	Ministry of Higher Education and Scientific Research - Iraq Al-Mustaqbal University College for engineering and technology Department of Biomedical Engineering	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Computer			**Module Delivery**	
Module Type	BASIC			**Theory** **Lecture** **Lab**	
Module Code	**UOMU0102044**				
ECTS Credits	**5**				
SWL (hr/sem)	125				
Module Level	2		**Semester of Delivery**	2	
Administering Department	Department of Chemical Engineering and Petroleum Industries		**College**	Engineering	
Module Leader	Abrar Falah Naji		**e-mail**	Abrar.falah.naji@uomus.edu.iq	
Module Leader's Acad. Title	Asst.lecturer		**Module Leader's Qualification**	master	
Module Tutor					
Peer Reviewer Name			**e-mail**		
Review Committee Approval	1/3/2026		**Version Number**		

<table>
<tr><th colspan="4">Relation With Other Modules
العلاقة مع المواد الدراسية الأخرى</th></tr>
<tr><td>Prerequisite module</td><td>None</td><td>Semester</td><td></td></tr>
<tr><td>Co-requisites module</td><td>None</td><td>Semester</td><td></td></tr>
<tr><th colspan="4">Module Aims, Learning Outcomes and Indicative Contents
أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية</th></tr>
<tr><td>Module Aims
أهداف المادة الدراسية</td><td colspan="3">To introduce chemical engineering students to modern calculating tools used in the practice of engineering to:
1. Develop problem-solving skills through algorithmic thinking and problem decomposition
2. Apply programming concepts to solve real-world problems and implement solutions efficiently.</td></tr>
<tr><td>Module Learning Outcomes
مخرجات التعلم للمادة الدراسية</td><td colspan="3">By the end of the module students should be able to:
1. Use Matlab to import data and create 2D and 3D plots.
2. Perform vector and matrix operations and use them to solve mathematical equations.
3. Solve engineering problems using for and while loops, if and if else statements.
4. Apply the most basic numerical methods to solve chemical engineering problems.</td></tr>
<tr><td>Indicative Contents
المحتويات الإرشادية</td><td colspan="3">Introduction to MATLAB (2 h)
1. Starting MATLAB
o MATLAB windows
o Menus and the toolbar
o Working in the command window
2. Basic Operations
o Arithmetic operations with scalars
o Useful commands
o Script files and the Editor Debugger
Symbolic Math (2h)
1. Symbolic Objects and Expressions
o Creating symbolic objects
2. Calculus
o Representing a function
o Solving algebraic equations
o Differentiation
o Integration</td></tr>
</table>

- Limits

One-dimensional Arrays (Vectors) (2h)

1. **Array Operations**
   - The transpose operator
   - Array addressing
   - Using a colon in addressing arrays
   - Adding and deleting elements
   - Built-in functions for handling arrays

Strings and String Variables

Two-dimensional Arrays (Matrices) (2h)

1. **Matrix Operations**
   - The transpose operator
   - Array addressing
   - Using a colon in addressing arrays
   - Adding and deleting elements
   - Built-in functions for handling arrays
   - Strings and strings as variables

Solving Simultaneous Algebraic Equations (2h)

Mathematics with Arrays

1. **Array Operations**
   - Addition and subtraction
   - Array multiplication and division
   - Element-by-element operations
   - Using arrays in MATLAB built-in math functions
   - Built-in functions for analyzing arrays
   - Generation of random numbers
   - Solving algebraic equations

2. **Matrix Algebra (2h)**
   - Introduction
   - Solving Linear Equations Using Matrix Algebra
   - Exercises

Polynomials (2h)

1. **Curve Fitting and Interpolation**
   - Interpolation
   - Polynomials

- Polynomial Interpolation

Condition and Loop Structures (2h)

1. **Relational and Logical Operators**
   - if-elseif-end
   - Relational operators (<, >, >=, <=, ==, ~=)
2. **Loops**
   - For-end
   - While-end
   - Using loops for solving iterative equations

Two-dimensional Plots (2h)

1. **Basic Plotting**
   - The plot commands
   - Using the plot command
   - Line styles, markers, and colors
   - Adding grids, labels, text, or a legend
   - Customizing axes
   - The fplot command
   - Plotting multiple graphs in the same plot
   - Multiple figure windows
   - Formatting a plot
2. **Specialized 2-D Plotting Functions (2h)**
   - Plots with special graphics

Manipulating Plots and Sub-plotting

Three-dimensional Plots (2h)

1. **3-D Plot Commands**
   - plot3 command
   - mesh command
   - surf command
   - cylinder command
   - sphere command
2. **Engineering Examples**

Midterm Exam (2h)

