

العمليات الحسابية التي تتم على المصفوفات  
هناك مجموعة من العمليات الحسابية التي تتم على المصفوفات من جمع و طرح ... الخ . و من هذه  
العمليات هي :

1- ايعاز ايجاد حجم المصفوفة size

لمعرفة عدد الصفوف والأعمدة نستخدم ايعاز حجم المصفوفة وحيث يمثل العدد الأول عدد  
الصفوف والثاني يمثل عدد الأعمدة

```
>>b=[1 2 3;4 5 6]
```

```
B=
```

```
1 2 3
```

```
4 5 6
```

```
>> size(b)
```

```
Ans =
```

```
2 3
```

2- جمع المصفوفات :

لغرض جمع المصفوفات يجب ان يكون عدد صفوفها وعدد أعمدها متساوية فيما بينها

```
>> a=[1 2; 3 4];b=[0 5; -2 6]
```

```
A=
```

```
1 2
```

```
3 4
```

```
B=
```

```
0 5
```

```
-2 6
```

تجري عملية الجمع باضافة كل من المصفوفة الأولى الى الذي يقابله من نفس الموقع في المصفوفة  
الثانية و هكذا

```
>add=a+b
```

```
Add=
```

```
1 7
```

```
1 10
```

3- طرح المصفوفات

تتم عملية الطرح بطرح كل عنصر من المصفوفة الأولى من العنصر المقابل من حيث نفس الموقع  
من المصفوفة الثانية بشرط تطابق ابعاد المصفوفتين

```
>>ab=a - b
```

```
Ab =
```

```
1 -3
```

## 4- اضافة قيمة عددية الى مصفوفة

يمكن اضافة قيمة عددية الى عناصر مصفوفة ذات بعد واحد أو بعدين :

```
>>as= a + 10
```

```
As=
```

```
11 12
```

```
12 14
```

تكرار وتدوير المصفوفات

توجد مجموعة من الأيعازات التي تقوم بتدوير المصفوفة أو تكرارها وسوف نستعرض بعض من الأيعازات

## 1- ايعاز repmat تكرار المصفوفات :

لغرض تكرار المصفوفات نستخدم الصيغتين التاليتين :

```
Y=repmat(a,p).....1
```

```
Y=repmat(a, rows , cols).....2
```

هذه الدالة تقوم بعملية تكرار لمصفوفة مدخلة مسبقاً ،حيث تقوم بمعالجة المصفوفة على انها كتلة واحدة وكأنها عنصر وحيد في مصفوفة و نريد تكرار هذا العنصر حسب رغبتنا

```
>>a=[1 2 3 ;4 5 6 ;7 8 9]
```

```
A=
```

```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

```
>> y=repmat(a ,2)
```

```
y=
```

```
(
1 2 3 1 2 3
4 5 6 4 5 6
7 8 9 7 8 9
1 2 3 1 2 3
4 5 6 4 5 6
7 8 9 7 8 9
)
```

مثال :

```
>>y=repmat(a,2,3)
```

Y=

```
1 2 3 1 2 3 1 2 3
4 5 6 4 5 6 4 5 6
7 8 9 7 8 9 7 8 9
1 2 3 1 2 3 1 2 3
4 5 6 4 5 6 4 5 6
7 8 9 7 8 9 7 8 9
```

2-ايغاز اعادة تشكيل المصفوفة reshape

حيث تقوم هذه الدالة باعادة ترتيب عناصر المصفوفة واعادة تشكيل ابعادها بالشكل المناسب

```
Y=reshape(a , rowno, colon)
```

حيث يمثل rowno عدد الصفوف بعد التشكيل و colon عدد الأعمدة بعد التشكيل

مثال :

```
>>a=[1 2 3 4; 5 6 7 8; 9 10 11 12;13 14 15 16]
```

A=

```
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16
```

```
>>y= reshape(a,2,8)
```

Y=

```
1 9 2 10 3 11 4 12
```

5 13 6 1 7 15 7 16

### ضرب المتجهات

ليكن المتجه A و مكون من ثلاثة صفوف والمتجه B والمكون من ثلاثة اعمدة فحاصل ضربهما يساوي قيمة واحدة

```
>> A=[1 2 3], y=[4 ; 5 ; 6]
```

```
A=
```

```
1 2 3
```

```
Y=
```

```
4
```

```
5
```

```
6
```

```
>> dot(x,y)
```

```
32
```

ضرب عناصر اعمدة مصفوفة باستخدام الأيعاز prod لتكن A مصفوفة ذات بعدين فإن الأيعاز prod تجد حاصل ضرب قيم كل عمود من الأعمدة على حدة في المصفوفة A

