الدوال الرياضية :

1. الدوال الأسية :Exponential Functions

Example	Function in MATLAB form	Operation
>> exp(0) ans = 1	exp(x)	الدالة الاسية
>> log(1) ans = 0	log(x)	دالة اللوغارتيم الطبيعي ln
>> log10(2) ans = 0.3010	log10(x)	دالة اللوغارتيم للاساس10
>> log2(2) ans = 1	log2(x)	دالة اللوغارتيم للاساس2
>> pow2(3) ans = 8	pow2(x)	دالة الرفع الى قوة للاساس2
>> sqrt(4) ans = 2	sqrt(x)	دالة الجذر التربيعي
>>power(3,3) ans = 27	power(X,Y)	دالة الرفع للاساس x

2.الدوال المثلثية :Trigonometric Functions

مثال	الامر في برنامج MATLAB	الدالة المثلثية
>> sin(5) ans = -0.9589	sin(angle)	الدالة sin
>> cos(5) ans = 0.2837	cos(angle)	الدالة cos
>> tan(5) ans = -3.3805	tan(angle)	الدالة tan
>> sec(5) ans = 3.5253	sec(angle)	الدالة 1/cos
>> csc(5) ans =	csc(angle)	الدالة 1/sin

-1.0428		
>> cot(5) ans = -0.2958	cot(angle)	الدالة 1/tan
>> asin(5) ans = 1.5708 - 2.2924i	asin(angle)	معكوس الدالة sin
>> acos(5) ans = 0 + 2.2924i	acos(angle)	معكوس الدالة cos
>> atan(5) ans = 1.3734	atan(angle)	معكوس الدالة tan
>> asec(5) ans = 1.3694	asec(angle)	معكوس الدالة sec
>> acsc(5) ans = 0.2014	acsc(angle)	معكوس الدالة csc
>> acot(5) ans = 0.1974	acot(angle)	معكوس الدالة cot

***ملاحظة :-** جميع الدوال المثلثية اعلاه مقاسة بالتقدير الدائري Radian degree ، ولغرض تحويل التقدير الى الدرجات degrees فانه يمكن ذلك بثلاث طرق :

الاولى :- بإضافة الحرف d قبل اي دالة ،مثلا نكتب دالة sin بالشكل sind.

الثانية:- بضرب الزاوية قبل تنفيذ الدالة بالمقدار (pi/180) كما في المثال التالي :

```
>> sind(30)
ans =
   0.5000
OR
>> sin(30*(pi/180))
ans =
   0.5000
```

ولتحويل الزاوية من تقدير الدرجات الى التقدير الدائري فنضرب الزاوية قبل تنفيذ الدالة بالمقدار (180/pi) كما في المثال التالي :

```
>> sin(30)
ans =
  -0.9880
OR
>> sind(30*(180/pi))
ans =
```

-0.9880

الطريقة الثالثة :- يتم تحويل الزاوية في برنامج MATLAB باستخدام احد الامرين التاليين :-

1- **rad2deg :-** يستخدم هذا الامر لتحويل الزاوية نصف قطرية (radians) الى الدرجات (degrees) .

Example:-

There are 180^0 in π radians :

```
>>anglout=rad2deg(pi)
anglout=
180
```

2- **deg2rad :-** يستخدم هذا الامر لتحويل الزاوية من الدرجات (degrees) الى القياس نصف القطري (radians) .

Example:-

Show that there are 2 radians in full circle :

```
>>2*pi – deg2rad(360)
ans=
0
```

……..

❖ دوال التدوير والبقية : Routing & Remainder functions

1-fix :- Round toward zero

دالة التقريب الى اقرب رقم صحيح الى الصفر(اهمال الجزء الكسري)

Ex:-

```
>>a = [-1.9 , -0.2 , 3.4 , 5.6 , 7.0 , 2.4 + 3.6i]
a =
  Columns 1 through 4
  -1.9000       -0.2000       3.4000       5.6000
  Columns 5 through 6
   7.0000       2.4000 + 3.6000i
>>fix(a)
ans =
  Columns 1 through 4
  -1.0000       0            3.0000       5.0000
  Columns 5 through 6
   7.0         2.0000 + 3.0000i
```

2-round :- Round to nearest integer

دالة التقريب الى اقرب عدد صحيح .

