Computer Science Department

Second Grade – First Semester

VISUAL PROGRAMMING MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Visual Programming			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	CSC23117				
ECTS Credits	6				
SWL (hr/sem)	150				
Module Level	2		**Semester of Delivery**	3	
Administering Department	Computer Science		**College**	College of Science	
Module Leader	Adnan J. Jabir		**e-mail**	Adnan.jabir@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Ass. Prof.		**Module Leader's Qualification**	Ph.D.	
Module Tutor	Mays Mohammed Hobi		**e-mail**	mays.m@sc.uobaghdad.edu.iq	
Peer Reviewer Name	Name		**e-mail**	E-mail	
Scientific Committee Approval Date	11/6/2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	CSC12107	**Semester**	2
Co-requisites module		**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Objectives أهداف المادة الدراسية	The objective of a Visual Programming by C# module can vary depending on the context and intended audience. However, here are some common objectives that such a module might aim to achieve: Introduction to Visual Programming: Provide an overview of visual programming concepts and their benefits compared to traditional text-

	based programming approaches. Understanding C# Programming Language: Familiarize students with the fundamentals of the C# programming language, including syntax, variables, data types, control structures, and object-oriented programming principles. Visual Programming Environments: Introduce students to visual programming environments or frameworks that utilize C# as the underlying programming language. This may include tools like Visual Studio. Designing User Interfaces (UI): Teach students how to create visually appealing and interactive user interfaces using drag-and-drop features, controls, and layout tools available in the visual programming environment. Event-Driven Programming: Explore event-driven programming concepts, where actions or events trigger specific functions or operations, and how to implement event handling using visual programming techniques. Data Manipulation and Storage: Cover techniques for data input, validation, manipulation, and storage within the visual programming environment using C# code blocks or components. Integration with External Resources: Demonstrate how to integrate external resources such as databases, web services, or APIs into visual programming projects using C# code and visual components. Error Handling and Debugging: Teach students how to handle errors, debug code, and troubleshoot issues that may arise during visual programming development. Project Development: Provide hands-on project-based learning experiences where students can apply visual programming concepts and C# language skills to develop complete applications or prototypes. Bottom of Form
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Here are some possible learning outcomes for a Visual Programming module using C#: Understand the concepts and advantages of visual programming compared to text-based programming languages. Demonstrate proficiency in using a visual programming environment, such as Visual Studio, for creating applications using C#. Develop user interfaces (UIs) using drag-and-drop tools, controls, and layout features available in the visual programming environment. Apply programming logic and concepts in a visual programming context, including variables, data types, control structures, and object-oriented programming principles. Implement event-driven programming by handling user interactions and triggering appropriate actions or functions in response. Manipulate and store data within the visual programming environment using C# code blocks or components. Integrate external resources, such as databases, web services, or APIs,

	into visual programming projects using C# code and visual components. Debug and troubleshoot visual programming projects, identifying and resolving errors or issues that may arise during development. Apply software engineering principles and best practices specific to visual programming and C# development, such as code organization, modular design, and readability. Demonstrate the ability to create functional and visually appealing applications or prototypes using visual programming techniques with C#. Collaborate effectively in a team environment by participating in project-based learning activities involving visual programming with C#. Reflect on the challenges and limitations of visual programming and propose potential solutions or workarounds. Adapt and learn new visual programming tools or environments as needed to continue exploring and expanding visual programming skills. Understand the ethical considerations related to visual programming, including issues of privacy, security, and intellectual property. Communicate effectively about visual programming concepts, techniques, and project outcomes both orally and in written form.
Indicative Contents المحتويات الإرشادية	Indicative content includes the following. Bottom of Form Indicative content for Windows application development can include various components and topics that are essential for building a Windows application. Here are some indicative content areas: Part1: Basic controls Properties and Events Forms (2hrs) Buttons (2hrs) Labels(2hrs) Textboxes(4hrs) Part2: List Controls ComboBox (6hrs) ListBox (6hrs) Part3: Components Timers (10hrs) Part4: Dialogs and scrollbars Colordialog, Fontdialog, openfiledialog, saveasdiaog (4hrs) Scrollbars, progressbar (4hrs) Part5: Data Structures Arrays and Strings (4hrs) Structures (2hrs) Files (2hrs) Part6: Practical Examples The use of timers and strings (8hrs) The use of comboboxes, listboxs and scrollbars (8hrs) Overall, the objective is to equip learners with the knowledge and skills to create visually appealing applications using visual programming

	techniques with C#.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	The main strategy that will be adopted in delivering this module is to encourage students' participation in the exercises, while at the same time refining and expanding their critical thinking skills. This will be achieved through classes, interactive tutorials, and by considering types of simple experiments involving some interesting sampling activities for the students.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	64	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	86	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	4
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	**150**		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	3	10% (10)	3,5 and 10	LO #1, #2 and #10, #11
	Assignments	2	10% (10)	2 and 12	LO #3, #4 and #6, #7
	Projects / Lab.	1	10% (10)	Continuous	All
	Report	1	10% (10)	13	LO #5, #8 and #10
Summative assessment	**Midterm Exam**	2hr	10% (10)	7	LO #1 - #7
	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 CSharp.NET
Week 2	User Interface Design
Week 3	Button, Textbox, Label controls
Week 4	Form events
Week 5	Button events and properties
Week 6	Check Box, Radio Button
Week 7	Mid-term Exam
Week 8	Working with timers
Week 9	Combobox Lists control
Week 10	Listbox and checked list
Week 11	Second mid-term exam
Week 12	Arrays and strings, Working with files and Object-oriented Programs
Week 13	Messagebox, scrollbars, and desktop notifications
Week 14	Menus, Common Dialog Boxes, and Multiform objects
Week 15	**Final Exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Introduction To Visual Studio 2010 (C#)
Week 2	Form Events and Properties
Week 3	Button Control Events and Properties
Week 4	Checkbox, Radiobuttons, TextBox and Label Control Events and Properties
Week 5	Timer Control Events and Properties
Week 6	ComboBox Control Events and Properties
Week 7	Mid Exam
Week 8	ListBox Control Events and Properties
Week 9	Array and String Operations
Week 10	Structures and Files Operations
Week 11	Messagebox, scrollbars, Progressbars
Week 12	Dialog Controls
Week 13	Month calendar, Tooltip, and desktop notifications
Week 14	MDI applications
Week 15	**Final Exam**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Mastering Microsoft CSharp 2010, 1st Edition by Evangelos Petroutsos, 2010	Yes
Recommended Texts	Turner, Ryan. *C#: 3 books in 1-The Ultimate Beginners, Intermediate and Expert Guide to Master C# Programming*. Publishing Factory, 2020.	No
Websites	The official Microsoft documentation and tutorials for Visual Studio and C# provide in-depth information, examples, and guidance for visual programming with C#. Visit the Microsoft Docs website (https://docs.microsoft.com) and search for relevant topics.	

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 failures" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.

OBJECT ORIENTED PROGRAMMING MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Object Oriented Programming			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	**CSC23114**				
ECTS Credits	6				
SWL (hr/sem)	150				
Module Level	2		**Semester of Delivery**	3	
Administering Department	Computer Science		**College**	College of Science	
Module Leader	Ass. Prof. Dunia Fadhel Safo		**e-mail**	dunia.f@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Asst. Prof.		**Module Leader's Qualification**	M.Sc.	
Module Tutor	Lect. Sura Abid Sarab		**e-mail**	sura.a@sc.uobaghdad.edu.iq	
Peer Reviewer Name			**e-mail**		
Scientific Committee Approval Date	11/6/2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	CSC12107	**Semester**	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	When designing an Object-Oriented Programming (OOP) module, the objective typically focuses on creating reusable, maintainable, and scalable software components. Here are some key objectives to consider for an OOP module:

	1. **Encapsulation**: Encapsulate data and operations within classes to protect the internal state of objects. This ensures that objects manage their state through methods, providing a clear interface and reducing the risk of unintended interference. 2. **Abstraction**: Use abstraction to simplify complex systems by modeling classes based on real-world entities or concepts. This allows users to interact with the system at a high level without needing to understand intricate details. 3. **Inheritance**: Implement inheritance to promote code reuse and establish hierarchical relationships between classes. This enables new classes to inherit properties and behaviors from existing ones, facilitating the creation of more specialized subclasses. 4. **Polymorphism**: Enable polymorphism to allow objects to be treated as instances of their parent class. This supports the ability to call methods on objects of different classes through a common interface, enhancing flexibility and interchangeability. 5. **Modularity**: Design the module to be modular, where classes and objects can be developed, tested, and debugged independently. This improves maintainability and allows for easier updates and enhancements. 6. **Cohesion and Coupling**: Aim for high cohesion within classes and low coupling between classes. Cohesion ensures that a class is focused on a single responsibility, while low coupling minimizes dependencies between classes, making the system more robust and easier to maintain. 7. **Scalability and Extensibility**: Ensure the module can scale and adapt to future requirements. This involves designing flexible interfaces and using design patterns that allow for easy addition of new features or modifications without significant rework. 8. **Code Reusability**: Promote the reuse of code through well-defined classes and interfaces. Reusable components save development time, reduce redundancy, and improve consistency across the application. 9. **Maintainability**: Write clean, readable, and well-documented code to facilitate maintenance. This includes adhering to coding standards, providing comprehensive comments, and creating meaningful class and method names. 10. **Performance and Efficiency**: Optimize the module for performance and efficiency, ensuring that it operates within acceptable resource limits and responds promptly to user interactions.

Module Learning Outcomes مخرجات التعلم للمادة الدراسية	The learning outcomes for an Object-Oriented Programming (OOP) module should clearly define what students will be able to understand and accomplish by the end of the course. Here are the key learning outcomes: Knowledge and Understanding 1. Fundamental Concepts: 2. Class Design and Implementation: Application 3. Creating and Using Objects. 4. Encapsulation and Abstraction. 5. Inheritance and Polymorphism. Analysis 6. Design Patterns. 7. Relationships between Classes. Synthesis 8. Building OOP Applications. 9. Advanced OOP Concepts. Evaluation 10. Code Quality and Best Practices. 11. Testing and debugging. Lifelong Learning 12. Continuous Improvement: o Explore further reading and resources to deepen understanding of OOP. o Participate in coding communities and contribute to open-source projects to stay updated with the latest developments in OOP.
Indicative Contents المحتويات الإرشادية	Indicative content includes the following. **Part A** Introduction to OOP. 2 hrs. Basic Principles Object-oriented (OO) programming: 2 hrs. Comparison with Procedural Programming. 2 hrs. Functions (definitions and parameters passing). 2 hrs. Functions overloading, Correspondence between software objects and real-world objects. 2 hrs. Classes and Objects: Definition and Usage, Attributes and Methods, Instance and Class Variables. 4 hrs. Encapsulation: Data Hiding, Access Modifiers (public, private, protected). 4 hrs. Members and access types. 2 hrs. **Part B** Classes Constructors and destructors. Default and Parameterized

	Constructors, Initialization of Objects. 4 hrs. Objects as passing parameters 6 hrs. Objects as return types. 4 hrs. **Part C** Friend function as a friend classes 6 hrs. Encapsulation; Data hiding and protection. 4 hrs. **Part D** Inheritance. 2hrs. Inheritance types. 4 hrs. Inheritance access types 6 hrs. Virtual inheritance. 4 hrs. **Part E** Overriding of methods 4 hrs. Polymorphism 2 hrs Method Overloading vs. Method Overriding 4 hrs. Runtime vs. Compile-time Polymorphism 4 hrs. Virtual functions 4 hrs. Operator overloading 2 hrs. Abstract classes and methods 3 hrs.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	**classroom hours** Number of hours per week in the classroom for explaining the fundamentals of object programming with examples, discussion, and answering students' questions. Students are allowed to participate in problem-solving. Lab. Hours: number of hours are considered to implement programs. **Non-classroom hours** Number of hours per week, students are expected to read, research, and do other individual work to support engagement in the classroom and computer Lab.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
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	71	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	5

