Academic Program Description Form

University Name: Tikrit University

Faculty/Institute: College of Computer Science and Mathematics

Scientific Department: Cybersecurity Department (Established in the academic year 2024-2025)

Academic or Professional Program Name: Bachelor of Cybersecurity

Final Certificate Name: Bachelor in Cybersecurity

Academic System: Semester-based

Description Preparation Date: January 20, 2025

File Completion Date: January 20, 2025

جامعة تكريت
كلية علوم الحاسوب والرياضيات
مكتب معاون العميد للشؤون العلمية

Signature:

Head of Department Name:
Lec. Dr. Moceheb Lazam Shuwandy

Date: 27/1/2025

Signature:

Scientific Associate Name:
Assist. Prof. Dr. Majid Hamid Ali

Date: 27/1/2025

The file is checked by:

Department of Quality Assurance and University Performance

Director of the Quality Assurance and University Performance Department:

Date: 27/1/2025 Lec. Yasmr Abdulateef Hussein

Signature:

Approval of the Dean

عميد كلية علوم الحاسوب والرياضيات

1. Program Vision

To become a leading department in cybersecurity at the local, regional, and international levels, with a focus on excellence in teaching, learning, scientific research, and community service. The program aims to graduate professionals capable of addressing emerging cybersecurity challenges in the digital era.

2. Program Mission

The Bachelor of Science in Cybersecurity program aims to:

- Prepare qualified graduates equipped with theoretical knowledge and practical skills for securing systems, protecting networks, and managing cybersecurity risks effectively.
- Enable graduates to have a strong presence locally and internationally in the field of cybersecurity while encouraging continuous learning and skill development in line with technological advancements.
- Promote scientific research in cybersecurity through graduates capable of pursuing advanced studies and supporting innovation and creativity in compliance with global standards.
- Build effective partnerships with the community and various institutions through student projects, training programs, and consultations to raise awareness about cybersecurity and data protection in the digital age.

3. Program Objectives

The Cybersecurity Department aims to achieve the following objectives for students earning a Bachelor's degree in Cybersecurity. Upon graduation, students will be able to:

1. Acquire advanced theoretical knowledge and practical skills in cybersecurity, enabling them to protect systems and networks, secure information, and contribute to digital security at national and regional levels.
2. Be prepared to pursue advanced education and postgraduate studies in cybersecurity fields, equipping them to conduct research and develop innovative solutions for cybersecurity challenges.
3. Understand and adhere to professional ethical standards expected of cybersecurity specialists, while appreciating the social and ethical implications of security technologies and data protection.
4. Develop critical thinking and problem-solving skills to address complex issues in cybersecurity and recognize the importance of lifelong learning to stay abreast of rapid advancements in the field.

5. Work effectively as members of interdisciplinary teams and contribute to developing cybersecurity strategies in various institutions.
6. Establish professional partnerships with public and private sectors to foster innovation and implement security solutions that meet local and international market demands.
7. Enhance their ability to conduct security analyses and make informed decisions to protect systems from cyber threats using the latest technologies and tools in cybersecurity.

4. Program Accreditation

The Cybersecurity Department was newly established for the academic year 2024–2025.

5. Other external influences

Is there a sponsor for the program?

The Cybersecurity Department recognizes the importance of external factors in enhancing its position and achieving its goals. Therefore, the department focuses on the following:

- **Strategic Partnerships:** Developing partnerships with supporting entities, academic institutions, and industrial organizations to strengthen the department's technical and laboratory infrastructure. This collaboration provides state-of-the-art technologies and software that enable students to receive exceptional practical training.
- **External Funding:** Seeking external funding through research grants and joint projects with stakeholders interested in the field of cybersecurity. These efforts ensure the sustainability and continuous development of academic programs in response to growing challenges in the field.
- **Community Engagement:** Emphasizing interaction with the community and raising awareness about the significance of cybersecurity by organizing workshops and educational events targeting local communities and institutions. This societal role enhances the department's standing as a leader in providing solutions and expertise in digital security.

6 Program Structure

(The Department of Cybersecurity was newly created for the academic year 2024-2025)				
Program Structure	Number of Courses	Credit hours	Percentage	Reviews•
Institution Requirements				
College Requirements				

Department Requirements				
Summer Training				
Other				

This can include notes whether the course is basic or optional.

7. Program Description

Year/Level	Course Code	Course Name	Credit Hours	
			theoretical	practical
First Year / First Semester	TUCY101	Programming (1)	6	2
	TUCY102	Discrete Structures (1)	2	2
	TUCY103	Computer Organization	6	2
	TUCY104	Probability and Statistics	4	2
	TUCY105	Data Security Principles	6	2
	TUCY106	Mathematics	2	2
	UOT002	English Language	2	2
First Year / Second Semester	TUCY111	Advanced Programming (2)	6	2
	TUCY112	Basics of Logical Design	6	2
	TUCY113	Cybersecurity Fundamentals	6	2
	TUCY114	Discrete Structures (2)	2	2
	TUCY115	Coding and Information Theory	2	2
	UOT001	Arabic Language	2	2
	UOT003	Democracy and Human Rights	2	2

8. Expected learning outcomes of the program

Knowledge:

1. Equip students with the theoretical knowledge and practical skills needed to analyze cybersecurity problems and design appropriate solutions using best practices, enabling them to excel and secure prominent positions in the job market.
2. Enhance students' ability to innovate while adhering to professional, legal, and ethical standards and working effectively in multidisciplinary teams.
3. Enable students to pursue continuous self-learning to stay updated with advancements in cybersecurity technologies and adopt new techniques

and methods.

4. Develop students' ability to analyze and discuss outcomes using the knowledge gained during their studies.
5. Increase students' research expertise, including verifying the validity of conclusions and findings.
6. Strengthen students' professional experience by applying the knowledge and skills acquired during their education.
7. Prepare students to enroll in postgraduate programs in cybersecurity and other related technical fields.

Skills:

1. Acquire essential skills in programming, encryption, and electronic security.
2. Develop the ability to conduct research addressing challenges and techniques in cybersecurity.

Ethics

1. Foster an understanding of cybersecurity principles and encourage teamwork and collaboration.
2. Instill ethical values in students, including loyalty, respect for rules, and adherence to administrative regulations.
3. Cultivate a spirit of initiative and positivity, enabling students to face future professional challenges.
4. Encourage participation in extracurricular, voluntary, and community activities to strengthen their societal role.
5. Promote values of citizenship, national identity, and altruism as part of their professional and social responsibilities.

This vision reflects the department's commitment to producing graduates with the knowledge, skills, and values needed for success in the dynamic and technical work environment.

9. Teaching and Learning Strategies

1. **Theoretical and Practical Lectures:** Utilize methodological and supplementary textbooks alongside modern educational tools to deliver comprehensive and innovative academic content.
2. **Skill Development:** Provide students with foundational skills in computing and e-learning while employing advanced devices and technologies to enhance the learning process and technical proficiency.
3. **Encouraging Creativity in Graduation Projects:** Grant students the freedom to choose graduation research topics, allowing them to explore their intellectual potential and reveal their scientific interests and preferences, aligning with cybersecurity requirements.
4. **Promoting Interaction and Discussion:** Create a learning environment that allows students to share ideas during lectures, engage in discussions, exchange opinions, and develop data analysis skills while drawing scientific conclusions.

These strategies aim to prepare graduates with competency, innovation, and the ability to tackle challenges in the field of cybersecurity.

10. Evaluation methods
1. Electronic examinations.
2. Central and monthly examinations.
3. Daily quizzes.
4. Daily assignments.
5. Scientific reports.
6. Computer-based laboratory examinations.
7. Graduation projects.

11. Faculty

Faculty Members

Academic Rank	Specialization		Special Requirements/Skills (if applicable)		Number of the Teaching staff	
	General	Special			Staff	Lecturer
Assistant Professor	Computer Science	Information Security			1	
Assistant Professor	Computer Engineering	Cybersecurity			1	
Assistant Professor	Computer Science	IoT Technologies			1	
Assistant Professor	Computer Science	Database Technology			1	
Assistant Professor	Computer Science	Informatics and Communications			1	
Assistant Professor	Computer Science	Software Engineering			1	
Lecturer	Computer Science	Mobile Security			1	
Lecturer	Computer Science	Cybersecurity			1	
Lecturer	Computer Science	Information Systems			1	
Lecturer	Computer Science	Machine Learning and Data Mining			1	
Lecturer	Computer Science	Software Engineering			1	
Assistant Lecturer	Computer Science	Information Security			1	
Assistant Lecturer	Computer Science	Computer Security			2	
Assistant Lecturer	Computer Science	Information Technology			1	

12. Acceptance Criterion

1. **Central Admission:**

Admission to the Cybersecurity Department is based on the central admission system of the Ministry of Higher Education and Scientific Research to ensure fair distribution among eligible students.

2. **Admission Criteria:**

Students are admitted based on their grades in the central admission lists. However, children of faculty members, martyrs, and those eligible for special privileges under ministry regulations are granted the right to choose their preferred department, ensuring their distribution aligns with departmental needs.

This policy aims to achieve fairness while accommodating specific groups with special privileges in accordance with ministry directives.

14. Program Development Plan

The Cybersecurity Department focuses on enhancing students' research and inquiry skills through various academic activities, including attending scientific discussions and preparing specialized research in cybersecurity. The department emphasizes developing students' abilities to analyze data, draw logical conclusions, and refine their critical thinking and argumentation skills.

To achieve this, students are encouraged to:

Visit the library weekly to explore academic resources, books, and scientific journals as primary information sources.

Utilize the internet, e-learning tools, electronic references, and specialized scientific websites in cybersecurity to deepen their knowledge and stay updated with technological advancements.

These efforts aim to prepare students as distinguished researchers capable of contributing to the advancement of cybersecurity and addressing its future challenges.

