

لغة (MATLAB) : هي لغة برمجة عالية الاداء تستخدم لاجراء الحسابات التقنية وتقوم بالحساب والاطهار ضمن بيئة سهلة البرمجة, حيث لايعتبر المسالة وحلها باشكال رياضية مشهورة. كما يمكنها القيام بما يلي:

1. اجراء العمليات الرياضية والحسابية
2. تطوير الخوارزميات
3. النمذجة والمحاكاة
4. تحليل واطهار المعطيات
5. اجراء الرسوم البيانية والهندسية
6. تطوير التطبيقات

سطح مكتب MATLAB

نوافذ سطح المكتب	
الوصف	النافذة
تمتلك من كتابة الاوامر لينفذها برنامج MATLAB	نافذة الاوامر (Command Window)
تمتلك من اعادة تنفيذ الاوامر السابقة المنفذة في نافذة الاوامر	الاوامر السابقة (Command History)
وهي عبارة عن بنية شجرية للادوات والبرامج التوضيحية والوثائق	لوحة البرامج التنفيذية (Launch Pad)
تسمح لك هذه النافذة باستعراض وتحميل وحفظ متغيرات لغة MATLAB	ساحة العمل (Workspace)
تمتلك هذه النافذة لتعديل محتوى متغيرات لغة MATLAB	محرر المصفوفة (Array Editor)
محرر ومنتبع الاخطاء للملفات الخاصة بلغة MATLAB	محرر-منتبع الاخطاء (Editor-Debugger)
يتم في هذه النافذة تحديد الدليل الحاي للملف الذي يتعامل معه برنامج MATLAB	الدليل الحالي (Current Directory)
تمتلك من البحث واستعراض الوثائق التعليمية بشكل مباشر	(Help) المساعدة

معلومات حول المتغيرات

يملك برنامج **MATLAB** قواعد خاصة باسماء المتغيرات كما هو الحال بالنسبة الى اللغات الاخرى, وسوف نذكر اهم تلك القواعد في الجدول ادناه مع ذكر بعض الامثلة عليها:-

امثلة	قواعد تسمية المتغير
Cost, cosT, COST هذه المتغيرات كلها مختلفة	اسماء المتغيرات حساسة لحالة الاحرف من حيث الاحرف الكبيرة والصغيرة
The-first-value	يمكن استخدام متغير يحوي على 63 رمزا, وسيهمل اي رمز زائد عن 63
First-value X542 a-s-e-t-o	يجب ان تبدأ أسماء المتغيرات بحرف متبوعا بأي عدد من الارقام او الاحرف او الخطوط السفلية (_) (under line) والفراغات غير مسموح بها بين الاحرف

هناك بعض الاستثناءات للقواعد في الجدول اعلاه, اذ ان للغة **MATLAB** عدة اسماء لا يمكن استخدامها كأسماء متغيرات, حيث تمثل هذه الاسماء كلمات مفتاحية في **MATLAB**, مثل:

For, end, if, while, function, return, elseif, case, otherwise, switch,
continue, else, try, catch, persistent, break

الدوال المثلثية (Trigonometric Function)

مثال	توضيح	اسم الدالة
>> acos(0.5) ans=1.0472	معكوس الدالة cos	acos
>> acot(1) ans=0.7854	معكوس الدالة cot	acot
>> acsc(1) ans=1.5708	معكوس الدالة csc	acsc
>> asec(1) ans=0	معكوس الدالة sec	asec
asin(1) ans=1.5708	معكوس الدالة sin	asin
>> atan (1) ans=0.7854	معكوس الدالة tan	atan
>> cos (1) ans=0.5403	الدالة cos	Cos
>> cot (1) ans=0.6421	الدالة cot	Cot
>> csc (1) ans=1.1884	الدالة csc	Csc
>> sec (1) ans=1.8508	الدالة sec	Sec
>> sin (1) ans=0.8415	الدالة sin	Sin
>> tan (1) ans=1.5574	الدالة tan	Tan

Exponential Function

مثال	توضيح	اسم الدالة
>> exp (0) ans=1	الدالة الاسية e	Exp
>> log (1) ans=0	الدالة ln	log
>> Log10 (2) ans=0.3010	دالة اللوغاريتم للاساس 10	log10
>> Log2 (2) ans=1	دالة اللوغاريتم للاساس 2	log2
>> Pow2 (3) ans=8	دالة الرفع الى قوة للاساس 2	pow2
>> sqrt (4) ans=2	دالة الجذر التربيعي	sqrt

مثال: مثل المعادلة الاتية بلغة MATLAB

$$\sqrt[2]{x} + \ln x + \sin(x) + \cos^{-1}(x)$$

الحل

$$\text{Sqrt}(x)+\log(x)+\sin(x)+\text{acos}(x)$$

دوال التدوير والبقية (Rounding and Remainder Function)

