MODULE DESCRIPTION FORM

Module Information

Module Title	**Arabic Language**	**Module Delivery**
Module Type	**Supportive**	☒ **Lecture**
Module Code	**UOWA103**	
ECTS Credits	**2**	
SWL (hr/sem)	**50**	

Module Level	UG 1	**Semester of Delivery**	1
Administering Department	Information Technology	**College**	College of Science
Module Leader	Ayad Karim	**e-mail**	ayadalsalahi@uowa.edu.iq
Module Leader's Acad. Title	Professor	**Module Leader's Qualification**	Ph.D.
Module Tutor	Ayad Karim	**e-mail**	ayadalsalahi@uowa.edu.iq
Peer Reviewer Name	Asst. Lect Nabeel Sadeq	**e-mail**	nabeel.alshreefy@uowa.edu.iq
Scientific Committee Approval Date	**2024-12-09**	**Version Number**	V1

Relation with other Modules

Prerequisite module	-	**Semester**	-
Co-requisites module	-	**Semester**	-

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

جامعة وارث الانبياء
عمادة كلية العلوم

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Objective	The objectives of this course in Arabic Language are focused on introducing students to the fundamental rules of correct spelling and orthography, enabling them to avoid errors in written expression and to develop proficiency appropriate to their cultural and academic level. The main objectives include: 1. Understanding the fundamental principles of Arabic orthography necessary for academic study and future professional practice. 2. Developing the ability to apply these rules accurately and with ease, without the need for rote memorization. 3. Identifying and avoiding common spelling and linguistic errors. 4. Acquiring the ability to express ideas correctly and independently. 5. Recognizing the importance of Arabic language in personal and professional life, expanding linguistic repertoire, and diagnosing and addressing common difficulties and errors.
Module Learning Outcomes	Upon successful completion of this module, students will be able to: 1. Explain the fundamental rules of writing in Arabic. 2. Apply linguistic knowledge accurately and with confidence. 3. Analyze basic linguistic structures and simple texts relevant to daily life. 4. Demonstrate self-confidence and the ability to communicate effectively. 5. Correct and overcome their own basic linguistic errors.
Indicative Contents	1. Promoting the use of the Arabic language among members of society to open new horizons for linguistic development and support. 2. Addressing the challenges faced by society in education, particularly in the teaching of Arabic, and exploring effective solutions to enhance linguistic competence. 3. Utilizing modern communication tools such as the internet and digital resources in the learning process.

Learning and Teaching Strategies	
Strategies	A variety of simple yet effective strategies can be employed to enhance the learning process, making it both engaging and beneficial. These include: 1. Numbered Heads Together Strategy 2. Popsicle Sticks Strategy 3. Think–Pair–Share Strategy 4. Cube Strategy 5. Correct the Error Strategy 6. Hot Seat Strategy

Student Workload (SWL)			
Structured SWL (h/sem)	30	**Structured SWL (h/w)**	2
Unstructured SWL (h/sem)	17	**Unstructured SWL (h/w)**	1
Total SWL (h/sem)	47 + 3 Final Exam = 50		

Module Evaluation

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	3	4 marks	5,7,9	2,3,4
	Assignments	2	4 marks	3 ,5	2, 3
	Onsite Assignments	10	1 mark	all	all
	Reports	1	10 marks	6,7,8,9,10	all
Summative assessment	**Midterm Exam**	2hr	10% (10)	9	
	Final Exam	3hr	50% (50)	17	
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

	Material Covered
Week 1	**Introductory Overview**: Definition of Orthography, its founder, origin, and historical development.
Week 2	**Rules (ء)Initial Hamzah**
Week 3	**Medial Hamzah (Part I)**
Week 4	**Medial Hamzah (Part II)**
Week 5	**Final Hamzah Rules**
Week 6	**Final Hamzah with Tanwīn al-Fatḥ**
Week 7	**) ت) ’vs. Open Taa (ة(’Tied Taa**
Week 8	**)ظ) ’and Ẓā (ض (Differentiation between Ḍād**
Week 9	**Midterm Examination**
Week 10	**Alif Maqṣūrah (ى) Rules**
Week 11	**Additional and Omitted Letters (زيادة ونقصان)**
Week 12	**Numerals and Counted Nouns (Part I)**
Week 13	**Numerals and Counted Nouns (Part II)**
Week 14	**Numerals and Counted Nouns (Part III)**
Week 15	**Shaddah (ّ) and Maddah (آ)**
Week 16	**Preparation and Review for Final Examination**

Learning and Teaching Resources

	Text	Available in the Library?
Required Texts	• *Al-Imlāʾ al-Farīd*, Naoum Jirjis Zarazir, Arabic Language Library, Baghdad – Iraq, 6th Edition, 2017.	Yes

	- *Al-Imlāʾ al-Wāḍiḥ*, Abdul-Majid Al-Na’imi, Dar Al-Mutannabi Library, Baghdad – Iraq, 3rd Edition, 1967.	
Recommended Texts	Additional references may be consulted to support and extend the understanding of orthographic rules.	NO
Websites	Lisān al-ʻArab Digital Library (مكتبة لسان العرب الإلكترونية) Alukah Network (شبكة الألوكة) Fasih Platform (موقع فصيح) Nargis Digital Library (مكتبة نرجس الإلكترونية) Al-Waqfeya Digital Library (المكتبة الوقفية الإلكترونية) Noor Digital Library (مكتبة نور الإلكترونية)	

Grading Scheme

Group	Grade	Mark	Marks %	Definition
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX –** Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(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.

MODULE DESCRIPTION FORM

Module Information			
Module Title	**Calculus I**	**Module Delivery**	
Module Type	**Basic**	☒ **Lecture**	
Module Code	IT105		
ECTS Credits	**5**		
SWL (hr/sem)	**125**		
Module Level	UG1	**Semester of Delivery**	1
Administering Department	Information Technology	**College**	College of Science
Module Leader	**Saja Bassem Ali**	**e-mail**	Saja.b@uowa.edu.iq
Module Leader's Acad. Title	Assistant Lecturer	**Module Leader's Qualification**	MS.C
Module Tutor	**Saja Bassem Ali**	**e-mail**	Saja.b@uowa.edu.iq
Peer Reviewer Name	Asst. Lecturer Nabeel Sadeq	**e-mail**	nabeel.alshreefy@uowa.edu.iq
Scientific Committee Approval Date	2024-12-08	**Version Number**	1.0

Relation with other Modules			
Prerequisite module	-	**Semester**	-
Co-requisites module	-	**Semester**	-

جامعة وارث الانبياء
عمادة كلية العلوم
COLLEGE OF SCIENCES

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د. ضياء حسين نوفل
٢٠٢٥/٢٠٢٤

أ.م.د. ضياء حسين نوفل
٢٠٢٥/٢٠٢٤

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Objectives	1-Understand the concept of the derivative of a function and its geometrical and mechanical significance. 2- Criticize the basic rules of differentiation and be able to apply them to find first and higher derivatives of functions. 3- Know the elementary properties of the trigonometric functions, the inverse trigonometric functions, the exponential and logarithmic functions. Be able to differentiate expressions involving these functions. 4- Know about critical points of differentiable functions and their use in determining maxima and minima. Be able to apply these ideas in simple problems in optimization. 5- State the different methods of integration and their applications. 6- Understand the essential mathematics relevant to computer science. 7- Demonstrate basic knowledge and understanding of a core of analysis, algebra, applied mathematics and statistics.
Module Learning Outcomes	1- Handle techniques of differentiation and integration in solving practical problems 2- Use of standard numerical recipes and mathematical libraries in problem solving. 3-Explore, and where feasible solve, mathematical problems, by selecting appropriate techniques. 4- Evaluate systems in terms of general quality attributes and possible tradeoffs presented within the given problem. 5- Prove and disprove assertions using a variety of techniques
Indicative Contents	1-Summarize the proposed solutions and their results. 2- Verifying solutions. 3- Observing results and attitudes. 4 - Setting goals towards solving traditional and non-traditional problems. 5- Defining problems in precise scientific way. 6- Restrict solution methodologies upon their results. 7- Identify a range of solutions and critically evaluate and justify proposed design Solutions. 8- Criticize the methods of differentiation and integration.

Learning and Teaching Strategies	
Strategies	1- Manage time effectively. 2- Present a clear, logical argument. 3- Work independently. d4- Solve practical problems in course projects. 4- Speeding up the computation of conventional mathematical problems such as sorting, recursion, and matrix multiplication. 5- The ability to evaluate systems in terms of general and specific quality attributes. 6- Work within and contribute to a team, apply management skills such as coordination, project design and evaluation and decision processes.

Student Workload (SWL)			
Structured SWL (h/sem)	45	**Structured SWL (h/w)**	4
Unstructured SWL (h/sem)	74	**Unstructured SWL (h/w)**	5
Total SWL (h/sem)	122 + 3 (Final Exam) = 125		

Module Evaluation					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	3	10% (10)	3,6 and 9	1,2,3,4
	Assignments	2	5% (5)	4, 12	1,2,3,4
	H. W	5	10% (10)	2,4,6,8,10	1,2,3,4
	Attendance	1	10% (10)	Continues	1,2,3,4
Summative assessment	**Midterm Exam**	2hr	15% (15)	5,11	
	Final Exam	3hr	50% (50)	16	
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

	Material Covered
Week 1	Numbers and Sets. Representations of Functions.
Week 2	Domain; Codomain; Range of Functions. Test for Even and Odd Functions.
Week 3	Types of Functions and their Graphs.
Week 4	Definition of Limit.
Week 5	Finding Limits Graphically and Numerically
Week 6	Limit Laws
Week 7	One-Sided Limits
Week 8	Infinite Limits
Week 9	Continuity
Week 10	Introduction to Differentiation
Week 11	The Derivative of a Function
Week 12	Differentiability and Continuity
Week 13	basic derivative theorems
Week 14	Implicit Differentiation
Week 15	Applications of Differentiation
Week 16	Preparatory week before the final Exam

Learning and Teaching Resources

	Text	Available in the Library?
Required Texts	1. Calculus. Thomas. book 2. Calculus I. Paul Dawkins book	Yes
Recommended Texts	Ron Larson and Bruce Edwards 11 Edition	No
Websites	https://tutorial.math.lamar.edu/Classes/CalcI/CalcI.aspx	

Grading Scheme

Group	Grade	Mark	Marks %	Definition
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX –** Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(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.