Total SWL (h/Sem) الحمل الدراسي الكلي للطالب خلال الفصل	**150**

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	3	10% (10)	2,8,11	LO #3, # 4, 5
	Assignments	4	10% (10)	6,8,10,11	LO #3, # 4, 5,#7
	Projects / Lab.	1	10% (10)	Continuous	All
	Report	1	10% (10)	-	LO #9
Summative assessment	**Midterm Exam**	2hr/1	10% (10)	7	All
	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 OOP
Week 2	Functions (definitions and calls)
Week 3	Parameter types (value, reference)
Week 4	Classes and objects
Week 5	Members and access types
Week 6	Objects as passing parameters
Week 7	**Midterm exam**
Week 8	Objects as return types
Week 9	Encapsulation, abstraction, data hiding, and message passing
Week 10	Constructors and destructors

Week 11	Friend functions
Week 12	Inheritance (definition types, and access types)
Week 13	Polymorphism, and binding types
Week 14	Overriding and virtual functions, operator overloading
Week 15	**Final exam.**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Revision for structured programming
Week 2	Programming using different types of functions
Week 3	Programming using classes and objects
Week 4	Programming using an array of objects
Week 5	Using the objects as parameters to member and non-member functions.
Week 6	Classes with constructors and destructors
Week 7	**Mid-term exam.**
Week 8	Programming the friend functions and friend classes program
Week 9	Inheritance with classes in programs
Week 10	Programs with single, multi-level inheritance
Week 11	Programs with multiple, hierarchical inheritance
Week 12	Programs with hybrid inheritance and solving diamond problems
Week 13	Inheritance in classes with destructors and different types of constructors
Week 14	Virtual functions in classes
Week 15	**Final exam.**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Object-Oriented Programming Using C++ BY JOYCE FARRELL	Yes
Recommended Texts	OBJECT-ORIENTED PROGRAMMING BY ROBERT LAFORE	No

Websites	https://www.learncpp.com/

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 failures" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.

NUMERICAL METHODS MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Numerical Methods			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	CSC23115				
ECTS Credits	5				
SWL (hr/sem)	125				
Module Level	2		**Semester of Delivery**	3	
Administering Department	Computer Science		**College**	College of Science	
Module Leader	Dr. Najlaa Mohammed Hussein		**e-mail**	najlaa.alkhafaji@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Lecturer		**Module Leader's Qualification**	Ph.D.	
Module Tutor	Lect. Ghusoon Ghazi Mohammed		**e-mail**	ghusoon.g@sc.uobaghdad.edu.iq	
Peer Reviewer Name			**e-mail**		
Scientific Committee Approval Date	11/6/2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	CSC11004	**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	1. Understanding of numerical approximation techniques: The course aims to introduce students to various numerical approximation techniques used to solve mathematical problems that are difficult or impossible to solve analytically.

	2. Familiarity with numerical algorithms: Students are expected to learn and understand the underlying algorithms employed in numerical methods, such as root-finding methods, interpolation, and solving system of linear equations. 3. Application of numerical methods: The course aims to equip students with the ability to apply numerical methods to solve real-world problems encountered in scientific, engineering, and computational disciplines. 4. Error analysis: Students should gain an understanding of the types of errors that can arise during numerical approximations, including round-off errors, truncation errors, and propagation of errors. They should be able to analyze and estimate the accuracy of numerical solutions. 5. Programming skills: students can implement numerical algorithms using programming language such as MATLAB. The objective is to develop computational skills to implement and apply numerical methods effectively.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Proficiency in numerical techniques: Students should be proficient in a range of numerical techniques and algorithms. They should have a good understanding of how and when to apply these techniques to solve specific mathematical problems. 2. Analytical thinking: The course helps students develop analytical thinking skills by requiring them to understand the mathematical foundations of numerical methods, analyze the accuracy and convergence properties of algorithms, and evaluate the implications of their results. 3. Problem-solving abilities: Students should acquire problem-solving abilities in the context of mathematical modeling and computational analysis. They should be able to formulate problems, select appropriate numerical methods, and interpret and validate the results obtained from numerical computations. 4. Understand sources of error: The course seeks to help students understand the various sources of errors in numerical computations, such as truncation errors and round-off errors. Students should learn to analyze and estimate these errors and their impact on the accuracy of numerical solutions. 5. Programming proficiency: Students should gain proficiency in writing code to implement numerical algorithms, understand the syntax and structure of programming languages MATLAB, and debug and optimize their code.

Indicative Contents المحتويات الإرشادية	Indicative content includes the following: **Part A - Solutions of Equations in One Variable** Bisection method, [4 hrs]. Newton-Raphson method, [4 hrs]. Secant method, [4 hrs]. False Position method, [4 hrs]. Fixed Point method, [4 hrs]. **Part B - Solving Linear Systems** Gauss Elimination method, [5 hrs]. LU Decomposition method, [5 hrs]. Jacobi's method, [5 hrs]. Gauss-Seidel method, [5 hrs]. **Part C - Interpolation and Polynomial Approximation** Polynomial, [4 hrs]. Interpolation and the Lagrange Polynomial, [5 hrs]. Newton's Divided Differences method, [5 hrs]. **Part D – Programming in MATLAB** Learning MATLAB language, [13 hrs]. Programming numerical algorithms using MATLAB, [12 hrs].

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	The main strategy that will be adopted in delivering this module is to understand theoretical concepts and introduce various numerical methods by explain their mathematical principles, algorithms, and their applications through presentations, visual aids, and examples, which will encourage students' participation in the exercises and HomeWorks. Since numerical methods require implementation in computer programming languages, Students use programming languages such as MATLAB to write code and execute numerical methods and guide students through coding exercises.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
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**	3	10% (10)	2 , 7, 10	LO #1, #2 , #3, #4, #5
	Assignments	5	10% (10)	2 , 3, 4, 5,14	Lo #2, #3, #4 ,#5
	Projects / Lab.	2	10% (10)	Continuous	All
	Report	1	10% (10)	-	LO #5
Summative assessment	**Midterm Exam**	2hr / 1	10% (10)	7	All
	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 Numerical Methods.
Week 2	Approximations and Errors.
Week 3	Solution of Equations of a Single Variable.
Week 4	Bisection method.
Week 5	Newton-Raphson method.
Week 6	Secant method.
Week 7	Mid-term Exam
Week 8	Systems of Linear Equations.
Week 9	Gauss Elimination Method.
Week 10	LU Decomposition Method.
Week 11	Jacobi's Method.
Week 12	Gauss-Seidel Method.
Week 13	Interpolation and the Lagrange Polynomial Method.
Week 14	Newton's Divided Differences Method.

Week 15	**Final Exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Lab 1: Introduction to MATLAB.
Week 2	Lab 2: M File.
Week 3	Lab 3: Program Control Statements
Week 4	Lab 4: Functions.
Week 5	Lab 5: Bisection method & Secant method.
Week 6	Lab 6: Newton-Raphson method.
Week 7	Mid-term Exam
Week 8	Lab 7: Vectors and Matrices and Cell array.
Week 9	Lab 8: Vector operation and Matrix operation.
Week 10	Lab 9: Functions applied on vectors and matrices.
Week 11	Lab 10: Gauss Elimination
Week 12	Lab 11: LU Decomposition.
Week 13	Lab 12: Jacobi's Method and Gauss-Seidel Method.
Week 14	Lab 13: Polynomial, Lagrange Interpolation and Newton's Divided Difference Method.
Week 15	**Final Exam**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	1. "Numerical Analysis", Richard. L. Burden, J. Douglas. Faires and Annette M. Burden, Brooks/Cole, Cengage Learning, 10th edition, 2017. 2. "Programming with MATLAB for Scientists: A beginner's Introduction", Eugeniy E. Mikhailov, CRC Press (Taylor & Francis Group), 2017.	No
Recommended Texts	"Numerical Methods for Engineers and Scientists Using MATLAB", Ramin S. Esfandiari, CRC Press (Taylor &	No

	Francis Group), 2nd edition, 2017.	
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 failures" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.				

DATA STRUCTURES 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	CSC23113				
ECTS Credits	6				
SWL (hr/sem)	150				
Module Level	2		**Semester of Delivery**	3	
Administering Department	Computer Science		**College**	College of Science	
Module Leader	Sarab M. Hameed		**e-mail**	Sarab.m@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Professor		**Module Leader's Qualification**	Ph.D.	
Module Tutor	Nasreen Jawad Kadhim		**e-mail**	nasreen.kadhim @sc.uobaghdad.edu.iq	
Peer Reviewer Name			**e-mail**		
Scientific Committee Approval Date	11/6/2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	CSC12107	**Semester**	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	A data structure course aims to provide students with the fundamentals of data structures and their associated algorithms. It gives an overview of various data structures and their applications and how to construct them in C++ code. The course begins with an introduction to essential data structures including arrays, linked lists, stacks, and queues. Array properties, allocation methods, and applications are covered. The fundamental principles of stack and queue data structures and their applications, operations, and algorithms are addressed. The basics of the

	linked list data structure, its memory allocation, types, operations, advantages, and disadvantages are also introduced. Then, non-linear data structures such as trees and graphs are introduced.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	The expected learning outcomes of a data structures course are as follows: 1. Students will be able to identify various data structures' characteristics, operations, and properties. 2. Students will learn how to implement data structures in C++. They can write code to efficiently create, manipulate, and manage data structures. 3. Students will develop problems using data structures. They will learn how to apply appropriate data structures to solve a problem. 4. Students will become familiar with standard libraries or APIs that provide pre-implemented data structures in C++. They will learn how to use these libraries effectively and efficiently.
Indicative Contents المحتويات الإرشادية	**Part A: Introduction to Data Structures and Linear Data Structures (40 hrs)** Definition and importance of data structures. Array types: One-dimensional arrays and multi-dimensional arrays. Stacks and Queues: Introduction to stacks and queues Implementation and applications, Operations on stacks and queues. Linked Lists and their types: Singly linked lists, doubly linked lists, Circular linked lists, and Operations on linked lists. **Part B: Non-Linear Data Structures (20 hrs)** Introduction to trees, Binary trees, Binary search trees, and tree traversals: in-order, pre-order, post-order. Basics of graphs and representation of graphs
Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Delivering lectures to introduce and explain essential concepts, principles, and theories related to data structures. This helps students build a strong foundation of knowledge. 2. Giving hands-on programming exercises helps them to apply the learned concepts and implement data structures. This helps their understanding and enhances their programming skills. 3. Providing code walkthroughs and examples demonstrating the implementation and use of various data structures. 4. Conducting problem-solving sessions, both in class and through assignments to allow students to apply data structures to solve problems 5. Encouraging group projects and activities among students. This promotes teamwork and enhances understanding. 6. Regular evaluations and examinations help gauge students'

	understanding and progress. These can include quizzes, assignments, and exams that assess theoretical knowledge and practical application of data structures. 7. Recommending textbooks, internet resources, and supplementary references might help students study more effectively.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	64	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	86	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	**150**		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	10% (10)	4 and 10	LO #1, #2, and #3
	Assignments	2	10% (10)	3 and 10	LO #1, #2, and #3
	Projects / Lab.	1	10% (10)	Continuous	All
	Report	5	10% (10)	12	All
Summative assessment	**Midterm Exam**	2hr	10% (10)	7	LO #1, #2, and #3
	Final Exam	4hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	Overview of data structures