1. Fix: دالة التقريب نحو الصفر

(Ex1) >> fix (3.1)

ans=3

(Ex2) >> fix (3.9)

ans=3

(Ex3) >> fix (3)

ans=3

(Ex4) >> fix(-3.9)

ans=-3

2. floor: دالة التقريب نحو اللانهاية السالبة ($-\infty$)

(Ex1) >> x=floor (-3.5)

x=-4

(Ex2) >> y=floor (6.9)

y=6

(Ex3) >> floor (-4.1)

ans=-5

(Ex4) >> floor (1.1)

ans=1

3. ceil: دالة التقريب نحو اللانهاية الموجبة ($+\infty$)

(Ex1) >> ceil (-1.1)

ans=-1

(Ex2) >> w= ceil (1.3)

w=3

(Ex3) >> ceil (2.9)

ans=3

4. round: دالة التقريب نحو اقرب عدد صحيح

(Ex1) >> round (-9.9)

ans =-10

(Ex2) >> round (3.2)

ans =3

(Ex3) >> round (-1.2)

ans=-1

(Ex4) >> round (4.6)

ans=5

5. rem : دالة الباقي من القسمة

(Ex1) >> rem (8,8) ans=0	% 8/8=1 remained 0
(Ex2) >> x=rem (10,6) x=4	% 10/6=1 remained 4
(Ex3) >> .rem (-15,4) ans=-3	% -15/4=-3 remained -3
(Ex4) >> y=rem (15,-4) y=3	% 15/-4=-3 remained 3
(Ex5) >> rem (-15,-4) ans=-3	% -15/-4=3 remained -3
(Ex6) >> rem (8.10) ans=8	% 8/10=0 remained 8
(Ex7) >> rem (-1.20) ans=-1	% -1/20=0 remained -1

ملاحظة: تلاحظ في الامثلة السبعة السابقة وجود الرمز (%) حيث يمثل رمز تعليق لاينفذ بعده اي دالة او رمز اونص, ولذلك يستخدم بكثرة لتوضيح الخطوات البرمجية المقابلة له

6. Sign: دالة الاشارة, حيث تعطي هذه الدالة ماياتي

1 اذا كان الرقم المعطى اكبر من الصفر

0 اذا كان الرقم المعطى مساوي من الصفر

-1 اذا كان الرقم المعطى اصغر من الصفر

(Ex1) >> Sign (0)

ans=0

(Ex2) >> Sign (-9)

ans=-1

(Ex3) >> Sign (0.55)

ans=1

ملاحظة: يمكن ان تكون المدخلات للدوال السابقة على شكل مصفوفة واليك هذه الامثلة

```
>> x= [0 -1.2 3.9 4 -5];
```

```
>> fix (x)
```

```
ans=0 -1 3 4 -5
```

```
>> floor(x)
```

```
ans=0 -2 3 4 -5
```

```
>> ceil(x)
```

```
ans=0 -1 4 4 -5
```

```
>> round (x)
```

```
ans=0 -1 4 4 -5
```

```
>> sign (x)
```

```
ans=0 -1 1 1 -1
```

اما اذا اردنا ان نستخدم مصفوفات مع الدالة rem فيجب مراعاة تساوي عدد الصفوف والاعمدة في المصفوفتين كما في المثال الاتي

```
>> x= [2 -5 10.6 8 -1];
```

```
>> y= [1 3 4.2 5 6];
```

```
>> x(x,y)          %2/1 -5/3  10.6/4.2   8/5   -1/6
```

```
Ans=0-2.0000      2.2000      3.0000      -1.0000
```

المصفوفات

تتكون المصفوفة من اعمده بوصفوف ويمكن الوصول الى قيم تلك المصفوفة عن طريق الفهرسه المباشره (اي تحديد موقع العنصر من ناحيه الصفوف والاعمده), وكما نعلم ان ابعاد اي مصفوفه تحدد عبر عدد الاعمده والصفوف التي تمتلكها تلك المصفوفه. الشكل التالي يوضح مصفوفه تمتلك N من الاعمده وM من الصفوف

		Column				
		1	2	3	...	N
ROW	1					
	2					
	3					
	.					
	.					
M						

يتعامل MATLAB مع المصفوفات بسهولة ومرونة, حيث تتوفر عدة طرق لتكوين مصفوفات تمتلك ابعاد مختلفة (مكونة من صفوف واعمده) ولكن هناك بعض الاساسيات التي وضعها MATLAB للتعامل مع المصفوفات.

لقد حدد MATLAB رمز الفارزة (, comma) او الفراغ (space) للفصل بين قيم الاعمده وتحديد عددها, اما الفارزة المنقوطة (; semicolon) فانها تستخدم للفصل ما بين قيم الصفوف وتحديد عددها, اما الاقواس المربعة [] فانها تدل على بدايه ونهاية عناصر المصفوفة, حيث تستخدم هذه الاقواس في عملية انشاء المصفوفات فقط ويستعاض عنها بالاقواس الهلالية () لعملية الفهرسة والوصول الى تلك العناصر (كما سنرى في فقرة ادرارة المصفوفات).