MODULE DESCRIPTION FORM

Module Information			
Module Title	**Computer Organization**	**Module Delivery**	
Module Type	**Core**	☒ **Lecture** ☒ **Practical**	
Module Code	**IT103**		
ECTS Credits	**6**		
SWL (hr/sem)	**150**		
Module Level	UG1	**Semester of Delivery**	1
Administering Department	Information Technology	**College**	College of Science
Module Leader	Makki Hussein Abdel Rahim	**e-mail**	maky.h@uowa.edu.iq
Module Leader's Acad. Title	Lecturer	**Module Leader's Qualification**	PhD
Module Tutor	Makki Hussein Abdel Rahim	**e-mail**	maky.h@uowa.edu.iq
Peer Reviewer Name	Asst. Prof. Dr Hayder Mohammed Ali	**e-mail**	hayder.alghanami@uowa.edu.iq
Scientific Committee Approval Date	2024-11-01	**Version Number**	1.0

Relation with other Modules			
Prerequisite module	-	**Semester**	-
Co-requisites module	-	**Semester**	-

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

جامعة وارث الانبياء
عمادة كلية العلوم

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Objectives	1. Equip students with a fundamental understanding of different computer types, including their structure and hardware components. 2. Foster an understanding of the functionality and operation of various input/output devices. 3. Provide in-depth knowledge about computer memory structures, including ROM, RAM, virtual memory, and cache memory. 4. Facilitate understanding of various storage options, their properties, and disk partitioning techniques. 5. Impart a comprehensive understanding of operating systems, their types, functionalities, and history.
Module Learning Outcomes	1. Identify and distinguish between different types of computers and their associated hardware components. 2. Understand and describe the functionality of various input/output devices. 3. Demonstrate knowledge about different memory types, their functions, and hierarchy. 4. Understand and explain various data storage options, including HDDs, SSDs, and the concept of disk partitioning. 5. Analyze and compare various operating systems, describing their functions, types, and historical developments.
Indicative Contents	1. Introduction to Computers: Definitions and types of computers, including supercomputers, server computers, workstation computers, personal computers, and microcontrollers. 2. Computer Hardware: Detailed analysis of hardware components such as input/output units, memory units, CPUs, motherboards, expansion cards, and power supply units. 3. Input Devices: In-depth study of devices like keyboards, mice, scanners, barcode and QR code scanners, and speech recognition technology. 4. Output Devices: Exploration of devices such as speakers, printers (laser and inkjet), and monitors, including resolution, color depth, and refresh rates. 5. Memory: Examination of ROM, RAM, virtual memory, CPU cache, and the hierarchy of memory. 6. Storage: Detailed look at HDDs, SSDs, disk partitioning techniques, file systems, and related tasks. 7. Operating Systems: Study of the functions and types of operating systems, with examples and history of UNIX, MacOS, Linux, and Microsoft Windows.

Learning and Teaching Strategies	
Strategies	The learning and teaching strategies for studying the database subject in an IT department involve a balanced approach of theoretical understanding and practical 3 application. Lectures, interactive discussions, and case studies provide the necessary theoretical foundation. Practical exercises, group work, and projects enable hands-on experience with database management systems. Workshops, demos, and industry examples offer real-world insights. Online resources, assessments, and feedback aid in reinforcing learning. Virtual labs and continuous learning emphasize practical skills development and staying updated with industry trends. These strategies ensure a comprehensive understanding of databases and their relevance in the IT field.

Student Workload (SWL)			
Structured SWL (h/sem)	60	**Structured SWL (h/w)**	5
Unstructured SWL (h/sem)	87	**Unstructured SWL (h/w)**	6
Total SWL (h/sem)	**147 + 3 (Final Exam) = 150**		

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

Delivery Plan (Weekly Syllabus)	
	Material Covered
Week 1	Introduction to Computers: What is a Computer, Types of Computers (Supercomputer, Server Computer, Workstation Computer, Personal Computer or PC, Microcontroller.
Week 2	Introduction to Computer Hardware (Input Unit and Output Unit (I/O), Memory Unit, CPU, Motherboard
Week 3	More on Computer Hardware (Expansion Cards, Power Supply)
Week 4	Input Devices (Keyboard, Pointing Devices including Mouse, Trackball, Touchpad/Pointing Stick, Touch Screen, Stylus)
Week 5	More Input Devices (Scanners, Bar-code and QR Code Scanners, Microphone, Speech Recognition)
Week 6	Output Devices (Sound and Speakers, Printers including Laser and Inkjet)
Week 7	Output Devices (Sound and Speakers, Printers including Laser and Inkjet)
Week 8	More on Output Devices (Monitors, including an understanding of Resolution, Color Depth, Refresh Rate, Difference between CRT, LCD, OLED)
Week 9	Memory (ROM, RAM, Virtual Memory, CPU Cache (Cache Memory), Memory Hierarchy)
Week 10	Storage (Hard Disk Drive (HDD), HDD Geometry, HDD Logical Blocks)
Week 11	More on Storage (Solid State Disk (SSD), SSD Controller, Disk Partitioning including MBR. Partitioning and GPT, File Systems and Typical Tasks for File Systems)
Week 12	Introduction to Operating Systems, Functions of OS, OS Types (Batch, Single-Tasking and Multitasking, Single- and Multi-User, Real Time OS, Distributed Operating System, Mobile OS
Week 13	More on Operating Systems (OS Examples and History: UNIX and UNIX-like Operating Systems, BSD and its Descendants, MacOS, Linux Family)
Week 14	More on Operating Systems (Linux, Mac OS)
Week 15	Preparatory week before the final Exam

Delivery Plan (Weekly Lab. Syllabus)	
	Material Covered
Week 1	Introduction to computer architecture and organization.
Week 2	Understand BIOS' role in booting the laptop and finding out the laptop model number.
Week 3	Explore how to change the boot device.
Week 4	Explore the importance of having a healthy chair and desk to work on a laptop or a PC
Week 5	Explore the importance of learning to type correctly.
Week 6	Introduction to computer components (CPU, Motherboard, RAM, HDD, Power supply, Case, Graphic card, Sound card, monitor, keyboard, mouse, speaker).
Week 7	Have practical experience with assembling and disassembling PC components.
Week 8	Explore Windows sandbox feature.
Week 9	Explore Oracle virtual box and Hyper-V.
Week 10	Download Windows ISO file and create a bootable flash disk using Rufus.

Week 11	Explore computer management and local users and groups.
Week 12	Explore Task scheduler, Event viewer, Services, Disk management, and Device manager.
Week 13	Learn about Windows users and groups and file permissions.
Week 14	Explore Task manager and startup programs.
Week 15	Explore disk encryption

Learning and Teaching Resources

	Text	**Available in the Library?**
Required Texts		No
Recommended Texts	"Computer Organization and Architecture" by William Stallings	No
Websites	https://www.tutorialspoint.com/basics_of_computer_science/index.htm	

Grading Scheme

Group	**Grade**	**Marks**	**Marks %**	**Definition**
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX –** Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(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.

MODULE DESCRIPTION FORM

Module Information			
Module Title	Digital Logic	**Module Delivery**	
Module Type	**Core**	**Lecture** **Practical**	
Module Code	**IT102**		
ECTS Credits	**6**		
SWL (hr/sem)	**150**		
Module Level	UG1	**Semester of Delivery**	1
Administering Department	Information Technology	**College**	College of Science
Module Leader	Nabeel Sadiq Abdel Abbas	**e-mail**	nabeel@uowa.edu.iq
Module Leader's Acad. Title	Assistant Lecturer	**Module Leader's Qualification**	Mcs
Module Tutor	Nabeel Sadiq Abdel Abbas	**e-mail**	nabeel@uowa.edu.iq
Peer Reviewer Name	Asst. Lect Karrar Sadiq	**e-mail**	karar.sadeq@uowa.edu.iq
Scientific Committee Approval Date	2024-11-01	**Version Number**	1.0

Relation with other Modules			
Prerequisite module	-	**Semester**	-
Co-requisites module	-	**Semester**	-

جامعة وارث الانبياء
عمادة كلية العلوم

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Objectives	1. Provide students with basic information about digital logic and logic circuits. 2. Increasing students' horizons in the fields of computer science and digital development. 3. Developing the students' English language by teaching the subject in English. 4. Providing students with applied and experimental skills through practical subjects and laboratories. 5. Familiarize students with the latest developments in the fields of different sciences and the technology emanating from them. 6. Developing the student's ability to research and providing him with scientific research contexts. 7. Develop students' ability to analyze and link information and conclusion. 8. Enhancing the scientific spirit in the interpretation of phenomena, discussion, and dialogue. 9. Consolidation of conviction in the integration of sciences and their universality towards the truth. 10. Working on refining the student's personality and discovering his inclinations and talents through scientific and cultural activities. 11. Enhancing the spirit of teamwork through the participation of students in laboratory work or the completion of joint scientific research. Establish values and ideals Higher among them is respect for instructions, discipline, respect for the institution to which the student belongs, and preservation of its property.
Module Learning Outcomes	1. Knowing the numerical number systems used in logical circuits and performing arithmetic operations on them. 2. Knowledge of logical circuits and their design methods. 3. Simplify logic circuits by simplifying their equations. 4. Full knowledge of digital meters, dividers, and other electronic circuits. 5. Full knowledge of the use of signs and their representation in binary numbers. 6. Full knowledge of how to convert between number systems used in numerical operations. 7. How to integrate digital portals together and methods of calculating their outputs. 8. Design counters and dividers and link them together
Indicative Contents	1. Introduction to Digital Logic and Logic Circuits ● Overview of digital logic and its significance in computer science and digital development ● Introduction to logic circuits and their role in processing digital information 2. Logic Gates and Circuit Design ● Exploration of fundamental logic gates (AND, OR, NOT, XOR, NAND, NOR) ● Designing and analyzing logic circuits using gates ● Application of De Morgan's theorem for circuit simplification 3. Combinational Logic Circuits ● Understanding the design and operation of combinational logic circuits

	● Implementation of multiplexers, demultiplexers, encoders, and decoders ● Building adders, subtractors, and comparators 4. Sequential Logic Circuits ● Introduction to sequential logic circuits and their behavior ● Study of flip-flops and latches for storing and transferring data ● Analysis and design of synchronous and asynchronous sequential circuits 5. Digital Integrated Circuits ● Types and characteristics of digital integrated circuits (TTL, CMOS, FPGA) ● Understanding IC packaging, pin configurations, and datasheets ● Testing, troubleshooting, and selecting appropriate ICs for specific applications. 6. Practical Applications and Research Focus ● Hands-on experiments in laboratory settings to apply learned concepts. ● Exploring emerging trends and advancements in digital logic and circuits ● Developing research skills and methodologies for investigating digital systems

Learning and Teaching Strategies	
Strategies	● Giving lectures ● Performing software tasks in laboratories ● Scientific discussions and dialogues and asking questions. ● The completion of tasks by student work teams in the laboratory

