IMAGE PROCESSING WITH MATLAB

ARRAY

Setting By : L. Waleed Rasheed

Second Lecture

المصفوفات Arrays

توفر برمجة ماتلاب قدرة وسهولة كبير في معالجة المتجهات Vectors وهي المصفوفة ذات البعد الواحد ان كانت عامود Column او صف row والمصفوفات Matrices وهي المصفوفات متعددة الابعاد والتي تملك عدد معين من الصفوف والاعمدة.

بناء المتجهات

يمكن انشاء متجه يحتوي على مجموعة مختلفة من الارقام، يتم الفصل بين تلك الارقام بواسطة فوارز commas او فراغات spaces، ويتم احاطة الارقام باقواس مربعة [square brackets]، كما موضح في المثال التالي...

```
>> Z = [2,4,6,8]
Z =
    2     4     6     8
>> Y = [4 -3 5 -2 8 1]
Y =
    4    -3     5    -2     8     1
```

بناء المتجهات

ويمكن بناء متجه بارقام متسلسلة وبدون الحاجة لطباعة كل الارقام ، فمثلا لبناء متجه يحتوي الارقام من 1 الى 9 تكون كتابة الامر بالشكل التالي..

```
>> X = 1:9
X =
    1    2    3    4    5    6    7    8    9
```

نلاحظ الزيادة بين الارقام في المثال السابق افتراضياً بمقدار 1 ، ويمكن التحكم بالزيادة او النقصان من خلال اضافة رقم وسطي بين رقم الاول والرقم الاخير، وكما موضح في المثال التالي..

```
>> X = 0:2:10
X =
     0     2     4     6     8    10
```

```
>> X = 0:0.2:1
X =
         0    0.2000    0.4000    0.6000    0.8000
1.0000
```

```
>> X = 10:-1:0
X =
    10     9     8     7     6     5     4     3
2     1     0
```

ويمكن التعامل مع محتويات اي عنصر داخل المتجه من خلال ذكر اسم المتجه وتسلسل العنصر او مجموعة عناصر، كما موضح في المثال التالي...

عرض العنصر الاول والرابع فقط	عرض اول ثلاث عناصر	عرض محتويات العنصر الثالث
>> X(1:3:4) ans = 0 6	>> X(1:3) ans = 0 2 4	**>> X(3)** **ans =** **4**

التعامل مع المتجهات

لتغيير المتجه X من متجه صف الى متجه عمود يتم اضافة علامة (') prime بعد اسم المتجه.

```
>> X = 0:2:10
X =
    0    2    4    6    8   10
```

```
>> X'
ans =
    0
    2
    4
    6
    8
   10
```

يمكن ان تؤدي العديد من العمليات الرياضية على المتجهات ، فمثلاً لتربيع عناصر المتجه X يتم اضافة نقطة قبل علامة الرفع لقوة (^) .

```
>> X.^2
ans =
    0    4   16   36   64  100
```

ان وجود النقطة مهم جدا قبل علامة الرفع لان المقصود هو تربيع عناصر المتجه ،اما اذا حذفنا النقطة من ذلك الامر سيكون المقصود تربيع المتجه بشكل كامل (ضرب المتجه في نفسه $X^{\wedge 2}$).

بناء المصفوفات

المصفوفة matrix هي عبارة عن مستطيل من الاعداد منظم على شكل صفوف واعمدة، ويتم الفصل بين الاعداد في الصف الواحد بواسطة الفوارز ويكون الفصل بين صف واخر بواسطة الفوارز المنقوطة semicolons ،فمثلا لبناء مصفوفة تتكون من 3 صفوف و 4 أعمدة يكتب الامر بالشكل التالي...

```
>> A = [1, 2, 3, 4; 5, 6, 7, 8; 9, 10, 11, 12]
A =
    1    2    3    4
    5    6    7    8
    9   10   11   12
```

العمليات الرياضية على المصفوفات

ان اجراء العمليات الرياضية لا يتطلب فقط ان نعرف الرمز المستخدم للعملية ولكن تحديد اذا كانت العملية تشمل المصفوفة بمجملها او عناصرها بشكل مفرد ومراعاة بعض الشروط الاخرى.

فمثلاً جمع وطرح المصفوفات يتطلب ان تكون كلتا المصفوفتين تملكان نفس عدد الصفوف والاعمدة

وفي ضرب عناصر المصفوفات يجب ان يكون عدد الاعمدة في المصفوفة الاولى مساويا لعدد الصفوفة في المصفوفة الثانية، حيث سيكون حجم المصفوفة الناتجة من عملية الضرب مساويا لعدد الصفوف في المصفوفة الاولى وعدد الاعمدة في المصفوفة الثانية.

العمليات الرياضية على المصفوفات

ومن خلال المثال التالي سنوضح بعض تلك العمليات..

