

## دستور شرطی while

در اینجا هدف بر این است که محاسبه انحراف معیار را با استفاده از while بر روی همین داده ها انجام دهیم. در ابتدا ساختار دستور را می نویسیم.

```
Data (1, -) = std (rain (:, -));
```

بعد از اینکه ساختار دستور را نوشتید حالا قالب دستور while را مجدداً می نویسیم برای جلوگیری از حجم کارتنها فرم کلاسیک آن را یادآور می شویم:

```
1—P=input ('pleas enter name file :')
```

```
2—[m n] = size (P);
```

```
3—j=1;
```

```
4— while j>=n
```

```
....
```

```
....
```

```
....
```

```
5— j=j+1;
```

```
6— end
```

```
5— clear
```

همان طوری که در بخش توضیحات مربوط به for هم اشاره شد در اینجا کافیسیت j را در ساختار دستور قرار دهیم. یعنی جای قرار دهیم که عملیات محاسبه بر روی آن انجام می شود از آنجای که عملیات محاسبه بر روی ستون ها انجام می شود j را در قسمت ستون ها قرار می دهیم که شکل کلاسیک برنامه به شکل زیر می باشد:

```
1—P=input ('pleas enter name file:')
```

```
2—[m n] = size (P);
```

```
3—j=1;
```

```
4— while j>=n
```

```
5—data(1,j)=std(p(:,j));
```

```
6— j=j+1;
```

```
7— end
```

```
8— clear j m n p
```

تفاوت حلقه For با while در خط سوم می باشد به طوری که در حلقه for شروع حلق بعد از تایپ خود for آورده می شود این در حالی می باشد که در while قبال از آن آورده می شود.

اگر قرار شد عملیات محاسبه بر روی تک تک درایه های (ستون ها یا سطرها) انجام شود نحوه دستور ذخیره متفاوت خواهد بود. برای مثال فرض کنید بخواهید در همین مثال تک تک سلوله های یک ستون را از میانگین همان ستون کم کنیم آنگاه شکل دستور ۵ عوض می شود:

```
5—data (:, j)= p(:,j)-mean(p(:,j));
```

توجه کنید دلیل اینکه قبل از تساوی به جای ۱ از : استفاده کردیم این بود که در دستور قبلی یا محاسبه انحراف معیار برای هر ستون فقط یک عدد یا همان انحراف معیار را خواستیم به همین خاطر در دستور قبلی دستورات

## جزوه کارگاه متلب ادامه جلسه چهارم- دوستکامیان

را در ردیف اول و ستون  $j$  تا  $n$  یا به تعداد ستون‌ها دخیره می‌کردند. در این دستور چون برای هر ستون عملیات بر روی تک تک درایه‌های آن انجام می‌شد بنابراین برای هرستون به تعداد ردیف‌های آن عدد می‌دهد به همین خاطر برای اینکه برنامه خطا ندهد به جای ۱ از : استفاده کردیم. این شرایط برای سایر حلقه‌ها هم صادق می‌باشد و فرقی نمی‌کند.

### دستور شرطی if و elseif

در بسیاری از مواقع می‌بایست گروهی از دستورات را به صورت شرطی مطابق با یک رابطه ایجاد کرد. این منطق در زبان برنامه نویسی با استفاده از شرط‌های مختلف if و elseif اجرا می‌شود. برای مثال فرض کنید در یک پهنه جغرافیایی قصد دارید که به جای بعضی از اعداد مثلاً از کد خاصی استفاده کنید یا اینکه اعداد بالاتر از یک آستانه خاصی را کد بندی کنید. ساختار دستور if به شرح زیر است.

```
clc
pp=input(' pleas enter name file :');
[m n]=size(pp);
for j=1:n
if
.....
elseif
.....
elseif
.....
else
.....
end
end
Clear pp m n H
```

ساختار ساده تر if

```
if expression
    statements
elseif expression
    statements
else
    statements
end
```

به مثال زیر توجه کنید. در این مثال قصد داریم تک تک درایه‌های یک ماتریس را با شرط‌های مورد نظر بسنجیم و شرط‌های مورد نظر را روی آن انجام می‌دهیم.