Student Workload (SWL)			
Structured SWL (h/sem)	60	**Structured SWL (h/w)**	5
Unstructured SWL (h/sem)	87	**Unstructured SWL (h/w)**	6
Total SWL (h/sem)	**147 + 3 (Final Exam) = 150**		

Module Evaluation

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	2	10% (10)	5 and 10	1,2,3,4
	Assignments	2	10% (10)	2 and 12	1-5
	Lab.	1	10% (10)	Continuous	1-5
	Report	1	10% (10)	13	1,2,3,4,5
Summative assessment	Midterm Exam	2hr	10% (10)	7	
	Final Exam	3hr	50% (50)	16	
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

	Material Covered
Week 1	Numbers system
Week 2	Binary, BCD, octal, Hex Numbers
Week 3	Converting Binary Arithmetic
Week 4	1’s and 2’s Complements of Binary Numbers Signed Numbers
Week 5	Logic Gate
Week 6	Boolean Algebra and Logic Simplification
Week 7	DE Morgan's Theorem
Week 8	Karnaugh Map
Week 9	Combinational Logic Circuit
Week 10	Functions of Combinational Logic
Week 11	Latches
Week 12	Flip-Flops
Week 13	Counters
Week 14	Counters
Week 15	Multiplexer and demultiplexer

Delivery Plan (Weekly Lab. Syllabus)

	Material Covered
Week 1	Introduction to Digital Logic and Logic Gates
Week 2	Logic Gates and Truth Tables
Week 3	Logic Gate Implementations

Week 4	Combinational Logic Circuits
Week 5	Multiplexers and Demultiplexers
Week 6	Encoders and Decoders
Week 7	Sequential Logic Circuits: Latches and Flip-Flops
Week 8	Sequential Logic Circuits: Counters
Week 9	Shift Registers
Week 10	Memory Units: RAM and ROM
Week 11	Introduction to Programmable Logic Devices
Week 12	Number Systems: Binary, Decimal, and Hexadecimal
Week 13	Number System Conversions
Week 14	Arithmetic Circuits: Adders and Subtractors
Week 15	Digital Logic Design Project

Learning and Teaching Resources

	Text	Available in the Library?
Required Texts	Digital Logic & Number System (Munich war Gulati & Mini) Gulati)	yes
Recommended Texts	Digital logic and computer design (Morris-Mano) 4th ed.	NO
Websites		

Grading Scheme

Group	Grade	Mark	Marks %	Definition
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX –** Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(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.

MODULE DESCRIPTION FORM

Module Information

Module Title	**Information Technology Fundamentals**	**Module Delivery**	
Module Type	**Core**	☒ **Lecture** ☒ **Seminar**	
Module Code	**IT101**		
ECTS Credits	**4**		
SWL (hr/sem)	**100**		
Module Level	UG1	**Semester of Delivery**	1
Administering Department	Information Technology	**College**	College of Science
Module Leader	Bandar Abdul abbas Almankoshi	**e-mail**	bandar@uowa.edu.iq
Module Leader's Acad. Title	Assistant Lecturer	**Module Leader's Qualification**	M.Sc.
Module Tutor	Bandar Abdul abbas Almankoshi	**e-mail**	bandar@uowa.edu.iq
Peer Reviewer Name	Asst. Lect Nabeel Sadeq	**e-mail**	nabeel.alshreefy@uowa.edu.iq
Scientific Committee Approval Date	2024-11-10	**Version Number**	1.0

Relation with other Modules

Prerequisite module	-	**Semester**	-
Co-requisites module	-	**Semester**	-

جامعة وارث الانبياء
عمادة كلية العلوم

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د. شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

Department Head Approval

أ.م.د. شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

Dean of the College Approval

<table>
<tr><th colspan="2">Module Aims, Learning Outcomes and Indicative Contents</th></tr>
<tr><td><b>Module Objectives</b></td><td>The module aims for information technology fundamentals in the Information. Technology department can vary depending on the specific educational institution or program. However, here are some general aims that are often covered in such a module:
1. Introduction to Information Technology: Provide an overview of the field of information technology, its importance, and its role in various industries.
2. Hardware and Software Fundamentals: Introduce the basic components of computer hardware, such as CPUs, memory, storage devices, and peripheral devices. Also, cover the basics of software, including operating systems, applications, and programming languages.
3. Networking Concepts: Familiarize students with the fundamentals of computer networks, including network architectures, protocols, network devices, and communication technologies.
4. Data Management and Databases: Introduce the principles of data management, including data types, data organization, database systems, and data security.
5. Information Systems: Explore the concept of information systems, including their components, functions, and the role of IT in supporting business processes.
6. Cybersecurity: Raise awareness about the importance of cybersecurity and introduce basic concepts of securing computer systems, networks, and data.
7. Web Technologies: Cover the basics of web development, including HTML, CSS, and JavaScript, as well as web design principles and website deployment.
8. Human-Computer Interaction (HCI) is a multidisciplinary field that focuses on the design, evaluation, and implementation of interactive computing systems for human use. In the IT field, HCI plays a crucial role in creating user-friendly and efficient software, websites, and other digital interfaces. Here are some key aspects of HCI in the IT industry.
9. System integration refers to the process of combining different subsystems, components, or software applications into a unified and cohesive system. It involves connecting and integrating various IT systems, databases, networks, and applications to enable seamless data flow, communication, and functionality across the organization. System integration plays a critical role in enabling interoperability, streamlining business processes, and maximizing the value of IT investments. Here are key aspects and considerations related to system integration.
10. IT Project Management: Provide an understanding of project management principles and practices in the context of IT projects, including planning, organizing, and controlling IT projects effectively.
11. Emerging Technologies: Discuss current trends and emerging technologies in information technology, such as cloud computing, artificial intelligence, Internet of Things (IoT), and big data analytics.
12. Ethical and Legal Considerations: Explore ethical issues related to IT, such as privacy, intellectual property, and responsible use of technology. Also, discuss legal frameworks and regulations relevant to IT.</td></tr>
</table>

	These aims are not exhaustive and can vary depending on the specific curriculum and institution. The module aims to provide students with a solid foundation in information technology concepts, principles, and skills, preparing them for further. studies or careers in the field of IT.
Module Learning Outcomes	Module Learning Outcomes for an Information Technology Fundamentals module in an Information Technology department can include the following: 1. Knowledge and Understanding: a. Demonstrate knowledge and understanding of the basic concepts, principles, and theories in information technology. b. Understand the fundamental components of computer hardware, software, and networking. c. Explain the importance of data management, information systems, and cybersecurity in organizations. 2. Technical Skills: a. Apply practical skills in using computer hardware and software effectively. b. Configure and troubleshoot basic computer networks. c. Use database management systems to organize and retrieve data. 3. Critical Thinking and Problem Solving: a. Analyze and solve basic technical problems related to hardware, software, and networking. b. Apply logical thinking and problem-solving skills to address IT-related challenges. c. Evaluate different information technology solutions and make informed decisions. 4. Communication: a. Communicate effectively with peers and instructors using appropriate IT terminology. b. Present technical information clearly and concisely. c. Collaborate with others in group projects and discussions related to IT concepts. 5. Ethical and Professional Conduct: a. Recognize and adhere to ethical guidelines and professional standards in IT. b. Understand the legal and regulatory frameworks related to IT. c. Demonstrate responsible and ethical use of technology and respect for intellectual property. 6. Lifelong Learning: a. Demonstrate a curiosity and enthusiasm for ongoing learning in the field of information technology. b. Engage in self-directed learning and stay updated with emerging trends and technologies. c. Adapt to changes in technology and apply new skills as needed. These learning outcomes are designed to provide students with a solid foundation in information technology fundamentals, preparing them for further studies or careers in the IT field. They encompass both knowledge-based understanding and practical skills, as well as critical thinking and ethical considerations.
Indicative Contents	The indicative contents for an Information Technology Fundamentals module in an Information Technology department may include the following topics: 1. Introduction to Information Technology: ● Definition and scope of information technology.

	● Evolution and history of information technology. ● Importance of information technology in various industries. 2. Computer Networks: ● Network architectures (LAN, WAN, client-server, peer-to-peer). ● Network protocols (TCP/IP, HTTP, FTP, etc.). ● Network devices (routers, switches, modems, etc.). ● Network security and common threats. 3. Data Management and Databases: ● Data types and data representation. ● Database concepts and models. ● Structured Query Language (SQL) and database operations. ● Data integrity, normalization, and database design principles. 4. Cybersecurity: ● Importance of cybersecurity and its challenges. ● Common security threats and vulnerabilities. ● Security measures and best practices. ● Cryptography and encryption techniques. 5. Emerging Technologies: ● Cloud computing and virtualization. ● Artificial intelligence and machine learning. ● Internet of Things (IoT) and its applications. ● Big data analytics and data-driven decision making. ● Ethical and Legal Considerations: 6. Ethical issues in information technology. ● Intellectual property rights and plagiarism. ● Privacy and data protection. ● Legal frameworks and regulations related to IT. 7. Human Computer Interaction: ● Show when human factors first became an issue in computer hardware and software design. ● Define the meaning of human-computer interaction or HCI. ● Define the meaning of user experience design or UXD. ● Describe the evolution from human factors to User Experience Design (UX). 8. Information Management (IM): IM refers to the process of ● organizing ● storing ● retrieving managing data and information within an organization. It involves various practices, technologies, and strategies to ensure that information is effectively captured, processed, stored, and utilized to support organizational goals and decision-making. Here are some key aspects of information management in the IT field. These indicative contents provide a broad overview of the topics that may be covered in an Information Technology Fundamentals module. The specific curriculum may vary based on the educational institution or program requirements.