Program Skills Outline															
				Required program Learning outcomes											
Year/Level	Course Code	Course Name	Basic or optional	Knowledge				Skills				Ethics			
				A1	A2	A3	A4	B1	B2	B3	B4	C1	C2	C3	C4
First Year / First Semester	TUCY101	Programming (1)	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	TUCY102	Discrete Structures (1)	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	TUCY103	Computer Organization	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	TUCY104	Probability and Statistics	Core	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓
	TUCY105	Data Security Principles	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	TUCY106	Calculus	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	UOT002	English Language	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
First Year / Second Semester	TUCY111	Advanced Programming (Programming 2)	Core	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓
	TUCY112	Basics of Logical Design	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	TUCY113	Cybersecurity Principles	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	TUCY114	Discrete Structures (2)	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	TUCY115	Coding and Information Theory	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	UOT001	Arabic Language	Core	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	UOT003	Democracy and Human Rights	Core	✓	✓			✓	✓	✓	✓	✓	✓	✓	✓

*** The scientific department was introduced for the current academic year 2024-2025, so it was limited to the learning outcomes of the program for the first year - the first and second semesters**

Course Description Form

1. Course Name:	
Programming 1	
2. Course Code:	
TUCY101	
3. Semester / Year:	
2024-2025	
4. Description Preparation Date:	
7/9/2024	
5. Available Attendance Forms:	
Theory, Practical, Tutorial	
6. Number of Credit Hours (Total) / Number of Units (Total)	
150 Hrs / 6 Units	
7. Course administrator's name (mention all, if more than one name)	
Name: Maytham Mustafa Hammood, Noor Sauod Abd Email: maythamhammood@tu.edu.iq	
8. Course Objectives	
Course Objectives	1. To introduce students to the fundamental principles and concepts of programming. 2. To familiarize students with the syntax and structure of the C++ programming language. 3. To develop students' problem-solving skills and algorithmic thinking. 4. To enable students to design, implement, and test programs using C++ to solve computational problems. 5. To provide students with hands-on experience in programming through practical exercises, assignments, and projects. 6. To promote the use of modular programming techniques for creating reusable and maintainable code. 7. To enhance students' ability to debug and troubleshoot programs effectively. 8. To develop students' communication skills in expressing programming concepts and solutions clearly and effectively. 9. To prepare students for advanced programming courses and real-world software development scenarios cipher algorithm programming.
9. Teaching and Learning Strategies	

Strategy	1. Lectures: The instructor will deliver lectures introducing and explaining programming concepts, C++ syntax, and problem-solving techniques. This will provide students with a solid theoretical foundation. 2. Interactive Discussions: Engaging students in interactive discussions allows them to ask questions, seek clarifications, and participate actively in learning. Discussions can include reviewing code examples, discussing programming best practices, and exploring real-world applications of programming concepts. 3. Laboratory Sessions: Laboratory sessions are dedicated practical sessions where students apply the concepts learned in lectures to hands-on programming exercises. Key strategies for the laboratory sessions include: a. Programming Exercises: Students will work on programming exercises and projects in the laboratory, providing them with practical experience in coding and problem-solving. b. Guided Practice: Lab instructors or teaching assistants will be available to provide guidance, assistance, and immediate feedback on students' code. They can help students debug their programs, identify errors, and improve their coding skills. c. Collaboration and Peer Learning: Students can collaborate with their peers in the laboratory, fostering teamwork and enabling knowledge sharing. Working together on programming tasks promotes discussions, problem-solving, and peer learning. d. Equipment and Resource Access: The laboratory should provide access to computers, necessary software tools, programming references, and relevant online resources. This ensures students have the necessary resources to effectively complete their lab exercises and assignments. 4. Programming Assignments: Assignments will be given to students to reinforce their understanding of programming concepts and encourage independent problem-solving. These assignments may involve implementing algorithms, designing software systems, or developing small-scale projects using C++. 5. Code Reviews and Feedback: The instructor will provide feedback on students' code, reviewing their solutions and offering suggestions for improvement. This feedback will help students enhance their coding skills and adhere to best practices. 6. Office Hours and Individual Support: The instructor should be available for individual consultations and support students who need additional help or guidance in understanding programming concepts or completing assignments.

10. Course Structure

Week	Hourse	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
Week 1	6	Demonstrate a solid understanding of the fundamental principles and concepts of programming.	Introduction to Computer Science, Computers Components, Binary and Info Representation	Cooperative learning strategy. Brainstorming learning strategy. Cooperative concept planning learning strategy. Real-time feedback learning strategy Notes series learning strategy.	Daily exams, oral and written, reports and discussions

				Exchange and discussion learning strategy. Information presentation learning strategy. Training and presentation of scientific developments learning strategy.	
Week 2	6	Utilize the syntax and structure of the C++ programming language to write well-structured and efficient code.	Algorithms Design and Writing pseudocode	=	=
Week 3	6	Apply problem-solving skills and algorithmic thinking to develop solutions for various computational problems.	Algorithms Design and Drawing Flowchart	=	=
Week 4	6	Design, implement, and test programs using C++ to solve specific tasks and challenges.	Variables, Datatypes, Output, and Input	=	=
Week 5	6	Utilize modular programming techniques to create reusable and maintainable code.	Operations (Arithmetic and Assignment) and Math Functions	=	=
Week 6	6	Debug and troubleshoot programs effectively using appropriate debugging techniques and tools.	Operations (Comparison and Logical)	=	=
Week 7	6	Collaborate and work effectively in teams to complete programming projects.	Flow Control (if – else)	=	=
Week 8	6	Communicate programming concepts, solutions, and ideas clearly and effectively, both orally and in written form.	Flow Control (switch – case)	=	=
Week 9	6	Demonstrate a readiness to progress to more advanced programming courses	Loops (counter and cumulative variables)	=	=

		or pursue a career in software development.			
Week 10	6	Utilize the syntax and structure of the C++ programming language to write well-structured and efficient code.	Uncountable Loops	=	=
Week 11	1	From week 1 to week 10	Midterm Exam	=	=
Week 12	6	Apply problem-solving skills and algorithmic thinking to develop solutions for various computational problems.	Nested Loops	=	=
Week 13	6	Design, implement, and test programs using C++ to solve specific tasks and challenges.	Functions	=	=
Week 14	6	Utilize modular programming techniques to create reusable and maintainable code.	building a TikTacToe Game	=	=
Week 15	2	From week 1 to week 14	Final exam.	/	/

11. Course Evaluation

Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc

12. Learning and teaching resources

Required textbooks (curricular books, if any)	Stroustrup, Bjarne - Programming_ principles and practice using C++-Addison-Wesley (2015)
Main references (Sources)	Olsson, Mikael - C++20 Quick syntax reference: a pocket guide to the language, apis, and library

Course Description Form

1. Course Name:	
Data Security Principles	
2. Course Code:	
TUCY105	
3. Semester / Year:	
2024-2025	
4. Description Preparation Date:	
7/9/2024	
5. Available Attendance Forms:	
Theory, Practical, Tutorial	
6. Number of Credit Hours (Total) / Number of Units (Total)	
150 Hrs / 6 Units	
7. Course administrator's name (mention all, if more than one name)	
Name: Moceheb Lazam Shuwandy Email: moceheb@tu.edu.iq	
8. Course Objectives	
Course Objectives	1. To provide students with a foundational knowledge of data security principles and practices. 2. To equip students with skills to recognize and mitigate various data security threats. 3. To introduce key data protection techniques, including encryption, authentication, and access control. 4. To ensure students understand the importance of data integrity, confidentiality, and availability. 5. To develop practical problem-solving skills related to securing sensitive data.
9. Teaching and Learning Strategies	
Strategy	**1. Theoretical Lectures** Objective: To deliver fundamental concepts related to data security in a structured and clear manner. Method: Traditional lectures supported by various teaching tools such as presentations, diagrams, and case studies. Approach: Explain key concepts such as encryption, access control, and data security principles. Use real-life examples to illustrate the practical applications of these concepts. Provide opportunities for students to ask questions and engage in discussions to enhance their understanding. **2. Practical Sessions (Labs)** Objective: To enable students to apply theoretical concepts practically using tools

	and technologies related to data security. Method: Lab sessions focused on implementing encryption, managing access control, and securing data. Approach: Provide students with hands-on experience using cybersecurity tools and software. Offer guidance and feedback during practical exercises to ensure students achieve the desired outcomes. Encourage teamwork and collaborative problem-solving through lab activities. **3. Interactive Tutorials** Objective: To reinforce students' understanding of specific topics through group discussions and in-depth analysis. Method: Interactive tutorial sessions involving case studies, security problem analysis, and solution discussions. Approach: Break down complex topics into digestible parts to make them easier for students to grasp. Facilitate student participation in discussions and encourage critical thinking. Address any misconceptions and provide further clarification on difficult topics. **4. Practical Assignments and Projects** Objective: To allow students to apply theoretical and practical knowledge to real-world data security problems. Method: Assignments such as implementing encryption systems or developing access control policies. Approach: Assign practical tasks that align with the topics covered in lectures and labs. Provide clear instructions and ongoing support to ensure successful completion of tasks. Encourage both individual and group work to foster collaboration and independent problem-solving. **5. Blended Learning** Objective: To offer a mix of face-to-face and online learning for flexibility and enhanced comprehension. Method: Use of online platforms to share resources, conduct quizzes, and facilitate discussions. Approach: Provide digital learning materials, such as recorded lectures, quizzes, and discussion forums. Encourage students to engage with supplementary materials, including videos and articles, on the online platform. Foster online discussions to complement in-class learning. **6. Formative and Summative Assessments** Objective: To measure students' progress in the module and identify areas for improvement. Method: Regular assessments including quizzes, lab exercises, and a final project. Approach: Use formative assessments like short quizzes and lab exercises to provide continuous feedback. Employ summative assessments, such as final exams and practical projects, to assess overall understanding. Provide clear and constructive feedback to help students improve their performance. **7. Introductory Lectures on Future Developments in Data Security** Objective: To motivate students by offering insights into how theoretical knowledge can be applied in future career paths.

	Method: Organize introductory lectures or workshops on the latest trends and future directions in data security. Approach: Offer an overview of current trends and emerging technologies in cybersecurity. Help students understand how this knowledge will be valuable for their future skillset. Open discussions on the importance of data security across different industries and sectors.

10. Course Structure

Week	Hourse	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
Week 1	6	Define basic concepts in data security, such as confidentiality, integrity, and availability.	Introduction to Data Security Concepts (CIA Triad)	Cooperative learning strategy. Brainstorming learning strategy. Cooperative concept planning learning strategy. Real-time feedback learning strategy Notes series learning strategy. Exchange and discussion learning strategy. Information presentation learning strategy. Training and presentation of scientific developments learning strategy.	Daily exams, oral and written, reports and discussions
Week 2	6	Identify different types of data security threats and vulnerabilities, and assess their potential impact.	Types of Data Security Threats and Vulnerabilities	=	=
Week 3	6	Explain and apply cryptographic techniques, including encryption and decryption.	Basics of Cryptography: Symmetric vs. Asymmetric Encryption	=	=
Week 4	6	Describe and implement authentication methods and access control models.	Key Management and Encryption Standards (AES, RSA)	=	=
Week 5	6	Analyze and evaluate different data protection protocols and techniques.	Authentication Methods: Passwords, Multi-factor Authentication	=	=

Week 6	6	Develop solutions to secure data in different scenarios, using appropriate security measures.	Access Control Models: Discretionary and Role-Based Access Control (RBAC)	=	=
Week 7	6	Understand the legal and ethical implications of data security and privacy, including regulatory compliance.	Data Integrity: Hashing, Checksums, and Digital Signatures	=	=
Week 8	6	Define basic concepts in data security, such as confidentiality, integrity, and availability.	Secure Communication Protocols: SSL/TLS, HTTPS	=	=
Week 9	6	Identify different types of data security threats and vulnerabilities, and assess their potential impact.	Midterm Review and Practical Lab Recap	=	=
Week 10	6	Explain and apply cryptographic techniques, including encryption and decryption.	Introduction to Firewalls and Intrusion Detection Systems (IDS)	=	=
Week 11	1	From week 1 to week 10	Midterm Exam	=	=
Week 12	6	Describe and implement authentication methods and access control models.	Data Privacy and Regulations: GDPR and Compliance	=	=
Week 13	6	Analyze and evaluate different data protection protocols and techniques.	Security Policies and Best Practices for Data Protection	=	=
Week 14	6	Develop solutions to secure data in different scenarios, using appropriate security measures.	Disaster Recovery, Backup Strategies, and Data Availability	=	=
Week 15	2	From week 1 to week 14	Final exam.	/	/
11. Course Evaluation					
Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc					
12. Learning and teaching resources					