Week 2	Array data structure, its properties, storage allocation, and applications
Week 3	Algorithm definitions
Week 4	Stack data structure
Week 5	Applications of stack data structure
Week 6	Queue data structure
Week 7	**Mid-term exam**
Week 8	Linked List, types, advantages, disadvantages
Week 9	Linked list operations
Week 10	linked lists Applications
Week 11	Introduction to the tree data structure
Week 12	Binary Tree data structure, its traversal method, and representation
Week 13	Binary search tree applications, operations, and algorithms
Week 14	Graph data structure, terminology, and representation
Week 15	**Final Exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Lab 1: Introduction to C++ IDE
Week 2	Lab 2: Array data structure: traverse an array, insert an element into an array at a specific location, remove an element from an array, Search for an element in an array, and modify an element in an array.
Week 3	Lab 3: Stack data structure operations (pop, push, IsEmpty, IsFull, Peek)
Week 4	Lab 4: Stack data structure applications (converting an expression from infix to postfix)
Week 5	Lab 5: Stack data structure applications (recursion function)
Week 6	Lab 6: Queue data structure operations (Enqueue: Inserts a new element at the rear of the queue, Dequeue: Removes the front element of the queue and returns it, Peek: Returns the front element present in the queue without dequeuing it, IsEmpty: Checks if the queue is empty, IsFull: Checks if the queue is full)
Week 7	**Mid-term exam**

Week 8	Lab 7: Single linked list operations: (insert a new node in a singly linked list, delete: remove a node from a singly linked list. − Traverse: displays the content of the linked list. Search: returns true if the specific data exists in the linked list otherwise returns false).
Week 9	Lab 8: Double-linked list
Week 10	Lab 9: Double-linked list operations
Week 11	Lab 10: Binary search tree
Week 12	Lab 11: Binary search tree traversal
Week 13	Lab 12: Binary search tree operations
Week 14	Lab 13: Graph data structure
Week 15	**Final Exam**

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	**Available in the Library?**
Required Texts		
Recommended Texts	1. Yedidyah Langsam, Moshe J. Augenstein, Aaron M. Tenenbaum, , Data Structures using C and C++, Prentice-Hall 2. Nell Dale, Chip Weems Tim Richards, C++ Plus Data Structures, Jones and Bartlet, Inc. 3. D.S. Malik, Data Structures Using C++, Second Edition 4. Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, Addison Wesley.	
Websites	https://tutorialink.com/ds/ https://www.tutorialspoint.com/data_structures_algorithms/index.htm	

Grading Scheme
مخطط الدرجات

Group	**Grade**	التقدير	**Marks %**	**Definition**
Success Group	**A -**	امتياز	90 -	Outstanding Performance

(50 - 100)	Excellent		100	
	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.

COMPUTATION THEORY MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Computation Theory			**Module Delivery**	
Module Type	C			☒ **Theory** ☒ **Lecture** ☐ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	CSC23116				
ECTS Credits	5				
SWL (hr/sem)	125				
Module Level	2		**Semester of Delivery**	3	
Administering Department	Computer Science		**College**	College of Science	
Module Leader	Assist Prof Wijdan Abdul Ameer Hassan		**e-mail**	wijdan@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Assistant Prof.		**Module Leader's Qualification**	MSC	
Module Tutor	Yasmin Alaa Hassan		**e-mail**	Yasmin.a@sc.uobaghdad.edu.iq	
Peer Reviewer Name	Name		**e-mail**	E-mail	
Scientific Committee Approval Date	11/6/2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	CSC12008	**Semester**	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	1-The objectives of computational theory in computer science are to study and understand the fundamental principles, capabilities, and limitations of computation. 2- Computational theory investigates the design and semantics of programming languages. 3- Computational theory investigates formal languages, which are sets of strings with specific properties, and the automata that recognize and generate these languages. This field explores the properties and

	relationships of different types of automata, such as finite automata, 4- Understand the application of language theory ideas, such as regular expressions and context free grammar in the design of software, such as compilers and text processors.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1- Computation theory provides a deep understanding of the fundamental principles that underlie computer science and information processing. 2- Computation theory helps develop strong problem-solving skills. 3- Computation theory provides insights into programming language design and implementation. Students gain an understanding of formal grammars, parsing techniques, and language semantics. 4- Computation theory serves as a foundation for advanced studies in computer science and related fields 5- Computation theory forms the theoretical backbone of computer science. By studying this field, the student will develop a strong foundation in the fundamental principles of computation, algorithms, and problem-solving.
Indicative Contents المحتويات الإرشادية	Indicative content includes the following: Part 1 Set, graph, tree, string and language, [2 hrs]. Deterministic Finite Automata (DFA), [5 hrs]. Nondeterministic finite automata (NFA), [4 hrs]. Finite automaton with output (Moore and Mealy machine), [4 hrs]. Part 2 Regular expression, [4 hrs]. Transition graph, Kleene's theorem, [3 hrs]. Part 3 Grammar, [7 hrs]. Canonical form of context free grammar [4]

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	The main strategy that will be adopted in delivering this module are: 1. Power point presentation (Data show). 2. Explanation on the white board using different color markers. 3. Discussions with the student during teaching. 4. Interaction with students through lecture. 5. Solve different examples with more exercises. 7. Prepare reports that develop critical thinking for students. 8. Submit assignment that develop student learning.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem)	33	**Structured SWL (h/w)**	2

الحمل الدراسي المنتظم للطالب خلال الفصل		الحمل الدراسي المنتظم للطالب أسبوعيا	
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.1
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**	5	10% (10)	3 and 12	LO #3, #4 and #5, #6 and #7, #8 and #9, #10 and #11, #12
	Assignments	2	10% (10)	2 and 10	LO #3, #4,#5 and #7,#8,#9
	Projects	1	10% (10)	Continuous	All
	Report	1	10% (10)	-	
Summative assessment	**Midterm Exam**	2hr	10% (10)	7 and 15	LO #1 - #7 and LO #8 - #14
	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 computation theory • Set, graph, tree, string and language
Week 2	• Finite State system • Deterministic finite automata (DFA)
Week 3	• Nondeterministic finite automata (NFA)
Week 4	• Equivalence of DFA and NFA
Week 5	• Nondeterministic finite automata with Є • Equivalence of NFA's with and without ϵ -moves

Week 6	• Finite automation with output Moore and Mealy machine • Equivalence of Moore and Mealy machines
Week 7	• **Mid-term Exam**
Week 8	• Regular expression
Week 9	• Equivalence of finite automata and regular expression
Week 10	• Transition graph • Kleene's theorem
Week 11	Grammar • Formal definition of grammar • Derivation and language • Derivation tree
Week 12	Types of Grammar • Phrase Structure Grammar (PSG) • Context Sensitive Grammar (CSG) • Context Free Grammar (CFG)
Week 13	• Simplification of context free grammar
Week 14	Canonical form of context free grammar 1- Chomsky Normal Form (CNF) 2- Greibach Normal Form (GNF)
Week 15	A preparatory week before the Final Exam
Week 16	Final Exam

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	1-Introduction to automata theory languages and computation, John E. Hopcroft, 1979.	Yes
Recommended Texts	2- Introduction to computer theory "Daniel I. A. Cohen",1986.	No
Websites	Many reference from 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 -	متوسط	60 - 69	Fair but with major shortcomings

	Satisfactory			
	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.

ENGLISH LANGUAGE II MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	English Language II			**Module Delivery**	
Module Type	B			☒ **Theory** ☒ **Lecture** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	UOB202				
ECTS Credits	2				
SWL (hr/sem)	50				
Module Level	2		**Semester of Delivery**	3	
Administering Department	Type Dept. Code		**College**	Type College Code	
Module Leader	Dr. Muthana Hameed Khalaf		**e-mail**	muthana.khalaf@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Assistant Professor		**Module Leader's Qualification**	Ph.D.	
Module Tutor	Name (if available)		**e-mail**	E-mail	
Peer Reviewer Name	Name		**e-mail**	E-mail	
Scientific Committee Approval Date	11/6/2024		**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	A pre-intermediate level course aiming to build and further improve language proficiency for second year students/ college of science, 1. Listening Objectives: • Understand and respond appropriately to a variety of spoken English in familiar contexts. • Comprehend main ideas, specific details, and imply information in spoken texts. • Develop listening strategies to enhance understanding. 2. Speaking Objectives: • Engage in conversations on a range of topics using appropriate

	vocabulary and grammar. • Express opinions, preferences, and experiences. • Develop speaking strategies for effective communication, such as turn-taking and seeking clarification. 3. Reading Objectives: • Read and understand a variety of texts, including articles, stories, and informational passages. • Comprehend main ideas, detail, and imply information in written texts. • Develop reading strategies for comprehension and vocabulary acquisition. 4. Writing Objectives: • Write coherent paragraphs and short texts on different topics. • Express ideas clearly and logically using appropriate grammar and vocabulary. • Develop writing strategies for organization, coherence, and accuracy. 5. Grammar and Vocabulary Objectives: • Develop a solid understanding and usage of a wide range of grammatical structures appropriate for the pre-intermediate level. • Expand vocabulary knowledge to include a broader range of words, idiomatic expressions, and collocations. • Apply grammar and vocabulary knowledge to express oneself accurately and effectively. 6. Pronunciation and Intonation Objectives: • Improve pronunciation accuracy of individual sounds, stress patterns, and intonation. • Use appropriate rhythm, stress, and intonation for effective communication. • Recognize and produce connected speech features to enhance fluency and naturalness. 7. Cultural Awareness Objectives: • Develop an understanding of cultural practices, customs, and social norms in English-speaking countries. • Demonstrate cultural sensitivity and adapt communication accordingly. • Recognize the impact of culture on language use and communication styles.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Learning is essential to the achievement of the Learning Outcomes. 1. Listening and Speaking: • Understand and respond appropriately to a range of everyday spoken English in familiar contexts. • Engage in conversations and discussions on a variety of topics using appropriate language and strategies. • Comprehend and extract information from spoken texts, such as

	interviews, dialogues, and narratives. 2. Reading: • Read and understand a variety of texts, including articles, stories, and informational passages. • Comprehend main ideas, details, and specific information from the texts. • Apply reading strategies to infer meaning from context and make predictions. 3. Writing: • Write coherent and well-organized paragraphs and short texts on various topics. • Express ideas and opinions clearly and concisely. • Demonstrate control of grammar, vocabulary, and sentence structures appropriate for the pre-intermediate level. 4. Grammar and Vocabulary: • Understand and use a wide range of grammatical structures and tenses, including present perfect, past simple, future forms, and conditionals. • Expand vocabulary knowledge to include a broader range of words, idiomatic expressions, and collocations. • Apply grammar and vocabulary in context to enhance communication skills. 5. Pronunciation and Intonation: • Develop accurate pronunciation of individual sounds and common word stress patterns. • Use appropriate intonation and stress patterns to convey meaning effectively. • Understand and produce connected speech features, such as linking sounds and contractions. 6. Cultural Awareness: • Gain insights into cultural practices, traditions, and customs in English-speaking countries. • Develop intercultural competence and sensitivity in communication. • Understand cultural influences on language use and behavior.
Indicative Contents المحتويات الإرشادية	Indicative content includes the following. 1: Greetings and Introductions • Vocabulary: Greetings, introductions, personal information • Grammar: Present simple, present continuous, subject pronouns, possessive adjectives • Skills: Listening to and giving personal information, role-playing introductions, writing short personal profiles 2: Daily Routines • Vocabulary: Daily activities, time expressions • Grammar: Present simple, adverbs of frequency, prepositions of

	time • Skills: Talking about daily routines, describing habits and schedules, writing a daily routine diary 3: Family and Relationships • Vocabulary: Family members, relationships, adjectives to describe people • Grammar: Possessive 's, can/can't, imperatives • Skills: Talking about family members, describing people's appearance and personality, writing about family members 4: Free Time and Hobbies • Vocabulary: Leisure activities, hobbies, sports • Grammar: Present simple vs. present continuous, question words • Skills: Discussing leisure activities, talking about hobbies and interests, writing about favorite pastimes 5: Shopping and Money • Vocabulary: Shops, money, prices, clothes • Grammar: Countable and uncountable nouns, plurals, quantifiers • Skills: Role-playing shopping conversations, describing clothes, writing a shopping list 6: Travel and Transportation • Vocabulary: Means of transport, travel destinations, directions • Grammar: Present perfect, past simple, adverbs of time • Skills: Discussing travel experiences, giving and following directions, writing about a memorable trip 7: Food and Eating Habits • Vocabulary: Food items, meals, cooking, restaurants • Grammar: Countable and uncountable nouns, articles, some/any • Skills: Talking about food preferences, ordering in a restaurant, writing a recipe 8: Health and Well-being • Vocabulary: Health issues, symptoms, remedies • Grammar: Should/shouldn't, modals for advice and obligation • Skills: Discussing health problems, giving advice, writing a health blog post 9: Jobs and Careers • Vocabulary: Professions, job descriptions, skills • Grammar: Past continuous, comparatives and superlatives • Skills: Talking about jobs and career aspirations, describing job experiences, writing a resume 10: Future Plans and Ambitions