Functions (2h)

1. **Creating and Using Functions**
   - Creating a local function file
   - Local and global variables
   - Saving a local function file
   - Examples of simple user-defined functions

	○ Anonymous Functions ○ Function Handle **Numerical Analysis and Differential Equations(2h)** 1. **Introduction and Initial Value Problems** ○ Numerical solution of differential equations ○ Euler's method ○ Error types ○ Exercises and examples **Solving Ordinary Differential Equations (ODEs)(2h)** 2. **Higher Order ODEs** ○ Runge-Kutta method ○ Integration of two or more coupled first-order ODEs ○ MATLAB built-in routines for solving ODEs ○ ode45 using local function ○ ode45 using local function as Anonymous function after main program ○ ode45 using Anonymous function with ode45 in one main program ○ Solving higher order ODEs using ODE45 ○ Solving non-linear equations using MATLAB's built-in function: (fsolve)

Learning and Teaching Strategies

استراتيجيات التعلم والتعليم

Strategies	Incorporate interactive lectures where students can engage with the material through discussions, demonstrations, and real-world examples. Use multimedia presentations to illustrate programming concepts and demonstrate their application in chemical engineering. Problem-Based Learning: Present students with real-world engineering problems that can be solved using programming. Encouraging students to work collaboratively in small groups to analyze the problem, develop algorithms, and implement solutions using computational tools. This approach helps students develop problem-solving skills and apply programming concepts in context.

Student Workload (SWL)

الحمل الدراسي للطالب

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	78	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	5
Unstructured SWL (h/sem)	47	**Unstructured SWL (h/w)**	3.1

الحمل الدراسي غير المنتظم للطالب خلال الفصل		الحمل الدراسي غير المنتظم للطالب أسبوعيا	
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	125		

Module Evaluation

تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	3	15% (5)	3-11	LO # 1-4
	Assignments	5	10% (2)	2-13	LO # 1-4
	Lab.	10	10% (1)	Continuous	
	Report	1	5% (5)	13	LO # 1-4
Summative assessment	**Midterm Exam**	2 hr	10% (10)	10-12	LO # 1-4
	Final Exam	3 hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

	Material Covered
Week 1	***Introduction to MATLAB*** • Starting MATLAB, MATLAB windows • Menus and the toolbar • Working in the command window • Arithmetic operations with scalars • Useful commands for managing variables • Script files and the Editor Debugger
Week 2	***Symbolic Math***

	• Symbolic objects, and symbolic expressions • Calculus • Representing a function • Solving algebraic equations • Differentiation • Integration • Limits Solving an ordinary differential equation
Week 3	***One-dimensional array (vector)*** • The transpose operator • Array addressing • Using a colon: in addressing arrays • Adding elements to existing variables • Deleting elements • Built-in functions for handling arrays Strings and strings as variables
Week 4	***Two-dimensional array (matrix)*** • The transpose operator • Using a colon: in addressing arrays • Adding elements to existing variables • Deleting elements • Built-in functions for handling arrays • Strings and strings as variables • Solving simultaneous algebraic
Week 5	***Mathematics with arrays*** • Addition and subtraction • Array multiplication • Array division • Element-by-element operations

	• Using arrays in MATLAB built-in math functions • Built-in functions for analyzing arrays • Generation of random numbers • Solving Algebraic Equations • Matrix Algebra • Solving Linear Equations Using Matrix Algebra
Week 6	***Polynomials*** • Curve fitting • Interpolation • Polynomials • Polynomial Interpolation
Week 7	***Condition and Loop*** Relational and logical operators • if-elseif-end • < less than • > =greater than or equal • greater than • = = equal • < =less than or equal ~ =not equal Loops • For-end • While-end Using Loops for solving iterative equations
Week 8	***Two-dimensional plot 1*** • The plot command • Using the plot Command • Line styles, Markers, and Colors • Adding Grids, Labels, Text, or a Legend

	• Customizing Axes • The fplot command • Plotting multiple graphs in the same plot • Multiple figure windows Formatting a plot
Week 9	***Two-dimensional plot 2*** Specialized 2-D plotting functions: • Plots with special graphics • Manipulating Plots and Sub-plotting
Week 10	***Three dimensional plot*** • plot3 command • mesh command • surf command • cylinder command • sphere command • Engineering examples on three dimensional plot
Week 11	**Midterm Exam**
Week 12	***Functions*** • Creating a function file • Local and global variables • Saving a function file • Examples of simple user-defined functions • Anonymous Functions • Function Handle
Week 13	***Introduction to Numerical analysis and Differential Equations of initial value problem 1*** • Numerical Solution of Differential Equations • Euler's Method • Error Types • Exercise and examples