ولكي نوضح الكلام السابق, اليك المثال التالي:

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

```
X = 1 2 3
```

```
4 5 6
```

```
7 8 9
```

لاحظ في بداية ادخال المصفوفة وضعنا القوس المربع [دلالة على بداية المصفوفة ومن ثم ادخلنا القيمة الرقمية 1 ووضعنا فراغ (space) ((يمكن وضع فارزة بدلا عن الفراغ)) لكي ندخل القيمة الرقمية 2 والتي تمثل العمود الثاني من الصف الاول ونكرر العملية بالنسبة للقيمة 3 بعد ذلك وضعنا فارزة منقوطة لكي نتمكن من ادخال قيم الصف الثاني وهكذا حتى تنتهي من ادخال القيم ونضع القوس المربع الثاني] لكي نشير الى نهاية المصفوفة.

انشاء المصفوفات

لقد تطرقنا الى الاساسيات التي يضعها MATLAB لكي نتمكن من انشاء مصفوفة متكونة من عدة صفوف واعمدة, والان نتطرق لنوع ثاني من المصفوفات والذي يطلق عليه المتجه (Vector) الذي يتكون اما من صف واحد و عدة اعمدة حيث يسمى بالمتجه الصفي او الافقي (Row Vector) اما الثاني فيتكون من عمود واحد و عدة صفوف ففي هذه الحالة يسمى بالمتجه العمودي (Column Vector)

طرق انشاء المتجه الصفي (Row Vector)

الطريقة الاولى: تعتمد هذه الطريقة نفس الاساسيات المتبعة في تكوين المصفوفة ذات اعمدة و صفوف, حيث نستخدم الفارزة (, comma) او الفراغ (space) للفصل بين قيم الاعمدة وتحديد عددها (اي انشاء صف), اما الاقواس المربعة [] فانها تدل على بدايه ونهاية المتجه.

```
>> V= [1 2 3 4 ]
```

```
V= 1 2 3 4
```

```
>> X= [-1,0,1 ]
```

```
X= -1 0 1
```

الطريقة الثانية: تعتمد هذه الطريقة لانشاء نوع خاص من المتجه الصفي الصيغة التالية:

```
X=first: last
```

حيث يتم انشاء متجه X يبدأ بقيمة تحدد عبر first و بزيادة 1 حتى تنتهي قبل او بالقيمة last

```
(Ex1) >> w=1:7
```

```
w=1 2 3 4 5 6 7
```

```
(Ex2) >> x=-1:2
```

```
x=-1 0 1 2
```

```
(Ex3) >>w=0.1:2
```

```
w=0.1000 1.1000
```

```
(Ex4) >> s=4:2
```

```
s=Empty matrix: 1-by-0
```

ملاحظة:

** لاحظ المتجه في المثال الثالث قد حدد 0,1 للبداية و 2 للنهاية لكن كيف كانت عناصر هذا المتجه ((العنصر الاول 0,1 وبزيادة واحد يصبح العنصر الثاني 1,1 الى هنا تنتهي العناصر؟ لان لو نستمر بالزيادة لاصبح العنصر الثالث 2,1 وهذا العدد خارج عن القيمة 2 التي تمثل last لذلك قلنا في وصف هذه الطريقة انه قد ينتهي قبل او بالقيمة last))

** لاحظ المتجه بالمثال الرابع لا يوجد فيه اي عنصر Empty matrix؟ لان قيمة first (4) اكبر من قيمه last (2)

** ان الصيغة x=[first: last] تعطي نفس النتائج لكن تستغرق زمنا اطول في التنفيذ.

الطريقة الثالثة: لو استرجعنا الطريقة الثانية لوجدنا ان الزيادة تكون بمقدار 1 على المقدار first لكن الصيغة الاتية والتي تمثل الطريقة الثالثة فنستطيع ان نحدد مقدار الزيادة التي نرغب فيها ولك الصيغة العامة:

X= first: increment: last

```
>> q=1:2:5
```

```
q=1 3 5
```

```
>> w=0.1:0.1:1
```

```
w=0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
```

```
>> -2:2:5
```

```
ans = -2 0 2 4
```

طرق انشاء المتجه العمودي (Column Vector)

لكي نعرف كيفية انشاء متجه عمودي, سوف نضع ملاحظات على الطرق الثلاث السابقة وبالتالي يعتبر هذه الطرق الثلاث هي المتبعة في تكوين المتجه العمودي

الطريقة الاولى:

**في هذه الطريقة سوف نستخدم الفارزة المنقوطة (; semicolon) للفصل ما بين قيم الصفوف وتحديد عددها(اي انشاء عمود), اما الاقواس المربعة [] فانها تدل على بدايه ونهاية عناصر المتجه.