	• Vocabulary: Future forms (will, going to, present continuous), ambitions, goals • Grammar: Future forms, time clauses • Skills: Discussing future plans, setting goals, writing a letter to your future self 11: Technology and Communication • Vocabulary: Communication devices, social media, technology-related terms • Grammar: Present perfect continuous, future continuous, indirect questions • Skills: Discussing technology and its impact, describing communication habits, writing an email or text message 12: Environment and Sustainability • Vocabulary: Environmental issues, natural disasters, conservation • Grammar: Conditional sentences, passive voice • Skills: Discussing environmental concerns, expressing opinions on sustainability, writing an article on environmental conservation 13: Culture and Traditions • Vocabulary: Festivals, customs, cultural practices • Grammar: Reported speech, relative clauses • Skills: Talking about cultural events, comparing traditions, writing a description of a cultural celebration 14: Education and Learning • Vocabulary: School subjects, learning methods, educational institutions • Grammar: Past perfect, modals for possibility and certainty • Skills: Discussing educational experiences, describing favorite subjects, writing an opinion essay on the benefits of education 15: Travel and Tourism • Vocabulary: Tourist attractions, accommodation, travel experiences • Grammar: Comparative and superlative adjectives, phrasal verbs • Skills: Talking about travel preferences, recommending destinations, writing a travel blog post or itinerary

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Communicative Approach: Emphasize communicative activities that promote interaction among students. Encourage pair and group work, role-plays, and discussions to practice language skills in meaningful

	contexts. 2. Integrated Skills: Integrate the four language skills (speaking, listening, reading, and writing) in lessons to create a balanced approach to language learning. Provide opportunities for students to use and develop these skills simultaneously. 3. Vocabulary Expansion: Incorporate vocabulary-building exercises and activities throughout the course. Use real-life contexts, visuals, and practical examples to help students learn and remember new words. 4. Grammar Focus: Teach and reinforce grammar structures in a systematic and progressive manner. Provide clear explanations, examples, and practice exercises to ensure students understand and can apply the grammar rules correctly. 5. Authentic Materials: Include authentic texts, such as articles, newspaper clippings, songs, and videos, to expose students to real-world language usage. This helps develop their reading and listening comprehension skills and exposes them to cultural aspects of English-speaking countries. 6. Cultural Awareness: Integrate cultural topics and discussions into the lessons to foster cultural awareness and sensitivity. Encourage students to share their own cultural backgrounds and experiences to promote understanding and appreciation of diverse perspectives. 7. Error Correction: Provide constructive feedback and error correction during speaking and writing activities. Help students identify and correct their mistakes, focusing on accuracy while encouraging fluency and self-expression. 8. Technology Integration: Utilize technology tools, such as interactive whiteboards, online resources, and language learning apps, to engage students and enhance their language learning experience. Incorporate multimedia materials for listening and speaking practice. 9. Regular Assessment: Assess students' progress regularly through quizzes, tests, and assignments. Provide timely feedback to guide their learning and address areas that need improvement. 10. Individualization: Cater to the individual needs and learning styles of students. Offer differentiated tasks and activities to ensure all learners are appropriately challenged and supported. 11. Cooperative Learning: Promote collaboration and teamwork among students through pair work, group projects, and peer feedback. This encourages active participation and a supportive learning environment.

	12. Review and Revision: Schedule regular review sessions to consolidate previously learned material. Encourage students to revise and practice independently, providing resources for self-study and additional practice.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	33	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	17	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	1.25
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	**50**		

Module Evaluation تقييم المادة الدراسية					
As		**Time/Number**	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	10% (10)	5 and 10	LO #1, #2 and #10, #11
	Assignments	2	10% (10)	2 and 12	LO #3, #4 and #6, #7
	Projects	1	10% (10)	Continuous	All
	Report	1	10% (10)	13	LO #5, #8 and #10
Summative assessment	**Midterm Exam**	2hr	10% (10)	7	LO #1 - #7
	Final Exam	3hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري

	New Headway Plus provides an integrated skills course with each unit divided into grammar, vocabulary, skills work and everyday English segments as follows:
Week 1	Getting to know you p6 **Tenses** *Present, past, future* p6 Questions *Where were you born?* *What do you do?* p6 Question words *Who …?, Why …?,* *How much …?* p7 **Right word, wrong word** Verbs of similar meaning *speak/talk*, *say/tell* Adjectives and nouns that go together Prepositions *to, from, at, about, of, on, in,* etc. Words with two meanings *I met my husband on a blind date.* *Dates are good for you.* p12 **Social expressions** *Have a good weekend!* *Same to you.* p13
Week 2	Whatever makes you happy p14 **Present tenses** Present Simple *She lives alone in Bristol.* p14 Present Continuous *She's planning …* p14 *have/have got* *He has his own company.* *I've got an idea for …* p15 **Things I like doing** *play games* *have a lie-in* *get up late* p17 **Making conversation** *What a lovely day it is* *today!* *Are you having a good* *time in London?*

	Have a good weekend! p21
Week 3	What's in the news? p22 **Past tenses** Past Simple *How far did he walk?* *I had a shower last night.* p23 Past Continuous *I was having a shower when …* p23 **Adverbs** *drive carefully* *speak furiously* *work hard* p28 **Saying when** *What's the date today?* *It's June the twentysecond.* *When did you last go* *to the cinema?* *Two weeks ago.* p29
Week 4	Eat, drink, and be merry! p30 **Quantity** *much* and *many* *How much milk?* *How many eggs?* p31 *some* and *any* *some apples, any bananas* p31 *a few*, *a little*, *a lot/lots of* p31 *something* / *someone* / *somewhere* p32 Articles *a shopkeeper, an old village,* *the north of England, He came* *by bus.* p32 **Food** *apples, beer, bread, cake* p36 **Shopping** *newsagent's, chemist's,* *off-licence* p36 **Can you come for dinner?** *Would you like some* *more rice?* *Could you pass the* *salt, please?* *How would you like* *your coffee?*

	This is delicious! p37
Week 5	Looking forward p38 **Verb patterns** *want/hope to do* *like/enjoy doing* *looking forward to doing* *'d like to* p38 **Future forms** *going to*, *will* and Present Continuous *I'm going to stay with a friend.* *I'll call or text you.* *I'm working late this evening.* p40 **Phrasal verbs – literal** *move back* *take away* *grow up* p44 **Phrasal verbs – idiomatic** *give up* *take off* *look after* p44 **Expressing doubt and certainty** *Of course he will.* *He might do.* *Mmm … maybe.* *I doubt it.* *No chance.* p45
Week 6	The way I see it p46 ***What … like?*** *What's your teacher like?* p46 **Comparative and superlative adjectives** *big, bigger, biggest* *good, better, best* p47 *as … as* *It isn't as hot as Dubai.* p47 **Relative pronouns** *who/that/which/where* p110 **Synonyms and antonyms** *lovely, beautiful* *brilliant, terrible* p52 **What's on?** *How much is it to go* *in the museum?* *Is it open on Sunday?* *What film is suitable*

	for children? p53
Week 7	Mid-term Exam
Week 8	Living history p54 **Present Perfect** *John has lived there for three years.* p55 *for* and *since* *for two hours* *since six o'clock* p55 *ever* and *never* *Have you ever been …?* *I've never been to South America.* p56 **Present Perfect or Past Simple** *Have you had an ordinary job?* *I worked in a restaurant.* p57 **Word endings** Jobs *philosopher, historian, economist* p57 Nouns and adjectives *competition, famous* p57 **Word stress** *danger, dangerous* *invite, invitation* p57 **Agree with me!** *It's wonderful, isn't it?* *You come from Scotland, don't you?* *It wasn't easy, was it?* *You've lived here for years, haven't you?* p61
Week 9	Girls and boys p62 ***have to*** *She has to train hard.* *I don't have to train every day.* *Do you have to work at weekends?* p63 ***should*** *You should show him this letter.* p64 ***must*** *He must get professional help.* p64 **Things to wear** *belt, cap, boots, jumper, make-up* p68 Materials

	leather, wool, denim, *cotton* p68 Situations *job interview, party, beach* *holiday* p68 **At the doctor's** *a sore throat, flu, food* *poisoning* *I've got a fever.* *My body aches.* *My glands are* *swollen.* p69
Week 10	Time for a story p70 **Past Perfect** *They had walked twenty miles.* p71 **Narrative tenses** *They saw a bear.* *They were looking for work.* p71 **Joining sentences** *although, because* *when, while, before, after, as, until,* *as soon as* p72 **Feelings** *angry, nervous, delighted,* *stressed* p76 **Exclamations with so and such** *I was so scared!* *It was such a shock!* *We had such terrible* *weather!* *I've got so much work!* p77
Week 11	Our interactive World p78 **Passives** *Mobile phones are used by almost* *6 billion people.* *The first mobile phone call was made* *in 1973.* *Camera phones have been sold since* *2002.* *Landline telephones will be replaced* *by mobile phones.* p79 **Words that go together** Noun + noun *text message,*

	businessman p81 Verb + noun *take notes,* *send a text message* p81 Adverb + adjective *well-known,* *badly-behaved* p81 **On the phone** *07700 900333* *Can I speak to* *Patrick, please?* *I'm calling because …* *Sorry, you're breaking* *up …* p85
Week 12	Life's what you make it! p86 **Present Perfect Continuous** *He's been making programmes* *since 2007.* *How long has she been working* *there?* p87 Present Perfect Simple versus Continuous *He's made three programmes.* *He's been teaching for three years.* p87 **Birth, marriage, death** *pregnant, born* *engaged, divorced* *funeral, died of* p92 **Good news, bad news** *Congratulations!* *That's fantastic news!* *What a shame!* *I'm so sorry.* p93
Week 13	Just wondering … p94 **First conditional *if* + *will*** *If it's sunny, we'll go for a picnic.* *We won't go out if it rains.* p95 *going to* and *might* *What are you going to do tonight?* *I might go out …* p95 **Second conditional *if* + *would*** *If I had a brother, I'd play with him.* *If I were you, I'd stop smoking.* p96 **Prepositions** *connected to* *on a date*

	listen to *think about* p100 **Thank you and goodbye!** *It's late. I must be going now.* *Thank you for a lovely evening.* *My pleasure!* p101
Week 14	**Living in a stately home** *Living history* Chatsworth House and the family who calls it home p58 **A family history** David Taylor Bews from Perth, Australia researches his family history p60 **What do you think?** Stately homes Aristocracy Inherited wealth p58 **Talking about you** Have you ever …? p57 The lives of your grandparents p60 **What do you think?** Family history p60 **A biography** Ordering paragraphs: Two Kennedys Researching facts about a famous person and writing a biography p111 **Families with all boys or all girls** *Sons and daughters* The parents of four daughters swap homes with the parents of four sons p66 **Heptathlon champion** An interview with Jessica Ennis – Britain's first world heptathlon champion p65 **What do you think?** Talking about successful people p65

	Pros and cons of all-girl or all-boy families The ideal family p66 **Dress person X** Describing an outfit p68 **Letters and emails** Formal and informal expressions *Dear Sir or Madam,* *Yours sincerely,* *Hi Cathy,* *Love Steve* Writing a formal letter to a language school and an email to an English friend p112
Week 15	**Final Exam**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	The core textbook is *Soars, John and Liz, (2011), New Headway Plus Pre-Intermediate Student's Book, Special Edition, Oxford University Press*	Yes
Recommended Texts	New Headway Plus provides an integrated skills course with each unit divided into grammar, vocabulary, skills work and everyday English segments	No
Websites	Oxford University Press: The New Headway series is published by Oxford University Press. Visit their website at www.oup.com and search for "New Headway Plus, Special Edition, pre-Intermediate" or browse their English language teaching section for information on the course.	

