MODULE DESCRIPTION FORM

Module Information					
Module Title	**Data Structures**			**Module Delivery**	
Module Type	**Basic**			☑ **Theory** ☑ **Lecture** ☑ **Lab** ☑ **Tutorial** ☑ **Practical** ☐ **Seminar**	
Module Code	AIE211				
ECTS Credits	**5**				
SWL (hr/sem)	**125 H/Sem.**				
Module Level	2	**Semester of Delivery**	3		
Administering Department	Artificial Intelligence Eng.	**College**	College of Engineering		
Module Leader	Sarmad Kadhim Alkhfaji	**e-mail**	sarmad.kazem@alayen.edu.iq		
Module Leader's Acad. Title	Assistant Prof.	**Module Leader's Qualification**	Ph.D.		
Module Tutor	Firas Sabar Miftan	**e-mail**	firas@utq.edu.iq		
Peer Reviewer Name		**e-mail**	Hasnaa.Mohammad@alayen.edu.iq		
Scientific Committee Approval Date	27/9/2025	**Version Number**	1.0		

Relation with other Modules			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents	
Module Objectives	**Introduction to Data Structures – Data representation – Data structures – Arrays and matrices - Stack and Queues – Linked lists, trees, and tables – Recursion techniques – Data structures for external storage – Searching, sorting, and ordering algorithms – Analysis and evaluation of algorithm efficiency using Python**

Module Learning Outcomes	Knowledge	A1	A1.1 Define the fundamental concepts of data structures and data representation. A1.2 Recognize different types of data structures such as arrays, matrices, stack, queues, linked lists, trees, and tables. A1.3 Describe recursion techniques and their applications in problem solving. A1.4 Explain the use of data structures for external storage and efficient data management. A1.5 Choose appropriate searching, sorting, and ordering algorithms for specific computational problems. A1.6 Illustrate data manipulation and algorithm implementation using the Python programming language. A1.7 Compare the performance and efficiency of different algorithms and data structures. A1.8 Distinguish between various methods of organizing and accessing data in memory and storage.
		A4	A4.1 Label the basic concepts of data structures and understand their importance in programming. A4.2 Identify various methods of data representation and their impact on efficiency. A4.3 Apply fundamental data structures including arrays and matrices to solve practical problems. A4.4 Demonstrate the implementation of queues, linked lists, trees, and tables using Python. A4.5 Categorize recursion techniques and utilize them in solving complex computational tasks. A4.6 Differentiate data structures suitable for external storage from those used in memory. A4.7 Organize searching, sorting, and ordering algorithms and analyze their efficiency in Python.
	Skills	B1	B.1.1 Function efficiently as an individual B.1.2 Function as a member of multidisciplinary and multicultural teams.

		B3	B.3.1 Acquire and apply new knowledge. B.3.2 Practice the strategies of searching for information in life – long self-learning
	Ethics	C3	C.3.1 Utilize contemporary technologies C.3.2 Make use of codes of practice and standards, quality guidelines.

Indicative Contents	Indicative content includes the following. Unit 1: Introduction to Data Structures and Data Representation [4 hours] Unit 2: Arrays, Matrices, Stacks, and Queues [8 hours] Unit 3: Linked Lists, Trees, and Tables [8 hours] Unit 4: Recursion Techniques and Data Structures for External Storage [4 hours] Unit 5: Searching, Sorting, Ordering Algorithms and Analysis of Algorithm Efficiency Using Python [6 hours]

Learning and Teaching Strategies

Strategies	• Presentation • Interactive learning • Discussions • Brainstorming • Group Solving Problems • Problem-Based Learning • Self-learning

