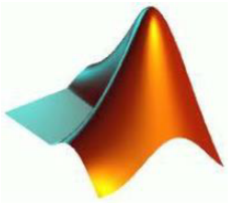
الجمهورية الجزائرية الديمقراطية الشعبية
People's Democratic Republic of Algeria
وزارة التعليم العالي والبحث العلمي
Ministry of Higher Education and Scientific Research

جامعة الشهيد حمه لخضر الوادي
كلية العلوم الدقيقة
قسم علوم المادة
شعبة الكيمياء

محاضرات في مقياس طرق عددية و برمجة
Numerical methods and programming



MATLAB

لطلبة ثانية كيمياء

من اعداد الاستاذة: منصوري اكرام

السنة الجامعية
2022 - 2021

الفهـــــرس

مقدمة **1**

1 اساسيات الماتلاب **3**
1.1 مكونات سطح مكتب:MATLAB 3
2.1 MATLAB كألة حاسبة: 4
3.1 جدول لبعض الدوال الرياضية المعرفة في برنامج MATLAB: 4
1.3.1 الدالة الاسية و اللوغارتيم: 4
2.3.1 الدوال المثلثية: 5
3.3.1 الدوال الزائدية: 5
4.1 المتغيرات: 6
5.1 دالة الادخال و الاخراج : 7
1.5.1 بعض التنسيقات: 7
2.5.1 fprintf دالة الاخراج: 7
3.5.1 input دالة الادخال: 8
6.1 الأشعة و المصفوفات: 9
1.6.1 الأشعة: 9
2.6.1 المصفوفات: 10
7.1 الاشتقاق و التكامل : 11
1.7.1 الاشتقاق: 11
2.7.1 التكامل: 12
8.1 الشروط: 13
9.1 البرمجة في الماتلاب: 13
1.9.1 كتابة البرامج في الماتلاب: 14
2.9.1 الحلقات في الماتلاب: 14
3.9.1 الدوال في الماتلاب: 15

2 التكامل العددي **16**
1.2 طريقة شبه المنحرف TRAPEZOIDAL: 16
2.2 طريقة سيمسون : SIMPSON 17

3 حل المعادلات غير الخطية **19**
1.3 طريقة التقسيم الثنائي : BISSECTION 19
2.3 طريقة نيوتن : NEWTON 20

4 الحل العددي للمعادلات التفاضلية العادية من الرتبة الاولى **22**
1.4 طريقة اولر (طريقة المماسات) : Euler 22
2.4 طريقة رانج كيتا من الرتبة الثانية : RUNGE-KUTTA 24

مقدمة

مقدمة حول MATLAB: ماتلاب بالإنجليزية: MATLAB (Matrix–Laboratory) (مختبر المصفوفات) هو برنامج رائد في التطبيقات الهندسية والرياضية من إنتاج شركة ماثووركس.

يسمح MATLAB بالتلاعب حسابيا بالمصفوفات، بالرسم البياني للتوابع الرياضية، بتنفيذ الخوارزميات المختلفة، إنشاء واجهات المستخدم الرسومية، والتواصل مع البرامج المكتوبة بلغات أخرى، بما في ذلك $C - C++$ جافا، وفورتران.

لمحة حول MATLAB:

مؤسسو برنامج الماتلاب

- **كليف مولر (Cleve Moler)**

هو أستاذ الرياضيات وعلوم الحاسب لأكثر من عشرين عاما في جامعة متشيجين وجامعة ستانفورد وجامعة نيومكسيكو ، وقد أمضى خمس سنوات عند اثنين من مصنعي الـ Hardware وهما Intel Hypercube organization و Ardent قبل أن يقوم بالانتقال إلى شركة Mathworks الشركة الأم لبرنامج الماتلاب ، كما أنه هو المؤلف لأول برنامج للماتلاب .

- **جاك ليل (jack little)**

هو المؤسس لشركة Mathworks كما أنه المساعد في وضع تخطيط برنامج الماتلاب ، وهو حاصل على بكالوريوس الهندسة الكهربية وعلوم الحاسب من جامعة MIT عام 1978 ، كما أنه حصل على شهادة M.S.E.E من جامعة ستانفورد عام 1980 .

تاريخ MATLAB:

- في منتصف السبعينات عام 1970 ، قام كليف مولر والذي كان رئيسا لقسم علوم الحاسب في جامعة نيومكسيكو وعدد من زملائه بتطوير مكتبات الفورتران والتي كانت تدعى EISPACK و LINPACK تحت منحة من المؤسسة القومية للعلوم.
- LINPACK و EISPACK هما اللبنة البرمجية الأساسية في برمجيات حسابات المصفوفات .
- بدأ كليف بكتابة برنامج للوصول إلى EISPACK و LINPACK بكل بساطة وسمي برنامجه MATLAB والتي تعني (Matrix Laboratory) أي مختبر المصفوفة .
- انتشر برنامج الماتلاب للجامعات الأخرى ووجد هذا البرنامج جمهور قوي من قبل المختصين بالرياضيات التطبيقية .
- في عام 1983 عُرض على المهندس جون ليتل برنامج الماتلاب أثناء زيارة مولر لجامعة ستانفورد ، وبعد ذلك انظم ليتل إلى مولر وستيف بانجرت وأعيدت كتابة الماتلاب بلغة السي .
- أسست في عام 1984 شركة Mathworks لمواصلة تطوير برنامج الماتلاب .

