	Ministry of Higher Education and Scientific Research - Iraq University of Technology-Iraq Department of Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	CODING TECHNIQUES			**Module Delivery**	
Module Type	CORE			**Theory** **Lecture** **Tutorial**	
Module Code	COTE124				
ECTS Credits	4				
SWL (hr/sem)	100				
Module Level	1		**Semester of Delivery**	2	
Administering Department	Type Dept. Code		**College**	Type College Code	
Module Leader			**e-mail**		
Module Leader's Acad. Title	Professor		**Module Leader's Qualification**	Ph.D.	
Module Tutor	Nuha jameel ibrahim		**e-mail**	Nuha.j.ibrahim@uotechnology.edu.iq	
Peer Reviewer Name			**e-mail**		
Review Committee Approval	01/06/2023		**Version Number**	1.0	

Relation With Other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	Information Theory	**Semester**	1
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	some common objectives that coding theory modules typically aim to achieve: 1. The course aims to teach students the properties of codes and their respective fitness for specific applications. 2. Developing the programming skills and knowledge of students in codes are used for data compression, cryptography, error-correction, and networking. 3. Understanding Error Correction: Coding theory aims to provide a solid understanding of error correction techniques in communication systems. 4. Code Decoding: Coding theory aims to provide techniques for decoding received codewords to recover the original information. Students learn about decoding algorithms. 5. Performance Analysis: The module aims to analyze the performance of error-correcting codes. Students learn how to calculate and evaluate metrics like error probability, bit error rate, and throughput. They also study trade-offs between coding complexity, code rate, and error correction capability. 6. Applications: Coding theory modules often aim to demonstrate the practical applications of coding theory in various fields. This may include applications in telecommunications, data compression , data storage, cryptography, and network coding. Students learn how coding theory contributes to improving data reliability, security, and efficiency in real-world scenarios. 7. Mathematical Foundations: Coding theory relies on mathematical concepts and tools. The module aims to provide a solid mathematical foundation for understanding and analyzing coding theory. This includes learning relevant mathematical techniques, proofs, and theorems. 8. Problem Solving and Implementation: Coding theory modules often include practical assignments or projects to enhance problem-solving skills and implementation abilities. Students may be tasked with designing and implementing error-correcting codes, simulating their performance, and analyzing the results.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	After studying the coding course, students will be able to: 1. Explain concept of Dependent & Independent Source, measure of information, Entropy, Rate of lnformation and Order of a source 2. Represent the information using Shannon Encoding, Shannon Fano, binary Huffman Encoding, ternary Huffman Encoding Algorithms 3. Determine a codeword comprising of the check bits computed using Linear Block codes, cyclic codes & convolutional codes 4. Understanding of basic concepts: Students should develop a solid understanding of fundamental concepts in coding theory, such as error detection and correction, coding schemes, and encoding and decoding algorithms.

	5. Students should be able to analyze and apply different error detection and correction codes, including parity codes, Hamming codes. 6. Proficiency in coding theory algorithms: Students should develop the skills to implement encoding and decoding algorithms for different coding schemes. They should be able to apply these algorithms to detect and correct errors in data transmission or storage systems. 7. Application of coding theory in practical scenarios: Students should be able to apply coding theory principles to real-world scenarios, such as data communication channels, wireless networks, storage systems, and error-prone environments. They should understand how coding theory techniques can improve the reliability and efficiency of these systems. 8. Critical thinking and problem-solving skills: Students should develop critical thinking skills to analyze complex problems in coding theory and propose solutions. They should be able to evaluate the performance of different coding schemes, identify potential issues, and propose improvements or optimizations.
Indicative Contents المحتويات الإرشادية	some indicative contents that you might find in a coding theory module: 1. Introduction to coding theory: • Basic concepts and definitions • History and applications of coding theory • Importance of error detection and correction 2. Coding techniques • Fixed length coding • Variable length coding 3. Linear codes: • Definition and properties of linear codes • Generator and parity-check matrices • Syndrome decoding • Hamming codes • Reed-Solomon codes 4. Cyclic codes: • Definition and properties of cyclic codes • Generator and parity-check polynomials • Syndrome decoding for cyclic codes • BCH codes 5. Error detection and correction techniques:

	• Hamming distance and error detection/correction capabilities • Minimum distance of a code 6. Convolutional codes: • Basic structure and properties of convolutional codes • Encoding and decoding using the Viterbi algorithm • Maximum likelihood decoding 7. Coding theory applications: • Channel coding and error control in communication systems • Error correction in storage systems • Error detection and correction in digital transmission
Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	some effective strategies: 1. Lecture-Based Instruction: Conducting traditional lectures can be a valuable strategy for introducing coding theory concepts and theories. Provide clear explanations, examples, and visual aids to help students grasp the fundamental concepts. 2. Active Learning: Incorporate active learning strategies to engage students in the learning process. This can include group discussions, problem-solving activities, and hands-on coding exercises. Encourage students to participate actively, ask questions, and collaborate with their peers. 3. Practical Coding Assignments: Assign coding projects or assignments that allow students to apply the coding theory principles they have learned. This hands-on experience helps reinforce their understanding and develops their coding skills. Provide feedback and guidance throughout the process. 4. Real-World Applications: Showcasing real-world applications of coding theory can enhance students' motivation and understanding. Discuss how coding theory is used in error correction, data compression, cryptography, and other relevant fields. Explore case studies or examples that demonstrate the practical applications of coding theory. 5. Online Resources and Interactive Tools: Utilize online resources and interactive tools to supplement classroom instruction. Point students to coding theory tutorials, simulations, and coding platforms where they can practice coding techniques and experiment with different algorithms. 6. Collaborative Learning: Encourage collaborative learning by assigning group projects or problem-solving tasks. This fosters teamwork and communication skills while allowing students to explore coding theory concepts together.

	7. Assessment and Feedback: Provide regular assessments, such as quizzes or exams, to gauge students' understanding of coding theory. Offer constructive feedback to help students identify areas for improvement. Consider incorporating both individual and group assessments to assess both individual comprehension and teamwork skills.

Student Workload (SWL)
الحمل الدراسي للطالب

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	63	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	37	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	3
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	100		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	15% (15)	5, 10,12	LO #1, 2, 7 and 8
	Assignments	2	10% (10)	2, 11	LO # 3, 4, 6 and 7
	Projects / Lab.				
	Report	1	10% (10)	13	LO # 5, 6 and 8
Summative assessment	**Midterm Exam**	2hr	15% (15)	7	LO # 1-7
	Final Exam	2hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

	Material Covered
Week 1	Principles of information theory.
Week 2	Introduction to coding techniques.
Week 3	Entropy, Average length of a code.
Week 4	code efficiently, code redundancy.
Week 5	Fixed length coding.

Week 6	Variable length coding, Shannon-Fano coding algorithm.
Week 7	Mid Exam
Week 8	Lempel-Ziv coding algorithm.
Week 9	Arithmetic coding algorithm
Week 10	Huffman coding • Huffman Binary coding. • Huffman Ternary coding.
Week 11	Extension of a source
Week 12	Error Control Types of errors, Single-bit Error, Burst Error
Week 13	Hamming code. • Error detection • Error Correction
Week 14	Error detection: • One-dimension Parity Check • Two-dimension Parity Check • Checksum • Cyclic Redundancy Checks (CRC), CRC performance.
Week 15	Single-bit error correction algorithm
Week 16	**Final Exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	
Week 2	
Week 3	
Week 4	
Week 5	
Week 6	
Week 7	

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Coding and Information Theory, Richard Wesley Hamming Prentice-Hall, 1986	No
Recommended Texts	Information Theory and Coding, Dr. J. S. Chitode Technical Publications, Jan 1, 2021 An introduction to Information theory by fazlollah	No
Websites		

APPENDIX:

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

NB Decimal places above or below 0.5 will be rounded to the higher or lower full mark (for example a mark of 54.5 will be rounded to 55, whereas a mark of 54.4 will be rounded to 54. The University has a policy NOT to condone "near-pass fails" so the only adjustment to marks awarded by the original marker(s) will be the automatic rounding outlined above.