Student Workload (SWL) for 15 weeks

Structured SWL (h/sem)	93	**Structured SWL (h/w)**	5.2
Unstructured SWL (h/sem)	32	**Unstructured SWL (h/w)**	4.8
Total SWL (h/sem)	125		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	10% (10)	5 and 10	A1, A4
	Assignments	2	10% (10)	2, 4,6,8,10,12 and 14	A1, A4
	Projects / Lab.				A1, A4
	Report	1	10% (10)	13	A4, B1, B3
Summative assessment	**Midterm Exam**	1hr	10% (10)	7	A1
	Final Exam	3hr	50% (50)	15	A1, A4
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري		
	Material Covered	**Los**
Week 1	Introduction to Data Structures	A1.1, A4.1
Week 2	Data Representation and Abstract Data Types	A1.2, A4.1, B1.1, B3.1, B3.2, C3.1
Week 3	Arrays and Matrices	A1.2, A4.3, B1.1, B3.1, B3.2, C3.1
Week 4	Linked Lists	A1.2, A4.4, B1.1, B3.1, B3.2, C3.1
Week 5	Stacks	A1.2, A4.4, B1.1, B3.1, B3.2, C3.1
Week 6	Queues	A1.2, A4.4, B1.1, B3.1, B3.2, C3.1
Week 7	Trees and Tables	A1.2, A4.4, B1.1, B3.1, B3.2, C3.1
Week 8	Recursion Techniques	A1.3, A4.5, B1.1, B3.1, B3.2, C3.1
Week 9	Data Structures for External Storage	A1.4, A4.6, B1.1, B3.1, B3.2, C3.1
Week 10	Mid -term EXAM	A1, B1
Week 11	Searching, Sorting, and Ordering Algorithms	A1.5, A4.7, B1.1, B3.1, B3.2, C3.1
Week 12	Analysis and Evaluation of Algorithm Efficiency	A1.6, A1.7, A1.8, A4.7, B1.1, B1.2,

		B3.1, B3.2, C3.1,C3.2
Week 14	Performance Evaluation and Final Review	A1, A4, B1.1, B3.1, B3.2, C3.1
Week 15	Rubric Exam Preparatory week before the final Exam	A2, A4, B3

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر		
	Material Covered	**Los**
Week 1	Introduction to Data Structures – basic concepts and Python setup	A1.1, A4.1
Week 2	Arrays and Matrices – creation, access, and manipulation	A4.3, B1.1, B3.1, B3.2, C3.1
Week 3	Linked Lists – singly and doubly linked lists implementation	A4.3, B1.1, B3.1, B3.2, C3.1
Week 4	Stacks – stack operations and applications	A4.3, B1.1, B3.1, B3.2, C3.1
Week 5	Queues – queues, circular queues, and priority queues	A4.3, B1.1, B3.1, B3.2, C3.1
Week 6	Trees and Tables – tree construction, traversal, and table operations	A4.3, B1.1, B3.1, B3.2, C3.1
Week 7	Recursion Techniques – solving problems recursively and iteratively	A4.5, B1.1, B3.1, B3.2, C3.1
Week 8	Data Structures for External Storage – file handling and external data management	A4.6, B1.1, B3.1, B3.2, C3.1
Week 9	Mid-Term Exam	A1, B1
Week 10	Searching, Sorting, and Ordering Algorithms – implementation and comparison	A4.7, B1.1, B3.1, B3.2, C3.1
Week 11	Analysis and Evaluation of Algorithm Efficiency – measuring time and space complexity	A4.7, B1.1, B3.1, B3.2, C3.1
Week 12	Mini Project	A1, A4, B1, B3, C1
Week 13	Performance Evaluation and Final Review	A1, A4, B1, B3, C1

Week 14	Rubric Exam / Preparatory Week	A1, A4, B1

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Miller, B. & Ranum, D. (2017). *Problem Solving with Algorithms and Data Structures*, Release 3.0.	NO
Recommended Texts	Karumanchi, N. (2017). *Data Structures and Algorithms Made Easy* (5th Edition).	NO
Websites	https://eng.alayen.edu.iq//colleges/e/	

Grading Scheme مخطط الدرجات				
Group	**Grade**	التقدير	**Marks (%)**	**Definition**
Success Group (50 - 100)	**A** - Excellent	امتياز	90 - 100	Outstanding Performance
	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 (0 – 49)	**FX** – Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required

Note: Marks 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.