Required textbooks (curricular books, if any)	"Principles of Data Security" by Ernst L. Leiss, 2012.
Main references (Sources)	"Principles of Information Security", 7th Edition, by Michael E. Whitman, and Herbert J. Mattord, Kennesaw State University, 2022.
Recommended books and references (scientific journals, reports…)	"Kali Linux Revealed: Mastering the Penetration Testing Distribution", Raphaël Hertzog, Jim O'Gorman, Mati Aharoni, 2017. "The Web Application Hacker's Handbook", by Dafydd Stuttard & Marcus Pinto, 2008.
Electronic references, Websites	SANS Cyber Security Resources: www.sans.org OWASP (Open Web Application Security Project): www.owasp.org

Course Description Form

1. Course Name:	
Computer Organization	
2. Course Code:	
TUCY103	
3. Semester / Year:	
2024-2025	
4. Description Preparation Date:	
17/9/2024	
5. Available Attendance Forms:	
Theory, Practical, Tutorial	
6. Number of Credit Hours (Total) / Number of Units (Total)	
150 Hrs / 6 Units	
7. Course administrator's name (mention all, if more than one name)	
Name: Saif Muhannad Mahir, Mohammed Taher Ahmed Email: saif.muhannad1985@tu.edu.iq, mohammed.t.a@tu.edu.iq	
8. Course Objectives	
Course Objectives	This module aims to provide students with an in-depth understanding of how computers are organized and how they process information at the hardware level. The course covers the basic components of computer systems, system architectures, memory hierarchies, and input/output systems. By the end of this module, students should have a comprehensive knowledge of the functioning of modern computer systems and be able to apply performance evaluation techniques to analyze and compare various system configurations. The specific aims of the module are: 1. To introduce the fundamental components of a computer system, including the CPU, memory, I/O devices, and storage. 2. To explain different computer architectures (e.g., von Neumann, Harvard, RISC) and how this affect system performance. 3. To explore processor design, memory management, and input/output mechanisms. 4. To develop students' ability to analyze the performance of computer systems using benchmarking tools. 5. To lay the foundation for understanding more advanced topics in computer systems, including parallel processing and virtualization.
9. Teaching and Learning Strategies	
Strategy	1. **Basic Components of Computer Systems**: a. CPU (Central Processing Unit)

b. Memory (RAM, ROM, Cache)
c. Input/Output Devices
d. Storage Devices (Hard drives, SSDs)

2. **System Architectures**:
   a. Von Neumann Architecture
   b. Harvard Architecture
   c. Reduced Instruction Set Computing (RISC)
   d. Instruction Sets and their impact on performance
3. **Processor Design**:
   a. Control Unit and Arithmetic Logic Unit (ALU)
   b. Pipelining and its challenges
   c. Multi-core Processors and Parallel Processing
4. **Memory Systems**:
   a. Memory Hierarchy: Cache, Main Memory (RAM), Secondary Storage (Hard drives, SSDs)
   b. Virtual Memory and Paging
   c. Memory Management Techniques
5. **Input/Output Systems**:
   a. I/O Devices (Input devices, Output devices)
   b. Interrupt Handling
   c. Direct Memory Access (DMA)
   d. I/O Performance Optimization Techniques
6. **Parallel Processing and Multiprocessing**:
   a. Parallel Algorithms
   b. Multi-core Processors
   c. Parallel Architectures
   d. Challenges of Parallel Computing
7. **Performance Evaluation**:
   a. Benchmarking Techniques
   b. Performance Metrics
   c. Performance Analysis Tools and Techniques
8. **Operating Systems and System Software**:
   a. Process Management
   b. Memory Management
   c. File Systems
   d. Device Management
   e. Scheduling Algorithms
9. **System Maintenance and Administration**:
   a. Basic Troubleshooting and Performance Tuning
   b. System Monitoring and Resource Management
   c. Backup and Recovery Methods

10. Course Structure

	Hourse	Required Learning			Evaluation

Week		Outcomes	Unit or subject name	Learning method	method
Week 1	6	**Basic Components of Computer Systems**: CPU (Central Processing Unit) Memory (RAM, ROM, Cache) Input/Output Devices Storage Devices (Hard drives, SSDs)	Introduction to Computer Organization	Cooperative learning strategy. Brainstorming learning strategy. Cooperative concept planning learning strategy. Real-time feedback learning strategy Notes series learning strategy. Exchange and discussion learning strategy. Information presentation learning strategy. Training and presentation of scientific developments learning strategy.	Daily exams, oral and written, reports and discussions
Week 2	6	**System Architectures**: Von Neumann Architecture Harvard Architecture Reduced Instruction Set Computing (RISC) Instruction Sets and their impact on performance	Computer Functions and System Architectures	=	=
Week 3	6	**Processor Design**: Control Unit and Arithmetic Logic Unit (ALU) Pipelining and its challenges Multi-core Processors and Parallel Processing	Instruction Set Architectures (ISA)	=	=
Week 4	6	**Memory Systems**: Memory Hierarchy: Cache, Main Memory (RAM), Secondary Storage (Hard drives, SSDs) Virtual Memory and Paging Memory Management	Processor Design – Control Unit and ALU	=	=

		Techniques			
Week 5	6	**Input/Output Systems**: I/O Devices (Input devices, Output devices) Interrupt Handling Direct Memory Access (DMA) I/O Performance Optimization Techniques	Memory Systems – Cache and Main Memory	=	=
Week 6	6	**Parallel Processing and Multiprocessing**: Parallel Algorithms Multi-core Processors Parallel Architectures Challenges of Parallel Computing	Secondary Storage and Virtual Memory	=	=
Week 7	6	**Performance Evaluation**: Benchmarking Techniques Performance Metrics Performance Analysis Tools and Techniques	Input/Output Systems – Part 1	=	=
Week 8	6	**Operating Systems and System Software**: Process Management Memory Management File Systems Device Management Scheduling Algorithms	Input/Output Systems – Part 2 (DMA and Interrupt Handling)	=	=
Week 9	6	**System Maintenance and Administration**: Basic Troubleshooting and Performance Tuning System Monitoring and Resource Management Backup and Recovery Methods.	Parallel Processing and Multi-core Processors	=	=
Week 10	6	**Processor Design**: Control Unit and Arithmetic Logic Unit (ALU)	Review and Midterm Preparation	=	=

		Pipelining and its challenges Multi-core Processors and Parallel Processing			
Week 11	1	From week 1 to week 10	Midterm Exam	=	=
Week 12	6	**Memory Systems**: Memory Hierarchy: Cache, Main Memory (RAM), Secondary Storage (Hard drives, SSDs) Virtual Memory and Paging Memory Management Techniques	Advanced Processor Design – Pipelining and Parallelism	=	=
Week 13	6	**Parallel Processing and Multiprocessing**: Parallel Algorithms Multi-core Processors Parallel Architectures Challenges of Parallel Computing	Memory Management and Optimization Techniques	=	=
Week 14	6	**Performance Evaluation**: Benchmarking Techniques Performance Metrics Performance Analysis Tools and Techniques	I/O System Design and Performance Optimization	=	=
Week 15	2	From week 1 to week 14	Final Project and Preparatory Week for Final Exam	/	/

11. Course Evaluation

Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc

12. Learning and teaching resources

Required textbooks (curricular books, if any)	1. Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and John L. Hennessy. 2. "Computer Organization and Architecture: Designing for Performance" by William Stallings.

	3. "Operating System Concepts" by Abraham Silberschatz, Peter B. Galvin, and Greg Gagne.
Main references (Sources)	1. "Structured Computer Organization" by Andrew S. Tanenbaum and Todd Austin. 2. "Exploring Microsoft Office 2019" by Mary Anne Poatsy, Keith Mulbery, Cynthia Krebs, and Lynn Hogan.
Recommended books and references (scientific journals, reports…)	1. "Operating System Concepts" by Abraham Silberschatz, Peter B. Galvin, and Greg Gagne. 2. "Operating Systems: Internals and Design Principles" by William Stallings.
Electronic references, Websites	https://ccms.tu.edu.iq/csd/electronic-lectures/409-stage1-8.html

Course Description Form

1. Course Name:	
Statistics and Probability	
2. Course Code:	
TUCY104	
3. Semester / Year:	
2024-2025	
4. Description Preparation Date:	
3/9/2024	
5. Available Attendance Forms:	
Theory, Practical	
6. Number of Credit Hours (Total) / Number of Units (Total)	
75 Hours / 3 Units	
7. Course administrator's name (mention all, if more than one name)	
Name: Saba Alaa Abdulwahhab **Email: saba.programmer12@tu.edu.iq**	
8. Course Objectives	
Course Objectives	1.Understand the laws of statistics and data distribution. 2.Enable the student to transform large data into understandable shapes and illustrations, and to deduce statistical data. 3.provide the students with details statistics and data population. 4.Define and explain the basic of probabilistic metrics like event, outcome, trial, simple event, sample space, Venn Diagram ,tree diagram, and calculate the probability that an event will occur. 5.Define and explain the basic of statistical measurements like Data Organization, variation, of central tendency. 6.Express the concepts and principal of counting techniques (factorial, combination) and the basic principles of Probability Theory 7. Solve the problems about permutation, combination and Binomial Theorem.
9. Teaching and Learning Strategies	
Strategy	1.The statistical in cyber-security developing data science techniques that enable large dynamic computer networks to identify intrusions and anomalous behavior and therefore protect against cyber-attacks and fraudulent activity. Using statistical methodology, machine learning and Big Data analytics the group develop tools to perform scalable anomaly detection in high volume data streams such as social networks, telecoms networks, network flow data, host-based sensor process-level data, cyber-physical and IoT data, pinpointing deviations from normal behavior. 2.Statistical techniques which have been so far been deployed include classification, data mining, streaming data analysis, cluster analysis, change point detection, graph analysis, topic modelling, penalized regression analysis, and machine learning. All of the work is

	motivated from real computer-network and internet data, with active government and industrial collaborators that include the Government's National Cyber Security Centre

10. Course Structure

Week	Hourse	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
Week 1	4	After completion of course, students will be able to apply Number theory in cyber security.	Introduction to probability concepts	Cooperative learning strategy. Brainstorming learning strategy. Cooperative concept planning learning strategy. Real-time feedback learning strategy Notes series learning strategy. Exchange and discussion learning strategy. Information presentation learning strategy. Training and presentation of scientific developments learning strategy.	Daily exams, oral and written, reports and discussions
Week 2	4	Understand the ideas of group, ring and an integral domain and be aware of examples of these structures in mathematics.	statistic, population, variable	=	=
Week 3	4	Understand probability and statical methods.	Data Organization	=	=
Week 4	4	Understand Statistics theory and its use in cyber security.	Data Description	=	=
Week 5	4	Have a better, grounded understanding of statistics and be able confidently to have statistical discussions.	Measures of Variation Variance & Standard Deviation	=	=
Week 6	4	Have gained skills in writing statistical projects, as the prelude to further project report writing at higher institutions or workplace report writing.	Graphic	=	=

Week 7	4	Have the basis in statistics to open a potential career path, as many jobs see understanding how to manipulate data and interpret statistics as a huge asset.	Counting techniques	=	=
Week 8	4	Understand the ideas of group, ring and an integral domain and be aware of examples of these structures in mathematics.	probability theory	=	=
Week 9	4	Understand probability and statical methods.	Theorems of Probability	=	=
Week 10	4	Understand Statistics theory and its use in cyber security.	Theorems of Probability	=	=
Week 11	1	From week 1 to week 10	Midterm Exam	=	=
Week 12	4	Understand Statistics theory and its use in cyber security.	Bayes' Theorem	=	=
Week 13	4	Have a better, grounded understanding of statistics and be able confidently to have statistical discussions.	Discrete Probability Distributions	=	=
Week 14	4	Have gained skills in writing statistical projects, as the prelude to further project report writing at higher institutions or workplace report writing.	transition probabilities	=	=
Week 15	2	From week 1 to week 14	Final exam.	/	/