Ex:-

```
>>a = [-1.9, -0.2, 3.4, 5.6, 7.0, 2.4+3.6i]
a =
```

```
  Columns 1 through 4
  -1.9000         -0.2000          3.4000          5.6000
  Columns 5 through 6
   7.0000          2.4000 + 3.6000i
>>round(a)
ans =
  Columns 1 through 4
  -2.0000               0          3.0000          6.0000
  Columns 5 through 6
   7.0000          2.0000 + 4.0000i
```

3- **floor** :- Round toward negative infinity

دالة التقريب باتجاه اللانهاية السالبة (∞-) , دالة التقريب نحو اليسار .

Ex:-

```
>>a = [-1.9, -0.2, 3.4, 5.6, 7.0, 2.4+3.6i]
a =
  Columns 1 through 4
  -1.9000         -0.2000          3.4000          5.6000
  Columns 5 through 6
   7.0000          2.4000 + 3.6000i
>>floor(a)
ans =
  Columns 1 through 4
  -2.0000         -1.0000          3.0000          5.0000
  Columns 5 through 6
   7.0000          2.0000 + 3.0000i
```

4- **ceil** :- Round toward positive infinity

دالة التقريب باتجاه اللانهاية الموجبة (∞+) , دالة التقريب نحو اليمين .

Ex:-

```
>>a = [ -1.9 , -0.2 , 3.4 , 5.6 , 7 , 2.4 + 3.6i ]
a =
  Columns 1 through 4
  -1.9000       -0.2000       3.4000       5.6000
  Columns 5 through 6
   7.0000       2.4000 + 3.6000i
>>ceil(a)
ans =
  Columns 1 through 4
  -1.0000       0            4.0000       6.0000
```

```
Columns 5 through 6
  7.0000        3.0000 + 4.0000i
```

5- **rem** :- Remainder after division

دالة الباقي من القسمة .

R = rem(X,Y) if Y ~= 0 , returns X – (n.*Y) where n = fix(X./Y)

Ex:-

```
>> rem(8,5)
ans =
    3
>> rem(8,8)
ans =
    0
>> rem(8,10)
ans =
    8
>> rem(-1,20)
ans =
   -1
```

6- **mod** :- Modulus after division

دالة الباقي من القسمة .

M = mod(X,Y) if Y ~= 0, returns X - (n.*Y) where n = floor(X./Y)

Ex:-

```
>>mod(13,5)
ans =
    3
>>mod([1:5],3)
ans =
    1    2    0    1    2
>>mod(magic(3),3)
ans =
    2    1    0
    0    2    1
    1    0    2
```

Note:- magic(n) returns an n-by-n matrix constructed from the integers 1 through n^2 with equal row and column sums .

```
>>magic(3)
ans =
```

8 1 6
3 5 7
4 9 2

Notes:-

- rem(X,Y) for X~=Y and Y~=0 has the same sign as X.
- mod(X,Y) for X~=Y and Y~=0 has the same sign as Y.
- rem(X,Y) and mod(X,Y) are equal if X and Y have the same sign, but differ by Y if X and Y have different signs.

Ex:-

```
>>rem(-5,2)
ans=
-1
>>mod(-5,2)
ans=
1
```

7-**sign :-** Signum function

دالة الاشارة :-

اذا كان الرقم المرسل اكـبـــرمن الصفر فان ناتج الدالة 1
اذا كان الرقم المرسل اصغـــرمن الصفر فان ناتج لدالة 1-
اذا كان الرقم المرسل مساوي الى الصفر فان ناتج الدالة 0

Ex:-

```
>> sign(0)
ans =
    0
>> sign(-9)
ans =
   -1
>> sign(0.1)
ans =
    1
```

الاعداد المركبة (العقدية) :Complex Numbers

تأخذ الاعداد المركبة صيغة واحدة وهي تواجد جزء للأعداد الحقيقية Real Numbers وجزء للأعداد التخيلية Imaginary Numbers وتكون على الصورة العامة التالية:

Z=X+Y*i

يعمل برنامج MATLAB على جراء العديد من العمليات على الاعداد المركبة مثل

1. ايجاد الجزء الحقيقي من العدد المركب (X) .
2. ايجاد الجزء الخيالي من العدد المركب (Y) .
3. ايجاد زاوية الطور ويتم الحصول عليها رياضيا من العلاقة التالية :

$$\text{angle} = \tan^{-1}\left(\frac{\text{Imaginary part}}{\text{Real part}}\right)$$

4. ايجاد القيمة المطلقة Absolute Value ويتم الحصول عليها من خلال العلاقة التالية

$$\text{Absolute Value} = \sqrt{x^2 + y^2}$$

ويمكن تلخيص الدوال التي تقوم بهذه العمليات الرياضية كما يلي بعد ادخال قيمة العدد المركب z :

```
>> z=2+4i
z =
   2.0000 + 4.0000i
```

Example	Operation	Function in MATLAB form
>> real(z) ans = 2	تستخدم لإيجاد الجزء الحقيقي من العدد المركب z	**real(z)**
>> imag(z) ans = 4	تستخدم لإيجاد الجزء الخيالي من العدد المركب z	**imag(z)**
>> abs(z) ans = 4.4721	تستخدم لإيجاد القيمة المطلقة للعدد المركب z	**abs(z)**
>> angle(Z) ans = 1.1071 >> angle(Z) ans = 1.1071	تستخدم لإيجاد زاوية الطور phase angle للعدد المركب z مقدرة بالراديان radian	**angle(z)**

ملاحظة : لحساب قيمة زاوية الطور بالدرجات يجب تحويل التقدير من radian الى degree بضرب قيمة الزاوية بالمقدار 180/pi .

مثال / اكتب برنامج بلغة MATLABلإيجاد ما يلي :-

1. الجزء الحقيقي .

2. الجزء الخيالي .
3. القيمة المطلقة العدد المركب .
4. زاوية الطور .

للعدد المركب التالي :- $C = 5\ \sqrt{-9}\ + 13$

Ex:-Write MATLAB program to calculate the following:-
1-Real part
2-Imaging part
3-Absolute Value
4-Angle phase for complex number
$C = 5\ \sqrt{-9} + \ 13$

Sol.

```
>> C = 5 * sqrt ( -9 ) + 13
C =
13.000 + 15.000 i
>> real ( C )
ans =
13
>> imag ( C )
ans =
15
>> angle ( C )
ans =
0.8567
>> angle ( C ) * 180 / pi
ans =
49.0856
OR
>> rad2deg ( angle ( C ) )
ans =
49.0856
>> abs ( C )
ans =
19.8494
```

اوامر الادخال والاخراج في برنامج MATLAB
Input I/P & Output O/P Commands

اوامر الادخال input :

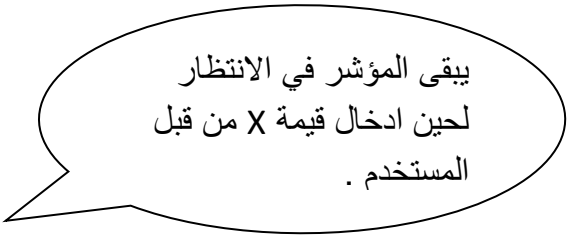
يطبع الامر input رسالة نصية للمستخدم على الشاشة كطلب إدخال بيانات عددية او حرفية وتعيينها الى متغير يعرفه المستخدم. ويستخدم الامر input على احدى الصورتين التاليتين :

اولا : ادخال بيانات عددية

X=input('displayed strings')

Displayed strings: هي مجموعة من الكلمات تمثل رسالة نصية يتم عرضها للمستخدم لتعبر عن القيمة التي سيقوم المستخدم بإدخالها ، X هو المتغير الذي يتم ادخال قيمته.

```
>> x=input('x=');
```




```
x=
x =5
```

ملاحظة: تستخدم عبارة الادخال هذه عوضا عن الطريقة السابقة للإدخال المباشر للمتغيرات في برنامج MATLAB للتحكم بالقيم المعطاة عند كل تنفيذ للبرنامج، طريقة الادخال المباشر سوف تعطي قيمة ثابتة لا يمكن تغييرها ولكن باستخدام طريقة الادخال هذه يمكن اعطاء قيم اخرى.

