MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Data Structures and Algorithms II**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☐ **Lecture** ☒ **Lab** ☒ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	**SOFT2402**				
ECTS Credits	**6**				
SWL (hr/sem)	**150**				
Module Level	2	**Semester of Delivery**	4		
Administering Department	SOFT	**College**	CIT		
Module Leader	بلقيس طلال حسن	**e-mail**	balqees.hasan@uoninevah.edu.iq		
Module Leader's Acad. Title	Lecturer	**Module Leader's Qualification**	Ph.D.		
Module Tutor		**e-mail**			
Peer Reviewer Name		**e-mail**			
Scientific Committee Approval Date	20/06/2023	**Version Number**	1.0		

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	SOFT2302	**Semester**	3
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Objectives أهداف المادة الدراسية	1. To define and apply asymptotic notation, time, and space complexity to analyze the efficiency of algorithms. 2. To analyze and compare the performance of different sorting algorithms. 3. To compare and evaluate the time complexity and efficiency of different search techniques. 4. To explore the principles and implementation of B-Tree, B+ Tree, and Trie structures. 5. To understand Depth-First Search (DFS) and Breadth-First Search (BFS) algorithms for graph traversal. 6. To explore the concepts and implementation of Extendible Hashing. 7. To apply the knowledge and skills acquired throughout the course to introduce a practical project involving data structures and algorithms.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Apply asymptotic notation (Big O, Big Omega, and Big Theta) to analyze and describe the efficiency of algorithms. 2. Understand the principles and applications of Radix Sort, Quicksort, Merge Sort and External Sorting algorithms. 3. Understand, implement, Compare and evaluate the efficiency and effectiveness of Fibonacci Search, Jump Search, Exponential Search, Ternary Search, and Interpolation Search. 4. Understand the properties and applications of Red-Black Trees, Fenwick , B-Trees and B+ Trees. 5. Understand Depth-First Search (DFS) and Breadth-First Search (BFS) algorithms for graph traversal.

Indicative Contents **المحتويات الإرشادية**	Algorithmic Efficiency Analysis Introduction to asymptotic notation (Big O, Big Omega, and Big Theta) Applying asymptotic notation to analyze the time and space complexity of algorithms Examples and exercises on efficiency analysis using asymptotic notation Sorting Algorithms Overview of sorting algorithms Principles and implementation of Radix Sort Principles and implementation of Quicksort Principles and implementation of Merge Sort Principles and implementation of External Sorting algorithms Search Techniques Introduction to various search techniques Implementing and comparing the efficiency of Fibonacci Search, Jump Search, Exponential Search, Ternary Search, and Interpolation Search Evaluating the effectiveness and efficiency of different search techniques through analysis and experiments Tree Structures Introduction to tree structures Properties and applications of Red-Black Trees Properties and applications of Fenwick Trees Properties and applications of B-Trees and B+ Trees Implementation and operations on tree structures Graph Traversal Introduction to graph traversal algorithms Depth-First Search (DFS) algorithm and its implementation Breadth-First Search (BFS) algorithm and its implementation Applications of DFS and BFS in graph traversal and problem-solving

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	The learning and teaching strategies for the data structures and algorithms module involve a combination of theoretical explanations, practical implementations, and active engagement. Students are introduced to the fundamental concepts, principles, and analysis techniques through clear explanations, visual aids, and examples. They are encouraged to actively participate in hands-on activities, such as coding exercises and problem-solving tasks, to reinforce their understanding and develop practical skills. Comparisons and evaluations of different algorithms and data structures are conducted to foster critical thinking and analytical abilities. Real-world applications and case studies are employed to demonstrate the relevance and practicality of the learned concepts. Overall, the module aims to provide a balanced approach that combines theoretical knowledge, practical implementation, and active engagement to facilitate effective learning and mastery of data structures and algorithms.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
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) الحمل الدراسي غير المنتظم للطالب خلال الفصل	72	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	5
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	**175**		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	10% (10)	2 and 7	LO #1, and #3, #4
	Assignments	2	10% (10)	3 and 11	LO #1, #3 and #4, #5
	Projects / Lab.	1	10% (10)	Continuous	All
	Report	1	10% (10)	13	LO #1, #2, #3, #4
Summative assessment	**Midterm Exam**	2hr	10% (10)	7	LO #1 - #2
	Final Exam	3hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered

Week 1	Introduction to Data Structures and Algorithms II ● Overview of the course ● Recap of basic data structures and algorithms
Week 2	● Recap of basic data structures and algorithms-Continued
Week 3	Analysis of Algorithms ● Asymptotic notation (Big O, Big Omega and Big Theta)
Week 4	Analysis of Algorithms- Continued ● Time and Space Complexity Analysis
Week 5	Sorting Algorithms ● Quicksort Analysis and Variants ● Merge Sort Analysis and Variants
Week 6	Sorting Algorithms-Continued ● Radix Sort ● External Sorting Algorithms
Week 7	Midterm Exam
Week 8	Search Techniques ● Fibonacci Search ● Jump Search ● Exponential Search
Week 9	Search Techniques-Continued ● Ternary Search ● Interpolation Search
Week 10	Trees ● B-Trees and B+ Trees ● Red-Black Trees
Week 11	● Trie structures ● Fenwick trees
Week 12	Graph Algorithms Depth-First Search (DFS) and Breadth-First Search (BFS) Algorithms

Week 13	Graph Algorithms-Continued Topological Sorting
Week 14	Extendible Hashing (Dynamic Hashing)
Week 15	Assign a practical project
Week 16	Reviewing previously covered topics

Delivery Plan (Weekly Lab. Syllabus) المنهاج الأسبوعي للمختبر	
	Material Covered
Week 1	Implement basic data structures such as arrays, linked lists, and stacks.
Week 2	Perform operations on basic data structures, such as insertion, deletion, and traversal.
Week 3	Write programs to demonstrate the concept of asymptotic notation (Big O, Big Omega, and Big Theta) for various algorithms.
Week 4	Analyze the time and space complexity of different algorithms and compare their efficiency.
Week 5	Implement and analyze the efficiency of Merge Sort.
Week 6	Implement Radix Sort and analyze its performance on different input sizes.
Week 7	Midterm Exam
Week 8	Implement Fibonacci Search and compare its performance with other search algorithms.
Week 9	Implement Ternary Search and evaluate its efficiency.
Week 10	Implement B-Trees and B+ Trees and perform operations like insertion, deletion, and searching.
Week 11	Implement Trie structures and perform operations like insertion, deletion, and searching.
Week 12	Implement Depth-First Search (DFS) and Breadth-First Search (BFS) algorithms for graph traversal.
Week 13	Implement Topological Sorting algorithm.
Week 14	Implement Extendible Hashing (Dynamic Hashing) and demonstrate its advantages in managing large datasets.
Week 15	Assign a practical project
Week 16	**Reviewing previously covered topics**

Learning and Teaching Resources

مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	Cormen et al. Introduction to Algorithms, 3rd edition, MIT Press, 2009.	No
Recommended Texts	Michael T. Goodrich et.al. Data Structures and Algorithms in Python 1st Edition	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.