Learning and Teaching Strategies	
Strategies	When it comes to the learning and teaching strategies for an Information Technology Fundamentals course in an Information Technology department, a combination of theoretical and practical approaches is often used to enhance students' understanding and application of the concepts. Here are some common strategies employed: 1. Lectures: In-class lectures provide an opportunity for the instructor to present theoretical concepts, explain complex topics, and provide an overview of key principles in information technology. 2. Interactive Discussions: Engaging students in discussions encourages active participation and critical thinking. It allows students to ask questions, share their perspectives, and collaborate with peers to deepen their understanding of the subject matter. 3. Hands-on Practical Exercises: Practical exercises and lab sessions provide students with the opportunity to apply the theoretical knowledge gained in lectures. It helps them develop technical skills, such as configuring computer systems, programming, database management, and networking. 4. Case Studies and Real-World Examples: Incorporating case studies and real-world examples helps students understand how information technology concepts are applied in practical scenarios. It enables them to analyze and solve problems and make connections between theory and real-world situations. 5. Group Projects and Collaborative Learning: Assigning group projects allows students to work together, enhancing their teamwork and communication skills. It also fosters collaborative problem-solving and encourages students to apply their knowledge to solve complex IT challenges. 6. Online Learning Resources: Utilizing online learning platforms, educational websites, and interactive multimedia resources can supplement classroom teaching. These resources can provide additional explanations, tutorials, quizzes, and simulations to enhance understanding and provide self-paced learning opportunities. 7. Guest Speakers and Industry Visits: Inviting guest speakers from the industry or organizing visits to IT companies can expose students to real-world practices, industry trends, and professional perspectives. It can help students understand the relevance of the course material to professional IT careers. 8. Assessments and Feedback: Regular assessments, such as quizzes, assignments, and exams, allow students to evaluate their understanding and progress. Constructive feedback from instructors on their performance helps students identify areas of improvement and reinforces their learning. 9. Online Discussion Forums: Establishing online discussion forums or platforms where students can ask questions, share resources, and engage in peer-to-peer learning can foster a collaborative learning environment outside the classroom. 10. Continuous Learning and Updates: Encouraging students to stay updated with the latest trends, technologies, and industry news through recommended readings, online resources, and professional development opportunities promotes lifelong learning and adaptability in the field of information technology.

	These strategies aim to create an engaging and immersive learning experience that combines theoretical knowledge with hands-on practice, critical thinking, and realworld applications. The specific strategies employed may vary based on the teaching style of the instructor, the resources available, and the educational institution's approach to IT education.

Student Workload (SWL)			
Structured SWL (h/sem)	52	**Structured SWL (h/w)**	4
Unstructured SWL (h/sem)	45	**Unstructured SWL (h/w)**	4
Total SWL (h/sem)	**97 + 3 (Final Exam) = 100**		

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

Delivery Plan (Weekly Syllabus)	
	Material Covered
Week 1	Definition of the Information Technology Academic Discipline.
Week 2	Data communication: Introduction about data communication, Components of data communication, Data communication basic terms, Signals, Transmission media, Effective data communication, Data rate, Bandwidth.
Week 3	Describe how integrating various modules can produce a working system, describe how integration is an important function of all IT professionals.
Week 4	Networking: a. Describe networking and the research scope of networking study. b. Identify some components of a network.

	c. Name several network devices and describe their purpose. d. Describe ways information technology uses or benefits from networks
Week 5	Networking: e. Illustrate the role of networks in information technology. f. Identify people who influenced or contributed to the area of networks. g. Identify several contributors to networks and relate their achievements to the area.
Week 6	The Internet: Internet Applications a. Describe how the world-wide web has impacted people’s lives over time. b. Illustrate the growth and changes in mobile devices and applications over time.
Week 7	Cybersecurity Principles: a. Make sense of the hard problem areas in cybersecurity that continue to make cybersecurity a challenge to implement. b. Describe how a significant cybersecurity event has led to increased organizational focus on cybersecurity. c. Tell a story of a significant cybersecurity advance.
Week 8	Cybersecurity Principles: a. Evaluate when the Confidentiality, Integrity, and Availability (CIA) of information has been or could be violated with regards to providing trust of information. b. Compare and evaluate different approaches/implementations of digital currencies.
Week 9	Human Computer Interaction: a. Show when human factors first became an issue in computer hardware and software design. b. Define the meaning of human-computer interaction or HCI. c. Define the meaning of user experience design or UXD. d. Describe the evolution from human factors to User Experience Design (UX).
Week 10	Human Computer Interaction: a. Contrast the physical and non-physical aspects of UXD. b. Identify several modern high-tech computing technologies that present UXD challenges. c. Describe several reasons for making UXD an essential part of the information technology discipline.
Week 11	Information Management (IM): ● organizing ● storing ● retrieving managing data and information within an organization. It involves various practices, technologies, and strategies to ensure that information is effectively captured, processed, stored, and utilized to support organizational goals and decision-making. Here are some key aspects of information management in the IT field
Week 12	Information Management (IM): ● Data Governance ● Data Integration ● Data Warehousing ● Database Management Systems (DBMS) ● Information Security ● Knowledge Management ● Information Lifecycle Management (ILM)

Week 13	System integration: ● Integration Technologies ● Data Integration ● Application Integration ● Enterprise Service Bus (ESB) ● Legacy Systems Integration
Week 14	System integration: ● Legacy Systems Integration ● Business Process Integration ● Cloud Integration ● Testing and Validation ● Security and Governance
Week 15	Preview
Week 16	Preparatory week before the final Exam

Learning and Teaching Resources		
	Text	**Available in the Library?**
Required Texts	Fundamentals of Information Technology, by: Salah Alkhafaji. Introduction of Information Technology, by V. Rajaraman, PHI Learning Private Limited	No
Recommended Texts		
Websites	http://www.sqlcourse.com/ http://www.db-book.com/	

Grading Scheme				
Group	**Grade**	**Mark**	**Marks %**	**Definition**
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX –** Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(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.

MODULE DESCRIPTION FORM

Module Information

Module Title	**Programming Fundamentals I**	**Module Delivery**	
Module Type	**Core**	**Lecture** **Practical**	
Module Code	**IT104**		
ECTS Credits	**7**		
SWL (hr/sem)	**175**		
Module Level	UG1	**Semester of Delivery**	1
Administering Department	Information Technology	**College**	College of Science
Module Leader	Mohsin Hassan Hussein	**e-mail**	mohsin.ha@uowa.edu.iq
Module Leader's Acad. Title	Assistant Professor	**Module Leader's Qualification**	Ph.D.
Module Tutor	Mohsen Hassan Hosein	**e-mail**	mohsin.ha@uowa.edu.iq
Peer Reviewer Name	Asst.Prof Hyder Mohammed Ali	**e-mail**	hayder.alghanami@uowa.edu.iq
Scientific Committee Approval Date	2024-11-01	**Version Number**	V1

Relation with other Modules

Prerequisite module	-	**Semester**	-
Co-requisites module	-	**Semester**	-

جامعة وارث الانبياء
عمادة كلية العلوم

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د ضياء حسين نوفل
٢٠٢٤/٢٠٢٥

أ.م.د ضياء حسين نوفل
٢٠٢٤/٢٠٢٥

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Objectives	The following are some key aims and benefits of studying Programming Fundamentals I: 1. Introduction to Programming: Introduce students to the fundamental concepts of programming, including the role of programming languages, the software development process, and basic programming principles. 2. Problem Solving: Teach students how to analyze problems and develop algorithms to solve them. Emphasize problem-solving techniques, algorithm design, and decomposition of complex problems into smaller, manageable parts. 3. Input and Output: Teach students how to interact with the user and handle standard input/output operations, including reading from keyboard and display to screen. 4. Programming Language Basics: Familiarize students with the syntax, semantics, and basic constructs of a programming language, such as variables, data types, control structures (loops, conditionals), and functions. 5. Debugging and Testing: Teach students how to debug and test their programs to identify and fix errors. Explore techniques for error detection, debugging tools, and strategies for writing effective test cases
Module Learning Outcomes	The following are some common learning outcomes for a Programming Fundamentals I: 1. Knowledge of Programming Concepts: Demonstrate a solid understanding of fundamental programming concepts, including variables, data types, control structures, and basic algorithms. 2. Problem Solving Skills: Apply problem-solving techniques to analyze and solve programming problems by decomposing them into smaller, manageable parts and designing appropriate algorithms. 3. Proficiency in Programming Language: Develop proficiency in using a specific programming language covered in the course, including understanding the language's syntax, semantics, and basic constructs. 4. Effective Code Writing: Write clear, well-structured, and readable code that follows coding standards and best practices, including proper indentation, meaningful variable names, and appropriate comments. 5. Debugging and Testing Skills: Use debugging techniques and tools to identify and fix errors in programs. Develop effective test cases and perform testing to ensure program correctness and reliability.
Indicative Contents	The indicative contents of a Programming Fundamentals I module have a list of common topics that shown below: 1-Introduction to Programming: Role of programming languages, Software development process, Basic programming principles and concepts. [15 hrs.] 2-Problem Solving and Algorithm Design: Problem analysis and requirements specification, Algorithm design techniques (e.g., topdown design, stepwise refinement), Flowcharts and pseudocode. [20hrs] 3-Input and Output: standard input/output operations, including reading from keyboard and display to screen. [10 hrs.] 4- Programming Language Basics: Variables and data types, Operators and expressions, Control structures (loops, conditionals). [30 hrs.]

	5- Modular Programming: Scope and lifetime of variables. [10 hrs.] 6-Debugging and Testing: Common types of programming errors, Debugging techniques and tools. [10 hrs.]

Learning and Teaching Strategies	
Strategies	To teach a Programming Fundamentals I module, various strategies can be employed to facilitate effective learning and engagement. Here are some learning and teaching strategies commonly used in Programming Fundamentals I module: 1- Lectures: Delivering lectures to present theoretical concepts, principles, and foundational knowledge of Programming Fundamentals I. Lectures can include visual aids, examples, and demonstrations to enhance understanding. 2- Interactive Discussions: Encourage students to actively participate in discussions by asking questions, sharing their thoughts, and engaging in peer-to-peer learning. Discussions can focus on challenging concepts, real-world applications, or case studies related to Programming Fundamentals I. 3- Hands-on Lab Sessions: Conduct practical lab sessions where students can gain hands-on experience with Programming Fundamentals I, 4 commands, and programming exercises. These sessions provide an opportunity to reinforce theoretical concepts and develop practical skills. 4- Group Projects: Assign group projects that involve designing, implementing, and evaluating components of Programming Fundamentals I. Group projects promote teamwork, problem-solving, and practical application of operating system concepts. 5- Online Resources and Tutorials: Provide access to online resources, tutorials, and interactive learning materials related to Programming Fundamentals I. This allows students to explore additional content, reinforce their understanding, and self-assess their progress. 6- Assessments and Feedback: Use a variety of assessment methods such as quizzes, assignments, projects, and exams to evaluate students' understanding of Programming Fundamentals I concepts. Provide timely and constructive feedback to help students improve their knowledge and skills.