مثال: لإدخال درجة الحرارة وعرض رسالة نصية لتدل على ان المتغير المدخل هو درجة الحرارة

```
>> T=input('Enter the temperature')
Enter the temperature
```

وبعد ظهور الجملة أعلاه قم بإدخال قيمة T الذي يمثل درجة الحرارة ولتكن 12

```
T =
   12
```

ثانيا : ادخال بيانات رمزية

X=input("displayed strings','s')

تستخدم هذه الصورة لاستقبال سلسلة حرفية يقوم المستخدم بإدخالها، حيث يستخدم الحرف 's' الذي يرمز للكلمة (string) ويفيد تحديد نوع البيانات في تحديد نطاق التخزين المستغل من الذاكرة المؤقتة للبرنامج مما يساعد على عدم اهدار الذاكرة المؤقتة للبرنامج.

مثال : نريد عرض رسالة نصية لتدل على ان درجة الحرارة مقاسة بالمقياس السيليزي

```
>> T=input('enter the measure of temperature ','s')
enter the measure of temperature
```

وبعد ظهور الجملة اعلاه قم بإدخال المقياس وليكن المقياس السيليزي Celsius

```
T =
Celsius
```

اوامر الاخراج : disp/display/fprintf

تستخدم اوامر الاخراج لعرض قيم واسماء المتغيرات او التعبيرات النصية في نافذة محرر الاوامر Command window .

1. **الامر disp :** يستخدم الامر disp في عرض قيمة المتغير فقط سواء كانت عددية او نصية ،ويستخدم هذا الامر على احدى الصورتين :

```
disp(x)
disp('displayed strings')
```

حيث يستخدم الامر الاول لعرض قيمة المتغيرX بينما يستخدم الامر الثاني لعرض تعبير نصي معين يتم ادخاله بين علامتي اقتباس single quotation marks .

مثال : لعرض قيمة عددية

```
>> x=100;
>> disp(x)
  100
```

OR او يكتب اسم المتغير مباشرة

```
>> x
x =
  100
```

OR لعرض تعبير نصي

```
>> disp(' the value of x is ')
the value of x is
```

ويمكن وضع القيمة العددية والتعبير النصي معا بالشكل التالي :

```
disp(['messege',num2str(variable)])
```

حيث تستخدم الدالة num2str والتي تعني numerical to string اي تحويل القيمة العددية الى سلاسل حرفية ، وتستخدم في اعطاء القيمة العددية بعد الرسالة (التعبيرالنصي) .

```
>> disp(['the value of x is ',num2str(x)])
the value of x is 100
```

2. **الامرdisplay** : يستخدم الامرdisplay في عرض اسم المتغير ثم قيمته سواء كانت رقمية او نصية ، ويستخدم على الصورة التالية :

```
display(x)
```

حيث يستخدم هذا الامر لعرض اسم المتغير x ثم قيمته على نافذة command window ، ولفهم الفرق الواضح بين الصورتين السابقتين لأوامر الاخراج لاحظ المثالين التاليين :

```
>> x=5;
>> disp(x)                    %display only variable value
    5
>> display(x)                 %display variable name and value
x =
    5
```

مثال/مصفوفة الوحدة (التي جميع عناصرها مكونة من رقم 1) لاحظ الفرق في طريقة عرض النتائج:

```
>> disp(ones(3))
    1    1    1
    1    1    1
    1    1    1
>> display(ones(3))
ans =
    1    1    1
    1    1    1
    1    1    1
```

نلاحظ انه عندما استخدم الامر disp(ones(3)) قام البرنامج بعرض قيم عناصر المصفوفة الناتجة فقط ، اما عندما استخدم الامر display(ones(3)) قام البرنامج بعرض اسم المتغير الافتراضي ans (لان المستخدم لم يقوم بتعيين متغير معين لتخزين المصفوفة الناتجة عن ones(3) ثم يقوم البرنامج بعرض قيم عناصر المصفوفة الناتجة .

3. الامر (File Print Format) fprintf:-

يستخدم هذا الامرلتنسيق طباعة النتائج على نافذة command window ،الحرف f في بداية الكلمة fprintf يخص التنسيق format حيث يمكننا اختيار الطريقة المناسبة لتنسيق البيانات لكي تسهل قراءتها .

