Academic Program Description Form

University Name: Tikrit University

Faculty/Institute: College of computer science and mathematics

Scientific Department: computer science

Academic or Professional Program Name: University Performance

Final Certificate Name: Bachelor's

Academic System: Semester

Description Preparation Date: 27 / 1 / 2025

File Completion Date: 27 / 1 / 2025

Signature:

Head of Department Name:

Asst.Prof.Dr. Mohamed Aktham Ahmed

Date: 2025/1/27

Signature:

Scientific Associate Name:

Asst. Prof. Dr. Majid Hamid Ali

Date: 2025/1/27

The file is checked by:

Department of Quality Assurance and University Performance

Director of the Quality Assurance and University Performance Department:

Lecturer: Yaamr abdulateef Hussein

Date: 27/1/25

Signature:

2025

Approval of the Dean

1. Program Vision

Providing distinguished academic programs in the field of computer science, both theoretical and applied, that comply with international standards of academic quality and meet the needs of the labor market. Encouraging and developing scientific research in the fields of computer science in general, and the fields of artificial intelligence, robotics, computational linguistics, and networks in particular.

2. Program Mission

The mission of the department is to prepare students professionally and academically with high-quality education through a model program equipped with all the teaching and learning tools the student needs and a highly qualified academic staff. The ultimate mission of the department is to provide distinguished education and a suitable academic environment in the fields of computer science.

3. Program Objectives

1. Providing modern and diverse educational programs in the fields of computer science, software and information technology, that meet international quality standards and keep pace with technical developments.
2. Graduating distinguished students with technical and creative skills and the ability to solve complex problems in the fields of computer science and software.
3. Providing an integrated educational environment based on the latest tools and technologies to improve
4. Supporting students and faculty members in adopting a culture of innovation and technological entrepreneurship, and encouraging the creation of innovative technical projects.
5. Providing distinguished graduate programs in software and computer science to qualify advanced academic and research competencies.
6. Instilling the values of integrity, credibility and social responsibility in all aspects of education and scientific research.
7. Providing training programs and workshops to raise the efficiency of students, graduates and faculty members and enhance opportunities for continuous learning.

4. Programmatic accreditation:
1. The student learns programming languages. 2. The ability to find scientific software solutions to community problems. 3. Develop the student's skills in building smart systems that depend on the basis of analysis and inference. 4. Provide the student with the basic rules in building software systems based on the basics of software engineering. 5. Increase the student's information on the basics of implementing software systems by understanding the mechanism of computer operation.

5. Other external influences:
1. Through the scientific conference for students. 2. The quarterly scientific seminar of the department. 3. Discussion sessions for professors. 4. Research seminars.

Notes	Unit of study	Number of courses	Program Structure
	12	6	Institutional Requirements
	18	6	Faculty Requirements
	84	20	Department Requirements
	Completed	1	Summer Training
			Other

* Notes can include whether the course is core or optional.

7. Program Description				
Credit Hours		Course Name	Course Code	Level
Lab	Theoretical			
2	2	Programming Basics	TU010101107	Stage 1/Semester 1
2	2	Principles of Computer Structure	TU010101109	Stage 1/Semester 1
	4	Discrete Structures 1	TU010101108	Stage 1/Semester 1
	4	Differential and Integral Calculus 1	TU010101110	Stage 1/Semester 1
	2	English Language 1	TU010101101	Stage 1/Semester 1
	2	Democracy and Human Rights	TU010101102	Stage 1/Semester 1
2	2	Advanced Programming	TU010101113	Stage 1/Semester 2
2	2	Website Programming	TU010101115	Stage 1/Semester 2
2	2	Algorithm Design	TU010101111	Stage 1/Semester 2
2	2	Logic Design	TU010101112	Stage 1/Semester 2
	4	Computational Theory	TU010101114	Stage 1/Semester 1
-	4	Computer Architecture	TU010102002	Stage 2/ First Semester
-	4	Systems Analysis and Design	TU010102003	Stage 2/ First Semester
2	2	Numerical Analysis	TU010102004	Stage 2/ First Semester
2	2	Data Structures 1	TU010102005	Stage 2/ First Semester
-	4	Computational Theory 1	TU010102006	Stage 2/ First Semester
2	2	Entity Programming	TU010102007	Stage 2/ First Semester
-	2	Baath Party Crimes in Iraq	TU010102008	Stage 2/ First Semester
-	2	Research Methodology	TU010102009	Stage 2/ Second Semester
-	4	Probability and Random Variables	TU010102010	Stage 2/ Second Semester
2	2	Internet of Things	TU010102011	Stage 2/ Second Semester
2	2	Data Structures 2	TU010102012	Stage 2/ Second Semester
2	2	Microprocessors	TU010102013	Stage 2/ First Semester