11. Course Evaluation

Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc

12. Learning and teaching resources

Required textbooks (curricular books, if any)	- Probability and statistics, theory and applications, Gunnar Blom

	- Probability and statistics for engineers, Richard L Scheaffer
Main references (Sources)	- Statistics: theories and applications, Joseph Inungo, 2006. - Introductory Statistics, Ronald J. Wonnacott
Recommended books and references (scientific journals, reports…)	Python for Probability, Statistics, and Machine Learning
Electronic references, Websites	https://www.spps.org/cms/lib/MN01910242/Centricity/Domain/859/Statistics%20Textbook.pdf

Course Description Form

1. Course Name:	
Discrete Structure1	
2. Course Code:	
TUCY102	
3. Semester / Year:	
2024-2025	
4. Description Preparation Date:	
01/09/2024	
5. Available Attendance Forms:	
Theory	
6. Number of Credit Hours (Total) / Number of Units (Total)	
100 Hrs / 4 Units	
7. Course administrator's name (mention all, if more than one name)	
Name: Rawan Adel Fawzi **Email: rawan_adel@tu.edu.iq**	
8. Course Objectives	
Course Objectives	1- Discrete structures provide the mathematical foundations for some fundamental discrete math subjects that are highly relevant for cybersecurity and cryptography. 2- Discrete structures are abstract mathematical structures used to represent discrete objects and relations lie between those objects. 3- Students' realization of the basic concepts of discrete structure, such as mathematics logic, and graphs. Such as graph types support specific analyses that can help cybersecurity professionals model and secure networks, detect vulnerabilities, and analyze encryption algorithms and protocols effectively. 4- Knowing the models of discrete structures and how to create them. 5- Developing students' ability to deal with transformations and their applications in constructing structures. 6- Giving the student the necessary experience to deal with the relations and applications 7- Giving students the necessary experience to understand graph types supports specific analyses that can help cybersecurity professionals model and secure networks, detect vulnerabilities, and analyze encryption algorithms and protocols effectively. 8- It is important for students to understand the structure of data sets, access control, and the mathematical foundation of database security.

9. Teaching and Learning Strategies

Strategy	• The teacher gives detailed theoretical lectures • The teacher requests periodic reports on the basic topics of the subject • The student is also assigned to self-read and to give the student a certain period to inquire and discuss the topics he has read. • Solve practical examples Evaluation modalities 1-Daily exams with practical and scientific questions. 2- Participation scores for difficult competition questions among students. 3- Setting grades for homework and the reports assigned to them.

10. Course Structure

Week	Hourse	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
Week 1	2	Subject-specific skills: - Learn to prove the correctness and accuracy of the given issue, whether it is solvable or not, before starting to think about solving it.	Mathematical logic	Cooperative learning strategy. Brainstorming learning strategy. Cooperative concept planning learning strategy. Real-time feedback learning strategy Notes series learning strategy. Exchange and discussion learning strategy. Information presentation learning strategy. Training and presentation of scientific developments learning strategy.	Daily exams, oral and written, reports and discussions
Week 2	2	Thinking skills: Giving the learner the skill to use logical hypotheses in building accurate software.	Logical operators	=	=
Week 3	2	Giving the learner the skills to have the ability to build relationships between components, models and theoretical structures with algorithms and computer programs.	Logical operators	=	=
Week 4	2	Enabling students to continue self-development after graduation.	Logical Equivalences, Compound Propositions Classification	=	=

Week 5	2	Making the learner well acquainted with all types of logical deductive proof and types of proof by other methods.	Set of theory, Properties of set	=	=
Week 6	2	Build basic causal skills in creating and validating algorithms and programs.	Sets of Numbers, Sets and elements, subsets	=	=
Week 7	1	From week 1 to week 6	Mid Exam	=	=
Week 8	2	Building skills to analyze and solve some important issues and the approximate time to solve them.	Set's Algebra	=	=
Week 9	2	Build skills on how to choose the appropriate solutions for some issues and designate the best algorithms to solve them	Relations	=	=
Week 10	2	Enabling students to continue self-development after graduation.	Properties of relations with examples	=	=
Week 11	2	Subject-specific skills: Learn to prove the correctness and accuracy of the given issue, whether it is solvable or not, before starting to think about solving it.	Review of graphs and types	=	=
Week 12	2	Build basic causal skills in creating and validating algorithms and programs.	Tree	=	=
Week 13	2	Making the learner well acquainted with all types of logical deductive proof and types of proof by other methods.	Basic number theory. Divisibility	=	=
Week 14	2	Build basic causal skills in creating and validating algorithms and programs.	Greatest Common Divisors, Least Common Multiples	=	=
Week 15	2	From week 1 to week 14	Final exam.	/	/
11. Course Evaluation					

Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc	
12. Learning and teaching resources	
Required textbooks (curricular books, if any)	Theory and problems of Discrete mathematics, by Seymour Lipschutz & Marc Lars Lipson, Schaum's Outline Series, third edition 2007.
Main references (Sources)	Discrete Mathematics and Its Applications, Seventh Edition, Kenneth H. Rosen, AT&T Laboratories, 2012.
Recommended books and references (scientific journals, reports…)	• Mathematical foundation of computer science, Y.N. Singh, 2005 • Discrete structures, Amin Witno, Revision Notes and Problems 2006, www.witno.com Discrete mathematical structures for computer science by Bernard Kolman & Robert C. Busby
Electronic references, Websites	http://en.wikibooks.org/wiki/Discrete_mathematics/Set_theory

Course Description Form

1. Course Name:
Calculus
2. Course Code:
TUCY106
3. Semester / Year:
1st Semester / 2025-2024
4. Description Preparation Date:
3/1/2025
5. Available Attendance Forms:
Theory
6. Number of Credit Hours (Total) / Number of Units (Total)
75 Hours / 3 Units
7. Course administrator's name (mention all, if more than one name)
Name: Dr. Ali Shebl Ajeel **Email: ali.shebl@tu.edu.iq**
8. Course Objectives

Course Objectives	The goal of studying differential calculus at university is to enable students to gain a deep understanding of this fundamental element of mathematics and its applications in different fields. By studying differential calculus, students learn how to calculate derivatives and understand the concept of a derivative as the instantaneous rate of change of a function. Students can apply the concepts of calculus to solve practical problems, analyze the behavior of functions, determine critical points, least and largest values of functions, and estimate changes of variable quantities. In addition, the study of differential calculus provides a foundation for the study of other topics in mathematics, science, and engineering, such as integration, calculus in multiple variables, and the solution of differential equations. Learning differential calculus aims to develop students' analytical thinking and mathematical reasoning capabilities and provide them with powerful mathematical tools to deal with complex technical and scientific problems.

9. Teaching and Learning Strategies	
Strategy	1. Student interaction: Active participation and interaction between students and the lecturer or teacher is encouraged. Small group discussions or collaborative sessions can be organized to solve various differential problems. Technology, such as online forums or distance learning tools, can be used to encourage communication and collaboration among students. 2. Practical Application and Projects: The course should include practical activities and application projects that allow students to apply differential concepts and skills in real-world contexts. For example, teams can be formed to solve multidimensional differential problems or applications in fields such as engineering and medical science. 3. Use of Technology: Calculus software and mathematical applications can be used to enhance interaction and interactive learning. Students can use graphing software or computer mathematics programs to analyze functions and graph their curves. Provide examples and practical exercises: A wide range of examples and practical exercises covering various differential calculus concepts should be provided. Students can practice solving the exercises.

10. Course Structure

Week	Hours	Required Learning Outcomes	Learning method	Evaluation method
1	2	Introduction. Sets, intervals, Inequalities.	Theoretical Lecture and discussion	Oral tests and quizzes
2	2	Functions, Types of functions	Theoretical Lecture and discussion	Oral tests and quizzes
3	2	Doman and range of functions	Theoretical Lecture and discussion	Oral tests and quizzes
4	2	Graph the functions	Theoretical Lecture and discussion	Oral tests and quizzes
5	2	Limit of function	Theoretical Lecture and discussion	Oral tests and quizzes
6	2	Continuous	Theoretical Lecture and discussion	Oral tests and quizzes
7	1	Mid-term exam + Trigonometric functions.	Theoretical Lecture and discussion	1st Midterm exam in previous weeks(1-7)
8	2	Derivatives	Theoretical Lecture and discussion	Oral tests and quizzes
9	2	Rules of differentiation	Theoretical Lecture and discussion	Oral tests and quizzes
10	2	Applications of Derivatives.	Theoretical Lecture and discussion	Oral tests and quizzes
11	2	The mean value theorem	Theoretical Lecture and discussion	Oral tests and quizzes
12	2	Derivatives of Trigonometric functions.	Theoretical Lecture and discussion	Oral tests and quizzes
13	2	Derivatives of Exponential Functions, Logarithm Functions	Theoretical Lecture and discussion	Oral tests and quizzes

14	2	Derivatives of inverse functions	Theoretical Lecture and discussion	Oral tests and quizzes
15	2	Final Exam		

11. Course Evaluation

Student evaluation in this course out of 100 consists of the sum of

1. Achievement assessment (striving) 50% • 2 tests in weeks (5, 10) 10%
   - 2 homework assignments in weeks (3, 8) 10%
   - 10 in-college assignments in weeks (14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2) 10%
   - 2 reports in weeks (6, 14) 10%
   - Midterm exam 10%
2. Summative assessment, which is the final exam score 50%

12. Learning and Teaching Recourses

Required textbook(Curricular book, if any):

Main References (Sources):

1. Courant, R., John, F., Blank, A. A., & Solomon, A. (1965). Introduction to calculus and analysis (Vol. 1). New York: Interscience Publishers.
2. Tall, D. (1996). Functions and calculus. International handbook of mathematics education, 1, 289-325.
3. Tall, D. (1996). Functions and calculus. International handbook of mathematics education, 1, 289.-
4. Marsden, J., & Weinstein, A. (1985). Calculus I. Springer Science & Business Media.
5. Thomas' Calculus, Early Transcendental, 12th ed.
6. Calculus and Analytic Geometric, Durfee. W.H ,1971 New York (3).