- في حالة طباعة تعبير نصي فان الامر يكتب بالصيغة التالية **fprinf('text')**

مثال :

```
>> fprintf('the amount of precipitation is')
the amount of precipitation is
```

- اما في حالة طباعة تعبير نصي وقيمة لمتغير فان الامر يكتب بالصيغة التالية :-

fprintf('format string', list of variable)

ونعني بكلمة format هنا تنسيق البيانات ، اما variable فهي القيمة العددية للمتغير .

لتوضيح التنسيق format تستخدم الصيغة التالية :

fprinf('text % -3.1g',variable)

text : يمثل النص المراد طباعته.

% : تمثل بداية تغيير تنسيق الرقم ، ويجب ملاحظة ان هذا الرمز هنا ليس للتعليقات كما تم توضيحه سابقا وانما يجب ان يكتب لكل متغير يراد طباعته في هذه الجملة .

- الاشارة : تمكننا من التحكم بتنسيق المخرجات ، كما موضح في الجدول التالي :

الرمز	المعنى	صيغة MATLAB	الامر وناتج التنفيذ
'-'	محاذاة نحو اليسار اي %-4g for 1 results in 1xxx اي ان العدد سيكون الى اليسار وعلى يمين العدد سيكون الفراغ الذي يرمز هنا بالرمز x	%-5.2f	>>fprintf('%-5.2f',9) 9.00 >>
'+'	يطبع اشارة العدد سواء كانت + او -	%+5.2f	>> fprintf('%+5.2f',9) +9.00>>
' '	يترك فراغ قبل طباعة العدد	% 5.2f	>> fprintf('% 5.2f',9)

			9.00>>
'0'	يطبع اصفار بدلا من الفراغات	%05.2f	>> fprintf('%05.2f',9) 09.00>>

3 : يمثل عرض الحقل ويمثل اقل عدد يمكن طباعته
.1 : يمثل عدد المراتب بعد الفارزة.
g : تمثل الصيغة الرياضية التي ستستخدم لتغيير صيغة الرقم كما في الجدول التالي :

الصيغة	الوصف	مثال / >>a=5.5;
%d Or %i	يطبع العدد كاملا فيطبعه بدلالة الدالة الاسية اذا كان عشريا ويطبعه كعدد صحيح اذا كان صحيحا	>> fprintf('%d',a) 5.500000e+000 OR 5.000000>> fprintf('%d',5) 5
%e	صيغة اسية باستخدام حالة الاحرف الصغيرة e	>> fprintf('%e',a) 5.500000e+000
%E	صيغة اسية باستخدام حالة الاحرف الكبيرة E	>> fprintf('%E',a) 5.500000E+000
%f	صيغة العدد الحقيقي (العشري)	>> fprintf('%f',a) 5.500000
%g OR %G	صيغة بين f و e اكثر اختصارا ، تظهر العدد كما هو فيطبعه كعدد صحيح او عدد عشري من دون استخدام الدالة الاسية	>> fprintf('%g',a) 5.5 >> fprintf('%G',a) 5.5
%s	يطبع سلسلة حرفية	>>n='MATLAB'; >> fprintf('%s \n',n) MATLAB

مثال :

```
>> fprintf('the amount of precipitation is %g',0.6)
the amount of precipitation is 0.6
```

تم طباعة التعبير النصي the amount of precipitation is ثم وضعنا العلامة % التي يجب ان تستخدم لتحديد صيغة العدد الذي سوف يستخدم لاحقا والمتمثلة بالرمز g ،يجب ان يكون كل ذلك بين علامتي اقتباس ' ' ، ثم وضعنا قيمة المتغير والتي تساوي 0.6 ويمكن ادخال قيمة المتغير مسبقا تحت اسم ما ووضعه بدلا من قيمة الرقم .

