

## آموزش نرم افزار MATLAB (آشنایی با نرم افزار و کاربرد آن در ریاضیات)

## \*بخش چهارم / برنامه نویسی در MATLAB:

نرم افزار MATLAB می تواند به عنوان یک محیط مناسب برای برنامه نویسی مورد استفاده قرار گیرد. برنامه نویسی در MATLAB شباهت زیادی به برنامه نویسی در محیط C دارد؛ البته بسیار ساده تر از آن است.

## آشنایی با حلقه ها و عبارات شرطی:

۱- حلقه ی `for`: در ابتدای این حلقه شرطی مطرح نمی شود، بلکه یک ماتریس سطری تعریف می شود و حلقه به تعداد درایه های این ماتریس تکرار می شود. روند کار بدین صورت است که پس از نوشتن کلمه ی `for` با یک فاصله در همان خط ماتریس سطری را تعریف می کنیم. به خط بعد می رویم و محتویات حلقه که می خواهیم تکرار شوند را می نویسیم تا آخرین خط. پس از آن برای نشان دادن پایان حلقه از کلمه ی `end` استفاده می کنیم. مثال:

```
>> x=0;
>> for a=1:20
x=x+a;
end
>> x
x=
    210
```

۲- حلقه ی `while`: یک حلقه ی شرطی است و تا زمانی که شرط نوشته شده در مقابل کلمه ی `while` برقرار باشد اجرای محتویات حلقه تکرار می شود. اینجا هم در پایان عبارات حلقه `end` می گذاریم. در واقع اگر ارزش منطقی عبارت شرط غیر صفر باشد می گوئیم شرط برقرار است. دستورات شرطی با استفاده از عملگرهای نسبت مانند `<`، `>` و `~=` و یا برخی توابع مخصوص MATLAB که با `is` نام آنها شروع می شود تعریف می شوند. در صورت برقرار بودن شرط ارزش آن ۱ و در غیر اینصورت 0 است. همچنین شما می توانید چند شرط را با عملگرهای منطقی مانند `&&` و `||` با هم ترکیب نمایید. در جدولهای زیر با برخی عملگرها و توابع شرط و علائم منطقی بین دو شرط آشنا می شویم:

عبارت شرطی نسبت	معنای عبارت
$A < B$	A کوچکتر از B باشد.
$A > B$	A بزرگتر از B باشد.
$A \leq B$	A کوچکتر یا مساوی با B باشد.
$A \geq B$	A بزرگتر یا مساوی با B باشد.
$A == B$	A مساوی با B باشد.
$A \sim= B$	A نا مساوی با B باشد.

## آموزش نرم افزار MATLAB (آشنایی با نرم افزار و کاربرد آن در ریاضیات)

معنای علامت	علامت بین دو عبارت شرطی
و (در صورتی که هر دو عبارت در دو سوی علامت درست باشند ارزش آن ۱ و در غیر اینصورت ۰ است).	&&
یا (در صورتی که حتی یکی از عبارات دو سوی علامت درست باشد ارزش آن ۱ و در غیر اینصورت ۰ است).	
علامت NOT منطقی است.	~

تابع شرط (is*)	شرط مورد بررسی که در صورت مثبت بودن پاسخ، ارزش تابع 1 و در غیر اینصورت 0 است
isreal(x)	آیا متغیر X یا تمامی درایه های ماتریس X عدد حقیقی هستند.
isempty(x)	آیا ماتریس X یک ماتریس تهی و بدون درایه است.
isfinite(x)	آیا متغیر X یا تمامی درایه های ماتریس X عدد محدود است.
isinf(x)	آیا متغیر X یا تمامی درایه های ماتریس X عدد بینهایت است.
isnan(x)	آیا متغیر X یا تمامی درایه های ماتریس X مبهم یا تعریف نشده است (مثل inf/inf یا inf-inf)
isequal(x,y,z,...)	آیا متغیرها یا تمامی درایه های نظیر ماتریسهای ورودی تابع از نظر عددی با هم برابر هستند.
iskeyword('str')	آیا کلمه ی داده شده به تابع از کلمات کلیدی MATLAB است.
isprime(x)	آیا متغیر X یا تمامی درایه های ماتریس X عدد اول هستند.