2	2	Visual Programming	TU010102014	Stage 2/ First Semester
2	2	Encryption	TU010103001	Stage 3/ First Semester
2	2	Compiler1	TU010103002	Stage 3/ First Semester
-	4	Operations Research	TU010103003	Stage 3/ First Semester
-	2	Principles of Management	TU010103004	Stage 3/ First Semester
2	2	Databases 1	TU010103005	Stage 3/ First Semester
2	2	Software Engineering	TU010103006	Stage 3/ First Semester
2	2	Artificial Intelligence	TU010103007	Stage 3/ Second Semester
2	2	Databases 2	TU010103008	Stage 3/ Second Semester
-	4	Mobile Computing	TU010103009	Stage 3/ Second Semester
2	2	Operating Systems 2	TU010103010	Stage 3/ Second Semester
-	2	English Language 2	TU010103011	Stage 3/ First Semester
2	2	compiler 2	TU010103012	Stage 3/ First Semester
2	2	Digital Signal Processing	TU010103013	Stage 3/ First Semester
2	2	Digital Image Processing	TU010104001	Stage 4/ First Semester
2	2	Operating Systems 2	TU010104002	Stage 4/ First Semester
2	2	Computer Security	TU010104003	Stage 4/ First Semester
-	4	Computer Networks	TU010104004	Stage 4/ First Semester
-	2	Modeling and Simulation	TU010104005	Stage 4/ First Semester
-	2	Graduation Project	TU010104012	Stage 4/ First Semester
-	4	Distributed Systems	TU010104007	Stage 4/ Second Semester
2	2	Computer Networks 2	TU010104008	Stage 4/ Second Semester
2	2	Multimedia and Network Security	TU010104009	Stage 4/ Second Semester
2	2	Networks Nervousness	TU010104010	Stage 4/ Second Semester
2	2	Data Mining	TU010104011	Stage 4/ First Semester
-	2	Graduation Project	TU010104012	Stage 4/ First Semester

<table>
<tr><th colspan="2">8. Expected Learning Outcomes of the Program</th></tr>
<tr><th colspan="2">Knowledge</th></tr>
<tr><td colspan="2">1. Enable students to gain knowledge and understanding of computer basics
2. Enable students to gain knowledge and understanding of computer applications
3. Enable students to gain knowledge and understanding of computer programming
4. Enable students to gain knowledge and understanding of computer networks
5. Enable students to gain knowledge and understanding of multimedia
Enable students to gain knowledge and understanding of data science and mining.</td></tr>
<tr><th colspan="2">Skills</th></tr>
<tr><td>Memorization and analysis skills</td><td>Scientific skills</td></tr>
<tr><td>Empowering students to obtain knowledge and understanding of computer programming
The ability to remember information accurately and in an organized manner.
The ability to retrieve information quickly and effectively when needed.
The ability to prioritize activities and allocate time effectively.</td><td>Proficiency in different programming languages such as Java, C, C++, etc.
Understanding the basic processes and concepts of software development in an effective and organized manner.
Understanding how to build and manage information networks.
Understanding security threats and how to protect data and networks from them.
Learn artificial intelligence and machine learning and understand the models and algorithms used in developing intelligent systems.
The ability to extract and analyze data effectively to obtain valuable insights.
Learn the basics of saving and viewing data using different databases.
Web application development</td></tr>
<tr><th colspan="2">Values</th></tr>
<tr><td>Learning Outcome Statement 4</td><td>Enable students to gain knowledge and understanding of computer networks</td></tr>
<tr><td>Learning Outcome Statement 5</td><td>Enable students to gain knowledge and understanding of multimedia</td></tr>
</table>

9. Teaching and learning strategies:

1- Providing students with the basics and additional topics related to the previous educational outcomes of skills to solve scientific problems.

2- Solving a set of scientific examples by the academic staff.

3- Asking students during the lecture to solve some scientific problems

4. Visiting all scientific computer science laboratories by the academic staff.

10. Evaluation methods:

1- Standard Quality Systems (standards Quality)

2- Daily exams with self-solved homework questions.

3- Points for sharing questions for a competition related to the subject matter.

4- Points for specific homework assignments.