❖ تستخدام الصيغ التالية لتنسيق المخرجات مع الامر fprintf كما في الجدول التالي :-

الصيغة	الوظيفة	الامر بصيغة MATLAB	ناتج التنفيذ
\n	يذهب لبداية سطر جديد	fprintf('hello') fprintf('\n') fprintf('bye') or >> fprintf('hello \n bye')	hello bye
\t	يترك مسافة افقية مساوية لـ Tab	fprintf('hello') fprintf('\t') fprintf('bye') or >> fprintf('hello \t bye')	hello bye
\\	يطبع الشكل \	fprintf('hello') fprintf('\\') fprintf('bye') or >> fprintf('hello \\ bye')	hello\bye
\%	يطبع الشكل %	fprintf('hello') fprintf('\%') fprintf('bye') >> fprintf('hello \% bye')	Hello%bye

❖ **بعض الامثلة المختلفة في طريقة اظهار النتائج :-**

Program	Results
clc a=[12 55.5 43]; fprintf('the format for d') fprintf('\n') fprintf('%d \t',a)	the format for d 12 5.550000e+001 43
clc a=[12 55.5 43]; fprintf('the format for 2.2d') fprintf('\n') fprintf('%2.2d \t',a)	the format for 2.2d 12 5.55e+001 43
clc a=[12 55.5 43]; fprintf('the format for e') fprintf('\n') fprintf('%e \t',a)	the format for e 1.200000e+001 5.550000e+001 4.300000e+001
clc a=[12 55.5 43];	the format for 2.2e 1.20e+001 5.55e+001 4.30e+001

fprintf('the format for 2.2e') fprintf('\n') fprintf('%2.2e \t',a)	
clc a=[12 55.5 43]; fprintf('the format for f') fprintf('\n') fprintf('%f \t',a)	the format for f 12.000000 55.500000 43.000000
clc a=[12 55.5 43]; fprintf('the format for 2.2f') fprintf('\n') fprintf('%2.2f \t',a)	the format for 2.2f 12.00 55.50 43.00
clc a=[12 55.5 43]; fprintf('the format for g') fprintf('\n') fprintf('%g \t',a)	the format for g 12 55.5 43
clc a=[12 55.5 43]; fprintf('the format for 2.2g') fprintf('\n') fprintf('%2.2g \t',a)	the format for 2.2g 12 56 43 يقرب النتائج لأقرب عدد صحيح

❖ **جد ناتج تنفيذ البرامج التالية :-**

command	Results
fprintf('%d %f %g %2.2e\t',5.5,5.5,5.5,5.5,5.5)	5.500000e+000 5.500000 5.5 5.500000e+000
x=97.5; fprintf('it "works"%g %%of the time\n',x)	it 'works'97.5 %of the time

x1=10;
x2=2;
x3=x1-x2;
fprintf('Difference of %g and %g is %g \n',[x1 x2 x3])
	Difference of 10 and 2 is 8

matrix=[1 2 3;4 5 6;7 8 9];
fprintf('+---+---+---+\n');
fprintf('| %g | %g | %g |\n+---+---+---+\n',matrix');
	+---+---+---+
	| 1 | 2 | 3 |
	+---+---+---+
	| 4 | 5 | 6 |
	+---+---+---+
	| 7 | 8 | 9 |
	+---+---+---+

❖ طباعة المخرجات الى ملف لحفظ البيانات باستخدام الامر fprintf :-

يستخدم الامر fprintf لحفظ مخرجات البرنامج بانشاء ملف يمكن استدعاءه فيما بعد بدون تنفيذ البرنامج مرة اخرى ، الصيغة العامة للامر تكتب بالشكل التالي :-

fprintf (' filename' , ' format string ' , list of variables)

e.g.

fprintf (' myfile ' , '%g ' , x)

يرسل القيمة x الى ملف اسمه myfile .

Ex:-

```
>>x=0:0.1:1;
>>A=[x ; exp(x)];
>> fileID=fopen('exp.txt','w');
>> fprintf(fileID,'%6s%12s\n','x','exp(x)');
>> fprintf(fileID,'%6.2f%12.8f\n',A);
>> fclose(fileID);
>> type exp.txt
```

x	exp(x)
0.00	1.00000000
0.10	1.10517092
0.20	1.22140276
0.30	1.34985881
0.40	1.49182470
0.50	1.64872127
0.60	1.82211880
0.70	2.01375271
0.80	2.22554093
0.90	2.45960311
1.00	2.71828183