Recommended book and references (Scientific journals, reports,…):

Grossman, Stanley I. Calculus Academic Press, 2014

Electronic Reference ,Web sites:

https://books.google.iq/books?hl=ar&lr=&id=0aziBQAAQBAJ&oi=fnd&pg=PP1&dq=calculus+book&ots=a1k4tINdCZ&sig=tmAQQ_yHi9mTDBLcx-qi7hy9uo8&redir_esc=y#v=onepage&q=calculus%20book&f=false

Course Description Form

1. Course Name:	
English Language1	
2. Course Code:	
UOT002	
3. Semester / Year:	
2024-2025	
4. Description Preparation Date:	
7/9/2024	
5. Available Attendance Forms:	
Theory	
6. Number of Credit Hours (Total) / Number of Units (Total)	
50 Hours / 2 Units	
7. Course administrator's name (mention all, if more than one name)	
Name: Ayham Mahmood **Email: Ahmed.f.saber@tu.edu.iq**	
8. Course Objectives	
Course Objectives	1. To enable the students to communicate effectively and appropriately in real life situation. 2. To develop and integrate the use of the four language skills i.e., Reading, Listening, Speaking and Writing. 3. To use English effectively for study purpose across the curriculum. 4. The ability to understand meaning of words, phrases and sentences in context. 5. The ability to speak and pronounce English Correctly and intelligibly 6. The ability to write English correctly and master the Mechanics of writing; the use of correct punctuation marks and capital letters. 7. To acquire the ability to use a suitable dictionary to understand labels, simple notices and written instructions. 8. To enable the students to know the contemporary strategies in teaching and learning English language.
9. Teaching and Learning Strategies	
Strategy	1- This course is characterized by the fact that it needs a special strategy that depends mainly on the development of English language and its skills. It also depends on previous courses in real analysis, situation, and some imagination. Teaching is mainly based on the home works that are given at the end of each week, and the student notes the interdependence between the serial topics of this course. In addition, to assigning the student (or a group of students) to make one seminar for the purpose of training in the use of scientific resources and the method of writing a subject in English language. 2- The purpose of this module is to develop students' linguistic ability by focusing

	on the key skills of reading, writing, speaking and listening, to encourage students to become independent learners and to introduce them to strategies and skills to enable them to cope with the demands, both academic and cultural, of undergraduate study in an English-speaking environment.

10. Course Structure

Week	Hourse	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
Week 1	2	Demonstrate through face-to-face conversations comprehension of simple words and phrases used in common everyday context.	An introduction to English language.	Cooperative learning strategy. Brainstorming learning strategy. Cooperative concept planning learning strategy. Real-time feedback learning strategy Notes series learning strategy. Exchange and discussion learning strategy. Information presentation learning strategy. Training and presentation of scientific developments learning strategy.	Daily exams, oral and written, reports and discussions
Week 2	2	Knowing the basic rules of the English language.	Unit (1) Hello am/ are/ is, my/your . This is … • How are you? • Good morning! What's this in English? • Numbers • Plurals	=	=
Week 3	2	Familiarity with the four English language skills: listening, reading, speaking and writing.	Unit (2) Your World Countries • he/she/ they, his/her • Where's he from? fantastic/ awful/ beautiful · Numbers 11-30	=	=
Week 4	2	Determine the negative effects of the mother tongue on the English language.	Unit (3) All about You Jobs • am/are/ is • Negatives and questions	=	=

			• Personal information • Social expressions		
Week 5	2	Interpreting the texts in different contexts.	Unit (4) Family and Friends our/their • Possessive 's • The family • has/have • The alphabet.	=	=
Week 6	2	Generate simple sentences containing learned vocabulary and using appropriate grammatical structures	Unit (5) The Way I live Sports/ Food/ Drinks • Present Simple • l/you/ we/ they • a/an Languages and nationalities • Numbers and prices.	=	=
Week 7	2	Express awareness of social and environmental issues.	Unit (6) Every day the time • Present Simple- he/she · always/sometimes/never Words that go together • Days of week.	=	=
Week 8	2	Acquire varied range of vocabulary; understand increased complexity of sentence structures both in reading and writing.	Unit (7) My favorites Question words •me/him/us/them • this/that Adjectives • Can I.?	=	=
Week 9	2	Obtaining a core competency such as, developing the ability to express student's thoughts orally and in-writing in a meaningful way in English language.	Unit (8) Where I live Rooms and furniture • There is/ are • Prepositions • Directions	=	=
Week 10	2	Enable students to communicate in English more confidently and effectively in their work or study environment.	Unit (9) Times past Saying years • was/were born • Past Simple - irregular verbs • have/do/go •When's your	=	=

			birthday?		
Week 11	1	From week 1 to week 10	Midterm Exam	=	=
Week 12	2	Generate simple sentences containing learned vocabulary and using appropriate grammatical structures	Unit (10) We had a great time! Past Simple - regular and irregular • Questions and negatives • Sport and leisure • Going sightseeing.	=	=
Week 13	2	Express awareness of social and environmental issues.	Unit (11) I can do that! can/can't · Adverbs • Adjective + noun • Everyday problems.	=	=
Week 14	2	Acquire varied range of vocabulary; understand increased complexity of sentence structures both in reading and writing.	Unit (12) Please and thank you I'd like - some/any • In a restaurant • Signs all around.	=	=
Week 15	3	From week 1 to week 14	Final exam.	/	/

11. Course Evaluation

Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc

12. Learning and teaching resources

Required textbooks (curricular books, if any)	Oxford Headway plus for Beginners.
Main references (Sources)	New Headway English Course (2002) by Julia Starr Keddle.
Recommended books and references (scientific journals, reports…)	English for Everyone

Electronic references, Websites	https://www.unionlearn.org.uk/english-and-maths-learning-resources-and-tools

Course Description Form

1. Course Name:	
Advanced Programming	
2. Course Code:	
TUCY111	
3. Semester / Year:	
2024-2025	
4. Description Preparation Date:	
18/01/2025	
5. Available Attendance Forms:	
Theory, Practical, Tutorial	
6. Number of Credit Hours (Total) / Number of Units (Total)	
150 Hrs / 6 Units	
7. Course administrator's name (mention all, if more than one name)	
Name: Maytham Mustafa Hammood, Mohammed Taher Ahmed **Email: maythamhammood@tu.edu.iq**	
8. Course Objectives	
Course Objectives	1. Understanding Advanced Data Structures: The module aims to provide students with a deep understanding of arrays, strings, and their manipulation techniques. Students will learn about multidimensional arrays, character arrays, and string handling functions. 2. Mastery of Pointers: The module aims to develop students' proficiency in using pointers in C++. Students will learn the concepts of memory addresses, pointer arithmetic, and dynamic memory allocation. They will understand how to manipulate data using pointers and how to utilize them for efficient memory management. 3. File Handling and Input/Output Operations: The module aims to introduce students to file handling concepts and techniques in C++. Students will learn how to read from and write to files, open and close files, handle file errors, and perform various input/output operations using file streams. They will understand file modes, buffering, and error handling. 4. File Management and Organization: The module aims to teach students how to manage and organize files effectively in C++. They will learn to create, modify, and delete files, organize file directories, and handle file-related operations. Students will understand the importance of file management in real-world programming scenarios. 5. Practical Application and Problem-Solving: Throughout the module, students will be exposed to practical programming exercises and problem-solving tasks.

	They will apply the concepts learned to solve real-world programming challenges, consolidating their understanding and enhancing their problem-solving skills. By focusing on arrays, strings, pointers, and file handling in C++, this advanced programming module aims to provide students with a comprehensive understanding of these concepts and their practical application. Students will develop the skills necessary to manipulate complex data structures, handle files, and write efficient and reliable code.
9. Teaching and Learning Strategies	
Strategy	1. Lectures: The instructor will deliver lectures introducing and explaining programming concepts, C++ syntax, and problem-solving techniques. This will provide students with a solid theoretical foundation. 2. Interactive Discussions: Engaging students in interactive discussions allows them to ask questions, seek clarifications, and participate actively in learning. Discussions can include reviewing code examples, discussing programming best practices, and exploring real-world applications of programming concepts. 3. Laboratory Sessions: Laboratory sessions are dedicated practical sessions where students apply the concepts learned in lectures to hands-on programming exercises. Key strategies for the laboratory sessions include: a. Programming Exercises: Students will work on programming exercises and projects in the laboratory, providing them with practical experience in coding and problem-solving. b. Guided Practice: Lab instructors or teaching assistants will be available to provide guidance, assistance, and immediate feedback on students' code. They can help students debug their programs, identify errors, and improve their coding skills. c. Collaboration and Peer Learning: Students can collaborate with their peers in the laboratory, fostering teamwork and enabling knowledge sharing. Working together on programming tasks promotes discussions, problem-solving, and peer learning. d. Equipment and Resource Access: The laboratory should provide access to computers, necessary software tools, programming references, and relevant online resources. This ensures students have the necessary resources to effectively complete their lab exercises and assignments. 4. Programming Assignments: Assignments will be given to students to reinforce their understanding of programming concepts and encourage independent problem-solving. These assignments may involve implementing algorithms, designing software systems, or developing small-scale projects using C++. 5. Code Reviews and Feedback: The instructor will provide feedback on students' code, reviewing their solutions and offering suggestions for improvement. This feedback will help students enhance their coding skills and adhere to best practices. 6. Office Hours and Individual Support: The instructor should be available for individual consultations and support students who need additional help or guidance in understanding programming concepts or completing assignments.
10. Course Structure	

	Hourse	Required Learning			Evaluation

Week		Outcomes	Unit or subject name	Learning method	method
1	6	Review of fundamental programming concepts, including variables, data types, control structures, and functions	Advanced Programming	Lecture + Lab	-
2	6	Introduction to Arrays (Linear arrays)	Advanced Programming	Lecture + Lab	-
3	6	Searching and sorting in linear arrays	Advanced Programming	Lecture + Lab	-
4	6	Multidimensional arrays and square arrays	Advanced Programming	Lecture + Lab	-
5	6	Multiplication of two arrays and re-writing Tic Tac Toe game using arrays	Advanced Programming	Lecture + Lab	Quiz (10%)
6	6	Introduction to Strings and their operations	Advanced Programming	Lecture + Lab	-
7	6	More examples on Strings	Advanced Programming	Lecture + Lab	Assignment (20%)
8	6	Introduction to Pointers	Advanced Programming	Lecture + Lab	-
9	6	Pointer to array and pointer arithmetic	Advanced Programming	Lecture + Lab	-
10	1	Midterm Exam	Logic Design	Exam	Midterm Exam (10%)
11	6	Introduction to Files and Directories	Advanced Programming	Lecture + Lab	Quiz (10%)
12	6	Working with text files (Read, Write)	Advanced Programming	Lecture + Lab	Assignment (20%)
13	6	Working with binary files	Advanced Programming	Lecture + Lab	-
14	6	Second project due (Student presentations - Part 1)	Advanced Programming	Project	Project (20%)

15	6	Final Exam	Covers All Units	Written & Practical Exam	Final Exam

11. Course Evaluation

Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc

12. Learning and teaching resources

Required textbooks (curricular books, if any)	Stroustrup, Bjarne - Programming_ principles and practice using C++-Addison-Wesley (2015)
Main references (Sources)	Olsson, Mikael - C++20 Quick syntax reference: a pocket guide to the language, apis, and library

Course Description Form

1. Course Name:	
	Cybersecurity Principles
2. Course Code:	
	TUCY113
3. Semester / Year:	
	2nd Semester/2024-2025
4. Description Preparation Date:	
	20-01-2025
5. Available Attendance Forms:	
	Theory, Practical, Tutorial
6. Number of Credit Hours (Total) / Number of Units (Total)	
	150 Hrs / 6 Units
7. Course administrator's name (mention all, if more than one name)	
	Name: Dr. Moceheb Lazam Shuwandy **Email: moceheb@tu.edu.iq**

8. Course Objectives

Course Objectives	The objectives of the **"Cyber Security Principles"** course for first-year students in the **Cybersecurity Department** are designed to provide foundational knowledge and skills in cybersecurity. The key objectives include: **1. Understanding Cybersecurity Concepts** • Introduce students to fundamental cybersecurity principles, including confidentiality, integrity, and availability (CIA Triad). • Explain the importance of cybersecurity in protecting digital assets and personal data. **2. Identifying Cyber Threats and Attacks** • Familiarize students with common cyber threats such as malware, phishing, denial-of-service (DoS) attacks, and social engineering. • Analyze real-world case studies of cyberattacks and their impact on organizations. **3. Security Mechanisms and Cryptography** • Introduce basic cryptographic principles, including encryption, hashing, and authentication methods. • Explain the role of encryption in securing communications and data storage. **4. Access Control and Authentication**

- Explore different authentication methods, including passwords, biometrics, and multi-factor authentication (MFA).
- Understand access control models such as discretionary access control (DAC) and role-based access control (RBAC).

