

	Ministry of Higher Education and Scientific Research - Iraq Al-Naji University College of Engineering AI Engineering Department	
---	--	---

MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information			
معلومات المادة الدراسية			
Module Title	Data Structures and Algorithms		Module Delivery
Module Type	Basic		☒ Theory ☐ Lecture ☒ Lab ☐ Tutorial ☐ Practical ☐ Seminar
Module Code	AIA1202		
ECTS Credits	5		
SWL (hr/sem)	125		
Module Level	1	Semester of Delivery	2
Administering Department	AI Engineering	College	Engineering
Module Leader	Gada Sabeeh Nassir	e-mail	ghada.sabeeh@alnaji-uni.edu.iq
Module Leader's Acad. Title	Lecturer	Module Leader's Qualification	MSc.
Module Tutor		e-mail	
Peer Reviewer Name		e-mail	
Scientific Committee Approval Date		Version Number	1.0

Relation with other Modules			
العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	Semester	
Co-requisites module	None	Semester	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Objectives أهداف المادة الدراسية	• Understand the fundamentals of data structures: Enable students to identify various types such as arrays, linked lists, stacks, queues, trees, and graphs. • Master basic algorithms: Learn how to design and analyze different algorithms for searching, sorting, and data manipulation. • Performance analysis: Teach students how to analyze time and space complexity using the Big O notation. • Develop programming skills: Apply theoretical concepts using programming languages (such as C++, Java, or Python) to implement data structures and algorithms. • Problem-solving: Enhance analytical and logical thinking through solving programming problems that require appropriate data structures and algorithms.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	(To be listed according to GO standards)
Indicative Contents المحتويات الإرشادية	(Key topics to be taught and the estimated time to cover them) • Course Duration: One full semester (15 weeks) • Weekly Hours: 5 hours (3 theoretical, 2 practical) Core Topics: • Algorithm concepts and analysis methods • Definition and use of arrays • Pointers and dynamic memory • Functions • Object-Oriented Programming • Linked Lists • Stacks and Queues • Graphs and Trees • Searching and Sorting

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	• Interactive Lectures: ○ Present theoretical concepts through slides and direct explanation. ○ Use real-world problems and examples to illustrate concepts. ○ Encourage students to ask questions and participate in discussions. • Problem-Based Learning: ○ Introduce coding problems that require specific data structures and

	algorithms. ○ Motivate students to research and investigate to find optimal solutions. • Inductive Teaching: ○ Present a practical problem or scenario and analyze it to extract theoretical concepts. ○ Start lessons with hands-on examples before explaining the academic theories. • Hands-On Activities: ○ Utilize programming labs to practically implement algorithms and data structures. ○ Provide a supportive environment for experimenting with code, finding bugs, and fixing them.
--	--

Student Workload (SWL)			
الحمل الدراسي للطلاب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطلاب خلال الفصل	78	Structured SWL (h/w) الحمل الدراسي المنتظم للطلاب أسبوعيا	5
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطلاب خلال الفصل	47	Unstructured SWL (h/w) الحمل الدراسي غير المنتظم للطلاب أسبوعيا	1
Total SWL (h/sem) الحمل الدراسي الكلي للطلاب خلال الفصل	125		

Module Evaluation					
تقييم المادة الدراسية					
		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	3	5% (5)		All
	Homeworks	1	5% (5)		All
	Lab.	4	10% (10)		All
	Insight Assignments	2	5% (5)		All
Summative assessment	Midterm Exam	2hr	10% (10)	7	All
	Final Exam	3hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	The concept of Algorithms: Algorithmic Thinking, Analysis and design of algorithms
Week 2	The concept of Algorithms: Block Diagrams construction
Week 3	Introduction to Array (One/Two Dimension) In C++
Week 4	Strings and Pointers
Week 5	Dynamic array
Week 6	Functions in C++
Week 7	Pointers and Functions
Week 8	Recursive function
Week 9	-----Midterm 01-----
Week 10	Introduction to OOP (Classes and Objects)
Week 11	Precondition, Postcondition, Documentation, and Big O Notion
Week 12	Link Lists
Week 13	Stack and Queue
Week 14	Graphs and Trees
Week 15	Sorting Algorithms
Week 16	

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	How to write the algorithm and convert it to code
Week 2	One Dimension array implementation
Week 3	Multiple Dimension array implementation
Week 4	Poitiers and their application
Week 5	Pointers and dynamic array
Week 6	Dynamic Array
Week 7	Introduction to Functions
Week 8	More applications for functions
Week 9	Recursive functions

Week 10	Introduction to OOP I (Classes and Objects)
Week 11	Introduction to OOP II (Classes and Objects)
Week 12	Link Lists
Week 13	Stack and Queue
Week 14	Search Techniques
Week 15	Sorting Techniques

Learning and Teaching Resources		
مصادر التعلم والتدريس		
	Text	Available in the Library?
Required Texts	Text book-1: Data Structures and Algorithm Analysis in C++ Author : Mark Allenn Wieis Edition & Year public : 2014 Peasron	
Recommended Texts	Text book-2: Data Structures and Algorithm Analysis in C++, 3rd Edition Author: Edition & Year public : 2013	
Websites		

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 (فيد 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.				