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.				

ALGORITHMS DESIGN AND ANALYSIS MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Algorithms Design and Analysis**			**Module Delivery**	
Module Type	**Core**			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	**CSC24119**				
ECTS Credits	**7**				
SWL (hr/sem)	**175**				
Module Level	2		**Semester of Delivery**	3	
Administering Department	Computer science		**College**	College of science	
Module Leader	Dr. Nasreen jawad kadhim		**e-mail**	nasreen.kadhim@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Lecturer		**Module Leader's Qualification**	Ph.D.	
Module Tutor	Dr. Tariq Zaid Hammood		**e-mail**	tarik.z @sc.uobaghdad.edu.iq	
Peer Reviewer Name			**e-mail**		
Scientific Committee Approval Date	11/6/2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	CSC23113	**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	Algorithms are at the heart of computer science, and the subject has both practical and intellectual applications. This course involves the principles of algorithm design and analysis of algorithms. The emphasis is on choosing appropriate data structures and designing correct and efficient

	algorithms to operate on these data structures. Topics covered in this course include asymptotic analysis, algorithm design paradigms and applications in sorting and searching, randomized algorithms, amortized analysis, and data structures (heaps and hash tables) and their analysis. The objective of this course can be summarized as follows: • Analyze the asymptotic performance of algorithms. • Exhibit knowledge of important algorithms and data structures. • Apply important algorithmic design paradigms (top down design, divide and conquer) to sorting and searching and learn the methods of analysis. • Apply the theoretical knowledge in practice through the practical part of the course. • Create efficient algorithms in common problem-solving scenarios.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Students who complete the course will have demonstrated the ability to do the following: • Learn how to analyze algorithms and estimate their complexities. • Compare different data structures and pick an appropriate data structure for a given algorithm design situation. • Learn the principles of algorithm design. Describe the divide-and-conquer paradigm and explain algorithms that employ this paradigm. Describe the performance of divide-and-conquer algorithms. • Compare between different data sorting and searching algorithms and pick an appropriate algorithm for a given data. • Explain the different ways to analyze randomized algorithms with an application to algorithms that employ randomization. • Explain what amortized running time is. Describe the different methods of amortized analysis. Perform an amortized analysis. • Explain what heaps and hash tables are and describe their performance.
Indicative Contents المحتويات الإرشادية	**1. Introduction to Algorithms:** Definition of algorithms and their Design Techniques **2. Algorithm Analysis** • Time and Space Complexity: Assessing the efficiency of algorithms in terms of the number of operations and analyzing the memory usage of algorithms. • Big-O Notation: Describing the upper bound on the asymptotic complexity of algorithms. • Best-case, Average-case, and Worst-case Analysis: Understanding performance under different scenarios. **3. Sorting algorithms** • Description and analysis of Greedy sorting algorithms (Comparison based Sorting Algorithms: Selection Sort, Bubble

	Sort, Insertion Sort and Shell Sort algorithm also, non-comparison based: Bin Sort, Counting Sort, and Radix Sort algorithm). ● Description and analysis of sorting algorithms that follow divide and conquer paradigm (Quick Sort algorithm and Merge Sort algorithm). ● Heap sort algorithm: description and analysis. **4. Description and analysis of Searching algorithms:** ● Linear search algorithm ● Binary search algorithm ● Hashing Techniques **5. Advanced analysis techniques** Probabilistic analysis, randomized algorithms, and amortized analysis.
Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Delivering lectures to introduce and explain essential concepts, principles, and theories related to algorithm design and analysis. This helps students build a strong foundation of knowledge. 2. Giving hands-on programming exercises helps them to apply the learned concepts and algorithm design and analysis method. This helps their understanding and enhances their programming skills. 3. Providing code walkthroughs and examples demonstrating the analysis of various algorithms. 4. Encouraging group projects and activities among students. This promotes teamwork and enhances understanding. 5. Regular evaluations and examinations help gauge students' understanding and progress. These can include quizzes, assignments, and exams that assess theoretical knowledge and practical application of data structures. 6. Recommending textbooks, internet resources, and supplementary references might help students study more effectively.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	64	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	111	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	7
Total SWL (h/sem)	**175**		

الحمل الدراسي الكلي للطالب خلال الفصل	

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	10% (10)	5 and 10	LO #1, #2, #3, #4 and #6, #8, #9
	Assignments	2	10% (10)	6 and 12	LO #1, #2, #3 and #8, #9, #10
	Projects / Lab.	1	15% (15)	Continuous	All
	Report	1	5% (5)	13	LO #5, #8, #10 and #12
Summative assessment	**Midterm Exam**	2hr	10% (10)	7	LO #1 - #6
	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 Algorithms and their importance in Computer Science
Week 2	Analysis of algorithms: Understanding Time complexity and Space complexity (best, worst, and average cases)
Week 3	Asymptotic Analysis: "Big-O" Notation
Week 4	Asymptotic Analysis: Running-Time calculations, the running time of a program, choosing the suitable data structure, and Examples of calculating the time complexity for different algorithms
Week 5	Sorting and Order Statistics: Comparison based sorting algorithms; Sorting in quadratic time, selection sort, bubble sort, shell sort, and insertion sort, and

	complexity analysis
Week 6	Sorting and Order Statistics: : Non-comparison based sorting algorithms; Sorting in linear time, Bin sort, counting sort, radix sort, bucket sort and complexity analysis
Week 7	**Midterm Exam**
Week 8	Sorting and Order Statistics: Heap sort and its complexity analysis
Week 9	Divide-and-conquer Paradigm: Defining the divide-and-conquer paradigm, description of Merge Sort algorithm, analysis of Merge Sort algorithm
Week 10	Divide-and-conquer Paradigm: Description of Quick Sort algorithm, analysis of Quicksort algorithm
Week 11	Searching and Order Statistics: Linear search algorithm, binary search algorithm, complexity analysis of searching algorithms
Week 12	Hashing: Hash tables, hash functions, collision resolution methods, analysis of hashing
Week 13	Probabilistic Analysis and Randomized Algorithms
Week 14	Amortized Analysis
Week 15	**Final Exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Lab 1: Measuring the time complexity of the primitive operations in simple C++ codes
Week 2	Lab 2: Implementation of solutions for different problems and computing the running-Time in addition to the comparison in terms of running time
Week 3	Lab 3: Speeding up the implementation of different problem solutions by picking the suitable data structures
Week 4	Lab 4: Implementation of selection sort, bubble sort, and measuring the running time
Week 5	Lab 5: Implementation of improved selection sort, improved bubble sort, insertion

	sort, and measuring the running time
Week 6	Lab 6: Implementation of Bin sort, counting sort, radix sort, bucket sort, and measuring the running time
Week 7	**Midterm Exam**
Week 8	Lab 7: Implementation of heap sort and measuring the running time
Week 9	Lab 8: Implementation of merge sort and measuring the running time
Week 10	Lab 9: Implementation of quick sort and measuring the running time
Week 11	Lab 10: Implementation of the linear search algorithm, the binary search algorithm and measuring the running time
Week 12	Lab 11: Implementation of hashing and measuring the running time
Week 13	Lab 12: Implementation of randomized algorithms and measuring the running time
Week 14	Lab 13: Implementation of amortized analysis
Week 15	**Final Exam**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, Introduction to Algorithms, 3rd edition, The MIT Press Cambridge, Massachusetts London, England, 2009.	Yes
Recommended Texts	Mark Allen Weiss, Data Structures and Algorithm Analysis in C++, 4th edition, Pearson Education, Inc., publishing as Addison-Wesley, United States of America, 2014.	Yes
Websites	https://ocw.mit.edu/courses/6-046j-design-and-analysis-of-algorithms-spring-2015/ https://www.cs.ox.ac.uk/teaching/courses/2020-2021/algdesign/	

	https://online.stanford.edu/courses/soe-ycsalgorithms1-algorithms-design-and-analysis-part-1

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.				

Second Grade – Second Semester

GENERAL-PURPOSE LANGUAGE MODULE DESCRIPTOR FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**General-Purpose Language** لغة للأغراض العامة			**Module Delivery**	
Module Type	**Elective**			☐ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	**CSC24120**				
ECTS Credits	**7**				
SWL (hr/sem)	**175**				
Module Level	2		**Semester of Delivery**	4	
Administering Department	Computer Science		**College**	College of Science	
Module Leader	Dr. Rawaa Dawoud Hassan		**e-mail**	rawaa.hassan@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Assistant Professor		**Module Leader's Qualification**	PhD	
Module Tutor	Dr. Najlaa Mohammed Hussein		**e-mail**	najlaa.alkhafaji@sc.uobaghdad.edu.iq	
Peer Reviewer Name			**e-mail**		
Review Committee Approval	11/6/2024		**Version Number**	1.0	

Relation With Other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	CSC23114	**Semester**	3
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	This module aims to provide students with a comprehensive introduction to the Python programming language, equipping them with essential programming skills and a solid foundation for further studies in computer science.