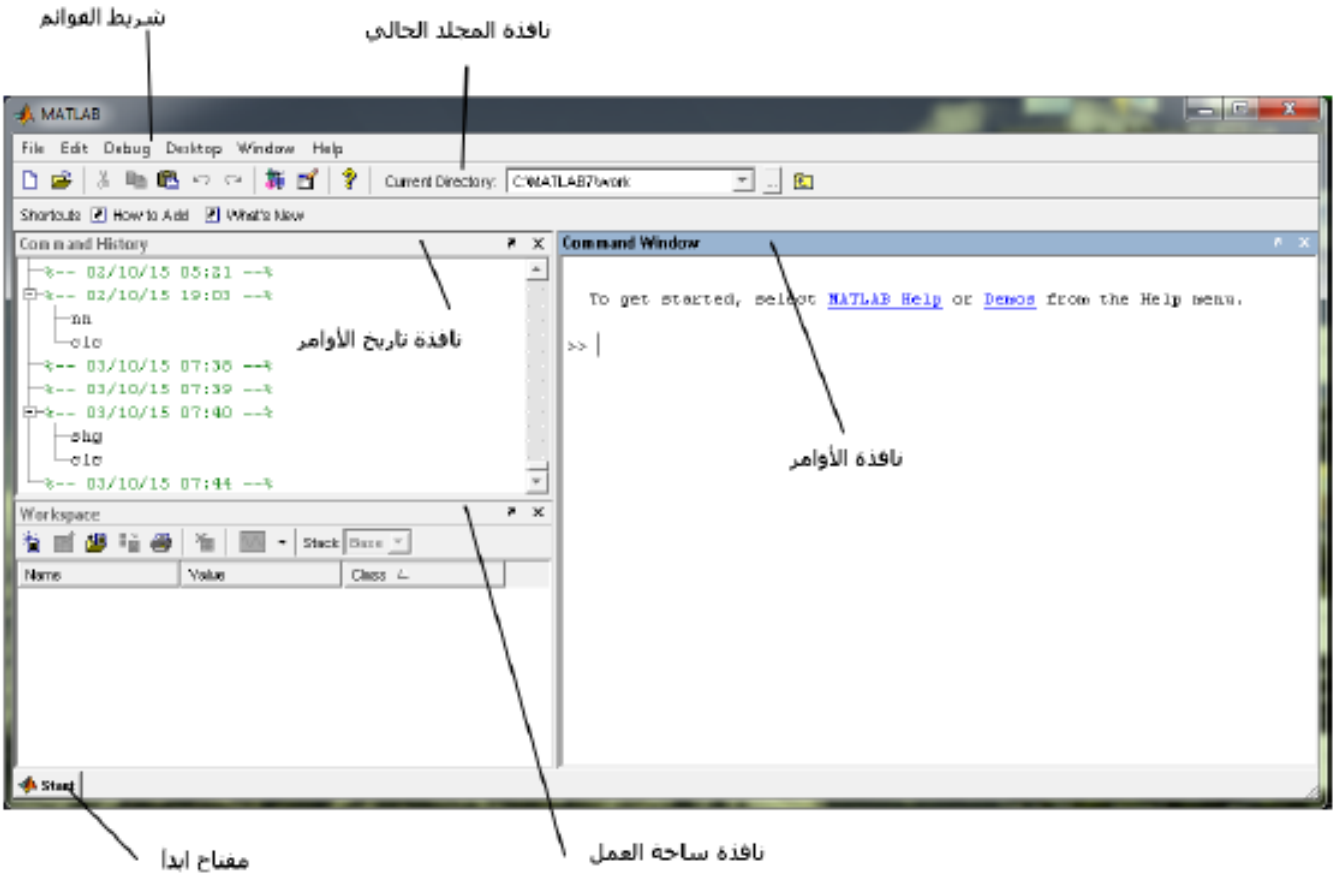
خصائص MATLAB:

- الماتلاب هي لغة ذات مستوى عالي للحسابات والبرمجة وتمتاز بوجود برامج تسهل عملية التعامل مع هذه اللغة ، وتشمل البرامج على :
  - الحسابات الرياضية
  - تطوير الخوارزميات
  - معالجة البيانات
  - النمذجة والمحاكاة وتصميم المخططات الأولية للمشاريع
  - تحليل البيانات وعرضها
  - عمل الرسومات الهندسية والعلمية
  - تطوير التطبيقات ضمن واجهات من نوع (GUI)
- ويعتمد الماتلاب على تنسيق البيانات في صورة مصفوفات ويكون البرنامج مثل أغلب لغات البرمجة .
- يشمل الماتلاب على مجموعة من البرامج والتي تدعى (toolbox) مما يتيح لك التعامل مع التطبيقات المتخصصة وهذه الأدوات عبارة عن برامج في صورة (M-files) .

الفصل الأول

اساسيات الماتلاب

1.1 مكونات سطح مكتب:MATLAB



2.1 MATLAB كألة حاسبة:

الموضوع الرئيسي	الموضوع الفرعي	تنفيذ المثال في الماتلاب	شرح المثال
استخدام الماتلاب كآلة حاسبة Operations on Variables	بعض العمليات الرياضية البسيطة Some math operations	>>6+2 >>6-2 >>6*2 >>6/2	إجراء بعض العمليات الرياضية البسيطة و هي بالترتيب الجمع ثم الطرح ثم الضرب ثم القسمة
	الدالة الأسية للأساس e Exponential Function	>>exp(1)	حساب قيمة الدالة الأسية للأساس (e) للعدد (1)
	الدالة اللوغارتمية الطبيعية Natural Logarithm	>>log(1)	حساب قيمة الدالة اللوغارتمية الطبيعية للعدد (1)
	الجذر التربيعي Square Root	>>sqrt(36)	حساب الجذر التربيعي للعدد (36)
	الدالة اللوغارتمية للأساس 10 Base 10 Logarithm	>>log10(20)	حساب قيمة الدالة اللوغارتمية للأساس (10) للعدد (20)
	القيمة المطلقة Absolute Value	>>abs(-100)	حساب القيمة المطلقة للعدد (-100)
	الدالة (sin) بالراديان Sine Function in Radians	>>sin(pi/2)	حساب قيمة الدالة (sin) للعدد (π/2) بالراديان
	الدالة (sin) بالدرجات Sine Function in Degrees	>>sind(90)	حساب قيمة الدالة (sin) للعدد (90) بالدرجات
	الدالة (cos) بالراديان Cosine Function (Radians)	>>cos(pi)	حساب قيمة الدالة (cos) للعدد (π) بالراديان
	دالة معكوس (tan) Tangent Inverse Function	>>atan(pi)	حساب قيمة دالة معكوس (tan) للعدد (100)
	الدالة (sec) Secant Function	>>sec(pi)	حساب قيمة الدالة (sec) للعدد (π)
	الدالة (sinh) Hyperbolic Sine Function	>>sinh(100)	حساب قيمة الدالة (sinh) للعدد (100)
ملاحظات Notes	حساب دالة داخل دالة أخرى Command Nesting	>>sin(abs(sqrt(90)))	يمكن في الماتلاب حساب الدالة لدالة أخرى وهكذا مع ضرورة مراعاة عدد الأقواس
	المعلومات عن دالة معينة Information about a Matlab Function	>>help sqrt >>help sin	للحصول على معلومات مفصلة عن دالة الماتلاب (sqrt) ومن ثم للدالة (sin) ، وتشمل معلومات التعريف وكيفية الاستخدام و غير ذلك
	معلومات مطولة عن دالة معينة Information about a Matlab Function	>>doc sind	للحصول على معلومات مطولة عن دالة الماتلاب (sind) مثل التعريف وكيفية الاستخدام ونحوه