Student Workload (SWL)			
Structured SWL (h/sem)	75	**Structured SWL (h/w)**	6

Unstructured SWL (h/sem)	97	**Unstructured SWL (h/w)**	5
Total SWL (h/sem)	**172 + 3 (Final Exam)= 175**		

Module Evaluation

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	5	5% (5)	3,5,7,9,11	LO #1, #3 and #4
	Home Work	5	10% (10)	2,4,6,8,10	LO #1, #3 and #4
	Lab	10	20% (20)	Continuous	All
	Onsite Assignments	5	5% (5)		LO #5, #8 and #10
Summative assessment	**Midterm Exam**	2hr	10% (10)	9	LO #1, #2 and #3
	Final Exam	3hr	50% (50)	17	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

	Material Covered
Week 1	Problem solving
Week 2	Algorithms and flow charts
Week 3	Introduction to programming Languages
Week 4	Variables, Constants, keywords, types, operators, expression, assignment
Week 5	Simple I/O Functions
Week 6	Conditional Statements
Week 7	If Statement
Week 8	Nested If
Week 9	Mid Exam
Week 10	Switch Statement
Week 11	Iterative control statements + for Statements
Week 12	While Statement
Week 13	Do while
Week 14	Nested Loops
Week 15	Nested while
Week 16	Preparatory week before the final Exam

Delivery Plan (Weekly Lab. Syllabus)	
	Material Covered
Week 1	IDE of Programming Language
Week 2	Examples for Algorithms and flow charts
Week 3	Using the IDE for writing sample of program
Week 4	Programs by using Variables, Constants, keywords, types, operators, expression, assignment
Week 5	Writing codes for 3 Programs Appling Simple I/O Functions
Week 6	Simple Conditional Statements programs
Week 7	Writing codes of If Statement programs
Week 8	Writing codes of Nested If programs
Week 9	Mid Exam
Week 10	Writing codes of Switch Statement programs
Week 11	Writing codes of Iterative control statements + for Statements programs
Week 12	Writing codes of While Statement programs
Week 13	Writing codes of Do while programs
Week 14	Writing codes of Nested Loops programs
Week 15	Writing codes of Nested while programs

Learning and Teaching Resources		
	Text	**Available in the Library?**
Required Texts	C++: The Complete Reference, Fourth Edition, Herbert Schildt.	Yes
Recommended Texts	The C++ Programming Language, Third Edition, Bjarne Stroustrup.	No
Websites	https://stackoverflow.com/	

Grading Scheme

Group	Grade	Marks	Marks %	Definition
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX –** Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(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.

MODULE DESCRIPTOR FORM

Module Information

Module Title	CALCULUS 2	**Module Delivery**	
Module Type	BASIC	**Theory** ✓ **Seminar** ✓ **Lecture** ✓	
Module Code	IT1211		
ECTS Credits	6		
SWL (hr/sem)	150		
Module Level	1	**Semester of Delivery**	2
Administering Department	Information technology	**College**	College of Sciences
Module Leader	**Saja Bassem Ali**	**e-mail**	Saja.b@uowa.edu.iq
Module Leader's Acad. Title	assistant Lecturer	**Module Leader's Qualification**	MSC
Module Tutor	Saja Bassem Ali	**e-mail**	Saja.b@uowa.edu.iq
Peer Reviewer name	Lecturer Maky H.Abdulraheem	**e-mail**	maky.h@uowa.edu.iq
Review Committee Approval	2024-1-20	**Version Number**	1.0

Relation With Other Modules

Prerequisite module	Calculus1	**Semester**	1
Co-requisites module	Calculus1	**Semester**	1

جامعة وارث الانبياء
عمادة كلية العلوم

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Aims	1-Understand the concept of the derivative of a function and its geometrical and mechanical significance. 2- Criticize the basic rules of differentiation and be able to apply them to find first and higher derivatives of functions. 3- Know the elementary properties of the trigonometric functions, the inverse trigonometric functions, the exponential and logarithmic functions. Be able to differentiate expressions involving these functions. 4- Know about critical points of differentiable functions and their use in determining maxima and minima. Be able to apply these ideas in simple problems in optimization. 5- State the different methods of integration and their applications. 6- Understand the essential mathematics relevant to computer science. 7- Demonstrate basic knowledge and understanding of a core of analysis, algebra, applied mathematics and statistics.
Module Learning Outcomes	1- Handle techniques of differentiation and integration in solving practical problems 2- Use of standard numerical recipes and mathematical libraries in problem solving. 3-Explore, and where feasible solve, mathematical problems, by selecting appropriate techniques. 4- Evaluate systems in terms of general quality attributes and possible tradeoffs presented within the given problem. 5- Prove and disprove assertions using a variety of techniques
Indicative Contents	1-Summarize the proposed solutions and their results. 2- Verifying solutions. 3- Observing results and attitudes. 4 - Setting goals towards solving traditional and non-traditional problems. 5- Defining problems in precise scientific way. 6- Restrict solution methodologies upon their results. 7- Identify a range of solutions and critically evaluate and justify proposed design solutions 8- Criticize the methods of differentiation and integration

Learning and Teaching Strategies	
Strategies	1-Manage time effectively. 2-- Present a clear, logical argument.

	3-- Work independently. d4- Solve practical problems in course projects. 4-- Speeding up the computation of conventional mathematical problems as sorting, recursion, and matrix multiplication. 5-- The ability to evaluate systems in terms of general and specific quality attributes. 6-- Work within and contribute to a team, apply management skills such as coordination, project design and evaluation and decision processes

Student Workload (SWL)			
Structured SWL (h/sem)	60	**Structured SWL (h/w)**	4
Unstructured SWL (h/sem)	87	**Unstructured SWL (h/w)**	6
Total SWL (h/sem)	147+3 final		

Module Evaluation

		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	5	4% (20)	1,2,3,4	2,5,7,8,9
	Assignments	2	5% (6)	6,11	All Outcome
	H.W	3	2% (6)	2,4,9,10	All Outcome
	Report	5	10% (10)	5,12	All Outcome
Summative assessment	**Midterm Exam**	2hr	15% (15)	5,11	
	Final Exam	3hr	50% (50)	16	
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)	
	Material Covered
Week 1	Antiderivatives.
Week 2	Indefinite Integrals.
Week 3	Basic Integration Rules.
Week 4	Integration by Substitution.
Week 5	Integration by Parts.
Week 6	trigonometric integrals
Week 7	Areas Between Curves
Week 8	Areas in rectangular coordinates
Week 9	Double Integrals
Week 10	Double Integrals over Rectangles
Week 11	Application of integrals
Week 12	Triple integrals (Volume)
Week 13	Area between two curves
Week 14	Odd and even powers of sine and cosine
Week 15	Odd and even powers of sine and cosine
Week 16	Preparatory week before the final Exam

Learning and Teaching Resources		
	Text	Available in the Library?
Required Texts	1. Calculus. Thomas. book 2. Calculus I .Paul Dawkins book	yes
Recommended Texts	Ron Larson and Bruce Edwards 11 Edition	no
Websites	https://tutorial.math.lamar.edu/Classes/CalcI/CalcI.aspx	

APPENDIX:

GRADING SCHEME				
Group	**Grade**	**Mark**	**Marks (%)**	**Definition**
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(0-44)	Considerable amount of work required
Note:				

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.

ملاحظة: هذا النموذج تم وضعه وتقديمه من قبل مديرية ضمان الجودة في وزارة التعليم العالي والبحث العلمي

MODULE DESCRIPTOR FORM

Module Information

Module Title	Human Rights and Democracy	**Module Delivery**	
Module Type	SUPPORTIVE	☒ **Lecture**	
Module Code	UOWA102		
ECTS Credits	2		
SWL (hr/sem)	50		
Module Level	UG 1	**Semester of Delivery**	2
Administering Department	Information technology	**College**	Science
Module Leader	Abbas Taher	**e-mail**	abbas.taher@uowa.edu.iq
Module Leader's Acad. Title	Asst. Lecturer	**Module Leader's Qualification**	MS.c
Module Tutor	Abbas Taher	**e-mail**	abbas.taher@uowa.edu.iq
Peer Reviewer Name	Asst. Lecturer Nabeel Sadeq Al-Shreefy	**e-mail**	nabeel.alshreefy@uowa.edu.iq
Review Committee Approval	2024-1-20	**Version Number**	1

Relation With Other Modules

Prerequisite module	None	**Semester**	None
Co-requisites module	None	**Semester**	None

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

جامعة وارث الانبياء
عمادة كلية العلوم
COLLEGE OF SCIENCES

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Aims	• Understand the topic of human rights and their significance in ancient and contemporary civilizations. • Learn about contemporary international and regional human rights frameworks. • Identify the main generations of human rights. • Study constitutional, judicial, and political guarantees of human rights. • Understand the role of the United Nations in protecting human rights.
Module Learning Outcomes	1. The relationship between democracy and human rights 2. Building a comprehensive framework for democratic governance based on human rights 3. Working to achieve the highest quality standards in the study of human rights and democracy 4. Enhancing students' cognitive abilities 5. Understanding their rights, duties, fundamental freedoms, and legal guarantees 6. Highlighting legal, international, regional, and national mechanisms for protecting human rights 7. Scientific perspective on the principles of democracy and its relationship to human rights and good governance 8. The impact of state institutions and civil society organizations on implementing democracy and good governance 9. Understanding the mechanisms of the United Nations 10. Recognizing the limitations imposed on human rights institutions 11. Applying student disciplinary laws within educational institutions 12. Concept and history of democracy 13. Understanding the characteristics and components of a democratic system 14. Understanding guarantees and public freedoms
Indicative Contents	1- General Introduction: Concept of Human Rights (3 hours) 2- Evolution of the Idea and Roots of Human Rights (2 hours) 3- Human Duties and Limitations (2 hours) 4- Professional Ethics (2 hours) 5- Student Disciplinary Law in the Ministry of Higher Education and Scientific Research (3 hours) 6- Concept and History of Democracy (2 hours) 7- Components of the Democratic System (2 hours) 8- Elections (2 hours) 9- Relationship between Democracy and Human Rights (2 hours) 10- Crimes of Genocide (2 hours) 11- Guarantees of Freedoms and Public Rights (2 hours) 12- Good Governance (2 hours) 13- Contemporary Democracy (2 hours)

Learning and Teaching Strategies	
Strategies	1- Explain the lecture through student discussion and participation to engage them. 2- Deliver the lecture in the form of a short quiz/exam. 3- Read the lecture material inside the classroom.

