MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Data Structures**			**Module Delivery**	
Module Type	**Core**			☒**Theory** ☒**Lecture** ☒**Lab** ☐**Tutorial** ☐**Practical** ☐**Seminar**	
Module Code	**MU010602101**				
ECTS Credits	**5**				
SWL (hr/sem)	**125**				
Module Level	**UGII**		**Semester of Delivery**	3	
Administering Department	Computer Science		**College**	College of Science	
Module Leader	Dr. Bassam Basim Jamil		**e-mail**	Dr.balkindy@uomustansiriyah.edu.iq	
Module Leader's Acad. Title	Assistant. **Prof**		**Module Leader's Qualification**		
Module Tutor	Assistant.Prof Bassam Basim Jamil		**e-mail**	Dr.balkindy@uomustansiriyah.edu.iq	
Peer Reviewer Name			**e-mail**		
Scientific Committee Approval Date	/ /2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	Programming Fundamentals (I) Programming Fundamentals (II)	**Semester**	1 2
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Objectives أهداف المادة الدراسية	This course presents the : 1- strategies and techniques that are required to store data (Data Structures) and perform operations on such data. 2- To teach efficient storage mechanisms of data for an easy access. 3- To design and implementation of various basic and advanced data structures. 4- To introduce various techniques for representation of the data in the real world. 5 - To develop application using data structures. Topics covered include this course : Survey basic abstract data types, their associated algorithms, and how they are implemented. structures of stacks, queues, lists, linked list and tree ; performance tradeoffs of different implementations; and asymptotic analysis of running time and memory usage.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Upon completion of the course, the students will be able to: 1- Student will be able to choose appropriate data structure as applied to specified problem Definition. 2- Student will be able to handle operations like searching, insertion, deletion, traversing mechanism etc. on various data structures. 3- Students will be able to use linear and non-linear data structures like stacks, queues, linked list , tree . 4- Compare and contrast the costs and benefits of dynamic and static data structure implementations. 5- Students will be able to Convert the types of Data structures to programs develop their coding skill.
Indicative Contents المحتويات الإرشادية	Indicative content includes the following. **UNIT I :** [8 hrs] Introduction to Data Structures , How to choose the suitable data structure Types of data structures , Memory Representation ,Introduction to Abstract Data Type , Introduction to Time Complexity (Space Complexity, Time Complexity) **UNIT-II** : [22 hrs] Stack ADT definition and operation , applications-infix to postfix conversion, Postfix expression evaluation, recursion implementation. Queue ADT definition and operations, Circular queues Insertion and deletion operations. **UNIT-III** : [28 hrs]

	- Array list ADT definition, Array list operations. - Linked List ADT definition , Linked List operations. -Linked List Design Modification : - Circular Linked List , Circular Linked List Operations, Traversing Circular Linked List. - Doubly Linked List . Doubly Linked List Operations , Traversing Doubly Linked List . - Linked Stack, Linked Queue, Linked Circular Queue Operations. - Recursion **UNIT IIII** : [15 hrs] Trees – Definitions, tree representation, properties of trees, Binary tree, Types of Binary tree Binary tree representation, binary tree properties, binary tree traversals, binary tree implementation, applications of trees. Binary Search Tree: Building a Binary Search tree , Finding Minimum and Maximum values in Binary Search tree , Delete value in Binary Search Tree Representation of Arithmetic Expressions using Binary Tree

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	**Learning Strategies:** The best way to learn and remember Data Structures is to practice them. You can also find books, courses, or tutorials that offer exercises and examples. 1- Hands-on Practice: Programming is a practical skill that requires practice. Allocate dedicated time to work on Data structures exercises, small projects, and problem-solving activities. 2- Reading types of data structures: Study and analyze existing data structures examples and programs written by experienced developers. Write the programs enhance your understanding of Data structures. 3- Collaborative Learning: Engage in discussions and collaborations with fellow learners or join programmer communities and forums to share knowledge, ask questions, and solve problems collectively. **Teaching Strategies:** 1- Conceptual Explanation: Start by providing clear and concise explanations of Data Structures concepts. Use real-life examples or analogies to help