47

3.1 جدول لبعض الدوال الرياضية المعرفة في برنامج MATLAB:

1.3.1 الدالة الاسية و اللوغارتيم:

Exponential and Logarithmic Functions

Matlab Function	Definition	Mathematical Form
exp(x)	Exponential	e^x
log(x)	Natural logarithm	$\ln(x)$
log10(x)	Common (base 10) logarithm	$\log(x)$
sqrt(x)	Square root	$\sqrt{x}$

2.3.1 الدوال المثلثية:

Trigonometric Functions		
Matlab Function	Definition	Mathematical Form
`cos(x)`	Cosine	$\cos(x)$
`sin(x)`	Sine	$\sin(x)$
`tan(x)`	Tangent	$\tan(x)$
`acos(x)`	Inverse cosine	$\cos^{-1}(x)$
`asin(x)`	Inverse sine	$\sin^{-1}(x)$
`atan(x)`	Inverse tangent	$\tan^{-1}(x)$
`cot(x)`	Cotangent	$\cot(x)$
`csc(x)`	Cosecant	$\csc(x)$
`sec(x)`	Secant	$\sec(x)$
`acot(x)`	Inverse cotangent	$\cot^{-1}(x)$
`acsc(x)`	Inverse cosecant	$\csc^{-1}(x)$
`asec(x)`	Inverse secant	$\sec^{-1}(x)$
`atan2(y,x)`	Four-quadrant inverse tangent.	

3.3.1 الدوال الزائدية:

Hyperbolic Functions		
Matlab Function	Definition	Mathematical Form
`cosh(x)`	Hyperbolic cosine	$\cosh(x)$
`sinh(x)`	Hyperbolic sine	$\sinh(x)$
`tanh(x)`	Hyperbolic tangent	$\tanh(x)$
`acosh(x)`	Inverse hyperbolic cosine	$\cosh^{-1}(x)$
`asinh(x)`	Inverse hyperbolic sine	$\sinh^{-1}(x)$
`atanh(x)`	Inverse hyperbolic tangent	$\tanh^{-1}(x)$
`coth(x)`	Hyperbolic cotangent	$\coth(x)$
`csch(x)`	Hyperbolic cosecant	$\operatorname{csch}(x)$
`sech(x)`	Hyperbolic secant	$\operatorname{sech}(x)$
`acoth(x)`	Inverse hyperbolic cotangent	$\coth^{-1}(x)$
`acsch(x)`	Inverse hyperbolic cosecant	$\operatorname{csch}^{-1}(x)$
`asech(x)`	Inverse hyperbolic secant	$\operatorname{sech}^{-1}(x)$

4.1 المتغيرات:

الموضوع الرئيسي	الموضوع الفرعي	تنفيذ المثال في الماتلاب	شرح المثال
إنشاء متغير Defining a Variable	طريقة التعريف	>>a=2 >>b=10	لإنشاء المتغيرين (a) ويساوي (2) و المتغير (b) ويساوي (10)
	ملاحظة هامة	>>A=3	لاحظ أن المتغير (a) لا يساوي المتغير (A) لأن الماتلاب Case Sensitive
العمليات على المتغيرات Operations on Variables	إنشاء ثلاث متغيرات Defining three variables	>>A=6; >>B=2; >>C=3;	لإنشاء ثلاث متغيرات مع عدم إظهارهم على الشاشة باستخدام الأمر (;)
	أمثلة لعمليات رياضية Examples of Simple Math Operations	>>A+B+C	المثال الأول للعمليات على المتغيرات
		>>(A*B)/C	المثال الثاني للعمليات على المتغيرات
		>>A^2+B^2	المثال الثالث للعمليات على المتغيرات
		>>(sin(A)+cos(B))/2	المثال الرابع للعمليات على المتغيرات
الأعداد المركبة Complex Numbers	العدد التخيلي i The complex number i	>>i	العدد التخيلي (i) معرف مسبقا في الماتلاب بشرط أن لا يتم اعادة تعريف المتغير (i) أثناء كتابة أي برنامج
	إنشاء متغيرين مركبين Defining two complex numbers	>>a=a+i >>b=a+2i	إنشاء متغيرين مركبين (a) و (b)
	بعض العمليات الرياضية على المتغيرين المركبين Some math operations	>>a+b >>a*b >>a/b >>a-b	إجراء بعض العمليات الرياضية البسيطة على المتغيرين المركبين وهي بالترتيب الجمع ثم الضرب ثم القسمة ثم الطرح
	إنشاء متغير مركب Defining a complex number	>>z=3+4i	إنشاء متغير مركب (z)
	الجزء الحقيقي لمتغير مركب Real part of a complex number	>>real(z)	الحصول على الجزء الحقيقي فقط للمتغير المركب (z)
	الجزء التخيلي لمتغير مركب Imaginary part of a complex number	>>imag(z)	الحصول على الجزء التخيلي فقط للمتغير المركب (z)
	زاوية الطور لمتغير مركب Phase angle of a complex number in radians	>>angle(z)	الحصول على زاوية الطور للمتغير المركب (z) بالراديان
	القيمة المطلقة لمتغير مركب Absolute value of a complex number	>>abs(z)	الحصول على القيمة المطلقة للمتغير المركب (z)
أوامر إضافية لبرنامج الماتلاب Extra Matlab Commands	عرض المتغيرات المستخدمة Display Variables	>>who	عرض كافة المتغيرات المستخدمة في نافذة الأوامر
	عرض مفصل للمتغيرات المستخدمة Display Variables with Information	>>whos	عرض مفصل لكافة المتغيرات المستخدمة في نافذة الأوامر مع عرض الحجم ومقدار الذاكرة المحجوزة بالبايت ونوع المتغير

