فرمول نویسی در متلب

همان طوری که هم اشاره شد برای نوشتن یک برنامه در متلب یک برنامه نویس باید حداقل فرمول نویسی را یاد گرفته باشد. در اینجا قصد داریم برای نمونه به چند مورد اشاره کنیم. ماتریس زیر را در نظر بگیرید:

```
a =
    17.587    20.443    19.479    20.991
    18.857    23.936    24.528    26.174
    15.438    15.007    12.569    14.823
    15.704    15.718    14.225    18.723
    19.115    18.092    16.384    20.207
    15.139    14.979    14.341    18.701
    12.074    10.798    10.336    13.403
    15.093     14.07    13.154    17.323
    16.509    19.043    16.815    17.692
    15.871    16.246    14.153    16.106
```

فرض کنید بخواهیم فرمول زیر را بر روی تک تک ستون‌های این ماتریس اجرا کنیم. به نحوه محاسبات دقت کنید:

تمرین شماره اول

$ano = x_i - \bar{x}$	

در این فرمول که به نوعی اشاره به آنومالی دارد هدف این است که برای هر ستون تک تک درایه‌ها آن ستون را از میانگین همان ستون کم کند. بنابراین اول باید میانگین هر ستون را حساب کنیم سپس تک تک درایه ها را از آن کم کنیم. البته به چند طریق می‌توان هر فرمولی را حل کرد. هم می توان آن را در یک خط و هم می توان آن را ابتدا جزء جزء کرد سپس آنومالی را حساب کرد. به مثال های زیر دقت کنید:

```
>> md=mean(a(:,1))
md =
    16.139
```

در ابتدا $\bar{x}$ را با استفاده از تابع mean حساب و در md ذخیره کردیم. می دانیم که ۱۶/۱۳۹ میانگین ستون اول می‌باشد بنابراین باید تک تک درایه های ستون اول را از میانگین همان ستون (۱۶/۱۳۹) کم کنیم. برای اشاره به ستون اول از ادرس a(:,1) استفاده می کنیم. بنابراین با در دست داشتن ادرس و تابع میانگین به راحتی فرمول آنومالی قابل حل می‌باشد. به ادامه برنامه توجه کنید:

```
>> ano1=a(:,1)-md
ano1 =
    1.4483
     2.718
  -0.70095
  -0.43419
```

```
    2.9763
   -0.99991
   -4.0649
   -1.0454
    0.37073
   -0.26797
```

برای حل این فرمول اول میانگین را حساب کردیم سپس آنومالی را حساب کردیم. اما ما می توانیم مستقیما بدون اینکه عملیات انتساب را انجام دهیم با استفاده از دستور زیر همین فرمول را حساب کرد. به مثال زیر توجه کنید:

```
>> ano1=a(:,1)-mean(a(:,1))
ano1 =
    1.4483
    2.718
   -0.70095
   -0.43419
    2.9763
   -0.99991
   -4.0649
   -1.0454
    0.37073
   -0.26797
```

حالا اگر بخواهیم برای ستون‌های دیگر انومالی را حساب کنیم به شرح زیر عمل خواهیم کرد. البته به این نکته توجه کنید که در این فرمول با وجود اینکه فرمول را حل می‌کنیم همه نتایج را در یک ماتریس به اسم ano1 ذخیر می‌کنیم. چون که ما به ازای تمام درایه های هر ستون آنومالی داریم در قسمت نتیجه از (:) استفاده کردیم. برای مثال در ستور زیر اشاره شد که آنومالی ستون دوم را حساب کن و در همه ردیف‌ها ولی ستون دوم ماتریس ano1 ذخیره کنید به همین منوال برای سایر ستون‌ها تفسیر مشابهی دارید با این تفاوت که شماره ستون‌ها در هر مرحله تغییر می‌کند:

```
>> ano1(:,2)=a(:,2)-mean(a(:,2))
>> ano1(:,3)=a(:,3)-mean(a(:,3))
>> ano1(:,4)=a(:,4)-mean(a(:,4))
```

در اینجا ۴ ستون داشتیم گاهی اوقات ممکنه بیش از ۲۰ هزار ستون داشته باشیم آنگاه برای حل این فرمول باید برنامه‌ای را نوشت. در این جا قصد داریم با استفاده حلقه for یا while برنامه‌ای را برای این فرمول نوشت. ابتدا باید ساختار دستور و ساختار for را بنویسیم:

ساختار دستور ano :

```
>> ano1(:,-)=a(:,-)-mean(:,-);
```

در اینجا چون هدف این است ک این فرمول بر روی تک تک درایه‌ها انجام شود به جای ستون‌ها – قرار دادیم این به این معنی می‌باشد که باید به جای – (خط تیره) اسم حلقه گذاشته شود. حالا ساختار for را به شرح زیر

مجدد یادآوری می‌کنیم:

```
for  j=1:n
ساختار دستور مورد نظر
end
```

جایگذاری ساختار در حلقه:

```
for  j=1:n
ano1(:,j)=a(:,j)-mean(:,j);
end
```

اگر دقت کنید مشاهده می‌شود که اول کافیست ساختار فرمول مورد نظر را نوشت بعد برنامه نوشتن آن بسیار ساده و آسان می‌باشد. حالا اگر همین برنامه را اجرا کنید نتایج زیر مشاهده می‌شود:

```
ano1 =

    1.45    3.61    3.88    2.58
    2.72    7.10    8.93    7.76
   -0.70   -1.83   -3.03   -3.59
   -0.43   -1.12   -1.37    0.31
    2.98    1.26    0.79    1.79
   -1.00   -1.85   -1.26    0.29
   -4.06   -6.04   -5.26   -5.01
   -1.05   -2.76   -2.44   -1.09
    0.37    2.21    1.22   -0.72
   -0.27   -0.59   -1.45   -2.31
```

در ادامه سعی می‌کنیم فرمول‌های را کمی پیچیده کنیم تا بتوانیم با نحوه فرمول نویسی بیشتر آشنا شویم.