5. **Network Security Fundamentals**
   - Provide an overview of network security principles, including firewalls, intrusion detection systems (IDS), and secure network protocols.
   - Explain common vulnerabilities in network security and how to mitigate them.
6. **Secure System and Application Development**
   - Introduce secure coding principles to prevent common vulnerabilities like SQL injection and cross-site scripting (XSS).
   - Discuss security best practices in software and web application development.
7. **Cybersecurity Policies and Regulations**
   - Explain the importance of cybersecurity laws, regulations, and ethical considerations.
   - Introduce international standards and frameworks such as ISO 27001, NIST, and GDPR.
8. **Hands-on Practical Skills**
   - Provide students with practical experience using cybersecurity tools such as Kali Linux for ethical hacking and penetration testing.
   - Conduct lab exercises to reinforce theoretical knowledge with real-world applications.
9. **Incident Response and Risk Management**
   - Introduce the fundamentals of incident response and digital forensics.
   - Teach students how to identify, analyze, and respond to cybersecurity incidents.
10. **Developing a Security Mindset**
    - Encourage students to adopt a proactive approach to cybersecurity and risk assessment.
    - Emphasize the importance of continuous learning and staying updated with emerging cyber threats.

9. Teaching and Learning Strategies

To ensure effective delivery of **Cyber Security Principles** in the **first level of the Cybersecurity Department**, the following **teaching and learning strategies** will be applied:

Strategy	**1. Interactive Lectures** **Objective:** Deliver fundamental concepts, theories, and principles of cybersecurity in an engaging manner. **Method:** Use **PowerPoint presentations** with real-world case studies. Include **live demonstrations** of cyber threats, attacks, and security mechanisms. Encourage **active participation** by posing thought-provoking questions and problem-solving scenarios.
	2. Practical Hands-on Labs

	Objective: Reinforce theoretical knowledge through real-world applications. **Method:** Conduct **laboratory sessions using Kali Linux** and security tools such as Wireshark, Nmap, and Metasploit. Simulate **cyberattacks and defense strategies** in a controlled environment. Assign **step-by-step lab exercises** to practice encryption, authentication, and penetration testing techniques.
	3. Problem-Based Learning (PBL) **Objective:** Enhance critical thinking and problem-solving skills. **Method:** Provide students with **real-world cybersecurity problems** and ask them to propose solutions. Use **case studies of major cyberattacks** and analyze how they could have been prevented. Encourage teamwork in solving **cyber incident response challenges**.
	4. Group Projects and Collaborative Learning **Objective:** Develop teamwork and research skills while applying cybersecurity principles. **Method:** Assign students to **small groups** to work on cybersecurity research projects. Have students **design and implement security solutions** for hypothetical organizations. Conduct **peer evaluations** to encourage collaboration and constructive feedback.
	5. Role-Playing and Simulated Cybersecurity Scenarios **Objective:** Prepare students for real-world cybersecurity challenges. **Method:** Conduct **cyber incident response simulations** where students act as security analysts. Use **capture-the-flag (CTF) competitions** to test cybersecurity skills in ethical hacking. Assign roles such as **attackers, defenders, and forensic analysts** to understand cyber operations from multiple perspectives.
	6. Flipped Classroom Approach **Objective:** Encourage self-directed learning and maximize in-class engagement. **Method:** Provide **pre-recorded video lectures** or reading materials before class. Use class time for **discussions, practical exercises, and Q&A sessions**. Implement **short quizzes** to assess pre-class learning.
	7. Use of Online Cybersecurity Platforms and Simulators **Objective:** Provide additional hands-on practice outside the classroom. **Method:** Use **Cyber Ranges** or online **security labs** (e.g., TryHackMe, Hack The Box). Integrate **interactive learning platforms** such as Coursera, Cybrary, or Open Web Application Security Project (OWASP). Assign **virtual penetration testing challenges**.
	8. Assessment through Real-World Application **Objective:** Evaluate students' ability to apply cybersecurity knowledge effectively.

Method:
Assign **practical cybersecurity tasks** rather than purely theoretical exams.
Include **mini-projects** where students build security solutions.
Use **graded security audits** where students assess vulnerabilities in simulated networks.

9. Industry Guest Lectures and Cybersecurity Seminars
Objective: Expose students to real-world cybersecurity challenges and industry practices.
Method:
Invite **cybersecurity professionals** to give guest lectures.
Organize **seminars on cybersecurity trends** and attack case studies.
Encourage **networking with industry experts** for career guidance.

10. Continuous Assessment and Feedback
Objective: Monitor student progress and provide timely support.
Method:
Use **weekly quizzes and assignments** to reinforce learning.
Provide **constructive feedback on lab exercises and projects**.
Conduct **student reflection sessions** to discuss learning challenges and areas for improvement.

By implementing these strategies, students will gain both **theoretical knowledge** and **practical expertise**, ensuring they are well-prepared for advanced cybersecurity courses and real-world challenges.

10. Course Structure

Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning Method	Evaluation
1	6	Understand cybersecurity principles and key terminologies	Introduction to Cybersecurity	Lecture + Discussion	Quiz + Participation
2	6	Explain the CIA Triad and its importance in security	Cybersecurity Fundamentals	Lecture + Case Study	Assignment
3	6	Identify types of cyber threats and vulnerabilities	Cyber Threats & Attack Vectors	Lecture + Hands-on Lab	Lab Report
4	6	Analyze common malware, phishing, and social engineering tactics	Malware & Social Engineering Attacks	Lab Simulation	Quiz
5	6	Understand encryption basics and hashing techniques	Cryptography & Data Protection	Lecture + Lab	Practical Test
6	6	Explore authentication methods and access control mechanisms	Authentication & Access Control	Lecture + Group Discussion	Assignment
7	6	Apply network security principles and analyze firewall rules	Network Security Basics	Hands-on Lab (Wireshark, Firewall)	Lab Report
8	6	Learn about web application security and common	Web Security & Secure Coding	Hands-on Lab (OWASP)	Quiz

		vulnerabilities			
9	6	Understand incident response steps and forensic analysis	Cyber Incident Response	Case Study + Role Play	Assignment
10	1	Mid-Term Exam	Covers Weeks 1-9	Written & Practical Exam	Mid-Term Exam
11	6	Learn about cybersecurity policies, ethics, and regulations	Cybersecurity Laws & Ethics	Lecture + Discussion	Assignment
12	6	Study penetration testing techniques and ethical hacking principles	Ethical Hacking & Pen Testing	Hands-on Lab (Kali Linux)	Lab Report
13	6	Identify cloud security risks and IoT security challenges	Cloud & IoT Security	Lecture + Case Study	Quiz
14	6	Conduct a security audit and risk assessment on a system	Security Risk Management	Practical Lab + Discussion	Practical Report
15	2	Final Exam	Covers All Units	Written & Practical Exam	Final Exam

11.Course Evaluation

The **evaluation structure** for the **Cyber Security Principles (TUCY105)** course is designed to comprehensively assess students' knowledge and practical skills, with a total course score of **100 marks**. The **final exam (50 marks)** is the most significant component, covering all course topics in both theoretical and practical formats to ensure students' mastery of cybersecurity principles. The **mid-term exam (10 marks)** provides an early assessment of students' progress and understanding. The **coursework (40 marks)**, also known as **semester work**, consists of quizzes, assignments, lab reports, projects, and class participation, ensuring continuous assessment throughout the semester. This structure maintains a balanced evaluation of **theory and practical application**, preparing students for real-world cybersecurity challenges.

12.Learning and Teaching Resources

Required textbooks (curricular books, if any)	
Main references (sources)	1- "Principles of Data Security" by Ernst L. Leiss, 2012. 2- "Principles of Information Security", 7th Edition, by Michael E. Whitman, and Herbert J. Mattord, Kennesaw State University, 2022.
Recommended books and references (scientific journals, reports…)	1- Kali Linux Revealed: Mastering the Penetration Testing Distribution", Raphaël Hertzog, Jim O'Gorman, Mati Aharoni, 2017. 2- "The Web Application Hacker's Handbook", by Dafydd Stuttard & Marcus Pinto, 2008.
Electronic References, Websites	

Course Description Form

1. Course Name:	
Discrete Structure2	
2. Course Code:	
TUCY114	
3. Semester / Year:	
2024-2025	
4. Description Preparation Date:	
02/02/2025	
5. Available Attendance Forms:	
Theory	
6. Number of Credit Hours (Total) / Number of Units (Total)	
100 Hrs / 4 Units	
7. Course administrator's name (mention all, if more than one name)	
Name: Rawan Adel Fawzi **Email: rawan_adel@tu.edu.iq**	
8. Course Objectives	
Course Objectives	1- Discrete structures provide the mathematical foundations for some fundamental discrete math subjects that are highly relevant for cybersecurity and cryptography. 2- Discrete structures are abstract mathematical structures used to represent discrete objects and relations lie between those objects. 3- Students' realization of the basic concepts of discrete structure, such as mathematics logic, and graphs. Such as graph types support specific analyses that can help cybersecurity professionals model and secure networks, detect vulnerabilities, and analyze encryption algorithms and protocols effectively. 4- Knowing the models of discrete structures and how to create them. 5- Developing students' ability to deal with transformations and their applications in constructing structures. 6- Giving the student the necessary experience to deal with the relations and applications 7- Giving students the necessary experience to understand graph types supports specific analyses that can help cybersecurity professionals model and secure networks, detect vulnerabilities, and analyze encryption algorithms and protocols effectively.

	8- It is important for students to understand the structure of data sets, access control, and the mathematical foundation of database security.

9. Teaching and Learning Strategies

Strategy	• The teacher gives detailed theoretical lectures • The teacher requests periodic reports on the basic topics of the subject • The student is also assigned to self-read and to give the student a certain period to inquire and discuss the topics he has read. • Solve practical examples Evaluation modalities 1-Daily exams with practical and scientific questions. 2- Participation scores for difficult competition questions among students. 3- Setting grades for homework and the reports assigned to them.