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

```
ans = 1
      2
      3
      4
```

**او ندخل القيم كمتجه صفي لكن مع وضع علامة اقتباس مفردة (!) عند القوس المربع الاخير

```
>>W=[ 1 2 3]'
```

```
W = 1
     2
     3
```

الطريقة الثانية:

نستخدم الصيغة التالية '(first: last) اي فقط وضعنا اقواس هلالية مع علامة الاقتباس المفردة في الاخير

```
>> (3:5)'
```

```
ans= 3
     4
     5
```

الطريقة الثالثة:

نستخدم الصيغة التالية '(first: increment: last)' ايضا الفارق فقط وضع اقواس هلالية مع علامة الاقتباس المفردة في الاخير.

```
>>A= (1:2:6)'
```

```
A=  1
    3
    5
```

لقد تطرقنا فيما سبق الى كيفية انشاء مصفوفة (Matrix) ذات اعمدة وصفوف متعددة وذلك عبر التخصيص المباشر للقيم الرقمية مع التقييد بالقواعد الخاصة بالمصفوفات من حيث الفصل بين الاعمدة والصفوف باستخدام الفارزة والفراغ لتحديد الاعمدة واستخدام الفارزة المنقوطة لتحديد الصفوف; وكذلك تطرقنا الى كيفية انشاء المتجه العمودي والافقي. والان نتطرق الى بعض الدوال التي يوفرها MATLAB والتي يمكن من خلالها انشاء مصفوفات قياسية تتمتع ببعض الخواص نذكر اهمها

1. ones: تستخدم هذه الدالة لتكوين مصفوفة مربعة $n \times n$ او مصفوفة ذات ابعاد $m \times n$ تاخذ كل عناصرها الرقم (1), هناك صيغتين لهذه الدالة:

*ones(n) (مصفوفة مربعة $n \times n$)

*ones(m, n) (مصفوفة ذات ابعاد $m \times n$)

(Ex1) >>a=ones(3)

(Ex2) >> ones(3,2)

a=

ans=

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

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

2. zeros: تستخدم هذه الدالة لتكوين مصفوفة مربعة $n \times n$ او مصفوفة ذات ابعاد $m \times n$ تاخذ كل عناصرها الرقم (0), هناك صيغتين لهذه الدالة:

* zeros (n) (مصفوفة مربعة $n \times n$)

* zeros (m, n) (مصفوفة ذات ابعاد $m \times n$)

>>a= zeros (2) >>a= zeros (2,3)

a= ans=

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

3. eye: تستخدم هذه الدالة لتوليد (Identity Matrix) التي تكون جميع عناصرها 0 عدا عناصر القطر الرئيسي التي تاخذ القيمة 1, تشابه الدالة eye في صيغتها الدالتين

ones & zeros

* eye (n) (مصفوفة مربعة $n \times n$)

* eye (m, n) (مصفوفة ذات ابعاد $m \times n$)

>>E= eye (4) >>E= eye (3,2)

E= E=

```
1 0 0 0      1 0
0 1 0 0      0 1
0 0 1 0      0 0
0 0 0 1
```

>>E= eye (2,3)

E=

```
1 0 0
0 1 0
```

ملاحظة: عند استخدام الدالة eye فانها تعطي جميع عناصر المصفوفة ذات الموقع (i,i) القيمة 1 وبقية العناصر القيمة 0 حيث i ياخذ القيمة الاقل من عدد الصفوف وعدد الاعمدة.

4. rand: تقوم هذه الدالة بإنشاء مصفوفة تحتوي عناصر بترقيم عشوائية تقع بين 0 و 1 وكما هو الحال في الدوال السابقة فإن هذه الدالة تأخذ أيضا صيغتين:

* rand (n) (مصفوفة مربعة $n \times n$)

* rand (m, n) (مصفوفة ذات ابعاد $m \times n$)

>>R= rand (3)

R=

```
0.9501  0.4321  0.9527
0.2311  0.7683  0.8735
0.6068  0.3862  0.0634
```

>>R= rand (4,3)

ans=

```
0.9501  0.4321  0.9527
0.2311  0.7683  0.8735
0.6068  0.3862  0.0634
0.8632  0.5972  0.0765
```

5. magic: تستخدم هذه الدالة في إنشاء المصفوفة السحرية حيث ناتج جمع عناصر اي صف او اي عمود او مجموع عناصر القطر الرئيسي او الثانوي يكون نفس الناتج.