Module Learning Outcomes مخرجات التعلم للمادة الدراسية	By the end of the module, students should be able to: **1. Understand the Basics:** ● Demonstrate a clear understanding of Python's syntax, basic data types, and operations. ● Apply fundamental programming concepts, including variables, loops, and conditional statements. **2. Develop Programming Logic:** ● Create algorithms and solve simple problems using Python. ● Understand and implement control flow mechanisms, such as if statements and loops. **3. Manipulate Data Structures:** ● Work with fundamental data structures, including lists, strings, and tuples. ● Perform operations on data structures, such as indexing, slicing, and iteration. **4. Build and Use Functions:** ● Design and implement functions for code modularity and reusability. ● Understand the concept of scope and practice good function design. **5. Introduction to Object-Oriented Programming (OOP):** ● Grasp the basics of OOP, including classes, objects, inheritance, and polymorphism. ● Apply OOP principles to solve simple programming problems. **6. Project Work and Collaboration:** ● Collaborate with peers on small projects to foster teamwork and communication. ● Complete a capstone project that demonstrates a holistic understanding of Python programming concepts.
Indicative Contents المحتويات الإرشادية	<u>Part A: Introduction and Fundamentals [20 hrs.]</u> Introduction to Python and setup; Basic syntax, variables, and data types; Control flow: conditionals and loops; Functions and modular programming; Lists, tuples, and strings; Dictionaries and Lists; Introduction to indexing and slicing; File handling and exception handling; Introduction to modules and libraries; Overview of time and space complexity. <u>Part B: Object-Oriented Programming (OOP) and Problem-Solving [20 hrs.]</u> Basics of object-oriented programming; Classes, objects, attributes, and methods; Inheritance and polymorphism; Advanced OOP concepts:

	encapsulation, abstraction; Developing problem-solving skills through exercises; Applying OOP in real-world scenarios; Further exploration of advanced concepts. Part C: Project Work, Review, and Conclusion [20 hrs.] Introduction to the capstone project; Forming project groups and selecting topics; Final project presentations; Code reviews, best practices, and reflections; Course summary and discussion of next steps.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	Here are some learning and teaching strategies that can be employed in this course: **1.** Encourage students to actively participate in class discussions, ask questions, and share their thoughts. **2.** Implement pair programming or group coding exercises to promote collaboration. **3.** Provide coding challenges with varying levels of difficulty to cater to different skill levels. **4.** Integrate real-world applications and projects into the curriculum. This could be a cumulative project that spans several weeks. **5.** Foster a collaborative learning environment by encouraging students to collaborate on coding projects. **6.** Utilize interactive resources such as online coding platforms, Python notebooks, and interactive coding environments. **7.** Visual representations of data structures, algorithms, and code flow can enhance comprehension. **8.** Encourage students to share their code during these walkthroughs to promote learning from each other. **9.** Conduct regular assessments, quizzes, and coding tests to gauge understanding. **10.** Share additional resources such as online tutorials, coding challenges, and relevant articles to encourage self-directed learning. **11.** Suggest Python-related books and documentation for deeper exploration. **12.** Emphasize the importance of documentation. Encourage students to write clear and concise comments in their code. **13.** Teach problem-solving strategies specific to programming, such as breaking down problems into smaller steps, pseudocode writing, and debugging techniques.

	14. Offer regular office hours for one-on-one or small-group discussions. **15.** Keep the course content updated with the latest Python features and best practices.

Student Workload (SWL) الحمل الدراسي للطالب			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	64	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	111	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	7
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	**175**		

Module Evaluation تقييم المادة الدراسية					
		Time/ Number	**Weight (Marks)**	**Weekly Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	10% (10)	5 and 10	1, 2, 3, and 4
	Assignments	2	10% (10)	2 and 12	1, 2, 3, 4, and 5
	Projects / Lab.	1	10% (10)	Continuous	All
	Report	1	10% (10)	10	All
Summative assessment	**Midterm Exam**	2 hr	10% (10)	7	1, 2, 3, and 4
	Final Exam	4 hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	**Introduction to Python** • Overview of Python and its applications • Setting up the Python environment (IDEs) • Running the first Python program (Hello World)

Week 2	**Programming Basics** ● Variables, data types, and basic operations ● Input and output (print(), input()) ● Comments and code structure ● Logical operators and basic loops
Week 3	**Strings and String Manipulation** ● String operations and methods ● Formatting strings ● String exercises
Week 4	**Lists and Tuples** ● Creating and manipulating lists ● Indexing and slicing ● Tuple introduction
Week 5	**Control Flow** ● Conditional statements (if, elif, else) ● While loops ● Loop control statements (break, continue) ● Practical exercises
Week 6	**Dictionaries and Sets** ● Introduction to dictionaries ● Key-value pairs and how they differ from lists. ● Creating dictionaries and adding key-value pairs ● Using loops to iterate keys, values, and items ● Definition and characteristics of sets ● How sets differ from other data structures ● Performing set operations such as union, intersection, difference, and symmetric difference. ● Advanced Set Operations
Week 7	**Mid-term exam**
Week 8	**Functions** ● Function definition and invocation ● Parameters and return values ● Scope and lifetime of variables ● Function exercises
Week 9	**Advanced Concepts** ● Common Python libraries overview (e.g., random, math) ● Time and space complexity ● Application of advanced concepts
Week 10	**Introduction to OOP** ● Understanding classes and objects

	• Defining classes and creating objects • Attributes and methods • Inheritance and polymorphism • Encapsulation and abstraction
Week 11	**File Handling** • Reading and writing files • Exception handling: try, except, finally • Introduction to modules and libraries
Week 12	**Best Practices and Code Style** • Developing problem-solving skills through practical exercises and coding challenges • Applying learned concepts to real-world scenarios • Understanding PEP 8 and other coding style guidelines • Writing clean, readable, and well-documented code
Week 13	**Collaborative Coding Projects** • Collaborating with peers on small projects to foster teamwork and communication. • Completing a capstone project that demonstrates a holistic understanding of Python programming concepts.
Week 14	**Final Project Presentation and Assessment** • Final project presentations and assessment • Concluding the module with a summary of learned skills and potential next steps.
Week 15	**Final exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	**Install and Setup.** Here are the main steps to install and set up Python: Download Python; Install Python; Verify the installation; Install a code editor
Week 2	**Stepping into the world of Python.** Our First Python Program; Printing in Python; Variables and Types; Numeric Data Types in Python; Numeric Operators; Expressions; Operator Precedence; Introduction to Blocks and Statements; if Statements; elif
Week 3	**Strings**. The str String Data Type; Negative Indexing in Strings; String Operators; String Replacement Fields; String Formatting; f-strings; Slicing
Week 4	**Lists and Tuples.** Introduction to Sequence Types; Lists; Immutable Objects; Mutable Objects; Binding

	Multiple Names to a List; Common Sequence Operations; Appending to a List; Iterating Over a List; The enumerate Function; Removing Items from a List; Tuples; Tuples are Immutable; Unpacking a Tuple; Nested Tuples and Lists
Week 5	**Program flow control in Python** More on if, elif and else; Conditional Operators; Using and, or, in Conditions; Boolean Expression True and False; in and not in; for loops; Stepping through a for loop; continue; break; while loops; Break in a while loop;
Week 6	**Dictionaries and Sets-part 1** Defining a Dictionary; Iterating over a dictionary; Adding items to a dictionary; Changing values in a dictionary; Removing items from a dictionary; Using `in` with a dictionary; Using a list with a dictionary; Using several dictionaries together
Week 7	**Mid-term exam**
Week 8	**Dictionaries and Sets-part 2** Python sets; Implications of sets being unordered; set membership; Testing set membership is fast; Adding items to a set; Set union; Set intersection; Set difference
Week 9	**Functions.** Defining a function; Parameters and arguments; Returning values; Returning None; Keyword arguments; Function annotations and type hints; Functions calling functions
Week 10	**Modules and Functions**. Modules and import; The standard Python library; Importing Techniques; The Random Module and Import
Week 11	**OOP1.** Object Orientated Programming and Classes; Instances, Constructors, Self and more; Class Attributes; Methods Part 1; Methods Part 2
Week 12	**OOP2.** Inheritance; Subclasses and Overloading; Overriding Methods; Polymorphism
Week 13	**Reading and Writing Files in Python.** Files and directories; Introduction to the command prompt or terminal; Paths; Text files; Reading from a text file; read, readline and readlines; Writing data to a text file
Week 14	**Final Project Presentation and Assessment**
Week 15	**Final Exam**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	**1.** Beazley, D., & Jones, B. K. (2013). *Python Cookbook*. O'Reilly Media.	No

	2. Matthes, E. (2015). *Python Crash Course*. No Starch Press.	
Recommended Texts	Sweigart, A. (2015). *Automate the Boring Stuff with Python*. No Starch Press.	No
Websites	Coursera Python for Everybody: A specialization by the University of Michigan on Coursera. (https://www.coursera.org/specializations/python)	

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	مقبول بقرار	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required

Note:

NB 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 failures" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.

COMPILERS MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Compilers**			**Module Delivery**	
Module Type	**Core**			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	**CSC24121**				
ECTS Credits	**6**				
SWL (hr/sem)	**150**				
Module Level	2		**Semester of Delivery**	4	
Administering Department	Computer Science		**College**	College of science	
Module Leader	Assist Prof Wijdan Abdul Ameer Hassan		**e-mail**	wijdan@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Assistant Prof.		**Module Leader's Qualification**	MSC	
Module Tutor	Yasmin Alaa Hassan		**e-mail**	Yasmin.a@sc.uobaghdad.edu.iq	
Peer Reviewer Name	Name		**e-mail**	E-mail	
Scientific Committee Approval Date	2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	CSC23116, CSC12107	**Semester**	3, 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	1- Compilers play a vital role in the design and implementation of programming languages. By studying compilers, the students gain insights into how programming languages are structured, how syntax and semantics are defined, and how languages are transformed into executable code.

	2- Understanding the principles and techniques used by compilers allows programmers to write code that can be optimized effectively, resulting in faster and more efficient programs. 3- Compiler course cover topics such as lexical analysis, parsing, and semantic analysis. These concepts are crucial for understanding how compilers detect and report errors in source code. 4- Compilers employ a range of optimization techniques to improve the performance of programs. By studying compilers, the students learn these optimization strategies and apply them in code, making it run faster and consume fewer resources. 5- Understanding program Execution.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1- Understand the basics and principles of the compiler work. 2- Know how to design compilers. 3- Know the importance and function of each of the six compiler stages. 4- Linking the basics of the subject with other knowledge subjects. 5- Know Debugging and Error Handling. 6- Know Programming Skills. 7- Know Performance Optimization
Indicative Contents المحتويات الإرشادية	Indicative content includes the following Part1 Language Processing System, [2 hrs]. Programming Languages, translator [2 hrs]. Part2 ● Introduction to compiler, [2 hrs]. ● Lexical Analysis (scanner), [5 hrs]. ● Syntax analyzer (parser) , [8 hrs]. ● Semantic analysis, [4 hrs]. ● intermediate code Generation)IR) , [5 hrs]. ● Code Optimization, [5 hrs]. ● Code Generation, [4 hrs]. Part3 programming in C++ Lexical analyzer program, [14] Lexical analyzer using file, [4] Syntax analyzer program, [8]

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	The main strategies that will be adopted in delivering this module are: 1. Power point presentation (Data show). 2. Explanation on the whiteboard using different color markers. 3. Discussions with the student during teaching. 4. Interaction with students through daily problems practice through lecture.

	5. Use a computer in the Lab to implement the compiler program like lexical and syntax analyzer. 6. Prepare reports that develop critical thinking for students. 7. Submit assignment that develop student learning.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	64	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	86	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	5.8
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	**150**		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	10% (10)	5 and 10	LO #3, #4 and #5, #6 and #7, #8 and #9, #10 and #11, #12
	Assignments	2	10% (10)	2 and 12	LO #3, #4,#5 and #7,#8,#9
	Projects / Lab.	1	10% (10)	Continuous	All
	Report	1	10% (10)	-	All
Summative assessment	**Midterm Exam**	1	10% (10)	7	LO #1 - #7 and LO #8 - #14
	Final Exam	4hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري

	Material Covered
Week 1	● Language processing system ● Programming languages ● Translator
Week 2	● Introduction to compiler ● Phases of compiler
Week 3	● Lexical Analysis(scanner) ● Recognition of token ● Lexical Error ● Symbol table
Week 4	● Syntax analyzer(parser) ● Syntax error handling ● Strategies for error recovery
Week 5	● Context free grammar, parse tree ● Ambiguity, left recursion, and left factoring
Week 6	● Top-down parsing ● Predictive parsing
Week 7	● Mid-term Exam
Week 8	● Bottom-up parsing, ● Shifts reduce parsing
Week 9	● Lr parsing
Week 10	● Semantic analysis ● Type system
Week 11	● Intermediate code Generation (IR) ● Implementation of three address code
Week 12	● Code optimization
Week 13	● Code optimization methods
Week 14	● Code generation
Week 15	● Final exam

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Lab1: review programs about string
Week 2	lexical analyzer program Lab2: split string into tokens
Week 3	Lab3: write a program with function to recognize keyword
Week 4	Lab4: write a program with function to recognize identifier
Week 5	Lab5: write a program with function to recognize digits

Week 6	Lab6: write a program with functions to recognize Arithmetic and logical operators
Week 7	Mid-term Exam
Week 8	Lab7: write a program with functions to recognize comparison operators and Assignment operators.
Week 9	Lab8: write a program of lexical analyzer with solve a problem of multi space between tokens
Week 10	Lab9: write a lexical program using multiline
Week 11	Lab10: write a program of Lexical analyzer by using a file
Week 12	Lab11: write a program of LR parsing table with function shift
Week 13	Lab12: complete a program of LR parsing with function Reduce
Week 14	Lab13: another example of LR parsing program
Week 15	Final Exam

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	1-Compilers Principles, Techniques, & Tools by Alfred V. Aho, 1986. 2- The designs and construction of compilers by JOHN WILEY & SONS,1981.	Yes
Recommended Texts	Compilers Principles, Techniques, & Tools by Alfred V. Aho, 2016.	No
Websites	Many references from website	

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 failures" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.