52

5.1 دالة الادخال و الاخراج :

1.5.1 بعض التنسيقات:

Some common format options:
•%s - print a string
•%c - print a single character
•%d - print a whole number
•%f - print a floating point number
•\n - print a new line (go to the next line to continue printing)
•\t - print a tab
•\\ - print a slash
•%% - print a percent sign

2.5.1 دالة الاخراج: fprintf

The fprintf function is used for printing information to the screen. The fprintf function prints an array of characters to the screen:

```
fprintf('Hello \n');
```

شكل 1.1: Writing in the Matlab

```
>> t2

Hello

>>
```

شكل 2.1: The result

3.5.1 دالة الادخال input:

```
age=input('donne age=')
name=input('donne name=', 's')
fprintf('my name is %s and my age is %d\n',name,age);
```

شكل 3.1: Writing in the Matlab

```
>> t2
donne age=
16

age =

    16

donne name=
akram

name =

    'akram'

my nane is akram and my age is 16

>>
```

شكل 4.1: The result

6.1 الأشعة و المصفوفات:

1.6.1 الأشعة:

الموضوع الرئيسي	الموضوع الفرعي	تنفيذ المثال في الماتلاب	شرح المثال
إنشاء متجه Vector Definition	المتجه الصفي Row Vector	`>>A=[1 2 3 4 5]`	إنشاء متجه صفي من خمسة عناصر بطريقة الفراغات بين العناصر
	طريقة أخرى لإنشاء المتجه الصفي Second Method	`>>A=[1:5]`	إنشاء نفس المتجه السابق باستخدام الأمر (:) والذي يعني كافة العناصر من (1) إلى (5) بزيادة واحد عن السابق
	إنشاء متجه صفي Row Vector	`>>D=[2 4 6 8 10]`	إنشاء متجه صفي من خمسة عناصر
	الطريقة أخرى لإنشاء المتجه الصفي Second Method	`>>D=[2:2:10]`	إنشاء نفس المتجه السابق باستخدام الأمر (:) والذي يعني كافة العناصر من (2) إلى (10) بزيادة (2) عن السابق
	المتجه العمودي Column Vector	`>>C=[1;2;3;4;5]`	إنشاء متجه عمودي من خمسة عناصر باستخدام الأمر (;) والذي يعني نهاية الصف في المتغير (C)
العمليات على المتجهات Operations on Vectors	إضافة عنصر جديد للمتجه Element Adding	`>>A=[1;3;5;7;8]` `>>A(6)=100`	إنشاء متجه عمودي (A) ومن ثم إضافة عنصر سادس بقيمة (100)
	إضافة عنصر جديد للمتجه Element Adding	`>>A(8)=150`	إضافة عنصر ثامن للمتجه (A) بقيمة (150) ويلاحظ بعد التنفيذ أن الماتلاب أضاف آليا القيمة صفر للعنصر السابع
	حذف عنصر من متجه Element Deletion	`>>B=[1;3;5;7;9]` `>>B(5)=[]`	إنشاء متجه عمودي (B) ومن ثم حذف العنصر الخامس فقط
	استبدال قيمة عنصر في متجه Element Replacing	`>>B=[1;3;5;7;9]` `>>B(3)=20`	إنشاء متجه عمودي (B) ومن ثم تغيير قيمة العنصر الثالث إلى القيمة (20)
	قيمة العنصر الأكبر لمتجه Maximum Value	`>>A=[7;8;3;4;5]` `>>max(A)`	إنشاء متجه عمودي (A) ومن ثم إيجاد قيمة العنصر الأكبر فيه
	قيمة العنصر الأصغر لمتجه Minimum Value	`>>min(A)`	إيجاد قيمة العنصر الأصغر للمتجه (A)
	طول المتجه Vector Length	`>>A=[1 7 9 8 6]` `>>length(A)`	إنشاء متجه صفي (A) ومن ثم إيجاد طوله ويساوي عدد العناصر فيه

	مقياس المتجه Vector Norm	`>>x=[1 2 3]`
		`>>norm(x,1)`
		`>>norm(x)`
		`>>norm(x,2)`
		`>>norm(x,inf)`
			إنشاء متجه صفي (x) ومن ثم إيجاد مقاييسه الأول والثاني (بطريقتين) وأخيرا المقياس اللانهائي

	مثال آخر لمقياس المتجه Vector Norm	`>>a=[-10 -5 3]`
		`>>norm(x,1)`
		`>>norm(x,inf)`
			إنشاء متجه صفي (a) ومن ثم إيجاد مقاييسه الأول واللانهائي

ضرب المتجهات Vector Product	الضرب المتجه Dot Product	`>>A=[1 2 3]`
		`>>B=[4 5 6]`
		`>>dot(A,B)`
			إنشاء متجهين صفيين (A) و (B) ومن ثم إيجاد قيمة الضرب (dot) بينهما