Week 14	***differential equations of initial value problem & Higher order ode's*** • Runge-Kutta Method • Integration two or more coupled first-order ODE's • MATLAB Built-In Routines for solving ODES • ode45 using local function • ode45 using local function as Anonymous function after main program • ode45 using Anonymous function with ode45 in one main program • Solving higher order ODEs using ODE45 • Solve Non-Linear Equation using MATLABs built-in function: (fsolve) • Exercise and examples
Week 15	**Review Week**
Week 16	**Final Exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Lab 1: Starting MATLAB, MATLAB windows, Menus and the toolbar, Working in the command window, Arithmetic operations with scalars, Useful commands for managing variables, Script files and the Editor Debugger.
Week 2	Lab 2: Symbolic objects, and symbolic expressions, Calculus, Representing a function, Solving algebraic equations, Differentiation, Integration, Limits, Solving an ordinary differential equation
Week 3	Lab 3: The transpose operator, Array Addressing, Using a colon: in addressing arrays, Adding elements to existing variables, Deleting elements, Built-in functions for handling arrays, Strings and strings as variables.
Week 4	Lab 4: The transpose operator , Array addressing ,Using a colon: in addressing arrays ,Adding elements to existing variables ,Deleting elements ,Built-in functions for handling arrays ,Strings and strings as variables, Solving simultaneous algebraic, Addition and subtraction, Array multiplication, Array division, Element-by-element operations, Using arrays in MATLAB built-in math functions.
Week 5	Lab 5: Matrix Algebra, Solving Linear Equations Using Matrix Algebra
Week 6	Lab 6: Curve fitting, Interpolation, Polynomials, Polynomial Interpolation.
Week 7	Lab 7: if-elseif-end, < less than, >= greater than or equal ,greater than , = = equal ,<= less than or equal, ~= not equal, For-end , Using Loops for solving iterative equations
Week 8	Lab 8: While-end, Using Loops for solving iterative equations.

Week 9	Lab 9: Creating a function file, Local and global variables, Saving a function file, Examples of simple user-defined functions, Anonymous function, handle function.
Week 10	Lab 10: The plot command ,Using the plot Command, Line styles, Markers, and Colors, Adding Grids, Labels, Text, or a Legend, Customizing Axes, the fplot command, Plotting multiple graphs in the same plot, Multiple figure windows ,Formatting a plot, Plots with special graphics ,Histograms Sub-plotting.
Week 11	Lab 11: Engineering examples on three dimensional plot.
Week 12	Lab 12: Numerical solution of algebraic equations, Numerical solution of multi simultaneous differential equations, Euler method Example, Runge-Kutta method Example.
Week 13	Lab 13: Numerical solution of higher order ordinary differential equation, use ODE45, Examples.
Week 14	Lab14: Nonlinear equation solve using fsolve command.
Week 15	Lab 15: final laboratory Exam.

Learning and Teaching Resources

مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	1. Chapra, Steven C. Applied numerical methods with MATLAB for engineers and scientists. Mcgraw-hill, 2018. 2. Yeo, Yeong Koo. Chemical engineering computation with MATLAB®. CRC Press, 2020. (ebook)http://www.taylorandfrancis.com 3. Kattan, Peter. Matlab for beginners. Vol. 1. Petra books, 2022.Schilling.	No Electronic file only
Recommended Texts	1. Otto, Stephen Robert, and James P. Denier. An introduction to programming and numerical methods in MATLAB. Springer Science & Business Media, 2005. 2. Yang, Won Y., et al. Applied numerical methods using MATLAB. John Wiley & Sons, 2020.	No
Websites	https://www.mathworks.com/	

APPENDIX:

GRADING SCHEME

مخطط الدرجات

Group	Grade	التقدير	Marks (%)	Definition
Success Group	**A -** Excellent	امتياز	90 - 100	Outstanding Performance

(50 - 100)	**B -** Very Good	جيد جدا	80 - 89	Above average with some errors
	C - Good	جيد	70 - 79	Sound work with notable errors
	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings
	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria
Fail Group	**FX –** Fail	مقبول بقرار	(45-49)	More work is required but credit awarded
(0 – 49)	**F –** Fail	راسب	(0-44)	A considerable amount of work required

Note:

NB Decimal places above or below 0.5 will be rounded to the higher or lower full mark (for example a mark of 54.5 will be rounded to 55, whereas a mark of 54.4 will be rounded to 54. The University has a policy NOT to condone "near-pass fails" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.

ملاحظة: هذا النموذج تم وضعه وتقديمه من قبل مديرية ضمان الجودة في وزارة التعليم العالي والبحث العلمي