الصيغة لهذه الدالة هي كالآتي:

magic(n)

حيث تولد مصفوفة سحرية لكل قيم $n > 0$ ما عدا $n = 2$ حيث لا تكون المصفوفة سحرية

>> magic (4)

ans=

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

```
>> magic (2)           %this Matrix not magic
```

```
ans=
```

```
1 3
```

```
4 2
```

فيما يلي بعض الدوال المهمة التي تتعامل مع المصفوفات

1. flipud: تعمل هذه الدالة على قلب المصفوفة من الاعلى للاسفل.

```
(Ex) >>a=[1 2; 3 4; 5 6; 7 8]
```

```
a=
```

```
1 2
```

```
3 4
```

```
5 6
```

```
7 8
```

```
>> flipud (a)
```

```
ans=
```

```
7 8
```

```
5 6
```

```
3 4
```

```
1 2
```

2. fliplr: تعمل هذه الدالة على قلب المصفوفة من اليسار لليمين.

```
(Ex) >>W=[1 2 3 4; 5 6 7 8]
```

```
W=
```

```
1 2 3 4
```

```
5 6 7 8
```

```
>> fliplr (a)
```

```
ans=
```

```
4 3 2 1
```

```
8 7 6 5
```

rot 90-3: تدوير المصفوفة 90 درجة.

حيث توجد صيغتان لهذه الدالة هما:

rot90 (A): تدوير المصفوفة A (90) درجة عكس عقارب الساعة

rot90 (A, k): تدوير المصفوفة A ($90*k$) درجة, حيث ان $k = -1, -2, \dots$

(Ex1) >>a=[1 2; 3 4]

a=

```
1 2
3 4
```

>> rot 90 (a)

ans=

```
2 4
1 3
```

(Ex2) >> rot 90 =(a,2) %rotate the Matrix(a) 90*2=180 Degree

ans=

```
4 3
2 1
```

(Ex3) >> rot 90 =(a,-1) %rotate the Matrix(a)90*-1=-90 Degree

ans=

```
3 1
4 2
```

diag -4 : دالة المصفوفات القطرية واقطار المصفوفة

diag(A, k) : تعطي هذه الصيغة متجه عمودي مكون من عناصر القطر (k)

للمصفوفة A.

diag(V) : تعطي هذه الصيغة مصفوفة قطرية وذلك بوضع المتجه V بالقطر الرئيسي لهذه المصفوفة, نذكر بان V هو متجه عمودي او افقي.

```
(Ex1) >>w=[1 2 3; 4 5 6; 7 8 9]
```

```
w=
```

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

```
>> diag (w)           %diag(w) is the same as diag(w,0)
```

```
ans=
```

```
1
5
9
```

```
(Ex2) >> diag =(w,1)
```

```
ans=
```

```
2
6
```

```
(Ex3) >> diag =(w,2)
```

```
ans=
```

```
3
```

```
(Ex4) >> diag =(w,-1)
```

```
ans=
```

```
4
8
```

```
(Ex5) >> diag =(w,-2)
```

```
ans=
```

```
7
```

```
(Ex6) >> diag =(w,-3)
```

```
ans=
```

```
Empty matrix: 0-by-1
```

```
(Ex7) >>v=[1 2 3]
```

```
v=
```

```
1 2 3
```

```
>> diag (v)
```

```
ans=
```

```
1 0 0
```

```
0 2 0
```

```
0 0 3
```

```
:triu-5
```

```
(Ex1) >>w=[1 2 3; 4 5 6; 7 8 9]
```

```
w=
```

```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

```
>> triu (w)
```

```
ans=
```

```
1 2 3
```

```
0 5 6
```

```
0 0 9
```

```
(Ex2) >>z=[1 2; 3 4; 5 6]
```

```
z=
```

```
1 2
```

```
3 4
```

```
5 6
```

```
>> triu (z)
```

```
ans=
```

```
1 2
```

```
0 4
```

```
0 0
```

```
(Ex3) >>y=[1 2 3 4; 5 6 7 8]
```

```
y=
```

```
1 2 3 4
```

```
5 6 7 8
```

```
>> triu (y)
```

```
ans=
```

```
1 2 3 4
```

```
0 6 7 8
```

:tril-6

```
(Ex1) >>w=[1 2 3; 4 5 6; 7 8 9]
```

```
w=
```

```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

```
>> tril (w)
```

```
ans=
```

```
1 0 0
```

```
4 5 0
```

```
7 8 9
```

```
(Ex2) >>z=[1 2; 3 4; 5 6]
```

```
z=
```

```
1 2
```

```
3 4
```

```
5 6
```

```
>> tril (z)
```

```
ans=
```

```
1 0
3 4
5 6
```

```
(Ex3) >>y=[1 2 3 4; 5 6 7 8]
```

```
y=
```

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

```
>> tril (y)
```

```
ans=
```

```
1 0 0 0
5 6 0 0
```

size-7: حجم المصفوفة

size(A)*: تعطي هذه الصيغة متجه صفي (Row Vector) متكون من قيميتين تمثل القيمة الاولى عدد الصفوف بينما تمثل الاخرى عدد الاعمدة للمصفوفة A.

size(A)=[R,C]: تعطي هذه الصيغة عدد الصفوف وعدد الاعمدة للمصفوفة A ولكن بشكل متغيرات منفصلة.