نمونه اول

```
1-clc
2-pp=input('pleas enter database:');
3-[m n]=size(pp);
4-nm=zeros(m,n);
5-for H=1:n*m;
6-if pp(H)>10
nm(H)= pp (H)/1000;
7-elseif pp (H)<10 && pp (H)>7
nm(H)= pp(H)^2;
8- elseif pp (H)<2
nm(H) = pp (H)==0;
9-else
nm(H) =555555;
10-end
11-end
Clear pp m n H
```

تشریح برنامه

۱- در این دستور فرمان clc آورده شده است. دستور clc در متلب، برای پاک کردن اطلاعات نمایش داده شده در پنجره command به کار می رود. یعنی هر زمان که خواستید تمامی اطلاعات نمایش داده شده در پنجره command (البته تا این لحظه و نه برای اطلاعاتی که در آینده نمایش داده می شوند)، پاک شوند، تنها کافی است که دستور clc را بنویسید. توصیه می شود که این دستور همیشه در اول برنامه آورده شود.

۲- همان طوری که اشاره شد دستور input برای فراخوانی یا وارد کردن داده می باشد. به عبارت دیگر در این برنامه هدف از این دستور وارد کردن فایل جهت انجام محاسبات میانگین می باشد. شکل این دستور همیشه به این طریق است که: input(' '); = اسم فایل.

در این دستور ما اسم فایل که قرار است فراخوانی بشود در این برنامه p گذاشتیم هر چند که هر اسم دیگری که شما بخواهید می تواند داشته باشد در محل (':') شما می تواند هر دستور دلخواه را وارد کنید برای مثال در دستور زیر بین دو تا کوتیشن عبارت pleas enter the file name data آورده شد. توجه کنید که برای این دستور قبل از کوتیشن دوم همیشه دونقطه (:) آورده شود. به عبارتی بهتر دستور input، پیامی را در پنجره Command نرم افزار متلب نمایش داده و سپس عدد وارد شده توسط کاربر را دریافت می کند و آن را در متغیر p ذخیره می کند.

۳- این دستور ابعاد ماتریس p را به ما می دهد. به عبارتی تعداد ردیف و ستون های ماتریس p را می دهد.

۴- در این دستور ماتریسی ایجاد می شود که تمام درایه های آن صفر می باشد و هم اندازه و هم ابعاد با ماتریس

## جزوه کارگاه متلب ادامه جلسه چهارم- دوستکامیان

- p می باشد. یا بع عبارت بهتر دارای m ردیف صفر و n ردیف ستون می باشد.
- ۵- در این دستور با استفاده از حلقه for تعداد پاس ها (تکرار حلقه) را مشخص شد. و برای این منظور چون قصد بر این است که تک تک درایه های ماتریس سنجیده شود حلقه از ۱ شروع شده و تا  $m \times n$  ادامه دارد. برای مثال اگر ماتریس ما  $4 \times 5$  باشد تعداد پاس های ما از ۱ تا ۲۰ ادامه می یابد. یعنی پاس ها به بر روی تعداد  $m \times n$  ماتریس p تکرار و شرط های که در ادامه می آید انجام می گیرد.
- ۶- در این دستور اولین شرط اجرا شده است به طوری که گفته شد که در پاس H مقدار آن بالاتر از ۱۰ باشد بر عدد ۱۰۰۰ تقسیم شود و در ماتریس mn ذخیره کند.
- ۷- در این دستور این طور بیان شد که اگر شرط اولی صادق نبود بیا شرطی دمی که همان ارش کوچکتر از ۰.۱ و بزرگتر از ۷ باشد اجرا کند. در دستور شش گفته شد که اگر اولین درایه این شرط برش حاک بود تقسم بر ۱۰۰۰ کن و اگر شرط دستور ۷ حاکم باشد به توان ۲ برسان.
- ۸- در این دستور گفته شده است که اگر دو شرط اولی برای اولین پاس حاکم نبود در غیر این صورت این شرط را (کوچکتر از ۲) را بر آن حاک کن در صورتی که شرط صادق بود مقدار آن را برابر با صفر قرار بده.
- ۹- در این دستور گفته شد که اگر هیچ کدام از دستورات یا شرط های بالا صادق نبود مقدار پاس H را برابر با ۵۵۵۵۵۵ قرار بده.
- ۱۰- ۱۱- توجه کنید این دستورات به معنی پایان حلقه می باشد. بنابراین به ازای هر شرط اصلی یک end آورده شود.