	الضرب المتجه Cross Product	`>>cross(A,B)`	إيجاد قيمة الضرب من نوع (cross) للمتجهين (A) و (B)

2.6.1 المصفوفات:

العمليات على مصفوفة واحدة Operations on a Single Matrix	تعريف مصفوفة كمثال	>>B=[1 -5 3;6 10 -9; 11 8 4]	لإنشاء المصفوفة (B) بحجم 3 صف في 3 أعمدة
	قلب الأعمدة كصفوف Matrix Transpose	>>B'	قلب الأعمدة كصفوف والصفوف كأعمدة للمصفوفة (B)
	جمع أعمدة المصفوفة Matrix Sum	>>sum(B)	جمع أعمدة المصفوفة (B)
	استخلاص عناصر قطر المصفوفة Matrix Diagonal Elements	>>diag(B)	استخلاص عناصر قطر المصفوفة (B)
	جمع عناصر قطر المصفوفة Sum Matrix Diagonal Elements	>>sum(diag(B))	جمع عناصر قطر المصفوفة (B)
	إيجاد معكوس المصفوفة المربعة Square Matrix Inversion	>>inv(B)	إيجاد المعكوس للمصفوفة المربعة (B)
	إيجاد درجة المصفوفة Matrix Rank	>>rank(B)	إيجاد درجة المصفوفة (B)
	إيجاد محدد المصفوفة المربعة Square Matrix Determinate	>>det(B)	إيجاد المحدد للمصفوفة المربعة (B)
تابع العمليات على مصفوفة واحدة Operations on a Single Matrix	إيجاد القيم الذاتية لمصفوفة مربعة Eigenvalues of Square Matrix	>>eig(B)	إيجاد القيم الذاتية للمصفوفة المربعة (B)
	إيجاد القيم والمتجهات الذاتية لمصفوفة مربعة Eigenvalues and Eigenvectors	>>[V,D] = eig(B)	إيجاد القيم الذاتية (V) والمتجهات الذاتية (D) للمصفوفة المربعة (B)
	اختيار صف محدد من مصفوفة Matrix Row Selection	>>B(2,:)	اختيار الصف الثاني فقط من المصفوفة (B)
	اختيار عمود محدد من مصفوفة Matrix Column Selection	>>B(:,2)	اختيار العمود الثاني فقط من المصفوفة (B)
	اختيار الصف الأخير من مصفوفة Matrix Last Row Selection	>>B(end,:)	اختيار الصف الأخير فقط من المصفوفة (B)
	اختيار العمود الأخير من مصفوفة Matrix Last Column Selection	>>B(:,end)	اختيار العمود الأخير فقط من المصفوفة (B)
	اختيار مصفوفة جزئية من مصفوفة Submatrix Selection	>>B(1:2,2)	اختيار العناصر المتقاطعة ما بين الصفين الأول والثاني مع عناصر العمود الثاني من المصفوفة (B)
	اختيار مصفوفة جزئية من مصفوفة Submatrix Selection	>>B(1:2,1:2)	اختيار العناصر المتقاطعة ما بين الصفين الأول والثاني مع عناصر العمودين الأول والثاني من المصفوفة (B)
	اختيار مصفوفة جزئية من مصفوفة Submatrix Selection	>>B(3,2:3)	اختيار العناصر المتقاطعة في الصف الثالث مع عناصر العمودين الثاني والثالث من المصفوفة (B)
العمليات على أكثر من مصفوفة واحدة Operations on Matrices	تعريف مجموعة مصفوفات	>>A=[3 4 6;5 1 8]; >>B=[7 9 4;8 1 1]; >>C=[1;3;4];	إنشاء ثلاث مصفوفات لغرض تطبيق الأمثلة التالية
	جمع مصفوفتين Matrix Addition	>>A+B	عملية جمع المصفوفتين (A) و (B)
	طرح مصفوفتين Matrix Subtraction	>>A-B	عملية طرح المصفوفتين (A) و (B)
	ضرب مصفوفتين Matrix Multiplication	>>A*C	عملية ضرب المصفوفتين (A) و (C)
	قسمة مصفوفة على عدد Matrix Multiplication	>>A/3	عملية قسمة المصفوفة (A) على عدد
حل منظومة معادلات خطية Solving a System of Linear Equations	إنشاء المعادلات وحلها Building and Solving the System	>>A=[1 -1 1;1 2 2;3 0 1] >>b=[5;10;1] >>x=inv(A)*b	قم بتطبيق الأوامر الموضحة وذلك لحل منظومة المعادلات الخطية التالية : $x-y+z=5$ $x+2y+2z=10$ $3x+z=1$
	التأكد من الحل	>>A*x	ناتج عملية الضرب (A*x) يجب أن يساوي المصفوفة (b)

7.1 الاشتقاق و التكامل :

1.7.1 الاشتقاق:

- **الخطوة الأولى:** تعريف المتغيرات بالتعليمة

```
syms x y z
```

- **الخطوة الثانية:** حساب الدالة المشتقة أو الاصلية

diff(f)	diff(cos(x))
diff(cos(x+y),y)	diff(cos(x),n)

```
syms x y z
diff(cos(x+y),y)
```

شكل 5.1: Writing in the Matlab

```
>> tpdiff

ans =

-sin(x + y)
```

شكل 6.1: The result

2.7.1 التكامل:

- ***الخطوة الأولى:*** تعريف المتغيرات بالتعليمة

```
syms x y z
```

- ***الخطوة الثانية:*** حساب الدالة المشتقة أو الاصلية

int(f)	int(f,0,pi)
int(g,y)	int(int(f,x,0,y),y,0,1)

```
clc
syms x y z
f=input('f=')
g=input('g=')
int(f,0,pi)
int(int(g,x,0,y),y,0,1)
```