>> D=[1 2 3; 4 5 6; 7 8 9];	بناء المصفوفة D تتكون من 3 صفوف و 3 اعمدة
>> P=[1 1 1; 4 7 5; 2 2 8];	بناء المصفوفة P تتكون من 3 صفوف و 3 اعمدة
>> Sum = D + P	جمع عناصر D و P وينتج عنه المصفوفة Sum
>>Sub = P - D	طرح عناصر المصفوفة D من P وينتج عنه المصفوفة Sub
>> D = D + 2	يضيف 2 الى كل عدد من عناصر المصفوفة D
>> P1 = P * 2	ينتج مصفوفة P1 من حاصل ضرب عناصر المصفوفة P في 2
>> M1 = P * D	ينتج عنها مصفوفة M1 من حاصل ضرب المصفوفتين P في D(طريقة ضرب الصف في العمود ثم جمع العناصر)
>> M2 = P.* D	ينتج المصفوفة M2 من حاصل ضرب العناصر في المصفوفة P مع العناصر المقابلة لها في المصفوفة D

ويمكن اجراء القسمة والقسمة العكسية على العناصر المتقابلة بين مصفوفتين على ان يكون عدد الصفوف والاعمدة في الاثنين متساوي، وكما موضح في المثال التالي...

```
>> x=[4 2 ; 6 8];
>> z=[2 2 ; 3 2];
```

```
>> x./z
ans =
     2     1
     2     4
```

```
>> x.\z
ans =
    0.5000    1.0000
    0.5000    0.2500
```

حذف الصفوف او الاعمدة

يمكن حذف صف او عمود من اي مصفوفة ثنائية الابعاد وبالشكل التالي...

بناء مصفوفة A من 3 صفوف و3 اعمدة	حذف العمود الاول	حذف الصف الثاني
>> A=[1 2 3 ; 4 5 6 ; 7 8 9] A = 1 2 3 4 5 6 7 8 9	>> A(:,1)=[ ] A = 2 3 5 6 8 9	>> A(2,:)=[ ] A = 2 3 8 9

وهنا نذكر انه لا يمكن حذف عنصر من المصفوفة بشكل مفرد.

ولتغيير قيمة اي عنصر في المصفوفة يمكن عمل ذلك من خلال كتابة الامر التالي..

>> A(2,1)=4

المصفوفات الخاصة

هنالك مجموعة من الدوال تستخدم لبناء مصفوفات رقمية خاصة بدون الحاجة الى كتابة الارقام داخل المصفوفة، ومنها ما يلي:

1-دالة pascal: تستخدم لبناء مصفوفات متماثلة symmetric matrices حيث كل رقم هو حاصل جمع الرقم السابق له في الصف والعمود ، ويجب ان تكون المصفوفة مربعة ويكفي ان نذكر رقم واحد لتحديد ابعاد المصفوفة، كما موضح في المثال التالي...

```
>> P = pascal(3)
P =
     1     1     1
     1     2     3
     1     3     6
```

2-دالة magic: تقوم بتوليد ارقام عشوائية صحيحة حاصل جمع عناصر اي عمود او صف فيها مساوي للصفوف والاعمدة الاخرى، ويجب ان تكون المصفوفة مربعة ايضا، وان اكبر رقم في تلك المصفوفة هو اصغر او يساوي تربيع الرقم الذي يمثل عدد الصفوف او الاعمدة(3^2=9)، كما موضح في المثال التالي

```
>> M= magic(3)
M =
     8     1     6
     3     5     7
     4     9     2
```

المصفوفات الخاصة

3-دالة zeros: تولد مصفوفة اصفار حسب ابعاد الصفوف والاعمدة، كما موضح في المثال التالي..

```
>> N=zeros(2,4)
N =
     0     0     0     0
     0     0     0     0
```

4-دالة ones : تولد مصفوفة كل عناصرها تساوي واحد وحسب الابعاد المخصصة لها،كما موضح في المثال التالي...

```
>> R=ones(2,3)
R =
     1     1     1
     1     1     1
```

5-دالة rand : تولد مصفوفة من الارقام العشوائية بين (0-1) وحسب الابعاد المخصصة لها، وكما يلي..

```
>> A=rand(3)
A =
    0.8147    0.9134    0.2785
    0.9058    0.6324    0.5469
    0.1270    0.0975    0.9575
```

المصفوفات الخاصة

6-دالة randn: تولد مصفوفة من الارقام العشوائية بارقام مختلفة موجبة وسالبة، وحسب الابعاد المخصصة لها، وكما موضح في المثال التالي...

```
>> B=randn(3)
B =
    2.7694    0.7254   -0.2050
   -1.3499   -0.0631   -0.1241
    3.0349    0.7147    1.4897
```

7-دالة eye : تولد مصفوفة القطر الرئيسي فيها مساوي للواحد وبقية المواقع تساوي الصفر وحسب ابعاد المصفوفة، كما موضح في ما يلي:

```
>> C=eye(3)
C =
     1     0     0
     0     1     0
     0     0     1
```

```
>> C1=eye(3,5)
C1 =
     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
```

التعامل مع المشغلات المنطقية والعلائقية (المقارنة)

يمكن تطبيق كل المشغلات العلائقية (المقارنة) والمنطقية مع المصفوفات وبنفس الشكل التي يتم فيها تطبيقها مع المتغيرات العادية، كما في الأمثلة التالية

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

```
>> A==B
ans =
     0     0     0
     0     0     1
     0     0     0
```

```
>> A>B
ans =
     0     0     0
     0     1     0
     0     1     1
```

```
>> A<=B
ans =
     1     1     1
     1     0     1
     1     0     0
```

دالة (sum)

تقوم بحساب مجموع الارقام في المتجه (المصفوفة الاحادية) اذا كانت بشكل صف او عمود، اما اذا كانت المصفوفة ثنائية فان الامر sumسيقوم بحساب مجموع كل عامود على حدة ، ولحساب مجموع جميع عناصر المصفوفة يتم استخدام امر sum مرتين، وذلك موضح من خلال الامثلة التالية..