	students grasp abstract concepts easily. 2- .Hands-on Coding Exercises: Include practical Data Structures exercises and assignments that reinforce the concepts taught. Convert the types of Data structures to programs . Encourage students to apply the algorithm into practice, helping them develop their coding skill. 3- Interactive Sessions: Conduct live sessions where students can actively participate, ask questions, and solve problems alongside the instructor. This approach promotes engagement and practical learning. In addition to the practical project 4- Resources and References: Share relevant Data Structures resources, recommended textbooks, and online tutorials to supplement the curriculum and encourage further exploration.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ أسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	79	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	5
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	46	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	3
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**	2	10% (10)	6 and 10	LO #1, #2, #3, #4, #5
	Assignments	2	8% (8)	4 and 12	LO #1, #2, #3, #4, #5
	Projects / Lab.	1	14% (14)	Continuous	All
	Report	1	8% (8)		LO #1, #2, #3, #4, #5
Summative assessment	**Midterm Exam**	2hr	10% (10)	8	LO #1, #2, #3, #4, #5
	Final Exam	4hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	Introduction to Data Structures: - How to choose the suitable data structure - Types of data structures - Memory Representation - Introduction to Abstract Data Type
Week 2	- Introduction to Time Complexity : - Space Complexity - Time Complexity (Big –O Notation)
Week 3	Stack - The Stack Abstract Data Type - Array Stack - Stack Operations - Time Complexity of these operations
Week 4	Applications of the stack: - Check for balancing of parenthesis - Convert infix expression to postfix - Evaluate postfix expression - Use of stack in Function calls
Week 5	Queue - The Queue Abstract Data Type - Queue operations - Time Complexity of these operations
Week 6	Circular Queue and Priority Queues: - The Abstract Data Type - Operations - Time Complexity of these operations
Week 7	**Midterm Exam /** Lists : - Array list - The array List Abstract Data Type - Time Complexity of these operations - Array list operation
Week 8	Linked List - Storage Allocation - Pointers - Linked List Abstract Data Type - Traversing a Linked List
Week 9	Linked List: - Linked List Operations

	- Time Complexity of these operations
Week 10	Linked List Design Modification : - Circular Linked List - Circular Linked List Operations - Traversing Circular Linked List - Time Complexity of these operations
Week 11	Linked List Design Modification : - Doubly Linked List - Doubly Linked List Operations - Traversing Doubly Linked List - Time Complexity of these operations
Week 12	Linked Stack, Linked Queue, Linked Circular Queue - Operations - Time Complexity of this operations - Recursion
Week 13	Binary Tree: - Types of Binary Tree - Traversing Binary Tree: (post order, in order, preorder)
Week 14	Binary Search Tree: - Building a Binary Search tree - Finding Minimum and Maximum values in Binary Search tree
Week 15	Binary Search Tree: - Delete value in Binary Search Tree - Representation of Arithmetic Expressions using Binary Tree
Week 16	**Preparatory week before the final Exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Review of VC# language
Week 2	Programming of stack
Week 3	Examples of stack
Week 4	Programming of Queue with Quiz
Week 5	Programming of Circular Queue
Week 6	Programming of Circular Queue
Week 7	**Midterm Exam /** Programming of array list
Week 8	Programming of array list
Week 9	Programming of Linked List

Week 10	Programming of Linked List With quiz
Week 11	Examples on Linked List
Week 12	Programming of Recursion Factorial and power, Sum and Fibonacci
Week 13	Programming of Binary Search Tree
Week 14	Programming of Binary Search Tree
Week 15	Programming of Binary Search Tree

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	**[1]: Author: MICHAEL McMillan. Title:" Data Structures and Algorithms Using C#" , 2007** **[2]: Author : Thomas H. Cormen , CHARLES E. LEISERSON Title :"Introduction to Algorithms ", third edition ,2009**	Yes
Recommended Texts	**[1]:** Author: Bruno R. Preiss Title:" Data Structures and Algorithms with Object-Oriented Design Patterns in C# " 2001 **[2]: Author: Deitle and Deitle. Title: "C Sharp How To Program", Prentic Hall, 2009**	No
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 (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.