شكل 7.1: Writing in the Matlab

```
f=
x

f =
x
g=
x+1

g =
x + 1

ans =
pi^2/2

ans =
2/3

>>
```

شكل 8.1: The result

8.1 الشروط:

أمثلة الاستخدام في الماتلاب

طريقة التمثيل في الماتلاب	القيم الرياضية
`if (x>=0) & (x<=5)`	$x \in [0,5]$
`if (x<0) & (x>5)`	$x \notin [0,5]$
`if (x>=0) & (x<5)`	$x \in [0,5)$
`while (x~=10)`	$x \neq 10$
`while (x==10)`	$x = 10$
`while (x==10)|(x==-10)`	$x = 10$ or $x = -10$
`while (x==10)xor(x==-10)`	$x = 10$ xor $x = -10$

9.1 البرمجة في الماتلاب:

جدول علاقات المقارنة (Relation Operators)

Description	وصف العلاقة	الرمز المستخدم
Equal to	يساوي	==
Not equal	لا يساوي	~=
Less than	أقل من	<
Lese than or equal	أقل أو يساوي	<=
Greater than	أكبر	>
Greater than or equal	أكبر أو يساوي	>=

جدول العلاقات المنطقية (Logical Operators)

Description	وصف العلاقة	الرمز المستخدم
And	العلاقة "و"	&
or	العلاقة "أو"	|
Not	العلاقة "لا"	~
XOR	العلاقة (xor)	xor

1.9.1 كتابة البرامج في الماتلاب:

الموضوع الرئيسي	الموضوع الفرعي	تنفيذ المثال في الماتلاب	شرح المثال
مقدمة في البرمجة Introduction to Programming	برنامج التحرير The Editor	`>>edit`	قم بفتح برنامج التحرير (Editor) وذلك لكتابة ملف (M-File)

	ملف الأوامر
	The Script File (M-File)
		`x=pi/100:pi/100:10*pi;`
		`y=sin(x)./x;`
		`plot(x,y)`
		`grid on`
			قم بكتابة الأوامر الموضحة في شاشة برنامج التحرير (Editor) وبعد الانتهاء قم بحفظ الملف باسم (script.m)

	تشغيل ملف الأوامر Executing the Script File	`>>script`	قم بتنفيذ البرنامج (script.m) وذلك بكتابة اسم الملف في نافذة الأوامر لبرنامج الماتلاب

2.9.1 الحلقات في الماتلاب:

الموضوع الرئيسي	الموضوع الفرعي	تنفيذ المثال في الماتلاب	شرح المثال

أوامر التحكم في المسارات
Control Flow
	الأمر (For)
	For Loop
		`for n=1:10`
		`x(n)=sin(pi*n/10);`
		`end`
		`x`
			لتعلم كيفية استخدام الأمر (for) قم قم بفتح برنامج التحرير (Editor) لإنشاء الملف (program1.m) والذي يحتوي على الأوامر الموضحة ثم قم بتنفيذ البرنامج

	الأمر (For)
	Nested For Loop
		`for n=1:10`
		`for i=1:10`
		`x(n,i)=sin(pi*n/10);`
		`end`
		`end`
		`x`
			قم بإجراء التعديلات الموضحة على نفس البرنامج السابق والمسمى (program1.m) بواسطة برنامج التحرير (Editor) ومن ثم قم بتنفيذ البرنامج

	الأمر (While)
	While Loop
		`a=1`
		`while a<100`
		`a=a*2`
		`end`
			لتعلم كيفية استخدام الأمر (While) قم بفتح برنامج التحرير (Editor) لإنشاء الملف (program2.m) والذي يحتوي على الأوامر الموضحة ثم قم بتنفيذ البرنامج

	الأمر (if)
	If Condition
		`a=2`
		`b=3`
		`if a<b`
		`j=-1`
		`elseif a>b`
		`j=2`
		`else`
		`j=3`
		`end`
			لتعلم كيفية استخدام الأمر (if) قم بفتح برنامج التحرير (Editor) لإنشاء الملف (program3.m) والذي يحتوي على الأوامر الموضحة ثم قم بتنفيذ البرنامج

الموضوع الرئيسي	الموضوع الفرعي	تنفيذ المثال في الماتلاب	شرح المثال

أوامر التحكم في المسارات
Control Flow
	الأمر (switch)
	Switch Case
		`x=ceil(10*rand);`
		`switch x`
		`case{1,2}`
		`disp('probability=20%')`
		`case{3,4,5}`
		`disp('probability=30%')`
		`otherwise`
		`disp('probability=50%')`
		`end`
			لتعلم كيفية استخدام الأمر (switch) قم بفتح برنامج التحرير (Editor) لإنشاء الملف (program4.m) والذي يحتوي على الأوامر الموضحة ثم قم بتنفيذ البرنامج

	تشغيل ملف الأوامر
	Executing the Script File
		`>>for i=1:10`
		`program4`
		`end`
			قم بتنفيذ البرنامج (program4) لعدد عشر مرات متتالية وذلك بكتابة هذه الأوامر في نافذة الأوامر لبرنامج الماتلاب

3.9.1 الدوال في الماتلاب:

قم بفتح برنامج التحرير (Editor) لإنشاء ملف الدالة (myfunc.m) والذي يحتوي على الأوامر الموضحة	`function`
	`[u,v]=myfunc(a,b)`
	`u=a+b;`
	`v=a-b;`
		كتابة دالة
		Writing a Function
			الدالة
			The Function

لاستدعاء الدالة قم بتنفيذ الأمر الموضح وذلك بكتابة اسم ملف الدالة في نافذة الأوامر لبرنامج الماتلاب	`>>[u,v]=myfunc(3,7)`	استدعاء الدالة Calling the Function	

قم بفتح برنامج التحرير (Editor) لإنشاء الملف (stat.m) والذي يحتوي على الأوامر الموضحة	`function y=stat(a,b)`
	`if a>b`
	`disp('a is greater than b')`
	`elseif a<b`
	`disp('a is less than b')`
	`else`
	`disp('a is equal b')`
	`end`
		كتابة دالة أخرى
		Writing another Function	