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

```
>> sum(A)
ans =
    12    15    18
```

```
>> sum(sum(A))
ans =
    45
```

وهذا ينطبق على العديد من الاوامر الاخرى مثل max لاستخراج اكبر قيمة و min لاستخراج اصغر قيمة و الامر sort الخاص بترتيب القيم.

منقول (مدور) المصفوفة

يمكن تحويل الاعمدة الى صفوف في المصفوفة الثنائية من خلال وضع علامة الاقتباس المفردة بعد اسم المصفوفة، وكما موضح في المثال التالي..

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

```
>> A'
ans =
     1     4     7
     2     5     8
     3     6     9
```

اعادة تشكيل المصفوفة

يستخدم الامر reshape لاعادة تغيير ابعاد المصفوفة الى أبعاد اخرى ، مثلاً تحويل مصفوفة ثنائية الى أحادية وكما موضح في المثال التالي..

```
>> B=[5 4 6; 7 3 6 ; 9 3 1 ]
B =
     5     4     6
     7     3     6
     9     3     1
```

```
>> B1=reshape(B,[1 9])
B1 =
     5     7     9     4     3     3     6     6     1
```

ملاحظة : ان الامر reshape يقوم بتحويل عمود بعد الاخر ، ولغرض تحويل سطر بعد الاخر نقوم بتطبيق المدور قبل استخدام امر اعادة التشكيل.

الامر find

يستخدم لإيجاد تسلسل عناصر في المصفوفة وحسب شرط معين، وفي المصفوفة الثنائية يكون التسلسل أستناداً الى الاعمدة، ويمكن تعويض القيم التي تم تحديد تسلسلها بقيم اخرى او التعامل معها بشكل مباشر، وكما موضح في المثال التالي...

```
>> B=[5 4 6; 7 3 6 ; 9 3 1 ]
B =
     5     4     6
     7     3     6
     9     3     1
```

```
>> i=find(B>=4)
i =
     1
     2
     3
     4
     7
     8
```

```
>> B(i)=5
B =
     5     5     5
     5     3     5
     5     3     1
```

دمج وتركيز المصفوفات

يمكن دمج اكثر من مصفوفة في مصفوفة واحدة او اعادة توليد مصفوفة جديدة من لمصفوفة الاصلية مع مراعاة ان تكون الابعاد بشكل قياسي، وكما موضح في المثال التالي...

```
>> C=ones(2)
C =
     1     1
     1     1
```

```
>> C1=[ C C+2 ; C+4 C+6]
C1 =
     1     1     3     3
     1     1     3     3
     5     5     7     7
     5     5     7     7
```

يمكن معرفة حجم المصفوفة اي لحظة من خلال استخدام الامر Size ، واذا طبقنا هذا الامر مع المصفوفة السابقة C1 فسيظهر لنا (عدد الصفوف ، عدد الاعمدة) على التوالي، وكما موضح في المثال التالي...

```
>> size(C1)
ans =
     4     4
```

المصفوفة متعددة الابعاد

ويقصد بها المصفوفة ثلاثية الابعاد فما فوق ، ويمكن تشكيلها من خلال دمج المصفوفات مع بعضها البعض، ويعتمد فيها نفس اسلوب التسلسل في المصفوفة الثنائية وتعرض في برنامج ماتلاب على شكل صفحات pages حيث تتكون كل صفحة من مصفوفة ثنائية ، ويجب ان تكون ابعاد الصفحة متساوية مع بقية الصفحات، وهنالك العديد من الطرق لتكوين المصفوفة المتعددة الابعاد فمثلاً لتكوين مصفوفة اصفار من صفحتين وكل صفحة تتكون من صفين و ثلاثة اعمدة يكتب الامر بشكل التالي..

```
>> Z=zeros(2,3,2)
Z(:,:,1) =
     0     0     0
     0     0     0

Z(:,:,2) =
     0     0     0
     0     0     0
```

ويمكن ان يكون لكل صفحة ارقام مختلفة عن الصفحة الاخرى، وكما موضح في المثال التالي...

```
>> X (:, :, 1) = zeros (2);
>> X (:, :, 2) = ones (2);
>> X (:, :, 3) = 3;
```

```
>> X
X(:,:,1) =
     0     0
     0     0

X(:,:,2) =
     1     1
     1     1

X(:,:,3) =
     3     3
     3     3
```