10. Course Structure

Week	Hourse	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
Week 1	2	Subject-specific skills: - Learn to prove the correctness and accuracy of the given issue, whether it is solvable or not, before starting to think about solving it.	Vectors and Matrices	Cooperative learning strategy. Brainstorming learning strategy. Cooperative concept planning learning strategy. Real-time feedback learning strategy Notes series learning strategy. Exchange and discussion learning strategy. Information presentation learning strategy. Training and presentation of scientific developments learning strategy.	Daily exams, oral and written, reports and discussions
Week 2	2	Thinking skills: Giving the learner the skill to use logical hypotheses in building accurate software.	Algebra in the Matrices	=	=
Week 3	2	Giving the learner the skills to have the ability to build relationships between components, models and theoretical structures with algorithms and computer programs.	Determinants	=	=

Week 4	2	Enabling students to continue self-development after graduation.	Cofactors & Minors	=	=
Week 5	2	Making the learner well acquainted with all types of logical deductive proof and types of proof by other methods.	Inverse Square Matrix	=	=
Week 6	2	Build basic causal skills in creating and validating algorithms and programs.	Rule Grammer	=	=
Week 7	2	From week 1 to week 6	Mid Exam	=	=
Week 8	2	Building skills to analyze and solve some important issues and the approximate time to solve them.	The Domain & the Range of a Relation	=	=
Week 9	2	Build skills on how to choose the appropriate solutions for some issues and designate the best algorithms to solve them	Binary Relation, Graph of the Relation & Photographer representation of the relations	=	=
Week 10	2	Enabling students to continue self-development after graduation.	Properties of relations with examples and Composition Relation	=	=
Week 11	2	Subject-specific skills: Learn to prove the correctness and accuracy of the given issue, whether it is solvable or not, before thinking about solving it.	Algebra of Function, Models of Functions and Draw Graphs Functions	=	=
Week 12	2	Build basic causal skills in creating and validating algorithms and programs.	Review of graphs and Examples of Graphs	=	=
Week 13	2	Making the learner well acquainted with all types of logical deductive proof and types of proof by other methods.	Tree, Examples of Trees and Pruning Algorithm for Minimal Path	=	=
Week 14	2	Build basic causal skills in creating and validating algorithms and programs.	Basic number theory, Divisibility Greatest Common Divisors and Least Common Multiples	=	=

Week 15	2	From week 1 to week 14	Final exam.	/	/

11. Course Evaluation

Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc

12. Learning and teaching resources

Required textbooks (curricular books, if any)	Theory and problems of Discrete mathematics, by Seymour Lipschutz & Marc Lars Lipson, Schaum's Outline Series, third edition 2007.
Main references (Sources)	Discrete Mathematics and Its Applications, Seventh Edition, Kenneth H. Rosen, AT&T Laboratories, 2012.
Recommended books and references (scientific journals, reports…)	• Mathematical foundation of computer science, Y.N. Singh, 2005 • Discrete structures, Amin Witno, Revision Notes and Problems 2006, www.witno.com Discrete mathematical structures for computer science by Bernard Kolman & Robert C. Busby
Electronic references, Websites	http://en.wikibooks.org/wiki/Discrete_mathematics/Set_theory

Course Description Form

1. Course Name:	
	Logic Design Fundamentals
2. Course Code:	
	TUCY112
3. Semester / Year:	
	2nd Semester/2024-2025
4. Description Preparation Date:	
	20-01-2025
5. Available Attendance Forms:	
	Theory, Practical, Tutorial
6. Number of Credit Hours (Total) / Number of Units (Total)	
	150 Hrs / 6 Units
7. Course administrator's name (mention all, if more than one name)	
	Name: Saif Muhand Maher Jassim **Email: saif.muhannad1985@tu.edu.iq**
8. Course Objectives	
Course Objectives	1. To develop problem-solving skills and an understanding of logic design 2. Teaching students the computer numerical systems 3. Teaching the student the logic gates, their structure, and the truth table 4. Teaching the student the sequential logic circuits, their analysis, and installation 5. Teaching students to design and analyze counters and registers and their types 6. Teaching the student the flip flop, its types, the state table, and the state diagram.
9. Teaching and Learning Strategies	
Strategy	To ensure effective delivery of **Logic Design Fundamentals** in the **first level of the Cybersecurity Department**, the following **teaching and learning strategies** will be applied: This course is characterized by the fact that it needs a special approach based mainly on the development of engineering thinking and the mathematical approach to thinking. Teaching depends mainly on homework that is presented at the end of each week, and the student notes the interdependence between the topics of the series in this course, in addition to assigning the student (or a group of students) to write one report and present it as a seminar for the purpose of training in the use of scientific resources, in addition to Assigning the student with a set of practical experiments that he implements in the laboratory and others that are given as

	homework, which will help the student to understand more broadly how logical circuits work.
	By implementing these strategies, students will gain both **theoretical knowledge** and **practical expertise**, ensuring they are well-prepared for advanced cybersecurity courses and real-world challenges.

10. Course Structure

Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning Method	Evaluation
1	6	Introduction to logic design and numerical systems	Logic Design	Lecture + Lab	Assignments
2	6	Convert between numerical systems and mathematical operations	Logic Design	Lecture + Lab	Assignments
3	6	Logical gates, how to draw them, and draw logical expressions	Logic Design	Lecture + Lab	Quizzes
4	6	Simplifying Boolean expressions using Boolean algebra	Logic Design	Lecture + Lab	Quizzes
5	6	Simplifying Boolean expressions using K-map	Logic Design	Lecture + Lab	Quizzes
6	6	Full Adder and Half Adder operations	Logic Design	Lecture + Lab	Assignments
7	1	Midterm Exam	Logic Design	Exam	Midterm Exam (10%)
8	6	Full Subtract and Half Subtract	Logic Design	Lecture + Lab	Lab work
9	6	Decoder and Encoder logic circuits	Logic Design	Lecture + Lab	Lab work
10	6	J-K Flip-Flop working principle	Logic Design	Lecture + Lab	Lab work
11	6	T Flip-Flop and D Flip-Flop	Logic Design	Lecture + Lab	Lab work
12	6	Shift Registers	Logic Design	Lecture + Lab	Lab work
13	6	Counters & Complex logical circuits	Logic Design	Lecture + Lab	Project Evaluation
14	6	Preparatory for Final Exam	Logic Design	Lecture	-
15	2	Final Exam	Covers All Units	Written & Practical Exam	Final Exam

11. Course Evaluation

The **evaluation structure** for the **Logic Design Fundamentals (TUCY112)** course is designed to comprehensively assess students' knowledge and practical skills, with a total course score of **100 marks**. The **final exam (50 marks)** is the most significant component, covering all course topics in both theoretical and practical formats to ensure students' mastery of Logic Design Fundamentals. The **mid-term exam (10 marks)** provides an early assessment of students' progress and understanding. The **coursework (40 marks)**, also known as **semester work**, consists of quizzes, assignments, lab reports, projects, and class participation, ensuring continuous assessment throughout the semester. This structure maintains a balanced evaluation of **theory and practical application**, preparing students for real-world cybersecurity challenges.

12. Learning and Teaching Resources

Required textbooks (curricular books, if any)	Principle of logic design (2020) by (Qasim Mohammed Hussein)
Main references (sources)	Digital logic and computer Design by Morris Mano
Recommended books and references (scientific journals, reports…)	
Electronic References, Websites	

Course Description Form

1. Course Name:	
	Coding and Information Theory
2. Course Code:	
	TUCY115
3. Semester / Year:	
	2nd Semester/2024-2025
4. Description Preparation Date:	
	20-01-2025
5. Available Attendance Forms:	
	Theory
6. Number of Credit Hours (Total) / Number of Units (Total)	
	100 Hrs / 4 Units
7. Course administrator's name (mention all, if more than one name)	
	Name: Asst. Prof. Dr. Mohaned Diab Mahdi **Email: mohaned@tu.edu.iq**

8. Course Objectives

Course Objectives	The objectives of the **" Coding & Information Theory "** course for first-year students in the **Cybersecurity Department** are designed to provide foundational knowledge and skills in cybersecurity. The key objectives include: 1. **Managing the fundamental components of an information system used in computer networks according to Shannon's theory.** This objective introduces students to how information systems operate within computer networks in terms of data flow, processing, and management. Based on Claude Shannon's principles, students will learn how to analyze, optimize, and improve the efficiency of data transmission and storage. 2. **Evaluating the amount of information a source contains or its redundancy and determining its efficiency using mathematical methods.** This objective helps students understand how to quantify information content using mathematical formulas, such as entropy, which measures uncertainty in data. Students will also learn how to detect redundant information and assess data efficiency to optimize data utilization. 3. **Differentiating between continuous and discrete information channels and mastering the methods of calculating their capacities.**

	Students will explore different types of information channels, including continuous channels (e.g., analog signals) and discrete channels (e.g., digital data). They will learn how to compute channel capacity, which determines the maximum data transmission rate without loss, based on Shannon's theorem. 4. **Acquiring detailed and practical knowledge of the fundamental types of source encoding and how to measure their efficiency.** This objective focuses on data encoding techniques that help compress and optimize data transmission. Students will study Shannon-Fano coding and Huffman coding, learning how to represent data with the minimum number of bits while maintaining its original meaning. 5. **Gaining detailed knowledge of basic channel encoding techniques, including error detection and correction methods.** These objective covers channel encoding techniques that ensure error-free data transmission by adding extra information for error detection and correction. Students will learn about techniques such as Error Detection and Correction Codes and Hamming Code, which help in maintaining data integrity during transmission. 6. **Understanding the fundamental causes of channel errors and learning techniques to minimize their impact.** Students will examine common issues affecting data transmission, such as noise, interference, and distortion, which can lead to data loss or corruption. They will explore error correction methods and channel selection strategies to reduce errors and improve communication quality.
9. Teaching and Learning Strategies	
Strategy	To ensure students achieve the required learning outcomes, the following **teaching and learning strategies** are implemented: 1. **Lectures and Explanations** ✓ Fundamental **concepts, theories, and mathematical models** in Coding and Information Theory are presented interactively. ✓ **Shannon's theory, entropy measurement, source encoding, and channel coding** are explained using real-world examples. 2. **Problem-Solving and Worked Examples** ✓ **Mathematical exercises** covering probability calculations, coding efficiency, and channel capacity are provided to strengthen students' analytical skills. ✓ Sample **exam-style questions** are discussed in class to help students prepare for assessments. 3. **Self-Learning and Independent Assignments** ✓ Students are assigned **independent research tasks**, such as studying error correction methods or analyzing the efficiency of compression techniques. ✓ **Homework assignments** include practical applications using **data analysis tools and coding techniques**. 4. **Use of Software and Practical Simulations** ✓ **Hands-on lab sessions** using software such as **MATLAB, Lab Mat, or Python** to implement **coding techniques, error detection, and correction methods**. ✓ Students develop **simulations of digital communication systems** and analyze the performance of various information channels.

5. **Class Discussions and Interactive Q&A**
   - ✓ **Spontaneous questions** are asked during lectures to test student engagement and encourage critical thinking.
   - ✓ **Group discussions** are held on modern applications of information theory, such as **cybersecurity, data compression, and digital communication**.
6. **Collaborative Learning and Group Projects**
   - ✓ Students work in teams on projects related to **data analysis, encryption, and error correction techniques**.
   - ✓ They are required to **present their reports and findings** to enhance communication and teamwork skills.
7. **Continuous Assessments and Quizzes**
   - ✓ **Regular and surprise quizzes** are conducted to evaluate students' understanding of theoretical concepts.
   - ✓ Students are assessed through **practical reports, assignments, midterm exams, and final evaluations**.
8. **Encouraging Creativity and Problem-Solving**
   - ✓ Students are encouraged to **design new coding techniques** or analyze methods to **improve channel efficiency**.
   - ✓ **Real-world challenges** are presented, requiring theoretical and practical solutions.

Assessment Methods

1. **Regular and surprise theoretical tests** to measure conceptual understanding.
2. **Assignments and practical applications** to evaluate problem-solving skills.
3. **Research reports and case studies** to enhance critical thinking and independent research.