حال به چند نمونه از کاربرد این عبارات در حلقه ی while توجه کنید:

```
>>x=0;y=1;
>>while x<10
x=x+1;
y=y*x;
end
>> y %y=10!
y=
    3628800
>>a=5;b=1;
>>while a~=0
b=b/a;
a=a-1;
end
>>b %b=1/5/4/3/2/1
b=
    0.0083
>>a=5;b=1;
>>while isfinite(b/a)
b=b/a;
a=a-1;
end
>>b %b=1/5/4/3/2/1
b=
    0.0083
```

## آموزش نرم افزار MATLAB (آشنایی با نرم افزار و کاربرد آن در ریاضیات)

```
>>a=5;b=1;
>>while ~isinf(b/a)&&b<=3
b=b+b/a;a=a-1;b=b+1;
end
>>a
a=
    2.5000
>>b
b=
    3.1429
```

۳- if: اگر شرط مقابل کلمه ی if برقرار باشد، خطوط بعد از آن اجرا می شود تا به کلمه ی else، elseif یا end برسد. اما اگر شرط برقرار نباشد خطوط بعد از آن را اجرا نشده رد می کند تا به یکی از این سه کلمه برسد. اگر شرط مقابل if برقرار بود و با اجرای خطوط پس از آن به else، elseif یا end رسید، به خط بعد از end می رود. در دستور if نیز مانند حلقه ی for و while حتما باید یک end برای نشان دادن پایان دستورات شرطی وجود داشته باشد. حال اگر شرط مقابل if برقرار نبود سه وضعیت پیش می آید. اگر با رد کردن خطوط پس از if به end رسید که به خط پس از آن رفته و دستورات بعدی را طبق روال عادی اجرا می کند. اگر به else برسد، دستورات پس از else را اجرا می کند تا به end برسد. اگر به elseif برخورد کند، در واقع وارد یک وضعیت شرطی مشابه با خود if شده و بسته به برقراری شرط مقابل خود در مورد اجرای دستورات خطوط بعدی مانند if عمل می کند تا به end، else یا یک elseif دیگر برسد. شما میتوانید داخل یک مجموعه ی شرطی if، یک یا چند if دیگر داشته باشید، اما دقت کنید که هر if داخلی باید یک end قبل از end مربوط به if خارجی و قبلی خود داشته باشد. به عبارت دیگر آخرین end مربوط به اولین if است. ضمناً elseif نیازی به end جداگانه ندارد. به چند مثال در رابطه با دستور if توجه کنید:

```
>> a=1;b=2;
>> if a<b
c=1;
else
c=2;
end
>> c
c=
    1
>> if a==b
c=3;
elseif a>b
c=4;
else
c=5;
end
>> c
c=
    5
```

## آموزش نرم افزار MATLAB (آشنایی با نرم افزار و کاربرد آن در ریاضیات)

۴- break و continue: این دو دستور داخل حلقه های for یا while قرار می گیرند و باعث جهشهایی می شوند. continue عمل جهش به ابتدای حلقه را بدون اجرای سطرهای باقیمانده از حلقه که پس از آن قرار دارند، انجام می دهد. دستور break باعث خروج از حلقه ی در حال اجرا شده و موجب می گردد که برنامه به خط پس از end حلقه جهش کند. اغلب دستورات continue و break تحت یک عبارت شرطی با if آورده می شوند که اگر شرط if برقرار باشد، این دستورات به عنوان محتویات زیر if اجرا گردیده و موجب انجام جهش در حلقه می شوند.