Student Workload (SWL)			
Structured SWL (h/sem)	30	Structured SWL (h/w)	2
Unstructured SWL (h/sem)	17	Unstructured SWL (h/w)	2.5
Total SWL (h/sem)	47 + 3 final = 50		

Module Evaluation					
		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	2	7%(14)	4,9	7,8,9,13
	Home Work	2	7%(14)	3,10	1,2,3,4,5,6,7,8,9
	Assignments in College	1	5%(5)	6	4,5
	Seminar	1	7%(7)	All Weeks	1,2,3,4,5,6,7,8,9,10,11,12
Summative assessment	Midterm Exam	2 h	10%(10)	7	1-6
	Final Exam	3h	50(50%)	16	all
Total assessment			100		

Delivery Plan (Weekly Syllabus)	
	Material Covered
Week 1	Introduction: General overview of the concept of Human Rights
Week 2	Roots of Human Rights and their evolution in human history / Development of Human Rights protection
Week 3	The International Community and Contemporary Human Rights / UN mechanisms for Human Rights protection
Week 4	Human duties and limitations on exercising Human Rights
Week 5	International organizations and bodies concerned with Human Rights / Professional ethics
Week 6	Student Disciplinary Law in the Ministry of Higher Education and Scientific Research Institutions
Week 7	Mid-course Exam
Week 8	Concept and History of Democracy
Week 9	Features and components of the Democratic system
Week 10	Civil society institutions and democracy / Elections
Week 11	The relationship between Human Rights and Democracy
Week 12	Crimes of Genocide
Week 13-14	Guarantees of Freedoms and Public Rights
Week 15	Good Governance
Week 16	Final Exam

Learning and Teaching Resources		
	Text	**Available in the Library?**
Required Texts	حقوق الانسان ، تطورها ومضامينها وحمايتها ، د. رياض عزيز هادي	نعم
Recommended Texts	حقوق الانسان والديمقراطية والحريات د. ماهر صبري	نعم
Websites	لا يوجد	

APPENDIX:

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

ملاحظة: هذا النموذج تم وضعه وتقديمه من قبل مديرية ضمان الجودة في وزارة التعليم العالي والبحث العلمي

MODULE DESCRIPTOR FORM

Module Information			
Module Title	DISCRETE STRUCTURES	**Module Delivery**	
Module Type	CORE	**Theory ✓** **Seminar ✓**	
Module Code	IT1202		
ECTS Credits	6		
SWL (hr/sem)	150		
Module Level	1	**Semester of Delivery**	2
Administering Department	Information technology	**College**	College of Sciences
Module Leader	ELAF ADIL	**e-mail**	Elaf.Adel.Abbas@uowa.edu.iq
Module Leader's Acad. Title	Lecturer	**Module Leader's Qualification**	PhD
Module Tutor	ELAF ADIL	**e-mail**	Elaf.Adel.Abbas@uowa.edu.iq
Peer Reviewer name	Asst. Prof. Dr Haider Mohammad	**e-mail**	hayder.alghanami@uowa.edu.iq
Review Committee Approval	2024-1-20	**Version Number**	1

Relation With Other Modules			
Prerequisite module	None	**Semester**	None
Co-requisites module	None	**Semester**	None

أ.م.د.ضياء حسين نوفل
٢٠٢٥/٢٠٢٤

جامعة وارث الانبياء
عمادة كلية العلوم
COLLEGE OF SCIENCES

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د.ضياء حسين نوفل
٢٠٢٥/٢٠٢٤

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Aims	1. Provide students with basic information about digital logic and logic circuits. 2. Increasing students' horizons in the fields of computer science and digital development. 3. Developing the students' English language by teaching the subject in English. 4. Providing students with applied and experimental skills through practical subjects and laboratories. 5. Familiarize students with the latest developments in the fields of different sciences and the technology emanating from them. 6. Developing the student's ability to research and providing him with scientific research contexts. 7. Develop students' ability to analyze and link information and conclusion. 8. Enhancing the scientific spirit in the interpretation of phenomena, discussion and dialogue. 9. Consolidation of conviction in the integration of sciences and their universality towards the truth. 10. Working on refining the student's personality and discovering his inclinations and talents through scientific and cultural activities. 11. Enhancing the spirit of teamwork through the participation of students in laboratory work or the completion of joint scientific research. Establish values and ideals Higher among them is respect for instructions, discipline, respect for the institution to which the student belongs, and preservation of its property.
Module Learning Outcomes	1. Knowing the numerical number systems used in logical circuits and performing arithmetic operations on them. 2. Knowledge of logical circuits and their design methods. 3. Simplify logic circuits by simplifying their equations. 4. Full knowledge of digital meters, dividers and other electronic circuits. 5. Full knowledge of the use of signs and their representation in binary numbers. 6. Full knowledge of how to convert between number systems used in numerical operations. 7. How to integrate digital portals together and methods of calculating their outputs. 8. Design counters and dividers and link them together
Indicative Contents	1. Foundational knowledge in digital logic and logic circuits for computer science and digital development: ○ Introduction to digital logic and its significance in computer science and digital development. ○ Understanding the principles and components of logic circuits ○ Exploring the role of logic circuits in data processing and information storage. 2. Broadening horizons in computer science and digital development: ○ Exploration of various fields and applications within computer science and digital development.

	○ Introduction to key concepts and technologies shaping the industry. ○ Understanding the impact of computer science on society and everyday life. 3. Practical application and experimental skills through hands-on work in laboratories: ○ Engaging in practical subjects and laboratory sessions to gain handson experience. ○ Applying theoretical knowledge to design and build logic circuits. ○ Developing skills in breadboarding, prototyping, troubleshooting, and circuit analysis. 4. Keeping students updated with the latest developments in science and technology: ○ Discussing recent advancements in various scientific fields related to digital logic and logic circuits. ○ Exploring emerging technologies and their impact on computer science and digital development. ○ Encouraging students to stay informed through literature review and research. 5. Enhancing research skills and providing scientific research contexts: ○ Developing research methodologies and skills necessary for scientific investigation. ○ Providing opportunities for students to conduct research projects related to digital logic. ○ Guiding students in collecting and analyzing data, drawing conclusions, and presenting research findings. 6. Developing analytical thinking, scientific spirit, teamwork, and instilling values of respect, discipline, and responsibility: ○ Cultivating analytical thinking skills to analyze and link information in the context of digital logic. ○ Promoting a scientific spirit by encouraging interpretation of phenomena and engaging in discussions and dialogues. ○ Fostering teamwork through collaboration in laboratory work and joint scientific research projects. ○ Instilling values of respect for instructions, discipline, and preservation of institutional property.

Learning and Teaching Strategies	
Strategies	▪ Giving lectures ▪ Scientific discussions and dialogues and asking questions

Student Workload (SWL)			
Structured SWL (h/sem)	45	**Structured SWL (h/w)**	3
Unstructured SWL (h/sem)	102	**Unstructured SWL (h/w)**	7
Total SWL (h/sem)	147 + 3 final = 150		

Module Evaluation

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	5%(10)	All Weeks	1,2,3,4
	Onsite Assignments	3	5%(15)	3,5,8,11	4,6,7
	Report	1	5%(5)	13	all
	Homework	5	2%(10)	4,7,9,10	1,2,3,4,5,6
Summative assessment	**Midterm Exam**	2h	10	7	
	Final Exam	3h	50	15	
Total assessment			100		

Delivery Plan (Weekly Syllabus)	
	Material Covered
Week 1	Introduction
Week 2	Mathematical logic
Week 3	Mathematical logic
Week 4	Functions
Week 5	Composition of Function
Week 6	Propositions
Week 7	Mathematical Proof
Week 8	Set Theory 1
Week 9	Set Theory 2
Week 10	Set Theory 3
Week 11	Representing Sets
Week 12	Combining Propositions 1
Week 13	Combining Propositions 2
Week 14	Combining Propositions 3
Week 15	Combining Propositions 4

Learning and Teaching Resources

	Text	**Available in the Library?**
Required Texts	Norman L. Biggs (2002-12-19). Discrete Mathematics. Oxford University Press. ISBN 978-0-19-850717-8.	no
Recommended Texts	Susanna S. Epp (2010-08-04). Discrete Mathematics With Applications. Thomson Brooks/Cole. ISBN 978-0-495- 39132-6.	no
Websites		

APPENDIX:

GRADING SCHEME				
Group	**Grade**	**Mark**	**Marks (%)**	**Definition**
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(0-44)	Considerable amount of work required
Note:				

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

MODULE DESCRIPTOR FORM

Module Information			
Module Title	ENGLISH LANGUAGE	**Module Delivery**	
Module Type	SUPPLEMENT	☒ **Lecture**	
Module Code	UOWA105		
ECTS Credits	2		
SWL (hr/sem)	50		
Module Level	UG1	**Semester of Delivery**	2
Administering Department	Information technology	**College**	College Sciences
Module Leader	Bandar Abdul abbas Almankoshi	**e-mail**	bandar@uowa.edu.iq
Module Leader's Acad. Title	Assistant Lecturer	**Module Leader's Qualification**	M.Sc.
Module Tutor	Bandar Abdul abbas Almankoshi	**e-mail**	bandar@uowa.edu.iq
Peer Reviewer Name	Lecturer Maky H.Abdulraheem	**e-mail**	maky.h@uowa.edu.iq
Review Committee Approval	2024-1-20	**Version Number**	1

Relation With Other Modules			
Prerequisite module	None	**Semester**	**None**
Co-requisites module	None	**Semester**	**None**

جامعة وارث الأنبياء
عمادة كلية العلوم

جامعة وارث الأنبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

أ.م.د شيماء حسين نوفل
٢٠٢٥/٢٠٢٤

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Aims	This course aims at: 1- Enhancing a mastery over the basic structure of a standard English Sentence. and the type of language used in scientific fields of study. 2- Knowing a good bit of information about the basic phrases in English Language regarding their formation, position in sentence word order, uses in real life situation as related to their field of work. 3- Focusing on the difference between simple and continuous present and past tenses as related to their study and career. 4- Enabling students to write certain types of expressions and texts useful for their field of study and future career. 5- Stimulating and directing students to speak and practice English language correctly, asserting the type of language used in real life situations and scientific field of study. 6- Specifying points of weakness in students' performance, trying to amend them. 7- Building a type of scheme in students' minds about what writing and speaking standard English language is supposed to be. 8- Forcing students to think critically while doing the assignments, quizzes and other similar activities.
Module Learning Outcomes	The student would be able to: 1- Speak and write a good standard sentence or type of English Language. 2- Differentiate between types of basic tenses. 3- Have a fluency while speaking the English Language. 4- Write acceptable formal and informal texts. 5- Comprehend the idea behind string of words in a sentence. 6- Work collectively within a teamwork.
Indicative Contents	Indicative content includes the following: - Word order: Statements, questions imperatives (command, request, instructions). - Phrases: Nouns, Adjectives, Adverbs, Verbs, Prepositions. - Verbs: Tenses (Form and basic uses), Passive. - Knowing how to say and write some useful texts. - Some text for reading comprehension and videos or recordings for listening. - Basic guide lines in writing a summary, letters, paragraphs, CV. - Topics for discussion.