size(A,1)*: تعطي هذه الصيغة عدد الصفوف للمصفوفة A.

size(A,2)*: تعطي هذه الصيغة عدد الاعمدة للمصفوفة A.

```
(Ex1) >>M=[0 -1; 3 4; 5,1]
```

```
M=
```

```
0 -1
3 4
5 1
```

```
>> size (M)
```

```
ans=
```

```
3 2
```

```
(Ex2) >>[The-Number-of-Rows, The-Number-of-Columns]= size (M)
```

```
The-Number-of-Rows=3
```

```
The-Number-of-Columns=2
```

```
(Ex3) >>R= size(M,1)
```

```
R=
```

```
3
```

```
(Ex4) >>C= size(M,2)
```

```
C=
```

```
3
```

```
(Ex5) >>A=[0 -1 3 4 5 1]
```

```
A=
```

```
0 -1 3 4 5 1
```

```
>> size (A)
```

```
ans=
```

```
1 6
```

```
(Ex6) >>E=[ ] %E is Empty Matrix
```

```
E=
```

```
[ ]
```

```
>> size (E)
```

```
ans=
```

```
0 0
```

length-8: طول المتجه

Length(x): تعطي هذه الصيغة طول المتجه x, في الحقيقة فان الدالة Length تعطي الاكبر من قيمتي حجم المصفوفة اي مكافئة الى (max(size(x))

(Ex1) >>q=[1,2;3,4;5,6]

q=

```

1 2
3 4
5 6

```

>> Length (q)

ans= 3

(Ex2) >>w=[1,2,3,4,0]

w=

```

1 2 3 4 0

```

>> Length (w)

ans= 5

numel-9: عدد العناصر الموجودة في المصفوفة

(Ex1) >>a=[1 2 3 4]

a=

```

1 2 3 5

```

>> numel (a)

ans= 4

```
(Ex2) >>E=[ ]
```

```
>> numel(E)
```

```
ans= 0
```

det-10: محدد المصفوفة المربعة

```
(Ex1) >>a=[1 2; 3 4]
```

```
a=
```

```
1 3
2 4
```

```
>> det(a)
```

```
ans=-2
```

```
(Ex2) >>a=[1 2 2; 3 4 5]
```

```
a=
```

```
1 2 2
3 4 5
```

```
>> det(a)
```

```
???Error using ==> det
```

Matrix must be square.

inv-11: معكوس المصفوفة المربعة (تظهر رسالة تحذير اذا كان المحدد للمصفوفة مساوي للصفر)

```
(Ex1) >>a=[1 2; 3 4]
```

```
a=
```

```
1 3
2 4
```

```
>> inv (a)
```

```
ans=
```

```
-2.0000  1.0000  
 1.5000 -0.5000
```

```
(Ex2) >>w=[1 2; 4 8]
```

```
w=
```

```
 1  2  
 4  8
```

```
>> inv (w)
```

```
Warning: Matrix is singular to working precision
```

```
(Type" Warning off MATLAB : singular Matrix" to suppress this  
Warning.)
```

```
ans=
```

```
 Inf  Inf  
 Inf  Inf
```

ادارة المصفوفات (عملية الفهرسة)

بعد معرفة طرق انشاء المصفوفات والتعرف على اهم الدوال التي تتعامل معها بقي لدينا ان نعرف كيف يمكن ان نسترجع او نصل الى القيم الداخلية للمصفوفة متمثلة بقيمة واحدة او قيم صف او عمود وكيف نتعامل معها من حيث التغيير او الحذف او الاضافة وما الى غيرها من العمليات التي يمكن ان نجربها على على قيم المصفوفة كل هذه العمليات تسمى (ادارة المصفوفات او عملية الفهرسة) وفي الجدول الاتي صيغ عمليات الفهرسة.

الوصف	عملية الفهرسة
الوصول الى القيمة الواقعة في الصف r والعمود c ضمن المصفوفة A	$A(r, c)$
الوصول الى كل قيم الاعمدة الواقعة ضمن الصف r في المصفوفة A (الوصول الى قيم الصف r)	$A(r, :)$
الوصول الى كل قيم الصفوف الواقعة ضمن العمود c في المصفوفة A (الوصول الى قيم العمود c)	$A(:, c)$
تحويل المصفوفة A الى متجه عمودي باخذ اعمدها واحدا تلو الاخر	$A(:)$
الوصول الى القيمة الواقعة في الموقع i للمصفوفة A (يجب ان تكون المصفوفة A من نوع متجه عمودي او افقي والافسوف يتم عدد المواقع من الاعلى للاسفل لكل عمود مبتدا بالعمود الاول الى ان يصل للموقع i حيث توجد في القيمة المطلوبة	$A(i)$

A(r, c) - 1

نتمكن بهذه الصيغة من اجراء الاتي:

((مع العلم بان r هو الصف وc هو العمود))

** الوصول الى احد قيم المصفوفة

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

```
w=
```

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

```
>> w(2,3)          الوصول الى القيمة الواقعة في الصف الثاني والعمود الثالث%
```

```
ans=
```

```
6
```

```
>> w(3,3)          الوصول الى القيمة الواقعة في الصف الثالث والعمود الثالث%
```

```
ans=
```

```
9
```

```
>> w(4,3)          اعطي موقع خارج ابعاد المصفوفة%
```

```
???Index exceeds matrix dimensions.
```

**تغير احد قيم المصفوفة

((الامثلة القادمة تفترض المصفوفة W المعرفة في اعلاه دون اجراء اي تغير عليها))

```
>> اعطيت القيمة 0 الى الصف الثاني والعمود الثاني بدلا من 5 %
```

```
w(2,2)=0
```

```
w=
```

```
1 2 3
4 0 6
7 8 9
```

** توسيع المصفوفة وذلك باعطاء قيمة الى موقع خارج ابعاد المصفوفة

>> w(4,5)=6

W=

```
1 2 3 0 0
4 5 6 0 0
7 8 9 0 0
0 0 0 0 6
```

نلاحظ ان ابعاد المصفوفة قد تغير من 3-by-3 الى 4-by-5 وجميع القيم الجديدة اعطيت القيمة 0 لها ماعدا القيمة الواقعة في الصف الرابع والعمود الخامس فاخذت القيمة 6 المعطاة لها اصلا.

A(r, :)-2

نتمكن بهذه الصيغة من اجراء الاتي:

** استدعاء قيم اي الصف من صفوف المصفوفة

((تذكر المصفوفة w ذات ابعاد 3-by-3))

>>W(2,:) %Recall the Row 2

ans=

```
4 5 6
```

>>S=W(1,:) % S is the Row 1

S=

```
1 2 3
```

** تغير قيم الصف

قد ياخذ هذا التغير قيمة واحدة لجميع قيم الصف

>>W(3,:)=10 % All values in the Row 3 are 10

W=

```
1 2 3
4 5 6
10 10 10
```

او قد يكون التغير بقيم مختلفة

```
>>W(2,:)= [1 0 1]
```

```
W=
```

```
1 2 3
1 0 1
7 8 9
```

لاحظ ان الحالة الاولى يمكن كتابتها بالشكل $W(3,:) = [10\ 10\ 10]$

**** حذف احد الصفوف**

يمكن حذف اي صف من صفوف المصفوفة وذلك بمساواة الصف بالمتجه الفارغ (Empty Vector) والذي يكون بالشكل []

```
>>W(1,:)= [ ]
```

```
W=
```

```
4 5 6
7 8 9
```

**** اضافة صف الى المصفوفة**

مثلا استطعنا من حذف صف فيمكننا الان اضافة صف وذلك عبر تحديد موقع الصف وعناصره

```
>>W(5,:)= [ 1 2 3 ]
```

```
W=
```

```
1 2 3
4 5 6
7 8 9
0 0 0
1 2 3
```

لاحظ ان المصفوفة W كانت ذات ابعاد 3-by-3 وفي مثالنا استطعنا من زيادة المصفوفة بصفتين احديهما (الصف الرابع) اخذ عناصره القيمة 0 والآخر (الصف الخامس) اخذ القيم المعطاة له اصلا.

A(:, c) -3

هذه الصيغة عكس الصيغة اعلاه (الصيغة 2) حيث يمكن اجراء

** استدعاء قيم اي عمود من اعمدة المصفوفة

>>W(:,2)

ans=

2

5

8

** تغيير قيم العمود

ايضا كما في الصيغة 2 فهناك نوعين من التغيير

*فقد ياخذ هذا التغيير قيمة واحدة لجميع قيم العمود

>>W(:,1)=1

W=

1 2 3

1 5 6

1 8 9

*اويكون التغيير بقيم مختلفة

>>W(:,3)= [1; 0 ;1]

W=

1 2 1

4 5 0

7 8 1

من الجدير بالملاحظة هو مساواة W(:,3) بالمتجه [1; 0 ;1] بدلا من [1 0 1]؟

ج| هذا يرجع الى ان المتغير هنا عمود وليس صف (كما في الصيغة 2) لذلك يتطلب منا

المساواة بمتجه عمودي [1; 0 ;1] وليس افقي [1 0 1]

** حذف عمود

>>W(:,2)=[]

W=

1 3

4 6

7 9

** اضافة عمود

>>W(:,4)=4

W=

```

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

```

A(:)-4

هذه الطريقة تمكننا من تحويل المصفوفة الى متجه عمودي باخذ عناصر الاعمدة واحد تلو الاخر