۵- switch: دستورات switch که با چند بند دستور دیگر شکل می گیرد باعث اجرای دستورات زیر خود به صورت انتخابی می گردد. جلوی کلمه ی switch نام یک متغیر عددی یا متنی قرار می گیرد و در خطوط پس از این خط حالات مختلفی که ممکن است آن متغیر اختیار کند به عنوان case قرار داده می شود. مقدار متغیر مقابل switch با مقدار هر یک از مقادیر قرار داده شده مقابل case ها برابر باشد، دستوراتی که بین آن case و case بعد قرار گرفته است اجرا می شود و پس از اجرا به خط بعد از end جهش می کند. پس در اینجا نیز وجود end به عنوان نشان دهنده ی پایان مجموعه دستورات تحت تاثیر switch الزامیست. علاوه بر case ها می توانید به عنوان آخرین بخش از بدنه ی دستور switch، عبارت otherwise را قرار دهید که اگر مقدار متغیر ورودی switch با هیچیک از مقادیر مقابل case ها برابر نبود، دستورات بین otherwise و end اجرا شود. البته وجود این قسمت اجباری نیست. شکل کامل بدنه ی دستور switch به شکل زیر است:

>> switch نام متغیر

case مقدار ممکن اول

**statement, ..., statement**

case مقدار ممکن دوم

**statement, ..., statement**

case { ..., مقدار ممکن nام, ..., مقدار ممکن چهارم, مقدار ممکن سوم }

**statement, ..., statement**

otherwise

**statement, ..., statement**

end

مثال زیر نمونه ای از چگونگی استفاده از دستور switch می باشد. تا اینجا برای تمامی مثالها فرض بر این بود که دستورات در پنجره ی Command Window خط به خط نوشته و اجرا می شود. اما در عمل ما از پنجره ی فرمان برای برنامه نویسی استفاده نمی کنیم و دستوراتمان را در پنجره ی Editor نوشته و سپس آنرا اجرا می کنیم. با پنجره ی Editor قبلا آشنا شدیم. حال به عنوان نمونه دستورات مربوط به مثال switch را در پنجره ی Editor نوشته و سپس با زدن کلید F5

## آموزش نرم افزار MATLAB (آشنایی با نرم افزار و کاربرد آن در ریاضیات)

آنرا save و اجرا می کنیم. ویرایش شدن متن دستورات به شکلی که می بینید در Editor به صورت خودکار انجام میشود و خود نرم افزار رنگ برخی عبارات را تغییر داده و دستوراتی که جزء بدنه ی یک دستور دیگر باشند را کمی جلوتر می نویسد. به چگونگی اجرای برنامه در Command Window نیز توجه کنید:

```

1 - x=input('x=')
2 - switch x
3 -     case 1
4 -         fprintf('it is number one')
5 -     case 2
6 -         fprintf('it is number two')
7 -     case {3,4,5}
8 -         fprintf('it is not 1 or 2')
9 -     otherwise
10 -         fprintf('it is not a number from 1 to 5')
11 - end

```

برنامه را اجرا می کنیم. خط اول که تابع input است با چاپ عبارت متنی ای که به عنوان ورودی به آن می دهیم در پنجره فرمان، منتظر وارد شدن یک عدد یا عبارت توسط کاربر می شود. عدد یا عبارتی که شما مقابل عبارت چاپ شده می نویسید و با زدن enter آن را به تابع می دهید، به عنوان خروجی تابع input بر می گردد و اگر تابع مساوی با یک متغیر قرار داده شده باشد، این عدد یا عبارت در آن متغیر ذخیره می شود. در این مثال عبارت x= چاپ می شود و ما عدد ۲ را تایپ و با زدن enter قرار گرفتن عدد ۲ در خروجی تابع را تایید می کنیم. این عدد در متغیر x که مساوی با دستور input('x=') قرار داده شده است ذخیره می شود:

```

x=2
x=
 2
it is number two>>

```

دستور fprintf همانگونه که در مثال فوق ملاحظه فرمودید برای چاپ یک عبارت متنی استفاده می شود. البته شما هر جای متن که بخواهید می توانید یک مقدار عددی را نیز چاپ کنید.

ادامه دارد...