Learning and Teaching Strategies	
Strategies	The program is designed to have two theoretical hours in points related to grammar and other three hours for the sake of practicing including doing the exercises. Before an exam, the student will have the chance to review the previous given materials. The practical hours include some basic information in pronunciation, reading, speaking, listening and writing skills. The program instructor will follow a mixture of traditional and communicative approaches to achieve the above-mentioned aims. The students will be asked to do some exercises and quizzes in relation to grammar. They could be divided into groups having certain duties related to different practical activities to be done by them. Each student will have his own evaluation which will raise the grade of each group work as a whole. The best group work will be rewarded at the end of the semester with some additional marks for their good performance during the course. Doing quizzes and assignments inside the classroom are very important to adjust some important grammatical points. To ensure self-learning, some websites and parts of texts related to the given lectures are going to be given to them. Certain activities such as speaking and listening are going to be given forward so as to be ready for the duties while practicing them inside the classroom.

Student Workload (SWL)			
Structured SWL (h/sem.)	26	**Structured SWL (h/w)**	1.75
Unstructured SWL (h/sem.)	21	**Unstructured SWL (h/w)**	1.4
Total SWL (h/sem.)	47 + 3 final = 50		

<table>
<tr><th colspan="6">Module Evaluation</th></tr>
<tr><th colspan="2"></th><th>Time/Number</th><th>Weight (Marks)</th><th>Week Due</th><th>Relevant Learning Outcome</th></tr>
<tr><td rowspan="4">Formative Assessment</td><td>Quizzes</td><td>3</td><td>5%(11)</td><td>3, 10</td><td>1, 2, 4, 6</td></tr>
<tr><td>Reports</td><td>1</td><td>10%(11)</td><td>9</td><td>2, 5</td></tr>
<tr><td>Onsite Assignments</td><td>1</td><td>5%(8)</td><td>6, 11</td><td>4,5</td></tr>
<tr><td>Homework</td><td>2</td><td>1%(10)</td><td>All Weeks</td><td>2,4</td></tr>
<tr><td rowspan="2">Summative Assessment</td><td>Midterm Exam</td><td>2hr</td><td>10</td><td>9</td><td></td></tr>
<tr><td>Final Exam</td><td>3hr</td><td>50% (50)</td><td>17</td><td></td></tr>
<tr><td colspan="3">Total Assessment</td><td colspan="3">100</td></tr>
</table>

<table>
<tr><th colspan="2">Delivery Plan (Weekly Syllabus)</th></tr>
<tr><th></th><th>Material Covered</th></tr>
<tr><td>Week 1</td><td>Word Order in Standard English – Statement: Positive vs. Negative, Questions.</td></tr>
<tr><td>Week 2</td><td>Word Order in Standard English – Imperative Sentence: Instructions, Request, Command.</td></tr>
<tr><td>Week 3</td><td>Nouns: singular nouns vs. plural nouns, Gender, Pure Nouns-Derived nouns, Articles.</td></tr>
<tr><td>Week 4</td><td>Nouns: Pronouns, Expressions of Quantity, Position in Word Order.</td></tr>
<tr><td>Week 5</td><td>Adjectives: Pure adjectives -Derived adjectives, Comparison Degrees, Position in Word Order.</td></tr>
<tr><td>Week 6</td><td>Adverbs: Pure adverbs-derived adverbs, Position in Word Order, Adverbs of Degree.</td></tr>
<tr><td>Week 7</td><td>Mid-Term Exam</td></tr>
<tr><td>Week 8</td><td>Expressing: Time, conditional, result, reason, purpose, contrast.</td></tr>
<tr><td>Week 9</td><td>Prepositions: Uses, position in Word Order.</td></tr>
<tr><td>Week 10</td><td>Verbs: Tenses-Present (Simple vs. Continuous).</td></tr>
<tr><td>Week 11</td><td>Verbs: Tenses-Past (Simple vs. Continuous).</td></tr>
<tr><td>Week 12</td><td>Verbs: Futurity, Modals (can, may, should, etc.).</td></tr>
<tr><td>Week 13</td><td>Verbs: Passive Voice.</td></tr>
<tr><td>Week 14</td><td>General Review and some Additional Notes.</td></tr>
<tr><td>Week 15</td><td>Final Exam</td></tr>
</table>

Delivery Plan (Weekly Practice Syllabus)	
Week	**Material Covered**
Week 1	Alphabetical Order, Word Order: Reforming Sentences, Introducing Oneself, Writing Simple Sentences.
Week 2	Jobs and Specialties in a Hospital. Listening 1, Writing Different Types of Sentences, Describing something around.
Week 3	Assignment 1, Reading and Writing Numbers in Different Situations. Reading passage 1
Week 4	Different Types of Derived Nouns and How to Use them in a Sentence. Listening 2, Writing a Summary.
Week 5	Countries, Nationalities, Languages, Parts of Human Body, listening 3, Writing a Short Report of an Experiment.
Week 6	Assignment 2, Days, Months, Colors, Reading Passage 2, Writing a Letter.
Week 7	Clothes, Continents, Pronouncing the suffix (s), Listening 4 Writing a Good Paragraph.
Week 8	Expressing: Time, conditional, result, reason, purpose, contrast.
Week 9	Things in the Lab\Hospital, Reading Passage 3, Pronouncing the suffix (-ed), Writing a Good paragraph.
Week 10	Verbs: Tell-Say, Reply-Answer-respond, Fill-Full, Listening 5, Punctuation Marks.
Week 11	Assignment 3, Some Silent Letters in English Words, Reading passage 4
Week 12	Like-love, listening 6, Performing Certain Situation 1, a Topic for Discussion.
Week 13	Performing Certain Situation 2, Reading Passage 5, Writing a Good CV.
Week 14	Performing Certain Situation 3, Writing about Future Dreams or Plans.
Week15	Final Exam

Learning and Teaching Resources		
	Text	**Available in the Library?**
Required Texts	John and Liz Soars, New Headway Plus, United Kingdom: Oxford University Press.	Yes
Recommended Texts	Baily, Stephen. 2011. *Academic writing*. London: Rutledge. Hewings, Martin. 2012. *Advanced grammar in Use.* United Kingdom: Cambridge university Press.	Yes Yes
Websites	- https://www.oxfordonlineenglish.com/ - https://www.grammarly.com/ - https://www.softschools.com/language_arts/reading_comprehension/science/8/magnetism/ - https://eslflow.com/	

GRADING SCHEME				
Group	**Grade**	**Mark**	**Marks (%)**	**Definition**
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(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 fails" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.

ملاحظة: هذا النموذج تم وضعه وتقديمه من قبل مديرية ضمان الجودة في وزارة التعليم العالي والبحث العلمي

MODULE DESCRIPTION FORM

Module Information			
Module Title	**Programming Fundamentals II**	**Module Delivery**	
Module Type	**Core**	☒ **Lecture** ☒ **Practical**	
Module Code	**IT203**		
ECTS Credits	**7**		
SWL (hr/sem)	**175**		
Module Level	UGI	**Semester of Delivery**	2
Administering Department	Information Technology	College	Science
Module Leader	Mohsin Hasan Hussein	**e-mail**	mohsin.h@uokerbala.edu.iq
Module Leader's Acad. Title	Assistant Prof.	**Module Leader's Qualification**	Ph.D.
Module Tutor	Assistant Prof. Dr. Mohsin Hasan Hussein	**e-mail**	mohsin.h@uokerbala.edu.iq
Peer Reviewer Name	Assist. Prof. Dr. Haider Mohammed	**e-mail**	hayder.alghanami@uowa.edu.iq
Scientific Committee Approval Date	2024-1-20	**Version Number**	V1

Relation with other Modules			
Prerequisite module	**Programming Fundamentals 2**	**Semester**	1
Co-requisites module	-	**Semester**	...

أ.م.د شيماء حسين نوفل
٢٠٢٤/٢٠٢٥

جامعة وارث الانبياء
عمادة كلية العلوم

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د شيماء حسين نوفل
٢٠٢٤/٢٠٢٥

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Objectives	The following are some key aims and benefits of studying Programming Fundamentals II: 1. Problem Solving: Teach students how to analyze problems and develop algorithms to solve them. Emphasize problem-solving techniques, algorithm design, and decomposition of complex problems into smaller, manageable parts. 2. Debugging and Testing: Teach students how to debug and test their programs to identify and fix errors. Explore techniques for error detection, debugging tools, and strategies for writing effective test cases. 3. Data Structures: Introduce students to fundamental data structures such as arrays, stacks, queues, structures. Explore their properties, implementation, and usage in solving programming problems. 4. Modular Programming: Introduce the concept of modular programming, including the use of functions, parameter passing, and code reuse. Emphasize the importance of modular design and writing reusable and maintainable code. 5. Programming Best Practices: Introduce students to programming best practices and coding standards, including code documentation, naming conventions, code formatting, and code optimization techniques. 6. Files Input and Output: Teach students how to interact with the user and handle input/output operations, including reading from and writing to files, standard input/output, and error handling. 7. Introduction to Object-Oriented Programming (OOP): Introduce the principles and concepts of OOP, including classes.

Module Learning Outcomes	The following are some common learning outcomes for an Programming Fundamentals II : 1. Effective Code Writing: Write clear, well-structured, and readable code that follows coding standards and best practices, including proper indentation, meaningful variable names, and appropriate comments. 2. Use of Data Structures: Apply appropriate data structures, such as arrays, linked lists, stacks, and queues, to store and manipulate data effectively in programming problems. 3. Modular Design and Reusability: Design and implement modular programs by breaking them into reusable functions or methods, facilitating code reuse, improving maintainability, and promoting good software engineering practices. 4. Debugging and Testing Skills: Use debugging techniques and tools to identify and fix errors in programs. Develop effective test cases and perform testing to ensure program correctness and reliability. 5. Understanding of Object-Oriented Programming (OOP) Concepts.
Indicative Contents	The indicative contents of an Programming Fundamentals II module have a list of common topics that shown below : 1- Modular Programming: **[25 hrs]** Functions and procedures, Scope and lifetime of variables, Parameter passing mechanisms. 2- Data Structures: **[25 hrs]** Arrays, Strings and lists, Structures, Stacks and queues. 3- Input and Output: **[15 hrs]** Standard input/output, Reading from and writing to files, Error handling and exception handling. 4- Debugging and Testing: Common types of programming errors, Debugging techniques and tools. **[20 hrs]** 5- Object-Oriented Programming (OOP) Concepts: Classes and objects. **[5 hrs]**

Learning and Teaching Strategies

Strategies	To teaching an Programming Fundamentals II module, various strategies can be employed to facilitate effective learning and engagement. Here are some learning and teaching strategies commonly used in Programming Fundamentals II module: 1- Lectures: Delivering lectures to present theoretical concepts, principles, and foundational knowledge of Programming Fundamentals II. Lectures can include visual aids, examples, and demonstrations to enhance understanding. 2- Interactive Discussions: Encourage students to actively participate in discussions by asking questions, sharing their thoughts, and engaging in peer-to-peer learning. Discussions can focus on challenging concepts, real-world applications, or case studies related to Programming Fundamentals II. 3- Hands-on Lab Sessions: Conduct practical lab sessions where students can gain hands-on experience with Programming Fundamentals II, commands, and programming exercises. These sessions provide an opportunity to reinforce theoretical concepts and develop practical skills. 4- Group Projects: Assign group projects that involve designing, implementing, and evaluating components of an Programming Fundamentals II. Group projects promote teamwork, problem-solving, and practical application of operating system concepts. 5- Online Resources and Tutorials: Provide access to online resources, tutorials, and interactive learning materials related to Programming Fundamentals II. This allows students to explore additional content, reinforce their understanding, and self-assess their progress. 6- Assessments and Feedback: Use a variety of assessment methods such as quizzes, assignments, projects, and exams to evaluate students' understanding of Programming Fundamentals II concepts. Provide timely and constructive feedback to help students improve their knowledge and skills.

