^4;  
y3=18*x1;  
figure(4)  
pause(2)  
plot(x1,y1,'-b')  
grid  
pause(2)
```

```

xlabel('\bf\itx-axis')
pause(2)
ylabel('\bf\ity-axis')
pause(2)
axis([-4 4 -30 70])
pause(2)
axis ij
pause(2)
title('\it\bf\fontname{titr}\fontsize{14} figure (3) plot')
pause(2)
text(-1.5,35,'\fontsize{12}x,y1')
hold on
pause(2)
plot(x2,y2,'-g')
pause(2)
gtext('\rm y=x^4')
axis square
pause(1)
plot(x1,y3,'-c')
pause(1)
gtext('\bf\alpha\sim\beta')
legend('y1=3*x1.^3-26*x1+6',
'26*x1+6','y2=x2.^4','y3=18*x1')
pause(1)
axis auto
pause(1)
axis normal xy
pause(5)
axis off
pause(2)
clf
pause(1)
close(4)
hold all

```

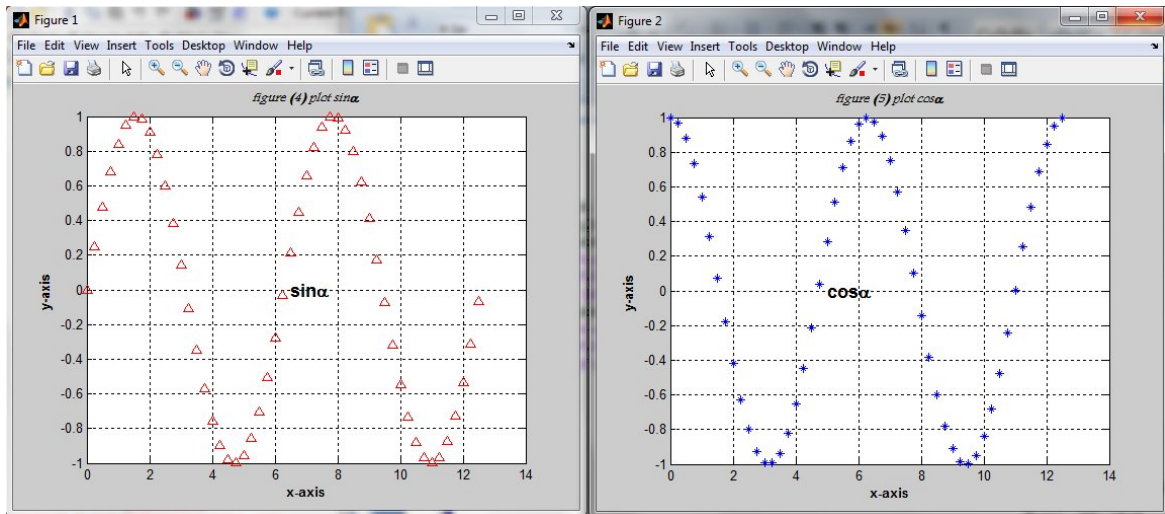


مثال:

```

x=0:.25:4*pi;
y1=sin(x);y2=cos(x);
figure(1);plot(x,y1,r'^');grid
xlabel('\bfx-axis');ylabel('\bfy-axis');
title('\it\bf\fontname{titr} figure (4) plot sin\alpha')
text(6.5,0,'\bf\fontsize{14}sin\alpha')
figure(2);plot(x,y2,b'*');grid
xlabel('\bfx-axis');ylabel('\bfy-axis');
title('\it\bf\fontname{titr} figure (5) plot cos\alpha')
text(5,0,'\bf\fontsize{14}cos\alpha')

```

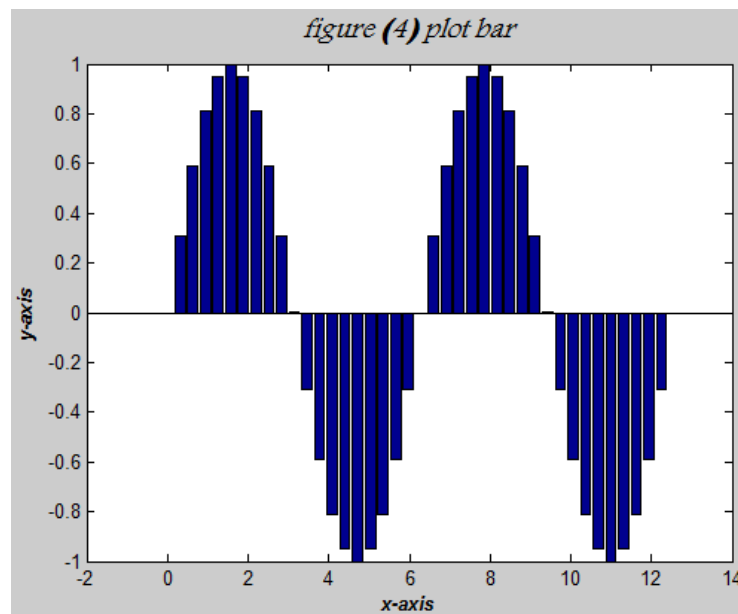
انواع نمودارهای دو بعدی دیگر

نمودار میله‌ای قائم (bar)

شکل کلی این دستور بصورت $\text{bar}(x,y)$ می‌باشد.

مثال:

```
x=0:pi/10:4*pi;
y=sin(x);
bar(x,y)
xlabel('\bf\itx-axis')
ylabel('\bf\ity-axis')
title('\it\bf\fontname{titr}\fontsize{14} figure (4) plot bar')
```

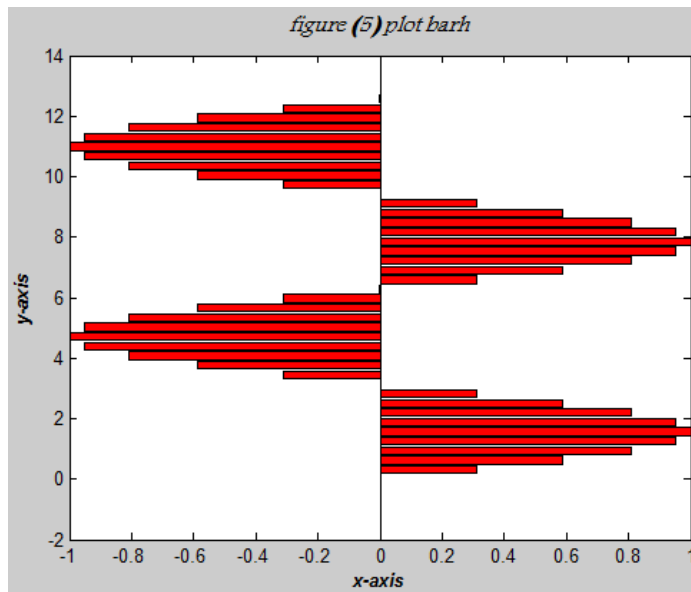


نمودار میله‌ای افقی (barh)

شکل کلی این دستور بصورت $\text{barh}(x,y)$ می‌باشد .

مثال:

```
x=0:pi/10:4*pi;  
y=sin(x);  
barh(x,y,'r')  
xlabel('\bf\itx-axis')  
ylabel('\bf\ity-axis')  
title('\it\bf\fontname{titr}\fontsize{12} igure (5) plot barh)
```

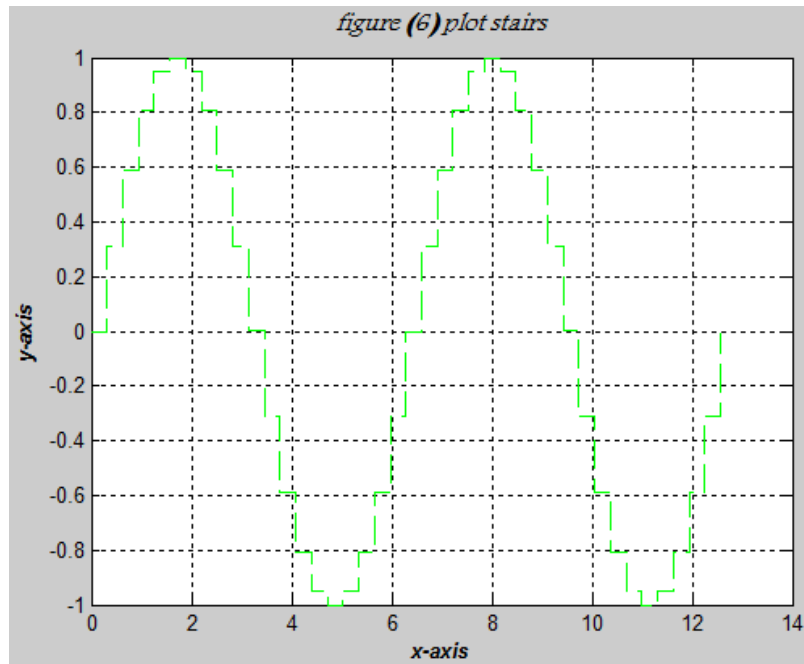


نمودار پلکانی (stairs)

شکل کلی این دستور بصورت $\text{stairs}(x,y)$ می‌باشد .

مثال:

```
x=0:pi/10:4*pi;  
y=sin(x);  
stairs(x,y,'-g')  
xlabel('\bf\itx-axis')  
ylabel('\bf\ity-axis')  
title('\it\bf\fontname{titr}\fontsize{12} igure (6) plot stairs')  
grid
```

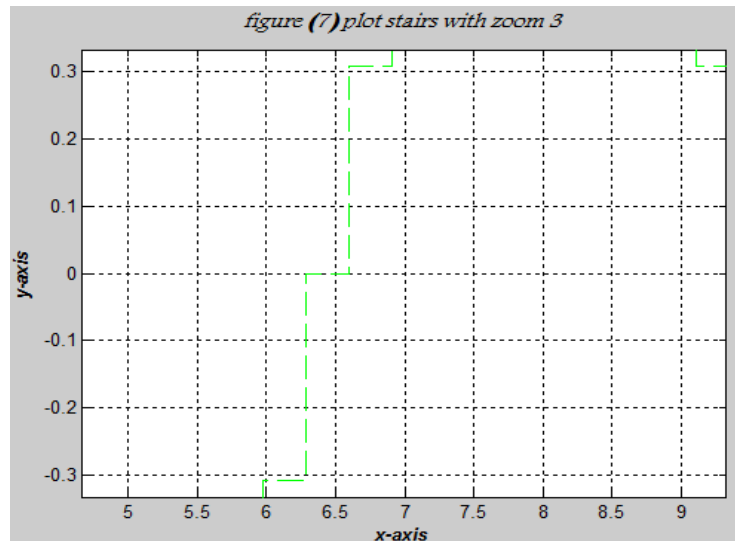


دستور zoom

به کمک این دستور می‌توان بزرگ‌نمایی پنجره گرافیکی را تنظیم کرد. شکل کلی این دستور بصورت `zoom(factor)` می‌باشد.

مثال:

```
x=0:pi/10:4*pi;
y=sin(x);
stairs(x,y,'-g')
xlabel('\bf\itx-axis')
ylabel('\bf\ity-axis')
title('\it\bf\fontname{titr}\fontsize{12} figure (7) plot stairs with zoom 3)
grid
pause(2)
zoom (3)
```

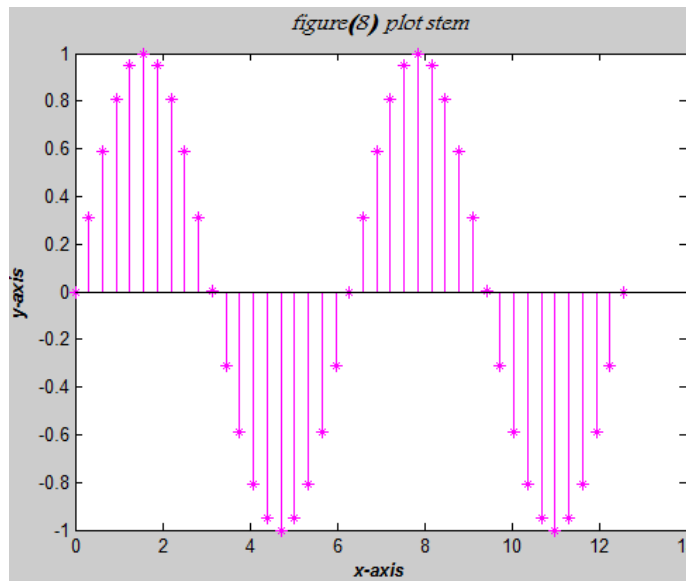


نمودار ساقه‌ای (stem)

شکل کلی این دستور بصورت $\text{stem}(x,y)$ می باشد .

مثال:

```
x=0:pi/10:4*pi;  
y=sin(x);  
stem(x,y,'* m')  
xlabel('\bf\itx-axis')  
ylabel('\bf\ity-axis')  
title('\it\bf\fontname{titr}\fontsize{12} igure(8) plot stem)
```



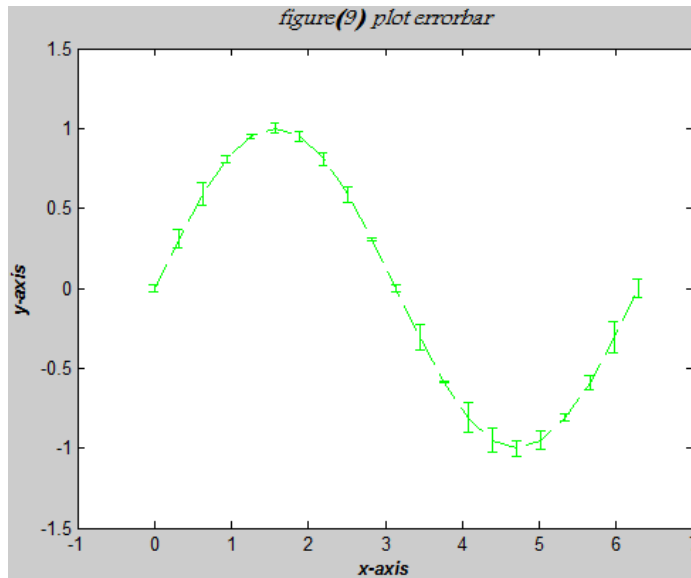
دستور errorbar

به کمک این دستور می توان نمودار میله ای رسم کرد که تلورانس خطا را نیز نشان دهد. صورت کلی این دستور بصورت ذیل می باشد .

$\text{errorbar}(x,y, \text{مقدار خطا برای هر نقطه})$

مثال:

```
x=0:pi/10:2*pi;  
y=sin(x);  
e=rand(size(x))/10  
errorbar(x,y,e,'--g')  
xlabel('\bf\itx-axis')  
ylabel('\bf\ity-axis')  
title('\it\bf\fontname{titr}\fontsize{12} igure(9) plot errorbar)
```

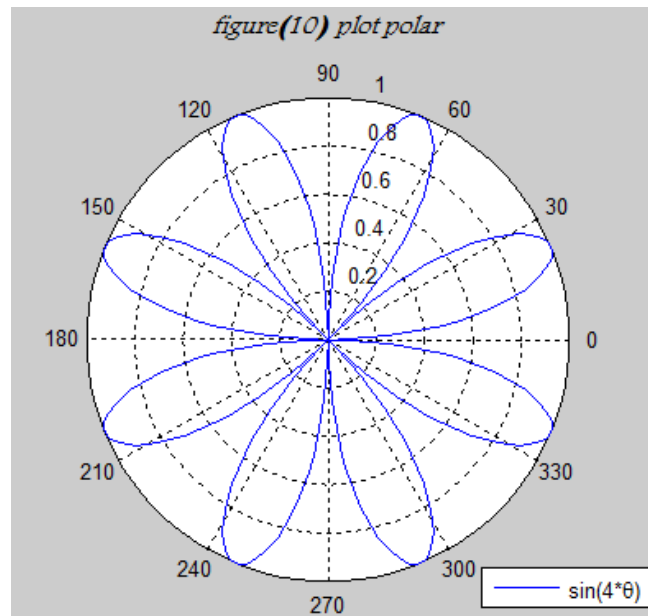


نمودار قطبی (polar)

شکل کلی این دستور بصورت (θ, r) می باشد .

مثال:

```
teta=0:pi/150:2*pi;
polar(teta,sin(4.*teta),'-b')
title('\it\bf\fontname{titr}\fontsize{12} igure(10) plot polar]
legend('sin(4*\theta)',0)
```

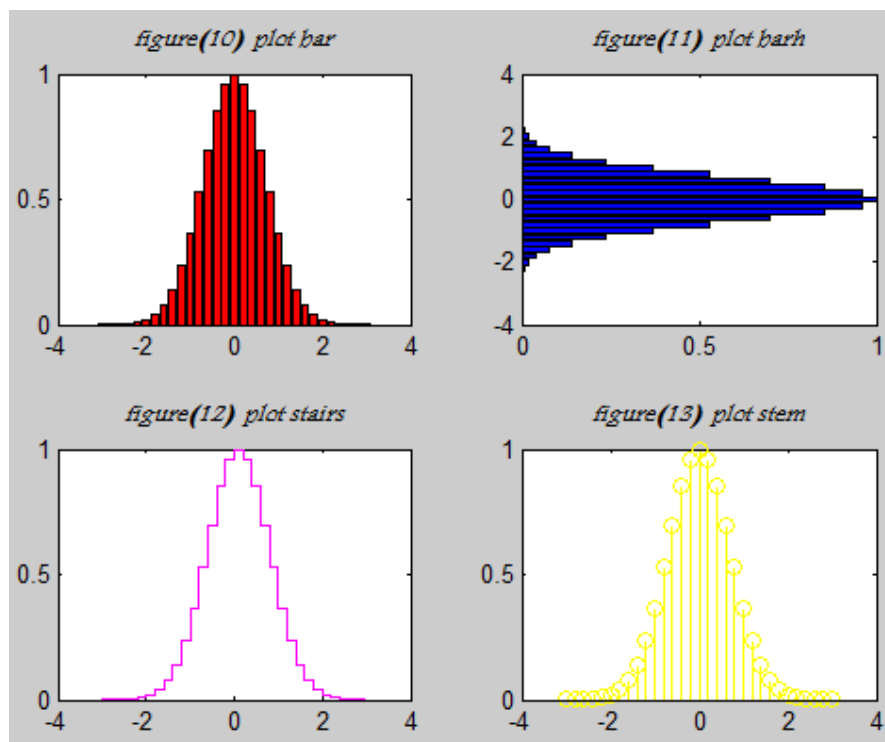


ترسیم زیر نمودار یا چندین نمودار در یک پنجره گرافیکی (subplot)

می‌توان در یک پنجره figure چندین نمودار را رسم کرد جهت این کار از دستور subplot استفاده می‌کنیم. که شکل کلی این دستور به شکل subplot(m,n,p) می‌باشد. که m و n در اصل مانند یک ماتریس تعداد نمودارها در سطر (m) و تعداد نمودارها در ستون (n) را مشخص کرده و p محل هر زیر نمودار را در این ماتریس (پنجره figure) مشخص می‌کند.

مثال:

```
x=-3:2:3;
y=exp(-x.*x);
subplot(2,2,1)
bar(x,y,'r')
title('\it\bf\fontname{titr} figure(10) plot bar)
subplot(2,2,2)
barh(x,y,'b')
title('\it\bf\fontname{titr} figure(11) plot barh)
subplot(2,2,3)
stairs(x,y,'m')
title('\it\bf\fontname{titr} figure(12) plot stairs)
subplot(2,2,4)
stem(x,y,'y')
title('\it\bf\fontname{titr} figure(13) plot stem)
```



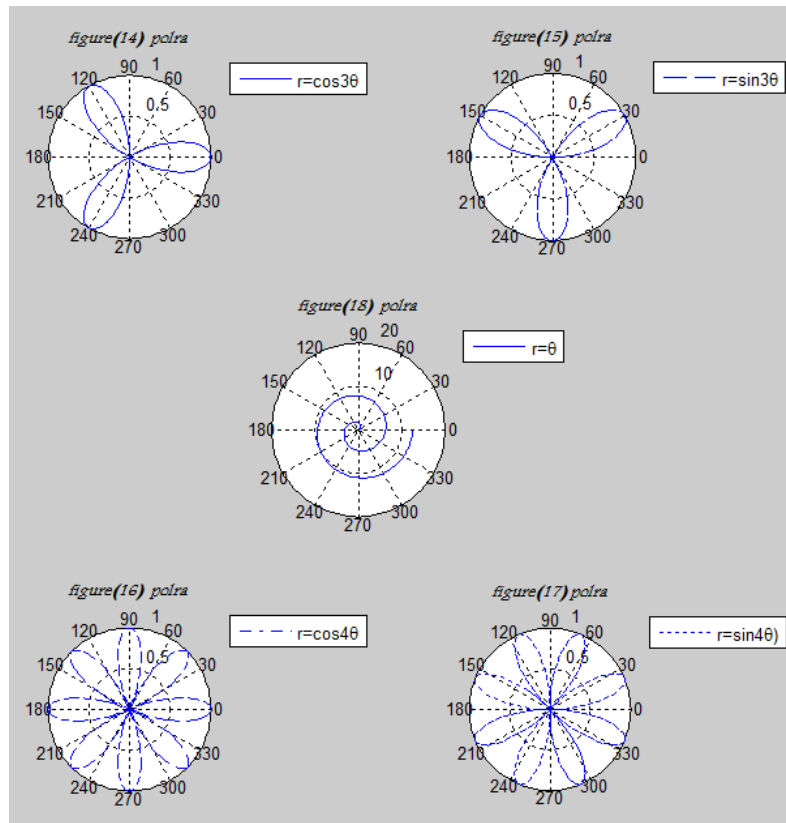
مثال:

```
clc
clg
clear
teta=0:pi/150:4*pi;
```

```

subplot(3,2,1)
polar(teta,cos(3.*teta),'-b')
title('\it\bf\fontname{titr} figure(14) polra)
legend('r=cos3\theta',-1)
subplot(3,2,2)
polar(teta,sin(3.*teta),'--b')
title('\it\bf\fontname{titr} figure(15) polra)
legend('r=sin3\theta',-1)
subplot(3,2,3.5)
polar(teta,teta,'-b')
title('\it\bf\fontname{titr} figure(18) polra)
legend('r=\theta',-1)
subplot(3,2,5)
polar(teta,cos(4.*teta),'-b')
title('\it\bf\fontname{titr}figure(16) polra')
legend('r=cos4\theta',-1)
subplot(3,2,6)
polar(teta,sin(4.*teta),':b')
title('\it\bf\fontname{titr} figure(17) polra)
legend('r=sin4\theta',-1)

```



مثال:

```

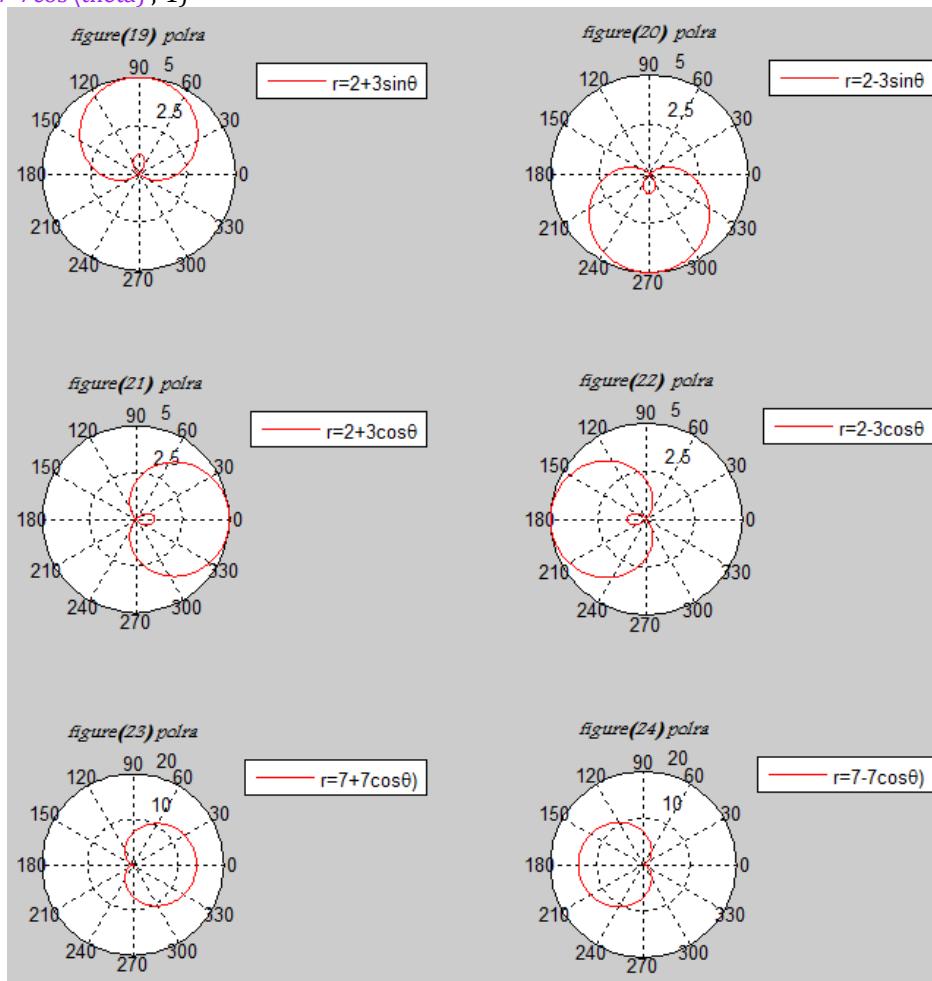
clc
clg
clear
teta=0:pi/150:4*pi;
subplot(3,2,1)

```

```

polar(teta,2+3.*sin(teta),'-r')
title('\it\bf\fontname{titr} figure(19) polra)
legend('r=2+3sin\theta',-1)
subplot(3,2,2)
polar(teta,2-3.*sin(teta),'-r')
title('\it\bf\fontname{titr} figure(20) polra)
legend('r=2-3sin\theta',-1)
subplot(3,2,3)
polar(teta,2+3.*cos(teta),'-r')
title('\it\bf\fontname{titr} figure(21) polra)
legend('r=2+3cos\theta',-1)
subplot(3,2,4)
polar(teta,2-3.*cos(teta),'-r')
title('\it\bf\fontname{titr} figure(22) polra)
legend('r=2-3cos\theta',-1)
subplot(3,2,5)
polar(teta,7+7.*cos(teta),'-r')
title('\it\bf\fontname{titr} figure(23) polra)
legend('r=7+7cos\theta',-1)
subplot(3,2,6)
polar(teta,7-7.*cos(teta),'-r')
title('\it\bf\fontname{titr} figure(24) polra)
legend('r=7-7cos\theta',-1)

```



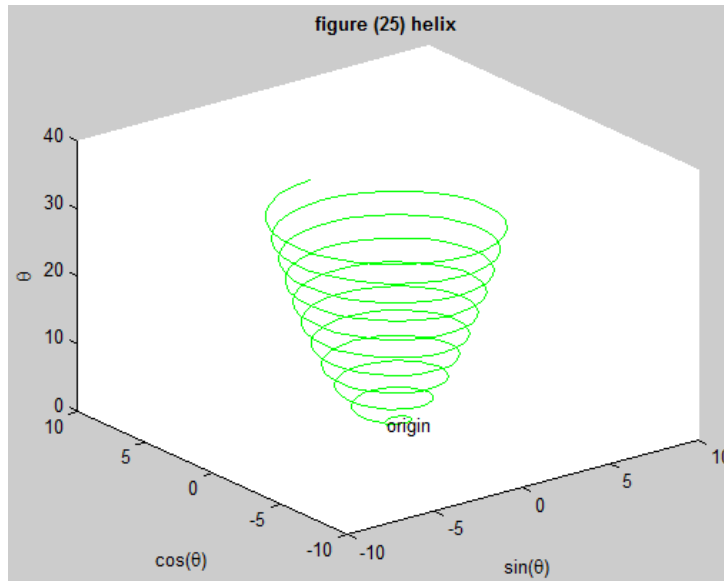
رسم گراف های سه بعدی

نمودار خطی سه بعدی (plot3)

این دستور کاملاً شبیه به دستور plot می باشد با این تفاوت که یک مقدار ارتفاعی (Z) به آن اضافه شده.

مثال:

```
clc
clf
clear
t=0:1:10*pi;
plot3((sqrt(t).*sin(2*t)),(sqrt(t).*cos(2*t)),t,'-g')
xlabel('sin(\theta)')
ylabel('cos(\theta)')
zlabel('\theta')
text(0,0,0,'origin')
title('\bf figure (25) helix')
```

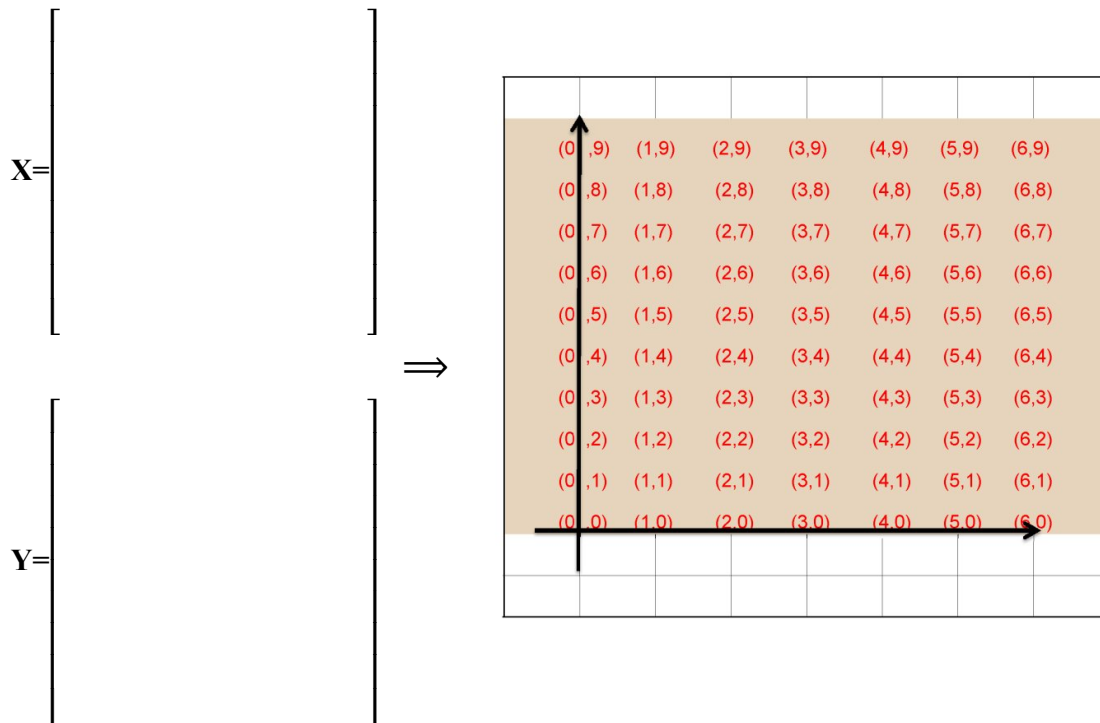


نمودار توری (mesh) و صفحه ای (surface) یا سطحی:

نمودارهای سه بعدی توری و سطحی نمودارهای Z هستند که مقدار Z آنها تابعی از X, Y می باشد ($Z=f(X, Y)$). به عبارتی برای ترسیم یک نمودار سه بعدی باید یک آرایه سه بعدی ایجاد کرد که تعداد در آیه های هر بعد آن با هم برابر باشند.

اولین کار برای رسم یک نمودار سه بعدی این است که یک صفحه X, Y تشکیل دهیم. و در نهایت به هر نقطه X, Y یک مقدار Z اختصاص دهیم.

ایجاد صفحه X, Y :



دستور meshgrid

برای ایجاد صفحه X, Y مانند شکل فوق نیاز به دو ماتریس X و Y می باشد . اما با استفاده از دستور `meshgrid` برای ایجاد صفحه X, Y تنها نیاز به درآیه های روی محور X و درآیه های روی محور Y می باشد .
مثال: شبکه صفحه X, Y بالا را با استفاده از دستور `meshgrid` ایجاد کنید.

```
[X,Y] = meshgrid(0:6,0:9)
```

یا
`X=0:6;Y=0:9`

```
[X,Y] = meshgrid(X,Y)
```

حال بعد از ایجاد دو آرایه X, Y نوبت به ایجاد آرایه Z می رسد. که این آرایه تابعی از X و Y می باشد . یعنی به ازاء هر X و Y باید یک مقدار Z داشته باشیم.

مثال: اگر $Z = \frac{X*Y}{X^2+Y^2}$ ماتریس Z را بدست آورید.

```
Z=(X.*Y.^2)/(X.^2+Y.^2)
```

صورت کلی دستور `surf` بصورت ذیل می باشد .

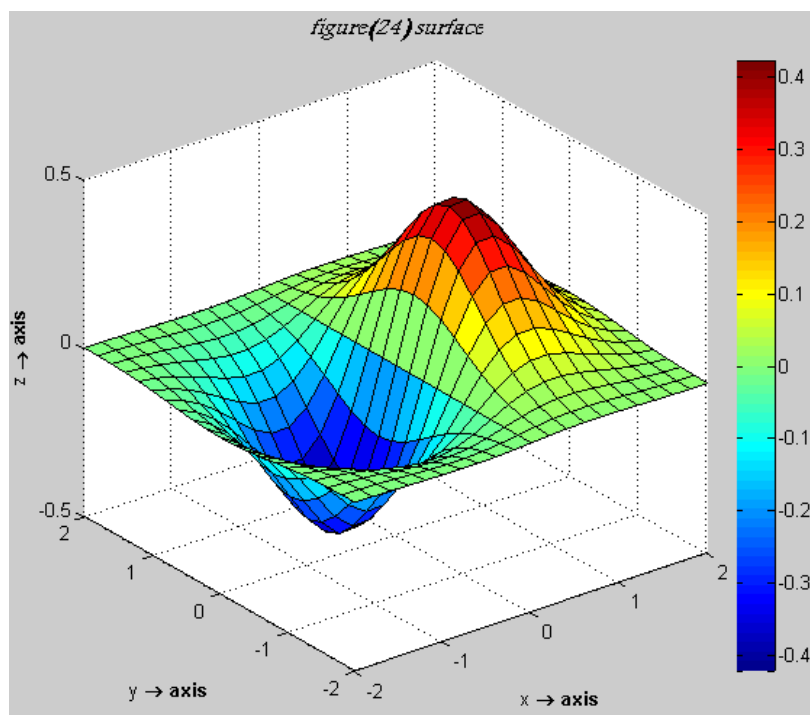
```
surf(x,y,z)
```

صورت کلی دستور `mesh` بصورت ذیل می باشد .

```
mesh(x,y,z)
```

مثال:

```
[X,Y] = meshgrid(-2:.2:2, -2:.2:2);  
Z = X.* exp(-X.^2 - Y.^2);  
surf(X,Y,Z)  
title('\it\bf\fontname{ttr}\fontsize{12} igure(24) surface)  
colorbar  
xlabel('x \bf\rightarrow axis')  
ylabel('y \bf\rightarrow axis')  
zlabel('z \bf\rightarrow axis')
```

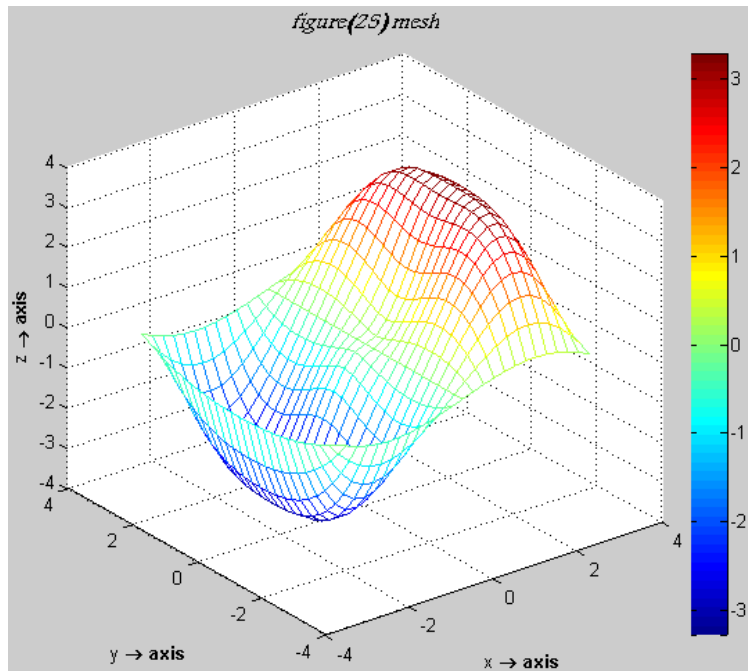


دستور colorbar

به کمک این دستور می توان در کنار گراف یک legend از رنگ ها ایجاد کرد.

مثال:

```
x=-3:.25:3;  
y=x;  
[x,y] = meshgrid(x,y);  
z=1.8.*sqrt(x.^2+y.^2).*cos(.5*y).*sin(x);  
mesh(x,y,z)  
title('\it\bf\fontname{ttr}\fontsize{12} igure(25) mesh)  
colorbar  
xlabel('x \bf\rightarrow axis')  
ylabel('y \bf\rightarrow axis')  
zlabel('z \bf\rightarrow axis')
```

دستور colormap

نمودار ترسیم شده رنگ بندی دارد که برحسب ارتفاع (Z) تغییر می کند تا بتوان از گراف یک تصویر سه بعدی تصور کرد. این رنگ ها را می توان تغییر داد.

سیستم بکار رفته برای نمایش گراف سیستم RGB بوده. پس برای ایجاد رنگ مورد نظر، باید از هر سه رنگ مقداری را انتخاب کنیم تا رنگ مورد نظر ایجاد شود. جهت این کار از فرمان colormap(c) استفاده می کنیم. که C یک ماتریس 3×1 می باشد که بترتیب از چپ به راست هر کدام یک از درآیه ها چگالی رنگ قرمز، چگالی رنگ سبز و چگالی رنگ آبی می باشد که در نهایت از ترکیب این سه رنگ جدیدی ایجاد می شود.

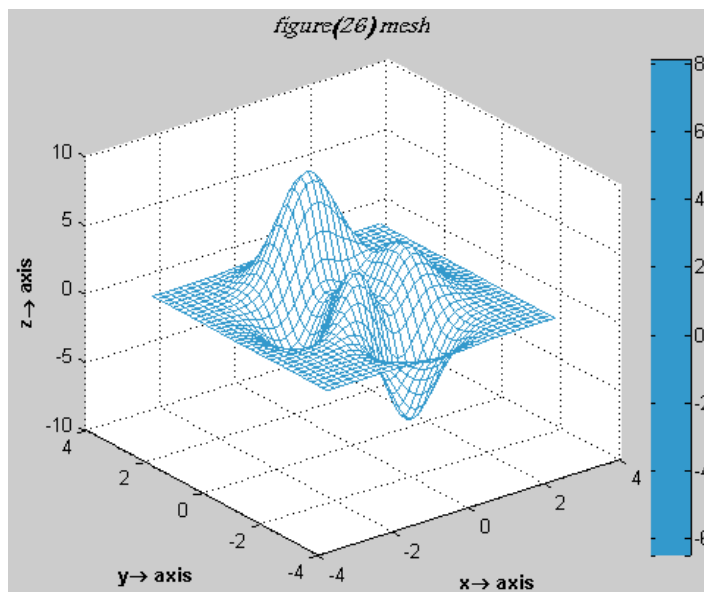
مقادیر چگالی رنگ	رنگ	Short name	مقادیر چگالی رنگ	رنگ	Short name
[0 0 1]	آبی	b	[0 0 0]	سیاه	k
[1 1 0]	زرد	y	[1 1 1]	سفید	w
[1 0 1]	زرشکی	m	[1 0 0]	قرمز	r
[.5 .5 .5]	زیتونی		[0 1 0]	سبز	g

یک سری ترتیب رنگی در برنامه matlab از پیش تعریف شده که می توان از آنها نیز بسته به نیاز استفاده کرد.



مثال:

```
[x,y,z]=peaks(35);
meshgrid(x,y);
colormap ([.2 .6 .8])
mesh(x,y,z)
title('\it\bf\fontname{titr}\fontsize{12} igure(26) mesh)
colorbar
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')
```



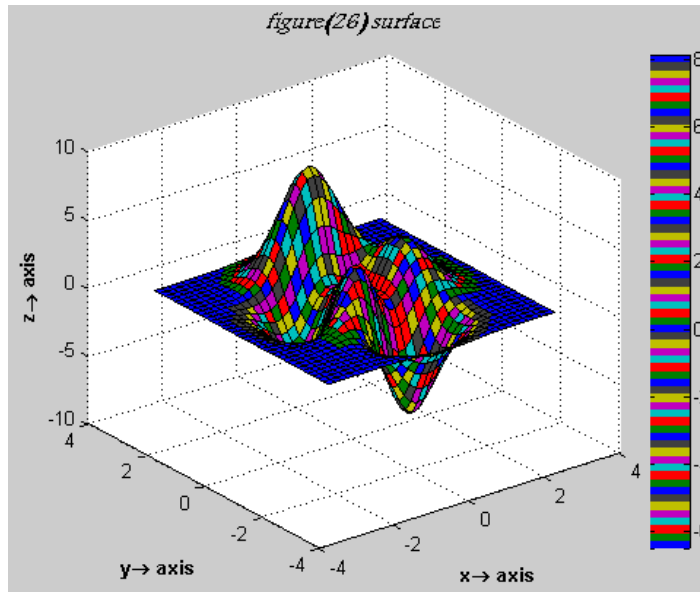
مثال:

```
[x,y,z]=peaks(35);
```

```

meshgrid(x,y);
colormap lines
surf(x,y,z)
title('\it\bf\fontname{titr}\fontsize{12} igure(26) surface)
colorbar
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')

```



نمودار توری (mesh curtain) یا (mesh zero)

به کمک دستور meshz می توان یک نمودار سه بعدی توری ایجاد کرد که سطح زیر صفر آن دیده نشود.

صورت کلی دستور meshz بصورت ذیل می باشد .

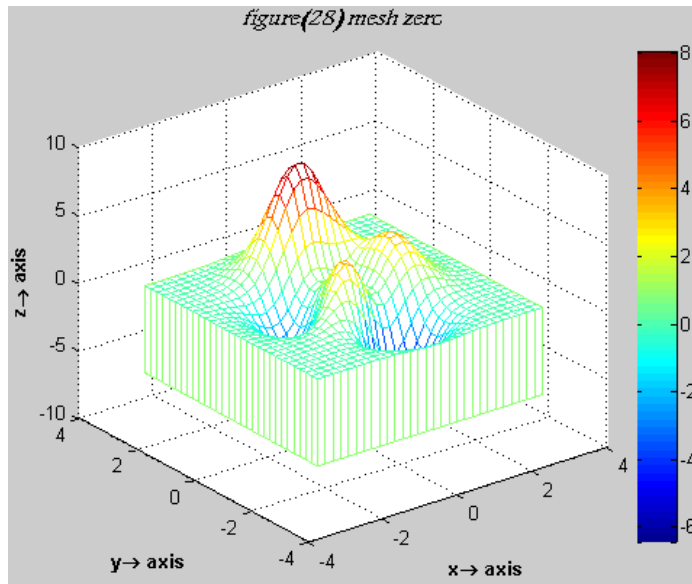
```
meshz(x,y,z)
```

مثال:

```

[x,y,z]=peaks(30);
meshgrid(x,y,z);
meshz(x,y,z)
title('\it\bf\fontname{titr}\fontsize{12} igure(28) mesh zero)
colorbar
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')

```



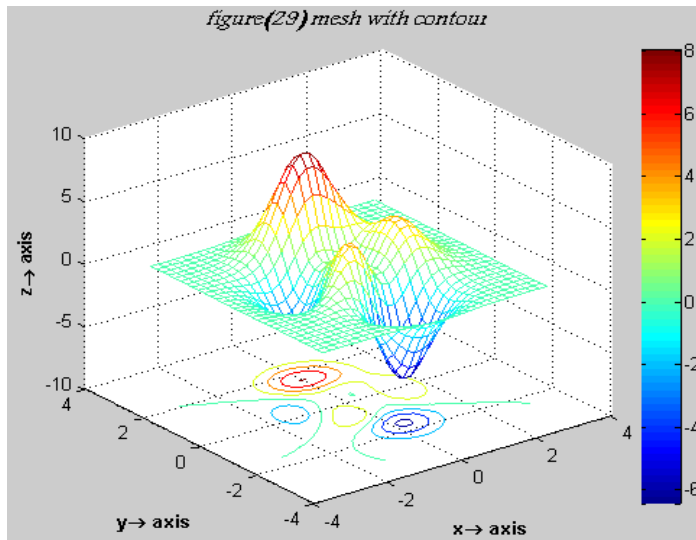
نمودار توری (mesh and contour)

به کمک دستور meshc می توان یک نمودار سه بعدی توری توری ایجاد کرد که بر روی صفحه x,y آن نیز منحنی میزان نمودار ترسیم گردد. صورت کلی دستور meshc بصورت ذیل می باشد.

meshc(x,y,z)

مثال:

```
[x,y,z]=peaks(30);
meshgrid(x,y,z);
meshc(x,y,z)
title('\it\bf\fontname{titr}\fontsize{12} igure(29) mesh with contour)
colorbar
xlabel('\bf\rightarrow axis')
ylabel('\bf\rightarrow axis')
zlabel('\bf\rightarrow axis')
```



نمودار آبشاری waterfall

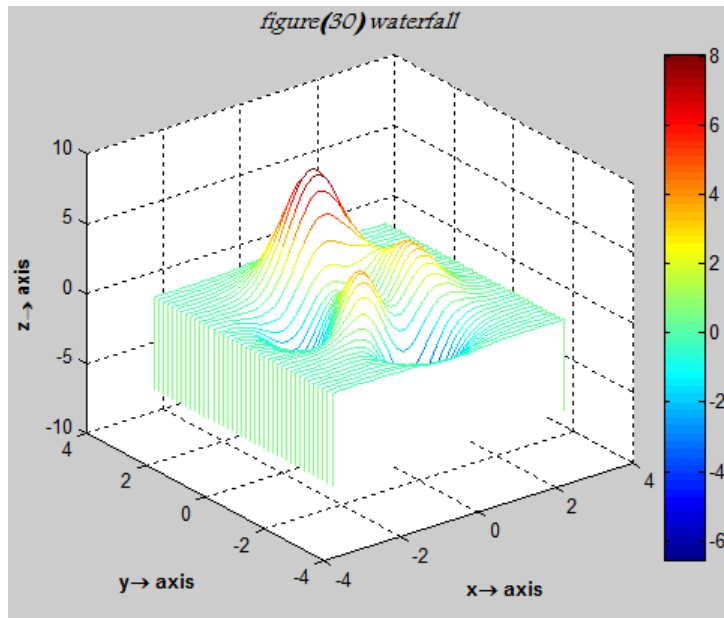
این نمودار نیز مانند نمودار mesh می‌باشد با این تفاوت که خطوط ایجاد کننده نمودار تنها در جهت محور Xها ایجاد می‌شود.

صورت کلی دستور meshc بصورت ذیل می‌باشد .

`waterfall(x,y,z)`

مثال:

```
[x,y,z]=peaks(40);
meshgrid(x,y,z);
waterfall(x,y,z)
title('\it\bf\fontname{ttr}\fontsize{12} igure(30) waterfall)
colorbar
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')
```



نمودار سطحی (surface and contour)

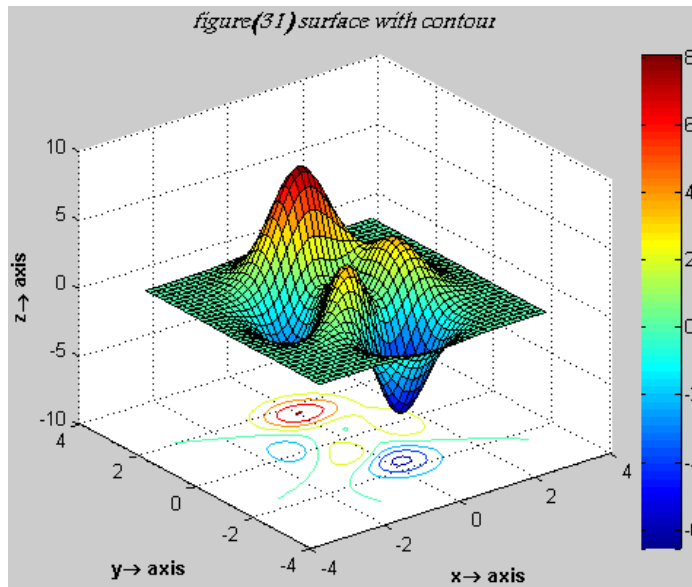
به کمک دستور surf می توان یک نمودار سه بعدی سطحی طوری ایجاد کرد که بر روی صفحه X, Y آن نیز منحنی میزان نمودار ترسیم گردد.

صورت کلی دستور surf بصورت ذیل می باشد .

`surf(x,y,z)`

مثال:

```
[x,y,z]=peaks(40);
meshgrid(x,y,z);
surf(x,y,z)
title('\it\bf\fontname{titr}\fontsize{12}  igure(31) surface with contour)
colorbar
xlabel('\bf\rightarrow axis')
ylabel('\bf\rightarrow axis')
zlabel('\bf\rightarrow axis')
```



نمودار سطحی درخشان (surface light)

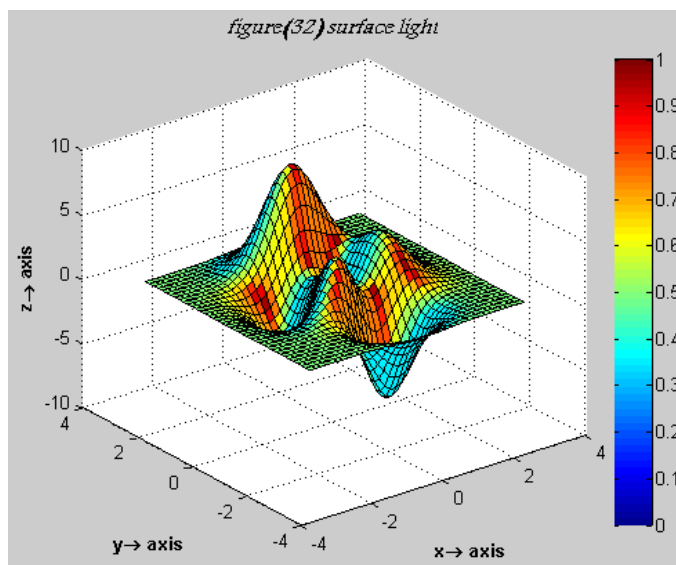
به کمک دستور `surf` می‌توان یک نمودار سه بعدی سطحی درخشان ایجاد کرد.

صورت کلی دستور `surf` بصورت ذیل می‌باشد.

`surf(x,y,z)`

مثال:

```
[x,y,z]=peaks(40);
meshgrid(x,y,z);
surf(x,y,z)
title('\it\bf\fontname{titr}\fontsize{12} figure(32) surface light)
colorbar
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')
```

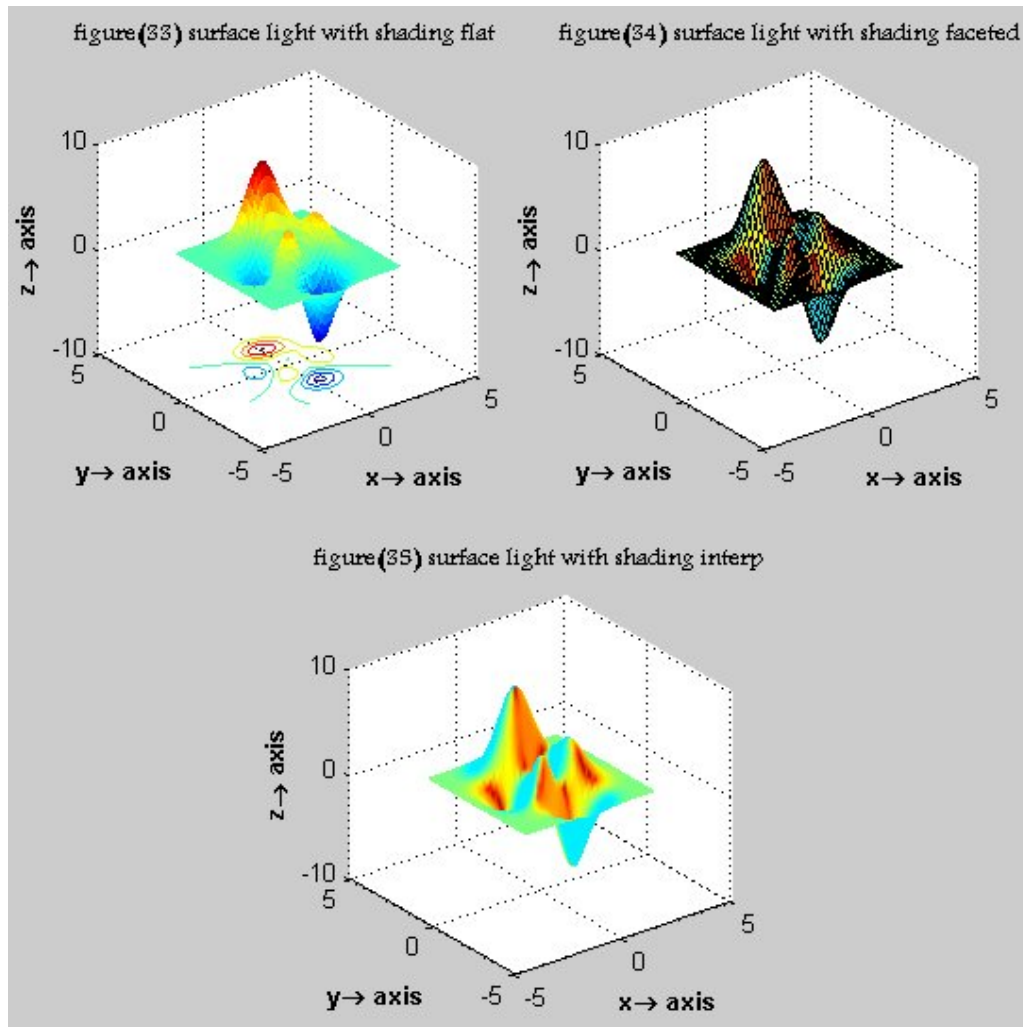


سایه زدنی نمودار (shading)

به کمک این دستور می توان نمودارهای سه بعدی را که ایجاد کرده ایم سایه زدنی کرد. سایه زدنی می تواند بصورت مسطح (flat) یا تراشیده شده (faceted) یا درونیایی شده (interp) باشد.

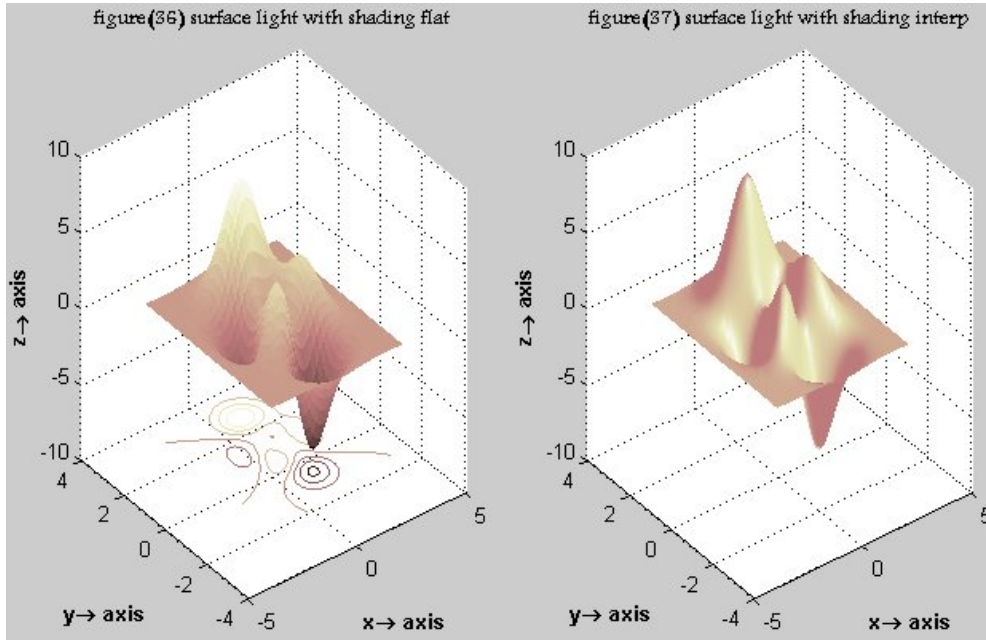
مثال:

```
[x,y,z]=peaks(40);
meshgrid(x,y,z);
subplot(2,2,1)
surf(x,y,z)
shading flat
title('\bf\fontname{titr}\fontsize{10} figure(33) surface light with shading la)')
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')
subplot(2,2,2)
surf(x,y,z)
shading faceted
title('\bf\fontname{titr}\fontsize{10} figure(34) surface light with shading la)')
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')
subplot(2,2,3.5)
surf(x,y,z)
shading interp
title('\bf\fontname{titr}\fontsize{10} figure(35) surface light with shading la)')
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')
```

مثال:

```
[x,y,z]=peaks(40);
meshgrid(x,y,z);
subplot(1,2,1)
surf(x,y,z)
shading flat
title('\bf\fontname{titr}\fontsize{10} figure(36) surface light with shading flat')
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')
colormap pink
subplot(1,2,2)
surf(x,y,z)
shading interp
title('\bf\fontname{titr}\fontsize{10} figure(37) surface light with shading interp')
xlabel('\bfx\rightarrow axis')
ylabel('\bfy\rightarrow axis')
zlabel('\bfz\rightarrow axis')
```



تنظیم زاویه دید نمودار (view)

جهت تنظیم زاویه دید نمودار از دستور view استفاده نمود. صورت کلی این دستور بصورت ذیل است.
`view(AZ,EL)`

این دستور دارای دو پارامتر می باشد، آزینوت دید (AZ) و زاویه ارتفاعی دید (EL).

در حالت پیش فرض $el=30^\circ$ و $az=-37.5^\circ$ می باشد.

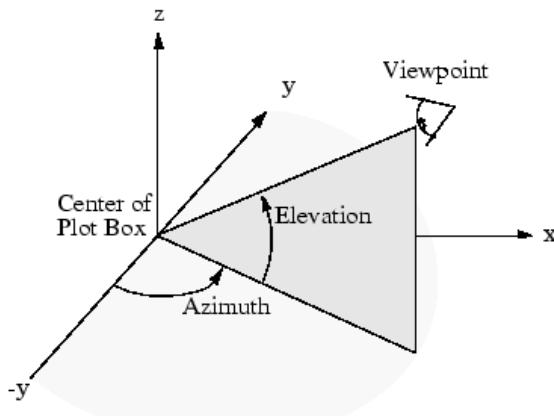
همان طور که در شکل نیز مشخص است مبدأ آزیموت

امتداد منفی محور yها می باشد که در جهت حرکت

عقربه های ساعت منفی و خلاف جهت حرکت عقربه های

ساعت مقدار آزیموت مثبت می باشد. و مبدأ زاویه

ارتفاعی دید صفحه X,y می باشد.



مثال:

```
[x,y,z]=peaks(40);
meshgrid(x,y,z);
surf(x,y,z)
shading interp
title('\fontname{titr}\fontsize{10} igure(38) view(az=60 el=30))
```

```
xlabel('\bfx\rightarrow axis')  
ylabel('\bfy\rightarrow axis')  
zlabel('\bfz\rightarrow axis')  
colormap copper  
view(60,30)
```