```
>>a
```

```
A=
```

```
2 3 5
```

```
4 7 1
```

```
9 3 8
```

```
>> prod(a)
```

```
Ans=
```

```
72 63 40
```

ضرب صفوف مصفوفة للحصول على حاصل ضرب صفوف كل مصفوفة نستخدم الأيعاز التالي

```
>> prod(a,2)
```

```
Ans=
```

```
30
```

28  
216

### ترتيب مصفوفة

يمكن ترتيب عناصر المصفوفة نوعين اما تصاعدي أو تنازلي.

1- الترتيب التصاعدي على المصفوفات  $y = \text{sort}(a, r)$

حيث تعمل هذه الدالة بترتيب عناصر المصفوفات بالشكل التصاعدي للعناصر ، والمتغير  $r$  يتم من خلاله الترتيب صفيا أو عموديا ، حيث الرقم 1 يطبق الترتيب على عناصر العمود ، و الرقم 2 يطبق الترتيب على عناصر الصف حيث لا يمكن قبول غير هذين الرقمين  
مثال :

```
>> a=[3 5 1;7 0 2;4 7 5]
```

```
A=
```

```
3 5 1
7 0 2
4 7 5
```

```
>> y=sort(a,1)
```

```
Y=
```

```
3 0 1
4 5 2
7 7 5
```

تم ترتيب المصفوفة تصاعديا اعتمادا على الأعمدة ، نجد كل عمود تم ترتيبه تصاعدياً

```
>> y=sort(a,2)
```

```
Y=
```

```
1 3 5
0 2 7
4 5 7
```

تم ترتيب المصفوفة تصاعديا اعتمادا على الصفوف ، نجد ان كل صف تم ترتيبه تصاعديا على حدة

```
>> sort(a,3)
```

```
??? error using ==> sort
```

الترتيب التنازلي ( من الأكبر الى الأصغر )  $y = \text{sort}(a,r,'decend')$

هنا يمكن - بواسطة الدالة اعلاه - ترتيب عناصر المصفوفة من الأصغر إلى الأكبر وضمن الخيارات التي يعطينا اياها المتغير r والتي تم شرحها سابقاً

```
>> a=[1 2 3; 4 5 6; 7 8 9]
```

```
A=
```

```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

```
>> y=sort(a,1,'decend')
```

```
Y=
```

```
7 8 9
```

```
4 5 6
```

```
1 2 3
```

```
>> y=sort(a ,2,'decend')
```

```
Y=
```

```
3 2 1
```

```
6 5 4
```

```
9 8 7
```

بما ان المصفوفة من النوع ثنائية الأبعاد فلا يتم قبول غير الرقمين 1 و 2 للمتغير r لأنه ضمن الأطار المسموح به فلو كانت المصفوفة ثلاثية الأبعاد لأضفنا البعد الثالث للمتغير r حيث سيتم ترتيب البعد الثالث

## 2.4 تكوين المصفوفات الخاصة

هنالك مجموعة من الأيعازات التي تكون مصفوفة و من هذه الأيعازات :

1- المصفوفات التي جميع عناصرها واحداث

هي المصفوفة التي كافة عناصرها يساوي واحد

مثال :

```
>> ones(3)
```

```
Ans=
```

```
1 1 1
```

```
1 1 1
```

```
1 1 1
```

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

```
Ans=
```

```
1 1 1  
1 1 1
```

```
>> ones(3)*5
```

```
Ans=
```

```
5 5 5  
5 5 5  
5 5 5
```

## 2- المصفوفات الصفرية

و هي المصفوفات التي جميع عناصرها يساوي صفر

```
>> zeros(3)
```

```
Ans=
```

```
0 0 0  
0 0 0  
0 0 0
```

```
>> zeros(2,3)
```

```
Ans=
```

```
0 0 0  
0 0 0
```

```
>> zeros(size(a))
```

```
Ans=
```

```
0 0 0  
0 0 0  
0 0 0
```

## 3- المصفوفات المحايدة

و هي المصفوفات التي قطرها الرئيسي يساوي واحدو بقية عناصرها اصفار . يمكن استخدام الأيعاز eye للحصول على مصفوفة محايدة

```
>> r=eye(3)
```

```
R=
```

```
1  0  0
```

```
0  1  0
```

```
0  0  1
```

```
>>a=[1  2;3  4]
```

```
A=
```

```
1  2
```

```
3  4
```

```
>> s1=eye(size(a))
```

```
S1=
```

```
1  0
```

```
0  1
```

```
>> q=eye(3,4)
```

```
Q=
```

```
1  0  0  0
```

```
0  1  0  0
```

```
0  0  1  0
```