10. Course Structure

Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning Method	Evaluation
1	2	Apply probability theory in information theory	Probability and Statistics Review, Definition of Alphabet, Random Variables	Lecture + Problem Solving	-
2	2	Understand independent and dependent events using probability rules	Joint Probability, Conditional Probabilities, Bayes' Rule, Venn's Diagram	Lecture + Exercises	-
3	2	Learn the fundamental model of an information transmission system	Information Transmission Model, Logarithmic Measure of Information, Self-Information	Lecture + Discussion	-
4	2	Compute mutual information and posterior probabilities in noisy channels	Definition of Information for Noisy Channels, Posteriori Probabilities	Lecture + Problem Solving	-

5	2	Describe information channels and their parameters	Shannon Representation Diagram, Discrete Channel Parameters	Lecture + Problem Solving	-
6	2	Compute entropy for discrete and continuous sources and analyze source efficiency	Average Information (Entropy) of Discrete and Continuous Sources, Maximum Entropy	Lecture + Exercises	-
7	2	Model discrete noiseless and noisy channels using transition probability matrices	Transition Probability Matrix, Noiseless and Noisy Channels, Uniform Channel, Ternary Symmetric Channel	Lecture + Lab	-
8	2	Understand and apply concepts of symmetric and binary symmetric channels	Transmission over Noiseless and Noisy Channels, Binary and Ternary Symmetric Channels	Lecture + Lab	-
9	1	Midterm Exam	Covers Previous Topics	Exam	Midterm Exam (10%)
10	2	Calculate channel capacity for noiseless and symmetric channels	Channel Capacity for Noiseless Channels, Efficiency and Redundancy	Lecture + Exercises	-
11	2	Calculate capacity for non-symmetric channels	Binary Non-Symmetric Channel, Channel Capacity	Lecture + Exercises	-
12	2	Differentiate between discrete and continuous information channels	Mutual Information and Efficiency of Continuous Channels	Lecture + Problem Solving	-
13	2	Understand entropy for continuous uniform and Gaussian distribution sources	Entropy for Continuous Uniform and Gaussian Distribution Sources	Lecture + Exercises	-
14	2	Apply Shannon-Hartley Theorem and Nyquist theorem in communication systems	Band-Limited Channels, Shannon-Hartley Capacity Theorem, Nyquist Theorem	Lecture + Discussion	-

15	2	Final Exam	Covers All Units	Written & Practical Exam	Final Exam (50%)

11.Course Evaluation

The evaluation structure for the **Coding and Information Theory (TUCY115)** course is designed to comprehensively assess students' theoretical understanding and analytical skills, ensuring a strong foundation in information theory and coding principles. The total course score is **100 marks**, with the **final exam (50 marks)** being the most significant component. This exam covers all course topics in a **theoretical format**, evaluating students' ability to apply probability theory, entropy calculations, channel capacity analysis, and coding techniques in real-world cybersecurity and data communication contexts. The **mid-term exam (10 marks)** serves as an early assessment of students' progress, focusing on fundamental concepts such as information transmission models, mutual information, and entropy measurements.

12.Learning and Teaching Resources

Required textbooks (curricular books, if any)	
Main references (sources)	**Forouzan, B. A.** (2007). *Data Communications and Networking*. McGraw-Hill, Forouzan Networking Series. © The McGraw-Hill Companies, Inc.
Recommended books and references (scientific journals, reports…)	**Cover, T. M., & Thomas, J. A.** (2006). *Elements of Information Theory* (2nd ed.). John Wiley & Sons.
Electronic References, Websites	

Course Description Form

1. Course Name:
Arabic Language
2. Course Code:
UOT001
3. Semester / Year:
2024-2025
4. Description Preparation Date:
18/01/2025
5. Available Attendance Forms:
Theory
6. Number of Credit Hours (Total) / Number of Units (Total)
50 Hrs/ 2 Units
7. Course administrator's name (mention all, if more than one name)
Name: Ahmed Mohammed Shareef **Email: ahmed.m.shareef@tu.edu.iq**
8. Course Objectives

Course Objectives	1- Strengthening students' linguistic ability. 2- The student acquires full knowledge of the basics of the Arabic language and provides them with the skill of correct expression. 3- Clarifying the importance of grammatical rules of the language 4- Strengthening students' linguistic ability and knowing common mistakes in the language. 5- Knowing the basic rules and being able to use and apply them.
9. Teaching and Learning Strategies	
Strategy	The learning and teaching strategy was developed in order for the student to obtain complete information covering the curriculum prepared for the subject and in order to achieve the basic goal of the curriculum, which is focused on the student's mastery and understanding of the basic concepts of the Arabic language subject. This course is distinguished by the fact that it requires a special approach that depends primarily on developing the student's ability to understand grammatical and spelling rules and how to apply them to Quranic and poetic texts, and not to fall into errors.

10. Course Structure

Week	Hourse	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
Week 1	2	Introducing the student to the terms related to the concept of spelling and grammar rules.	Speech and what it consists of	Cooperative learning strategy. Brainstorming learning strategy. Cooperative concept	Daily exams, oral and written, reports and discussions

				planning learning strategy. Real-time feedback learning strategy Notes series learning strategy. Exchange and discussion learning strategy. Information presentation learning strategy. Training and presentation of scientific developments learning strategy.	
Week 2	2	Introducing the student to grammar rules and the ability to use them	Verb types	=	=
Week 3	2	Introducing the student to grammar rules and the ability to use them	Noun and verb signs	=	=
Week 4	2	Introducing the student to the importance of accuracy in observation and distinguishing between right and wrong in what they hear or read, which helps them understand the meanings of sentences and styles	Ta marbuta and ta mabsuta	=	=
Week 5	2	Introducing the student to the importance of accuracy in observation and distinguishing between right and wrong in what they hear or read, which helps them understand the meanings of sentences and styles	The shortened noun, its dual and plural, and the extended noun, its dual and plural	=	=
Week 6	2	Introducing the student to speech and what it consists of	Kan and its sisters	=	=
Week 7	2	Introducing the student to speech and what it consists of	Kan and its sister’s types	=	=
Week 8	2	Introducing the student to distinguish between verbs, nouns and letters by displaying the signs of each section of speech	Verb-like letters and their meanings	=	=

Week 9	1	From Week 1 to Week 8	Mid-term exam	=	=
Week 10	2	Introducing the student to the importance of accuracy in observation and distinguishing between right and wrong in what they hear or read, which helps them understand the meanings of sentences and styles	Positions of breaking the hamza of in and opening it	=	=
Week 11	2	Introducing the student to speech and what it consists of	The subject and its states	=	=
Week 12	2	Introducing the student to distinguish between verbs, nouns and letters by displaying the signs of each section of speech	The inflected and the uninflected	=	=
Week 13	2	Introducing the student to non-declinable words	The five verbs and their inflection	=	=
Week 14	2	Introducing the student to non-declinable words	The non-declinable	=	=
Week 15	2	From Week 1 to Week 14	Final exam.	/	/

11. Course Evaluation

Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc

12. Learning and teaching resources

Required textbooks (curricular books, if any)	General Arabic for non-specialized departments / Compiled and prepared by M.M. Bushra Adel Saleh / Tikrit University / Computer Science and Mathematics
Main references (Sources)	Explanation of Ibn Aqil and Qatar Al-Nada
Recommended books and references (scientific journals, reports…)	

Course Description Form

1. Course Name:
Democracy and human rights
2. Course Code:
UOT003
3. Semester / Year:
2024-2025
4. Description Preparation Date:
18/01/2025
5. Available Attendance Forms:
Theory
6. Number of Credit Hours (Total) / Number of Units (Total)
50 Hrs/ 2 Units
7. Course administrator's name (mention all, if more than one name)
Name: Saad Hussain Ali **Email: Saad.h.ali@tu.edu.iq**
8. Course Objectives

Course Objectives	1- Learn about the human rights situation in ancient and modern civilizations and heavenly religions 2- Study how the true Islamic religion deals with religious and worldly human rights, and shows that the great Islam has granted the individual many great rights before his birth and after his death, which is something we do not find in the rest of the ancient and modern civilizations. 3- Distinguish between human rights topics in the theoretical aspect and their practical and realistic applications. 4- Understand how the human rights topic was dealt with in the League of Nations and the role of the United Nations and its specialized agencies. 5- The course studies the types of human rights and the guarantees included in the Constitution of the Republic of Iraq for the year 2005. 6- Study societal human rights and modern human rights.

9. Teaching and Learning Strategies	
Strategy	1- To be able to possess legal thinking independently 2- To develop and improve reporting skills 3- To have realistic experiences with cognitive perceptions 4- To have a clear goal that he seeks to achieve

10. Course Structure

Week	Hourse	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method

Week 1	2	Chapter One - Human Rights in Ancient Civilizations and Heavenly Religions	Human Rights	Scientific lecture method	Identification Test
Week 2	2	Human Rights in Judaism and Christianity	Human Rights	Discussion method by directing questions to students	Short Tests
Week 3	2	Chapter Two/ Ancient and Modern Human Rights Documents	Human Rights	Encouraging students to dialogue and ask questions	Identification Test
Week 4	2	International Sources of Human Rights	Human Rights	Discussion method	Short Tests
Week 5	2	Human Rights in International Organizations	Human Rights	Scientific lecture method	Identification Test
Week 6	2	Human Rights in Regional Organizations	Human Rights	Scientific lecture method	Identification Test
Week 7	2	Human Rights in the Arab League	Human Rights	Discussion method	Short Tests
Week 8	2	Chapter Three/ Human Rights in the Constitution of the Republic of Iraq for the year 2005	Human Rights	Scientific lecture method	Short Tests
Week 9	1	Mid-term Exam	/	/	/
Week 10	2	NGOs and Human Rights	Democracy	Scientific lecture method	Identification Test
Week 11	2	Chapter Four/ Mass Graves Crimes	Democracy	Scientific lecture method	Identification Test
Week 12	2	Crimes of Genocide	Democracy	Discussion method	Short Tests
Week 13	2	NGOs and Human Rights	Democracy	Scientific lecture method	Short Tests
Week 14	2	Chapter Four/ Mass Graves Crimes	Democracy	Scientific lecture method	Short Tests
Week 15	2	Final Exam	Final exam.	/	/

11. Course Evaluation	
Distributing the score out of 100 according to the tasks assigned to the student such as daily preparation, daily oral, monthly, or written exams, reports, etc	
12. Learning and teaching resources	
Required textbooks (curricular books, if any)	
Main references (Sources)	• Dr. Ghassan Karim Al-Mujtab and Amjad Zain Al-Abidin Taama, Human Rights and Democracy, 2018. • Zouina Al-Walid, The Crime of Genocide in Light of the Judicial Efforts of the International Criminal Tribunal for Rwanda, Unpublished Master's Thesis, (University of Algiers, Faculty of Law, Ben Aknoun, 2013). • David Betham and Kevin Boyle, Introduction to Democracy: Free and Fair Elections, Translated by: Gharib Awad, (Bahrain, Faradees Publishing and Distribution House, 2007). • The Iraqi Constitution of 2005.
Recommended books and references (scientific journals, reports…)	Mohamed Gharbi, Democracy and Good Governance: The Challenges of Political Participation and Achieving Development, Special Issue, (Algeria, Journal of Political and Legal Notebooks, April 2011).
Electronic references, Websites	United Nations Universal Declaration of Human Rights 1948, https://www.supremecourt.ge