

MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information			
معلومات المادة الدراسية			
Module Title	Programming Fundamentals (II)		Module Delivery
Module Type	Core		<input checked="" type="checkbox"/> Theory <input checked="" type="checkbox"/> Lecture <input checked="" type="checkbox"/> Lab <input type="checkbox"/> Tutorial <input type="checkbox"/> Practical <input type="checkbox"/> Seminar
Module Code	MU010601205		
ECTS Credits	7		
SWL (hr/sem)	175		
Module Level	1	Semester of Delivery	
Administering Department	Computer Science	College	College of Science
Module Leader	Nada Thanoon Ahmed	e-mail	nada_thanoon@uomustansiriyah.edu.iq
Module Leader's Acad. Title	Assistant. Prof	Module Leader's Qualification	MSc.
Module Tutor	Zied Othman Ahmed	e-mail	zied_othman@uomustansiriyah.edu.iq
Peer Reviewer Name	Dr. Jamal Nasir Hasoon	e-mail	jamal.hasoon@uomustansiriyah.edu.iq
Scientific Committee Approval Date	01/06/2023	Version Number	1.0

Relation with other Modules			
العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	Programming Fundamentals (I)	Semester	1
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 focuses on the increased levels of care and attention in the teaching of structured programming. This course will be covering: Arrays. string and string processing. Methods and parameter passing. Structured decomposition. File input and output methods, and Analyze and explain the behaviors of simple programs involving the fundamental programming constructs covered by this unit using C# programming language. Furthermore, modify and expand short programs that use standard conditional and iterative control structures and functions.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	<ol style="list-style-type: none">1. Understand the Method, Array, strings and file in C# language.2. Understand the Object –Oriented paradigms.3. Be able to write a program in C# language using array statement.4. Be able to write a program using method statement and introduction to object-oriented paradigms.5. Be able to write a program using file input and output.6. Apply computer programming concepts to problems or situations, and demonstrate appropriate design, coding, testing, and documenting of computer programs that solve these problems or situations using concept of structured programming.
Indicative Contents المحتويات الإرشادية	Indicative content includes the following. <ol style="list-style-type: none">1. Arrays and Lists: [30 hrs]<ul style="list-style-type: none">○ Introduction to arrays and lists.○ Array/List declaration and initialization.○ Array/List manipulation (accessing elements, adding, removing).○ Array/List traversal and searching.○ Multidimensional arrays.2. Methods and Modular Programming: [34 hrs]<ul style="list-style-type: none">○ Introduction to Methods.○ Method declaration and definition.○ Method parameters and return values.○ Scope and lifetime of variables.○ Modular programming concepts.3. Strings and Characters: [12 hrs]<ul style="list-style-type: none">○ String manipulation and operations.○ String comparison and searching.

	<ul style="list-style-type: none"> ○ String concatenation and splitting. ○ Character manipulation. <p>4. Input and Output: [12 hrs]</p> <ul style="list-style-type: none"> ○ Standard input and output. ○ Streams ○ File input and output. ○ Reading from a file ○ Writing to a File ○ Formatting output. ○ Error handling and exception handling.
--	---

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	<p>Learning Strategies:</p> <ol style="list-style-type: none"> 1. Hands-on Practice: Programming is a practical skill that requires practice. Allocate dedicated time to work on coding exercises, small projects, and problem-solving activities. 2. Code Reading and Analysis: Study and analyze existing code examples and programs written by experienced developers. Understand the logic, structure, and flow of the code to enhance your understanding of programming concepts. 3. Break Problems into Smaller Parts: When faced with complex programming problems, break them down into smaller, manageable tasks. This approach, known as "divide and conquer," helps tackle the problem systematically. 4. Collaborative Learning: Engage in discussions and collaborations with fellow learners or join programming communities and forums to share knowledge, ask questions, and solve problems collectively. 5. Debugging and Troubleshooting: Debugging is an essential skill in programming. Practice identifying and fixing errors in your code, as it enhances your problem-solving abilities and helps you understand the logic behind the code. <p>Teaching Strategies:</p> <ol style="list-style-type: none"> 1. Conceptual Explanation: Start by providing clear and concise explanations of programming concepts. Use real-life examples or analogies to help students grasp abstract concepts easily. 2. Hands-on Coding Exercises: Include practical coding exercises and assignments that reinforce the concepts taught. Encourage students to apply the theory into practice, helping them develop their coding skills.

	<ol style="list-style-type: none"> 3. Visualizations and Diagrams: Use visual aids like diagrams, flowcharts, or pseudocode to illustrate programming concepts and algorithms. Visual representations can aid understanding, especially for beginners. 4. Interactive Coding Sessions: Conduct live coding sessions where students can actively participate, ask questions, and solve problems alongside the instructor. This approach promotes engagement and practical learning. 5. Resources and References: Share relevant programming resources, recommended textbooks, and online tutorials to supplement the curriculum and encourage further exploration.
--	--

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	94	Structured SWL (h/w) الحمل الدراسي المنتظم للطالب أسبوعيا	6
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	81	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)	6 and 10	LO #1, #3, #4, #6
	Assignments	2	8% (8)	4 and 12	LO #1, #3,#4,#6
	Projects / Lab.	1	14% (14)	Continuous	All
	Report	1	8% (8)		LO #1, #2, #3, #4, #5.#6
Summative assessment	Midterm Exam	2hr	10% (10)	8	LO #1, #2, #3, #4, #6
	Final Exam	4hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

	Material Covered
Week 1	Array: One dimension array
Week 2	One dimension array
Week 3	Two-dimension array
Week 4	Two-dimension array
Week 5	Two-dimension array
Week 6	Methods
Week 7	Methods
Week 8	Midterm Exam / Method overloading
Week 9	Method overloading
Week 10	Working with structures
Week 11	Working with structures
Week 12	Strings and Characters
Week 13	Strings and Characters
Week 14	Input and Output: file input and output
Week 15	Input and Output: file input and output
Week 16	Preparatory week before the final Exam

Delivery Plan (Weekly Lab. Syllabus)

المنهاج الاسبوعي للمختبر

	Material Covered
Week 1	Examples One dimension array
Week 2	Examples 1D Array
Week 3	Examples Two-dimension array
Week 4	Examples Two-dimension array
Week 5	Examples Two-dimension array
Week 6	: first Exam / Examples Methods
Week 7	Examples Methods
Week 8	Examples Methods
Week 9	Examples Methods
Week 10	: second exam : Examples Structures.
Week 11	Examples Structures.

Week 12	Examples Strings and Characters
Week 13	Examples Strings and Characters
Week 14	Examples Input and Output: file input and output
Week 15	Examples Input and Output: file input and output

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	Available in the Library?
Required Texts	<ul style="list-style-type: none"> - "Programming Basics with C#" Book and Video Lessons Svetlin Nakov and TeameBook 2019 - Visual C# How to Program, 6th edition Published by Pearson (July 14th 2021) - Copyright © 2017 	Yes
Recommended Texts	<ul style="list-style-type: none"> - Deitle and Deitle. "C Sharp How to Program", Prentice 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.