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

W=

```

1 2 3
4 5 6
7 8 9

```

>>W(:)

ans=

```

1
4
7
2
5
8
3
6
9

```

سكيف يمكن تحويله من متجه عمودي الى افقي؟

جا وذلك بوضع عملية الاقتباس المفردة (') في نهاية القوس

>>W(:)'

ans=

```

1 2 3 4 5 6 7 8 9

```

A(i)-5

لنستعيد ماكتبناه في الجدول من وصف لرهذه العملية

((الوصول الى القيمة الواقعة في الموقع i للمصفوفة A (يجب ان تكون المصفوفة A من نوع متجه عمودي او افقي والافسوف يتم عد المواقع من الاعلى الى الاسفل لكل عمود مبتدأ بالعمود الاول الى ان يصل الى الموقع i حيث توجد في القيمة المطلوبة))

هذه العملية تفترض وجود نوعين من المصفوفات

** اذا كانت A عبارة عن متجه عمودي او افقي

هنا يكون الامر سهل جدا حيث نحدد موقع القيمة وسوف يرجع لنا القيمة نفسها

```
>>RV=[ 0 9 6 4 2 8 ];
```

```
>>RV(5)          سوف يعطي القيمة الواقعة في الموقع الخامس من المتجه%
```

```
ans=
```

```
2
```

```
>>RV(1)          سوف يعطي القيمة الواقعة في الموقع الاول من المتجه%
```

```
ans=
```

```
0
```

```
>>CV=[ 2;6;8 ];
```

```
>>CV
```

```
2
```

```
6
```

```
8
```

```
>>CV(3)
```

```
ans=
```

```
8
```

**اذا كانت A مصفوفة متعددة الصفوف والاعمدة

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

```
M=
```

```
1 3 4
5 7 9
```

```
>>M(4)
```

```
ans=
```

```
7
```

اذا تم عد المواقع من الاعلى الى الاسفل مبتدئين من العمود الاول فسوف نجد ان الموقع الرابع يحوي القيمة 7

```
>>M(6)
```

```
ans=
```

```
9
```

اذا تم عد المواقع من الاعلى الى الاسفل مبتدئين من العمود الاول فسوف نجد ان الموقع السادس يحوي القيمة 9

انشاء المصفوفات الجزئية

يمكن الاستفادة من بعض عمليات الفهرسة بتكوين مصفوفات جزئية من المصفوفة الاصلية المعطاة.

(Ex1) >>M=[1 3 4; 5 7 9]

M=

1 3 4
5 7 9

>> B=M(1:2,1:2)

B=

1 3
5 7

كيف يتم انشاء المصفوفة B ؟

انتبه اولاً الى الصيغة M(1:2,1:2) التي كونت المصفوفة الجزئية B

ماذا نجد؟

*اخذ رمز المصفوفة الاصلية وفتح اقواس لبدء تكوين المصفوفة الجزئية (,) M

*تحديد عدد الصفوف والاعمدة :

ففي مثالنا اعلاه نريد ان نكون مصفوفة جزئية تاخذ اول قيمتين من الصف الاول والثاني لذلك وضعنا 1:2 في مكان الصفوف (,) M(1:2,) وايضا نريد العمود الاول والثاني لذلك وضعنا 1:2 في مكان الاعمدة (, 1:2) M وبذلك اصبحت لدينا الصيغة النهائية للمصفوفة الجزئية التي اطلقنا اسميناها B بالشكل M(1:2,1:2). والتي تتكون من الصفين والعمودين الاوليين.

** ويمكن استخدام الصيغ الاخرى لايجاد مصفوفات جزئية مختلفة بتركيبها وحسب الحاجة , فعلى سبيل المثال اذا كانت لديك مصفوفة ذات ابعاد 3×3 وكنت تريد ان تنشأ مصفوفة جزئية من هذه المصفوفة ذات ابعاد 2×2 ولكن تاخذ القيمة الاولى والثالثة من الصف الاول والقيمة الاولى والثالثة من الصف الثالث (فماذا عليك ان تفعل)؟

ج| اين تقع هذه القيم من حيث الصفوف؟ (في الصف الاول والثالث) اذا نضع الصيغة 1:2:3 في مكان الصفوف (اي ياخذ 1 ومن ثم يزداد بمقدار 2 حتى ياخذ 3) اين تقع هذه القيم من حيث الاعمدة؟ (في العمود الاول والثالث) اذا نضع الصيغة 1:2:3 في مكان الاعمدة (اي ياخذ 1 ومن ثم يزداد بمقدار 2 حتى ياخذ 3).

ولتوضيح ذلك انظر المثال الاتي:

```
(Ex2) >>A=[1 2 3;4 5 6; 7 8 9]
```

A=

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

```
>> Z=A(1:2:3,1:2:3)
```

Z=

```
1 3
7 9
```

(Ex3)

انظر الى المصفوفة A اعلاه

```
>> D=A(3:-1:1,1:2)
```

D=

```
7 8
4 5
1 2
```