WEB DESIGN AND PROGRAMMING MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Web design and programming**			**Module Delivery**	
Module Type	**Elective**			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	**CSC24022**				
ECTS Credits	6				
SWL (hr./Sem)	150				
Module Level		2	**Semester of Delivery**		4
Administering Department		Type Dept. Code	**College**	Type College Code	
Module Leader	Asst. Prof. Dr. Adnan J. Jabir		**e-mail**	Adnan.jabir@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title		Asst. Prof.	**Module Leader's Qualification**		Ph.D.
Module Tutor	Dr. Maysa I Abdulhussain		**e-mail**	Maysaa.i@sc.uobaghdad.edu.iq	
Peer Reviewer Name			**e-mail**		
Scientific Committee Approval Date		11/06/2024	**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	Understanding **Web Technologies**: • Grasping the role and purpose of HTML, CSS, and JavaScript in web development. • Understanding how these technologies work together to create web pages and applications.

	HTML **(HyperText Markup Language)**: • Learning the basic structure of an HTML document. • Mastering the use of HTML tags and attributes to create and organize content. • Understanding semantic HTML to enhance accessibility and SEO. • Learning to embed images, videos, and other multimedia. CSS **(Cascading Style Sheets)**: • Understanding the basics of CSS syntax and selectors. • Learning how to style HTML elements with properties like color, font, layout, and spacing. • Mastering the use of advanced CSS features such as Flexbox, Grid, and animations. • Understanding responsive design principles to create web pages that look good on all devices. **JavaScript:** • Understanding the basics of JavaScript syntax and programming concepts (variables, data types, operators, control structures, functions). • Learning how to manipulate the Document Object Model (DOM) to create interactive web pages. • Mastering event handling to respond to user actions.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Vocational Outcomes: Upon successful completion of this course, the student will: **Fundamental Understanding**: • Demonstrate an understanding of the basic concepts of web technologies, including how the internet and web browsers work. • Explain the roles of HTML, CSS, and JavaScript in web development. **HTML Proficiency**: • Write well-structured HTML documents using appropriate tags and attributes. • Utilize semantic HTML to create accessible and SEO-friendly web pages. • Embed multimedia elements like images, videos, and audio files into web pages. **CSS Proficiency**: • Apply CSS to style and layout web pages, using selectors, properties, and values effectively. **JavaScript Proficiency**: can add features such as form validation, animations, pop-ups, carousels, and menus to websites, making them more engaging and intuitive. • **Bootstrap fundamentals**: Bootstrap includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many other, as well as optional JavaScript plugins.

	Project Development: using HTML, CSS, Javascript and Bootstrap to develop a full front-end project.
Indicative Contents المحتويات الإرشادية	1. course overview, 2. learning outcomes, 3. expectations What is a web site – 4. Concept Maps Finding a Web Host Module 5. Planning Information Architecture Module 6. The Markup Web Basics HTML Module 7. CSS Basic, CSS for text formatting, Working with Layout, Working with Images in CSS. 8. JavaScript fundamentals 9. Bootstrap framework fundamentals

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	The main strategies that will be adopted in delivering this module are: 1. Power point presentation (Data show). 2. videos, is posted online and available so that students can read or listen to it whenever and as many times it is desired or needed 3. Lecture Document: i.e., a PDF file containing between 25 and 40 PowerPoint slides, with pointers to required or recommended textbooks, posted online 4. Lecture Video: i.e., one or more video recordings of an explanation of the lecture slides, posted together with the lecture document 5. Explanation on the whiteboard using different color markers. 6. Discussions with the student during teaching. 7. Interaction with students through daily problems practice through lecture. 8. Solve different problems with more exercises. 9. Submit assignment that develop student learning.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	64	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	86	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	5.8
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	**150**		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	10% (10)	5 and 10	LO #1, #2 and #10, #11
	Assignments	2	10% (10)	2 and 12	LO #3, #4 and #6, #7
	Projects / Lab.	1	10% (10)	Continuous	All
	Report	1	10% (10)	13	LO #5, #8 and #10
Summative assessment	**Midterm Exam**	2hr	10% (10)	7	LO #1 - #7
	Final Exam	3hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
Week #	**Material Covered**
Week 1	Introduction to web design: The fundamental elements of web page structure and basic concepts of web development are introduced through this subject.
Week 2	Introduction to HTML: It includes the history of HTML and essential syntax and concepts revolving around HTML, the basic tags of HTML. Html-grouping using div span. HTML-Lists: The definition of HTML lists, their significance, types of HTML lists like ordered and unordered lists, etc.
Week 3	HTML Images: the image tag and its attributes. HTML Hyperlink: focuses on Uniform Resource Locator (URL) and related concepts. It also talks about URL encoding and the importance of hyperlinks in web pages. HTML Tables: It focuses on arranging data in tabular format in web pages.
Week 4	HTML Forms: Integrating forms into web pages and their significance in HTML are included in this subject. How Forms Work, From Data Entry to Response, The Action Attribute, The Method Attribute. The POST and GET methods.
Week 5	CSS fundamentals: introduction to CSS and its importance in HTML web page design, How CSS Work, CSS rules. The CSS syntax, single style sheets, multiple style sheets.
Week 6	CSS Background image, text fonts, List, tables, display position, and floats.

Week 7	Mid Exam
Week 8	Fundamentals of JavaScript: JavaScript methods, JavaScript Console, Error messages & troubleshooting JavaScript. Variables, Array, Conditional statements, and loops.
Week 9	Selecting HTML elements with getElementById(), Manipulating selected elements Getting & setting properties (such as adding a class). Defining & calling functions Defining parameters & passing arguments Using an event listener.
Week 10	Event handling in JavaScript, working with cookies, working DOM objects.
Week 11	Introduction to Bootstrap: Buttons, Images.
Week 12	Bootstrap Tables, Grids, Lists.
Week 13	Bootstrap Dropdown lists, Navbars and menus.
Week 14	Final Semester Projects
Week 15	Final Exam

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
Week #	**Material Covered**
Week 1	HTML basic tags, LINKS, Target attribute, Absolute URL vs. Relative URL, Image as link, Create Bookmark, and HTML Lists
Week 2	Images and HTML tables.
Week 3	HTML Forms.
Week 4	HTML Forms Actions.
Week 5	CSS Syntax, Properties, Values.
Week 6	CSS Backgruond images, Lists, Tables and display positions.
Week 7	Mid Exam
Week 8	Introduction, JS change CSS style and HTML, JS show and hide elements, JS change content
Week 9	JS in body, head and external file, JS Document Object Model (DOM).
Week 10	Introduction to Bootstrap (BS), What is Responsive Web design, Advantages, Examples.
Week 11	Containers, Fixed Containers, Fluid Containers, Button Styles.
Week 12	Bootstrap Tables
Week 13	Bootstrap Grids, Lists, Dropdown lists.
Week 14	Navbars and menus.

Week 15	Final Exam

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Krrish Ghindani 2022, Everything You Need To Know About Front-End: A Step-by-Step approach to getting started with Programming.	Yes
Recommended Texts	Anon Antonov, 2016. **Front-end developer book big nerd ranch**. **Big Nerd Ranch Guides publisher**	Yes
Websites	https://www.w3schools.com/	

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.

ARABIC LANGUAGE II MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Arabic Language II**			**Module Delivery**	
Module Type	**Basic**			☒ **Theory** ☐ **Lecture** ☐ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	UOB201				
ECTS Credits	**2**				
SWL (hr./Sem)	**50**				
Module Level	2		**Semester of Delivery**	4	
Administering Department	Type Dept. Code		**College**	Type College Code	
Module Leader	**Dr. Leqaa faleh owdaa**		**e-mail**	leqaa.falih@ircoedu.uobaghdad.edu.iq	
Module Leader's Acad. Title	Lecturer		**Module Leader's Qualification**	Ph.D.	
Module Tutor	Name (if available)		**e-mail**	E-mail	
Peer Reviewer Name	**Assistant lecturer. A'laa Sabah Hammood**		**e-mail**	alaa.sabah@sc.uobaghdad.edu.iq	
Scientific Committee Approval Date	11/06/2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module		**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	1- تهدف إلى تنمية روح الاعتزاز باللغة العربيَّة للمحافظة على الهوية العربيَّة. 2- تهدف إلى تأهيل الطلبة بالمعارف والمخرجات الخاصة علم النحو، والصرف، والإملاء؛ لتمكنه من الكتابة الصحيحة والتعبير السليم وتقويم لسانه. 3- تهدف إلى تنمية ذوق الطالب الأدبي وإثراء تحصيله وإغناء زاده من الفكر العربي والإسلامي. 4- تهدف إلى تطوير مهارات الطلاب اللغويَّة التي تؤهلهم للإبداع المتميز. 5- تهدف إلى تنمية مهارات التحدث بـ (اللغة العربيَّة). 6- تهدف إلى الارتقاء بمستوى الطلبة من الجانب المهني والبحثي.

Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1- التعرف على الظواهر اللغويَّة كونها إحدى خصائص اللغة العربيَّة التي تميزت بها. 3- التعرف على قواعد كتابة الألف اللينة في آخر الكلمة، والتمييز بين الألف الطويلة والقصيرة عن طريق ذكر مواضع كل منهما وتوضيح ذلك بذكر الأمثلة. 4- التعرف على الاستثناء من حيث تعريفه وأدواته وحكمه وبيان ذلك بالأمثلة التوضيحية. 5- التعرف على الحال من حيث تعريفه وحكمه وبيان ذلك بالأمثلة التوضيحية. 6- التعرف على التمييز من حيث تعريفه وحكمه وبيان ذلك بالأمثلة التوضيحية. 7- التعرف على المفاعيل الخمسة وبيان أحكامها بكونها من منصوبات الأسماء وبيان ذلك بالأمثلة التوضيحية. 8- التعرف على حروف الجر بكونها من مجرورات الأسماء، والتمييز بين معانيها، وبيان حكمها مع توضيح ذلك بذكر الأمثلة. 9- التعرف على الاسم المذكر والاسم المؤنث من حيث تعريفهما، وأقسامهما مع ذكر الأمثلة التوضيحية. 10- التمييز بين اللام الشمسية واللام القمرية من حيث النطق والكتابة، وذلك من حيث تعريفهما ومعرفة حروف كل منهما. 11- التعريف بحروف الحذف والزيادة في الكلمة، وبيان ذلك بالأمثلة التوضيحية. 12- تعريف الطالب بمواضع الوقف في اللغة العربيَّة لما فيه من أهمية لإيصال المعلومات إلى المتلقي بشكل صحيح فضلاً عن تمكنه من فهم النص فهماً صحيحاً. 13- تمكين الطالب من معرفة المواضع الإعرابية للكلمات داخل النص، ومعرفة معاني بعض الكلمات، فضلاً عن استخراج الأهداف منه. 14- التعرف على الشاعر المتنبي بكونه من شعراء العصر العباسي. 15- التعرف على الشاعرة نازك الملائكة بكونها إحدى رواد الشعر الحر الحديث في العراق.
Indicative Contents المحتويات الإرشادية تتضمن الكلمات المفتاحية المهمة للمحاضرات	• الظواهر اللغوية: التَّرادف، المشترك اللفظي، التَّضاد. • الألف اللينة: الألف الطويلة، الألف القصيرة. • الاستثناء. • الحال. • التمييز. • المفاعيل الخمسة: منصوبات الأسماء، المفعول به، المفعول فيه، المفعول المطلق، المفعول لأجله، المفعول معه. • حروف الجر: مجرورات الأسماء، معاني حروف الجر. • الاسم المذكر والمؤنث: تعريف الاسم المذكر، والاسم المؤنث، أقسام الاسم المذكر والمؤنث. • اللام الشمسية، اللام القمرية، الحذف والزيادة. • الوقف. • سورة لقمان، إعراب سورة لقمان، تفسير سورة لقمان. • الشاعر المتنبي: حياته، مؤلفاته. • - الشاعرة نازك الملائكة: حياتها، مؤلفاتها.
Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	الاستراتيجية الرئيسية التي سيتم تبنيها في تقديم هذه الوحدة هي تشجيع الطلاب على المشاركة في التمارين والتطبيقات النحوية والإملائية، مع تحسين مهارات التفكير والتحليل في الوقت نفسه. ويتم تحقيق ذلك عن طريق الفصول والبرامج التعليمية التفاعلية والنظر في أنواع التطبيقات التي تتضمن بعض الأنشطة التي تهم الطلبة.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/Sem) الحمل الدراسي المنتظم للطالب خلال الفصل	33	Structured SWL (h/w) الحمل الدراسي المنتظم للطالب أسبوعيا	**2**
Unstructured SWL (h/Sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	17	Unstructured SWL (h/w) الحمل الدراسي غير المنتظم للطالب أسبوعيا	**1.13**
Total SWL (h/Sem) الحمل الدراسي الكلي للطالب خلال الفصل	50		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Weekly Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	**2**	**10% (10)**	**3, 9**	**LO #1 ,2 and 8**
	Assignments	**2**	**10% (10)**	**5, 8**	**LO # 4 and 6**
	Projects / Lab.	**1**	**10% (10)**	**Continuous**	**All**
	Report	**1**	**10% (10)**	**10**	**LO #,1,2,3,4,5,6,8,9,10,11,12,13and14**
Summative assessment	**Midterm Exam**	**2hr**	**10% (10)**	**7**	**LO # 1-6**
	Final Exam	**3hr**	**50% (50)**	**16**	**All**
Total assessment			**100% (100 Marks)**		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	الظواهر اللغويَّة: الترادف، المشترك اللفظي، التضاد.
Week 2	قواعد كتابة الألف اللينة في آخر الكلمة.
Week 3	الاستثناء.
Week 4	الحال.
Week 5	التمييز.
Week 6	المفاعيل الخمسة: المفعول به، المفعول فيه، المفعول المطلق، المفعول لأجله، المفعول معه.
Week 7	امتحان نصف الفصل.

Week 8	حروف الجر ومعانيها.
Week 9	الاسم المذكر والمؤنث.
Week 10	الحروف من حيث النطق والكتابة: اللام الشمسية والقمرية، الحذف والزيادة.
Week 11	الوقف.
Week 12	نص من سورة لقمان.
Week 13	الشاعر المتنبي.
Week 14	الشاعرة نازك الملائكة.
Week 15	امتحان ختامي.

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	القرآن الكريم • الأدب العربي في العصر العباسي: د. ناظم رشيد. • إعراب القرآن وبيانه: محيي الدين درويش. • التطبيق الصرفي: د. عبده الراجحي. • تفسير الكشاف: للزمخشري. • جامع الدروس العربيَّة: الشيخ مصطفى الغلاييني. • ديوان المتنبي. • ديوان نازك الملائكة . • شرح ابن عقيل: ابن عقيل، تحقيق: محمد محي الدين عبد الحميد. • الشعر العراقي الحديث مرحلة وتطور: د. جلال الخياط • فقه اللغة العربيَّة وخصائصها: د. إميل بديع يعقوب. • المفيد في أحكام التلاوة والتجويد: القارئ الشيخ رافع العامري. • - الوجيز في اللغة العربيَّة: أ.د. محيي هلال السرحان	**Yes**

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.				

CRIMES OF THE DEFUNCT BAATH PARTY MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Baath Regime Crimes in Iraq**			**Module Delivery**	
Module Type	**Basic**			☒ **Theory** ☐ **Lecture** ☐ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	**UOB105**				
ECTS Credits	**2**				
SWL (hr/sem)	**50**				
Module Level	2		**Semester of Delivery**	4	
Administering Department	Type Dept. Code		**College**	Type College Code	
Module Leader	**Dr. Mohanad Ahmed Yaseen**		**e-mail**	mohannad.ahmed@sc.uobaghdad.edu.iq	
Module Leader's Acad. Title	Lecturer		**Module Leader's Qualification**	Ph.D.	
Module Tutor	Name (if available)		**e-mail**	E-mail	
Peer Reviewer Name	Name Dr farah diea hussain mubarak		**e-mail**	E-mail Farah@copolicy.uobaghdad	
Scientific Committee Approval Date	13/09/2023		**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	1- ان الأجيال الحالية لم تعيش فترة الدكتاتورية والكثير منهم لا يعرف معاناة الشعب والجرائم التي ارتكبها النظام المقبور 2- بيان مدى سوء حكم النظام الشمولي والذي لم يقتصر فقط على داخل العراق، بل على دول المجاور له 3- توعية الطلبة على الأضرار الكبيرة التي احدثها النظام البائد والجرائم التي ارتكبها بحق الشعب العراقي 4- أظهار الاضرار الاقتصادية والاجتماعية والتنموية التي أحدثها النظام السابق 5- بيان مدى وحشية النظام البائد والإعدامات الجماعية

أهداف المادة الدراسية	6- بيان الاساليب القمعية التي مارسها النظام البائد والتهجير القصري 7- كبح الحريات العامة وتدهور مستوى الاعلام والثقافة 8- توضيح الأضرار البيئية والزراعية التي ظهرت آثارها في السنوات السابقة والحالية 9- بيان مدى سوء حكم النظام الشمولي والذي لم يقتصر فقط على داخل العراق، بل على دول المجاورة ايضا 10- ان الهدف من تدريس هذه المادة لمعرفة تاريخ تلك الحقبة السوداء 11- الهدف من هذه المادة ان الحكم في العراق لن يدوم باستخدام أدوات العنف والقوة مهما كانت مفرطة والعراق يجب ان يحكم بنظام سياسي يحترم العراقيين ومعتقدات ودياناتهم والقومية التي ينتمون اليها وان يؤمن بالتعدد في المجتمع العراقي
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. التعرف على الجرائم النظام البائد في كبح الحريات العامة 2. دراسة الانظمة السياسية في العراق نبذة تاريخية 3. معرفة ابرز انتهاكات النظام البعثي للحقوق والحريات 4. معرفة اثر سلوكيات النظام البعثي المقبور على المجتمع العراقي 5. التوضيح للأجيال الحالية حقيقة حقبة تاريخية سوداء في تأريخ العراق المعاصر التي شهدت الظلم والاستبداد 6. الاطلاع على وحشية واستبداد وقمع النظام البائد للشعب العراقي 7. معرفة ان الظلم والاستبداد والحكم الدكتاتوري لن يدوم مهما كانت قسوته 8. تعليم الطلبة وارشادهم على النظام السياسي الصحيح لحكم هذا الشعب الطيب. والذي يجب ان يبتعد عن الدكتاتورية والظلم وان يكون مبنى على العدالة واحترام التعددية الدينية والمذهبية والقومية 9. توعية الطلبة الى حجم الدمار والتلوث البيئي الذي احدثته الحروب واستخدام اسلحة محرمة دوليا 10. بيان مدى قسوة النظام البعثي وقمعه للشعب والمقابر الجماعية التي ضمت رفاة آلاف الشهداء الأبرياء 11. توعية الطلبة الى ما قام به النظام السابق من تهجير ابناء هذا البلد وكفاءته العلمية والادبية
Indicative Contents المحتويات الإرشادية	يتضمن المحتوى الإرشادي ما يلي: مقدمة في البداية تتضمن نبذة تاريخية عن النظام السياسي في العراق من قبل بريطانيا وصولا حزب البعث المقبور الى السلطة وكذلك دراسة جرائم حزب البعث منذ توليه السلطة والعبث بها كذلك توضيح ما اصاب العراق من اثار وكوارث على يد هذا النظام الدكتاتوري المجرم الذي جسد اقسى انواع التعسف والظلم والطغيان والاستبداد كذلك ارشاد الطلبة الى ان الظلم والاستبداد يدمر الشعوب ويجر الويلات عليها وبيان الثأر التي حدثت نتيجة الحروب العبثية التي خلفت ورائها تدمير في كل مفاصل البلاد فدمرت البنى التحتية والتربة والمياه والسماء والشجار وكل شيء في هذه البلاد والتي كانت من افضل بلدان الشرق الوسط .كذلك تم تدمير حتى البيئة المائية من خلال تسريب النفط في حرب الكويت والخسائر الاقتصادية الهائلة وتضرر البار النفطية والبنى التحتية والصناعة وفرض حصار دمر البيئة الاجتماعية والاقتصادية التي لزلنا الى يومنا هذا نرفع اثار النظام البائد على الصعيد الدولي والداخلي.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	33	Structured SWL (h/w) الحمل الدراسي المنتظم للطالب أسبوعيا	2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	17	Unstructured SWL (h/w) الحمل الدراسي غير المنتظم للطالب أسبوعيا	1.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	50		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	**2**	**10% (10)**	**5 and 10**	**LO #1, #2 and #10, #11**
	Assignments	**8**	**10% (10)**	**2,3,6,8,10 and 12**	**LO #3, #4 and #6, #7**
	Projects / Lab.		**10% (10)**	**Continuous**	**All**
	Report	**1**	**10% (10)**	**13**	**LO #5, #8 and #10**
Summative assessment	**Midterm Exam**	**1hr:**	**10% (10)**	**8**	**LO #1 - #7**
	Final Exam	**3hr**	**50% (50)**	**16**	**All**
Total assessment			**100% (100 Marks)**		

Delivery Plan (Weekly Syllabus) المنهاج السبوعي النظري	
	Material Covered
Week 1	مقدمة عن انتهاكات الحقوق والحريات
Week 2	نبذة وصفية عن الانظمة السياسية في العراق
Week 3	انتهاكات النظام البعثي للحقوق والحريات العامة
Week 4	أثر سلوكيات النظام البعثي في المجتمع وتسلطه على الدولة
Week 5	أثر المرحلة الانتقالية في محاربة السياسة الاستبدادية
Week 6	الميدان النفسي والاجتماعي
Week 7	الدين والدولة
Week 8	امتحان نصف شهري
Week 9	عسكرة المجتمع والثقافة والاعلام
Week 10	أثر القمع والحروب على البيئة والسكان

Week 11	التلوث البيئي واستعمال الاسلحة المحرمة دوليا
Week 12	سياسة الارض المحروقة وتجفيف الاهوار
Week 13	المقابر الجماعية وتدمير البيئة الزراعية
Week 14	المحاضرة الأخيرة الممهدة للامتحان
	امتحان الشهري

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	**Available in the Library?**
Required Texts	منهاج جرائم حزب البعث البائد /2023جمهورية العراق/وزارة التعليم العالي والبحث العلمي/دائرة الدراسات والتخطيط	No

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 failures" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.