Student Workload (SWL)			
Structured SWL (h/sem)	75	Structured SWL (h/w)	5
Unstructured SWL (h/sem)	97	Unstructured SWL (h/w)	6.5
Total SWL (h/sem)	172 + 3 final = 175		

Module Evaluation					
		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	5	1% (5)	All Weeks	1,2,3,4
	Assignments	5	1%(5)	All Weeks	All Outcome
	Lab	5	4% (20)	All Weeks	All Outcome
	Home Work	5	2%(10)	All Weeks	All Outcome
Summative assessment	Midterm Exam	2hr	10%(10)	9	
	Final Exam	3hr	50% (50)	17	
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)	
	Material Covered
Week 1	Functions
Week 2	Function Types
Week 3	The concept of Recursion
Week 4	Array
Week 5	1D array

Week 6	2D array (Matrix)
Week 7	Array of Characters (Strings)
Week 8	String Processing
Week 9	Midterm Exam
Week 10	Arrays and functions
Week 11	Structures
Week 12	Array of structures and Nested Structures
Week 13	Stack and Queue
Week 14	Pointers
Week 15	Files
Week 16	**Preparatory week before the final Exam**

Delivery Plan (Weekly Lab. Syllabus)

	Material Covered
Week 1	Writing Codes using Functions
Week 2	Writing Codes using Function Types
Week 3	Writing Codes using The concept of Recursion
Week 4	Writing Codes using Arrays
Week 5	Writing Codes using 1D arrays
Week 6	Writing Codes using 2D array (Matrix)s
Week 7	Writing Codes using Array of Characters (Strings)
Week 8	Writing Codes using String Processing
Week 9	Midterm Exam
Week 10	Writing Codes using Arrays and functions
Week 11	Writing Codes using Structures
Week 12	Writing Codes using Array of structures and Nested Structures
Week 13	Writing Codes using Stack and Queue

Week 14	Pointers
Week 15	Files

Learning and Teaching Resources

	Text	Available in the Library?
Required Texts	**C++: The Complete Reference, Fourth Edition, Herbert Schildt.**	**Yes**
Recommended Texts	**The C++ Programming Language , Third Edition , Bjarne Stroustrup.**	**Yes**
Websites	**https://stackoverflow.com/**	

Grading Scheme

Group	Grade	Mark	Marks %	Definition
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX –** Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(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.

MODULE DESCRIPTOR FORM

Module Information			
Module Title	System Administration	**Module Delivery**	
Module Type	Core	Lecture ✓ Practical ✓	
Module Code	IT1204		
ECTS Credits	7		
SWL (hr/sem)	175		
Module Level	1	**Semester of Delivery**	2
Administering Department	Information technology	**College**	College of Sciences
Module Leader	Maki Hussein Abd Alraheem	**e-mail**	Maky.h@uowa.edu.iq
Module Leader's Acad. Title	Lecturer	**Module Leader's Qualification**	PhD
Module Tutor	Asst. Lecturer Ali Abdulhussein Ibrahim	**e-mail**	ali.abdulhussein19@uowa.edu.iq
Peer Reviewer name	Asst. Prof. Dr Haider Mohammed	**e-mail**	hayder.alghanami@uowa.edu.iq
Review Committee Approval	2024-1-20	**Version Number**	1

Relation With Other Modules			
Prerequisite module	None	**Semester**	None
Co-requisites module	None	**Semester**	None

أ.م.د ضياء حسين نوفل
٢٠٢٤/٢٠٢٥

جامعة وارث الانبياء
عمادة كلية العلوم

جامعة وارث الانبياء
كلية العلوم
قسم تكنولوجيا المعلومات

أ.م.د ضياء حسين نوفل
٢٠٢٤/٢٠٢٥

Department Head Approval **Dean of the College Approval**

Module Aims, Learning Outcomes and Indicative Contents	
Module Aims	1. To provide a comprehensive understanding of command-line interfaces, programming languages, open-source software and software licenses, data backup, and data encryption. 2. To differentiate and compare various elements within each topic, such as CLI types, elements of programming languages, different software licenses, backup methods, and encryption types. 3. To understand and evaluate the role and importance of these elements in the field of computer science and daily computing.
Module Learning Outcomes	1. Understand and articulate the concept of command-line interfaces, their types, and their comparison with graphical user interfaces. 2. Understand the basic elements of programming languages, including syntax, type systems, standard libraries, specifications, and implementations. 3. Understand the concept of open-source software and be able to distinguish between open-source and proprietary software licensing models. 4. Comprehend the importance of data backup and different backup methods. 5. Understand the fundamental principles of data encryption, the different types, and their application in operating systems and third-party programs.
Indicative Contents	1. Command-line interfaces: Definition, types, comparison with GUI, shell CLI. 2. Programming languages: Basic elements, syntax, type systems, standard libraries, specifications, and implementations. 3. Open-source software and software licenses: Definition of open-source software, comparison of open-source licenses, proprietary software licensing models, software cracking and piracy. 4. Data backup: Importance of data backup, data backup concepts, backup methods, backup media management. 5. Data encryption: Introduction to data encryption, importance of encryption, basics of encryption, types of data encryption on PC, OS built-in and thirdparty encryption programs.

Learning and Teaching Strategies	
Strategies	The learning and teaching strategies for studying the database subject in an IT department involve a balanced approach of theoretical understanding and practical application. Lectures, interactive discussions, and case studies provide

	the necessary theoretical foundation. Practical exercises, group work, and projects enable hands-on experience with database management systems. Workshops, demos, and industry examples offer real-world insights. Online resources, assessments, and feedback aid in reinforcing learning. Virtual labs and continuous learning emphasize practical skills development and staying updated with industry trends. These strategies ensure a comprehensive understanding of databases and their relevance in the IT field.

Student Workload (SWL)			
Structured SWL (h/sem)	60	**Structured SWL (h/w)**	4
Unstructured SWL (h/sem)	112	**Unstructured SWL (h/w)**	7.5
Total SWL (h/sem)	172 + 3 final = 175		

Module Evaluation					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	4% (8)	3,10	1,2,4
	Lab	4	5% (20)	3,5,7,10	1,2,3,4
	Project	1	4% (4)	13	All outcome
	Homework	4	2% (8)	6,11	All outcome
Summative assessment	**Midterm Exam**	2 h	10% (10)	7	
	Final Exam	3 h	50% (50)	15	
Total assessment			100% (100)		

Delivery Plan (Weekly Syllabus)	
	Material Covered
Week 1	Introduction, types of CLI, operating system command-line interfaces, application command-line interfaces.
Week 2	Comparison between CLI and GUI, shell CLI.
Week 3	Introduction to programming languages, elements of programming languages, syntax.
Week 4	Type systems, standard library.
Week 5	Specification and implementation in programming languages.
Week 6	Introduction to open-source software, common open-source licenses.
Week 7	Introduction to common open-source licenses.
Week 8	Proprietary software licensing models, software cracking and piracy.
Week 9	Introduction to data backup, data backup concepts, backup methods.
Week 10	More on backup types
Week 11	Backup media management. Backup media management.
Week 12	Introduction to encryption, the importance of encryption, basics of encryption.
Week 13	Introduction to encryption, the importance of encryption, basics of encryption.
Week 14	Types of data encryption on PC, OS built-in encryption programs.
Week 15	Preparatory week before the final Exam

Delivery Plan (Weekly Lab. Syllabus)	
weeks	**Material Covered**
Week 1	Familiarize with the Command Prompt and basic CLI commands such as dir, cd, copy, del, move.
Week 2	Practice creating, navigating, renaming, and deleting directories and files using CLI.
Week 3	Learn advanced file operations like finding files, comparing files, and using wildcards.
Week 4	Understand the concept of input and output redirection, learn to use pipes to combine commands.
Week 5	Introduction to batch files, create simple batch scripts.
Week 6	Learn to use variables in batch programming, receive input from users.
Week 7	Understand and implement if-else logic in batch programming.
Week 8	Understand and implement loop structures such as for and while loops in batch programming.
Week 9	Learn to create and use functions in batch programming.
Week 10	Understand error handling and exception management in batch programming.
Week 11	Write advanced batch scripts combining learned elements
Week 12	Learn how to automate repetitive tasks using batch scripts.
Week 13	Understand and use CLI commands for network operations such as ping, ipconfig, and netstat.
Week14	Learn to create batch scripts for network operations.
Week 15	Finalize and present a self-created project utilizing learned skills, review key learning points.

Learning and Teaching Resources		
	Text	**Available in the Library?**
Required Texts	"Computer Organization and Architecture" by William Stallings	no
Recommended Texts		
Websites	https://www.tutorialspoint.com/basics_of_computer_science/index.htm	

GRADING SCHEME				
Group	**Grade**	**Mark**	**Marks (%)**	**Definition**
Success Group (50 - 100)	**A -** Excellent	Excellent	90 - 100	Outstanding Performance
	B - Very Good	Very Good	80 - 89	Above average with some errors
	C - Good	Good	70 - 79	Sound work with notable errors
	D - Satisfactory	Fair / Average	60 - 69	Fair but with major shortcomings
	E - Sufficient	Pass / Acceptable	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	Fail (Pending)	(45-49)	More work required but credit awarded
	F – Fail	Fail	(0-44)	Considerable amount of work required
Note:				

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