لاستدعاء الدالة قم بتنفيذ الأوامر الموضحة وذلك بكتابة اسم ملف الدالة في نافذة الأوامر لبرنامج الماتلاب	`>>stat(5,3)`
	`>>stat(6,6)`
	`>>stat(2,8)`
		استدعاء الدالة
		Calling the Function	

الفصل الثاني
التكامل العددي

1.2 طريقة شبه المنحرف TRAPEZOIDAL:

البرمجة:

```
clc
clear all
close all
syms x
%methode of trapezoidal
f=input (' Give the function f(x)=') ;
f=inline(f);
a=input (' Enter the value of a=') ;
b=input (' Enter the value of b=') ;
n=input (' Enter the value of n=') ;
h=(b-a)/n;
s=0.5*(f(a)+f(b));
for i=1:n-1
    s=s+f(a+i*h);
end
I=h*s;
disp(I)
```

مثال: اكتب برنامج يقوم بايجاد القيمة التقريبية لتكامل التالي بطريقة شبه المنحرف

$$\int_0^{pi/2} xsin(x)dx$$

حيث $n = 10$

الحل : التكامل الحقيقي يساوي 1.

البرمجة:

```
clc
clear all
close all
%methode of trapezoidal
f=@(x) x*sin(x);
a=0 ;
b=pi/2 ;
n=10 ;
h=(b-a)/n;
s=0.5*(f(a)+f(b));
for i=1:n-1
    s=s+f(a+i*h);
end
I=h*s;
disp(I)
```

بعد تنفيذ البرنامج تظهر القيمة التقريبة لتكامل تساوي 1.0021.

2.2 طريقة سيمسون : SIMPSON

البرمجة:

```
clc
clear all
close all
syms x
%methode of simpson
f=input (' Give the function f(x)=') ;
f=inline(f);
a=input('Enter the value of a=');
b=input('Enter the value of b=');
n=input('Enter the value of n=');
h=(b-a)/n;
s=f(a)+f(b);
for i=1:2:n-1
    s=s+4*f(a+i*h);
end
for i=2:2:n-2
    s=s+2*f(a+i*h);
end
I=h/3*s;
disp(I)
```

مثال: اكتب برنامج يقوم بايجاد القيمة التقريبية لتكامل التالي بطريقة سيمسون

$$\int_0^{pi/2} xsin(x)dx$$

حيث $n = 10$
الحل : التكامل الحقيقي يساوي 1.

البرمجة:

```
clc
clear all
close all
%methode of simpson
f=@(x) x*sin(x);
a=0;
b=pi/2;
n=10;
h=(b-a)/n;
s=f(a)+f(b);
for i=1:2:n-1
    s=s+4*f(a+i*h);
end
for i=2:2:n-2
    s=s+2*f(a+i*h);
end
I=h/3*s;
disp(I)
```

بعد تنفيذ البرنامج تظهر القيمة التقريبة لتكامل تساوي 1.0000

الفصل الثالث
حل المعادلات غير الخطية

1.3 طريقة التقسيم الثنائي : BISSECTION

البرمجة:

```
clear all
close all
syms x
%methode of Bissection
y=input (' Give the function y=') ;
y=inline(y);
x1=input(' Enter the value of x1=');
x2=input(' Enter the value of x2=');
if y(x1)*y(x2)>0
    fprintf (' No roots exist with in the given interval \n')
    return
end
if y(x1)==0
    fprintf ('x1 is one of the roots \n')
    return
elseif  y(x2)==0
    fprintf ('x2 is one of the roots \n')
    return
end
for i=1:100
    xh=(x1+x2)/2;
    if y(x1)*y(xh)<0
        x2=xh;
    else
        x1=xh;
    end
    if abs(y(x1))<1.0E-6
break
end
    end
fprintf ('The root= %f \n The number of bisctions= %d\n',x1,i)
```

مثال: اكتب برنامج يقوم بايجاد تقريب لحل المعادلة $x^3 + 2x - 7 = 0$ على المجال $[2 : 1]$ بسبع تكرارات بطريقة التقسيم الثنائي.

البرمجة:

```
clc
clear all
close all
%methode of Bissection
y=@(x) x^3+2*x-7;
x1=1;
x2=2;
if y(x1)*y(x2)>0
    fprintf (' No roots exist with in the given interval \n')
    return
end
if y(x1)==0
    fprintf ('x1 is one of the roots \n')
    return
elseif  y(x2)==0
    fprintf ('x2 is one of the roots \n')
    return
end
for i=1:7
    xh=(x1+x2)/2;
    if y(x1)*y(xh)<0
        x2=xh;
    else
        x1=xh;
    end
    if abs(y(x1))<1.0E-6
break
end
    end
fprintf ('The root= %f \n The number of bisctions= %d\n',x1,i)
```

بعد تنفيذ البرنامج تظهر قيمة الحل التقريبي ب 7 تكرارات للمعادلة السابقة 1.562500.

2.3 طريقة نيوتن : NEWTON

البرمجة:

```
clc
clear all
close all
%methode of Newton
syms x
f=input (' Give the function f(x)=') ;
df=diff(f)
f=inline(f);
df=inline(df);
i=0;
Imax=input('Enter sup ofvalue of i Imax=');
eps=input (' Enter the value of eps=');
x0=input(' Enter the value of x0=');
err=inf;
while (err>eps) &&(i<Imax)
    if df(x0)==0
        break
    else
        x1=x0-f(x0)/df(x0);
        i=i+1;
        err=abs(x1-x0);
        x0=x1;
    fprintf ('\n x%d=%f;err=%f',i,x1,err);
    end
end
fprintf ('\n solution \n x%d=%f',i,x1);
fprintf('\n sure\n f(%f)=%f \n',x1, f(x1));
```

مثال: اكتب برنامج يقوم بايجاد القيمة التقريبية لحل المعادلة السابقة ب 8 تكرارات و بأخذ $x_0 = 1.75$ بطريقة نيوتن مع اخذ النتائج باربع ارقام بعد الفاصلة.

البرمجة:

```
clc
clear all
close all
%methode of Newton
syms x
f=@(x) x^3+2*x-7  ;
dif(x)=diff(f,x);
df=@(x) dif(x);
eps=0.0001;
x0=1.75;
Imax=8;
i=0;
err=inf;
while (err>eps) &&( i<Imax)
    if df(x0)==0
        break
    else
        x1=x0-f(x0)/df(x0);
        err=abs(x1-x0);
        x0=x1;
    i=i+1;
fprintf ('\n x%d=%f;err=%f',i,x1,err);
    end
end
fprintf ('\n solution \n x%d=%f',i,x1);
fprintf('\n sure\n f(%f)=%f \n',x1, f(x1));
```

بعد تنفيذ البرنامج يظهر ما يلي

```
x1=1.583799;err=0.166201
x2=1.569056;err=0.014743
x3=1.568946;err=0.000110
x4=1.568946;err=0.000000
solution
x4=1.568946
sure
f(1.568946)=0.000000

>>
```

الفصل الرابع

الحل العددي للمعادلات التفاضلية العادية من الرتبة الاولى

1.4 طريقة اولر (طريقة المماسات) : Euler

البرمجة:

```
clc
clear all
close all
%methode of Euler
syms x y
f=input (' Give the function f(x)=') ;
f=inline(f);
dy=input (' Give the function dy=') ;
dy=inline(dy);
x0=input(' Enter the value of x0=');
xn=input(' Enter the value of xn=');
y=input(' Enter the value of y=');
h=input(' Enter the value of h=');
err=0;
fprintf ('x \t \t y ( Euleur) \t y (analytical)  \t (error)\n')
fprintf (' %f \t %f \t %f \t %f \n', x0,y,f(x0),err);
for x=x0:h:xn-h
    y=y+dy(x,y)*h;
    x=x+h;
    err=abs(y-f(x))
    fprintf ('%f \t %f \t %f \t %f \n', x,y,f(x),err);
end
```

مثال: لنعتبر مسألة كوشي

$$\begin{cases} y' = x + y; \\ y(0) = 0, \end{cases}$$

مع $y : [0,1] \to \mathbb{R}$ و الحل الحقيقي هو $y = e^x - (x+1)$ اكتب برنامج يقوم بايجاد تقريب للحل y على المجال $[0,1]$ بخطوة $h = 0.1$ بطريقة اولر.

البرمجة:

```
clc
clear all
close all
%methode of Euler
f=@(x) exp(x)-(x+1) ;
dy=@(x,y) x+y;
x0=0;
xn=1;
y=0;
h=0.1;
err=0;
fprintf ('x \t \t y ( Euleur) \t y (analytical) \t (error) \n')
fprintf (' %f \t %f \t %f \t %f \n', x0,y,f(x0),err);
for x=x0:h:xn-h
    y=y+dy(x,y)*h;
    x=x+h;
    err= abs(y-f(x));
    fprintf ('%f \t %f \t %f \t %f \n', x,y,f(x),err);
end
```

بعد تنفيذ البرنامج يظهر ما يلي

```
x     y ( Euleur)   y (analytical)   (error)
 0.000000   0.000000   0.000000   0.000000
0.100000   0.000000   0.005171   0.005171
0.200000   0.010000   0.021403   0.011403
0.300000   0.031000   0.049859   0.018859
0.400000   0.064100   0.091825   0.027725
0.500000   0.110510   0.148721   0.038211
0.600000   0.171561   0.222119   0.050558
0.700000   0.248717   0.313753   0.065036
0.800000   0.343589   0.425541   0.081952
0.900000   0.457948   0.559603   0.101655
1.000000   0.593742   0.718282   0.124539


>>
```

2.4 طريقة رانج كيتا من الرتبة الثانية : RUNGE–KUTTA

البرمجة:

```
clc
clear all
close all
%methode of Runge-Kutta
syms x y
f=input (' Give the function f(x,y)=') ;
f=inline(f);
h=input(' Enter the value of h=');
a=input(' Enter the value of a=');
b=input(' Enter the value of b=');
x=a:h:b;
n=length(x);
y=zeros(1,n);
y(a)=input ('Enter the value of y(a)=');
fprintf ('y(%d)=%d',a,y(a));
for i=1:n-1
    k1=h*f(x(i),y(i));
    k2=h*f(x(i)+h,y(i)+k1);
    K=0.5*(k1+k2);
    y(i+1)=y(i)+K;
end
y
```

مثال: لنعتبر المسألة التالية:

$$\begin{cases} y' = \dfrac{2xy + e^x}{x^2 + xe^x}; \\ y(1) = 0, \end{cases}$$

مع $y : [1, 7/5] \to \mathbb{R}$ اكتب برنامج يقوم بايجاد تقريب للحل y على المجال $[1, 7/5]$ بخطوة $h = 0.2$ بطريقة رانج كيتا.

البرمجة:

```
clc
clear all
close all
%methode of Runge-Kutta
f=@(x,y) (2*x*y+exp(x))/(x^2+x*exp(x)) ;
h=0.2;
a=1;
b=1.4;
x=1:h:1.4;
n=length(x);
y=zeros(1,n);
y(1)=0;
fprintf (' y ( rung-kutta)  \n');
for i=1:n-1
    k1=h*f(x(i),y(i));
    k2=h*f(x(i)+h,y(i)+k1);
    K=0.5*(k1+k2);
    y(i+1)=y(i)+K;
end
fprintf (' %f  \n',y);
```

بعد تنفيذ البرنامج يظهر ما يلي

```
y ( rung-kutta)
0.000000
0.140785
0.271428

>>
```