11. The teaching staff

The teaching staff members

الرتبة العلمية	التخصص		المتطلبات/المهارات (ان وجدت)	اعداد الهيئة التدريسية	
	عام	خاص		ملاك	محاضر
ا.م.د. محمود ماهر صالح علاوي	علوم الحاسوب	ذكاء اصطناعي و تطبيقاته	ذكاء اصطناعي و تطبيقاته	ملاك	-
ا.م.د. محمد أكثم احمد عبد الحليم	علوم الحاسوب	ذكاء اصطناعي و تطبيقاته	ذكاء اصطناعي و تطبيقاته	ملاك	-
ا.م.د. زيدون طارق عبد الوهاب عبد اللطيف	علوم حاسبات	ذكاء اصطناعي	الذكاء الاصطناعي وتطبيقاته	ملاك	-
ا.م.د. د. سعدي حمد ثلج احمد	علوم حاسبات	تكنولوجيا قواعد البيانات وتطبيقاتها	قواعد البيانات وتطبيقاتها	ملاك	-
ا.م.د. مهند ذياب مهدي غثيث	علوم حاسبات	تكنولوجيا أنترنت الاشياء وتطبيقاتها	تكنولوجيا المعلومات والاتصالات	ملاك	-
ا.م.د. مشاري عايد عسكر ملاش	علوم الحاسوب	تكنولوجيا معالجة الإشارة الرقمية وتطبيقاتها	شبكلن واتصالات معالجه الأشاره	ملاك	-
ا.م.د. ميثم مصطفى حمود جزاع	علوم حاسوب	امنية معلومات	امنية معلومات	ملاك	-
ا.م. نايف محمد حماش خلف	علوم الحاسوب	تكنولوجيا المعلومات	علوم الحاسوب/ تكنولوجيا المعلومات وتطبيقات ذكية	ملاك	-

		وتطبيقاتها			
ا.م. مروى أديب محمد يحيى	علوم الحاسوب	هندسة برمجيات	هندسة برمجيات، نظم متعددة الوكلاء، الوكيل المتنقل، ادارة مشاريع، ذكاء اصطناعي	ملاك	-
ا.م. حارث عبدالله حسين عبدالله	علوم الحاسوب	تكنالوجيا الويب	تطبيقات الويب و الإنترنت	ملاك	-
ا.م. سلوى خالد عبد اللطيف	علوم حاسبات	الذكاء الاصطناعي	تعلم الالة- الذكاء الاصطناعي- معالجة الصور	ملاك	-
ا.م. احمد سعدي عبدالله عمر	علوم حاسوب	رؤية حاسوبية ومعالجة الصور الرقمية	ذكاء اصطناعي ، معالجة الصور الرقمية ، امنية	ملاك	-
ا.م. محمد وليد نوفان عسكر	علوم حاسوب	معالجة صور	الذكاء الاصطناعي - معالجة الصور الرقمية	ملاك	-
ا.م. وسام داود عبدالله خلف	علوم الحاسوب	شبكات وانترنت الاشياء وتطبيقاتها	انترنت الاشياء وتطبياقته كالذكاء الاصطناعي و تعلم الالة و التعلم العميق بالاضافة الى الاتمتة و الامن السيبراني وتحليل بيانات الانترنت	ملاك	-
ا.م. ماجد حامد علي جاسم	علوم حاسبات	تكنالوجيا المعلومات	شبكات	ملاك	-
م.د. اياد احمد ياس رومي	علوم حاسبات	نظم المعلومات	نظم المعلومات	ملاك	-
م.د. مصحب لزام شوندي مهاوش	علوم حاسوب	امنية الموبايل	امنية الحاسوب وتطبيقات الذكاء الاصطناعي	ملاك	-
م.د. خالد خالص ابراهيم عزت	علوم الحاسوب	هندسة البرمجيات والذكاء الاصطناعي	الذكاء الاصطناعي والامن السيبراني	ملاك	-
أ.م. ارمانيسة نعمان حسون علي	علوم حاسوب	ذكاء اصطناعي	ذكاء اصطناعي	ملاك	-
م. ليث رافع حازم علي	علوم حاسبات	تدريب الالة وتنقيب البيانات	ذكاء اصطناعي	ملاك	-
م. روان عادل فوزي شهاب	علوم حاسوب	الامن السيبراني	امنية الحاسوب	ملاك	-
م. مهند داود سلمان رومي	علوم الحاسوب	الذكاء الاصطناعي وتطبيقاته	الذكاء الاصطناعي	ملاك	-
م.م. لؤي ابراهيم خلف عبدالله	ماجستير علوم حاسبات	نظم معلومات	مستمر بدراسة الدكتوراه	ملاك	-
م.م. عمار فاروق عباس وهيب	هندسة حاسبات	هندسة حاسبات	انظمة مدمجة وانترنت الاشياء	ملاك	-
م.م. عقبه سالم عطيه احمد	علوم حاسوب	ذكاء اصطناعي	الذكاء الاصطناعي	ملاك	-
م.م. يحيى ليث خليل محمود	علوم حاسوب	تكنولوجيا المعلومات	ذكاء اصطناعي	ملاك	-
م.م. محمدطاهر أحمد محمد أحمد	علوم حاسوب	علوم حاسوب	-	ملاك	-
م.م. ريم ضياء اسماعيل هلال	علوم حاسوب	Web technology	الذكاء الاصطناعي	ملاك	-
م.م. تحسين علي نعيم محمود	هندسة حاسبات	هندسة حاسبات	الذكاء الاصطناعي وتمييز الانماط - معالجة الاشارة الرقمية - معمارية	ملاك	-

			الحاسوب		
م.م. حسن ناجي علي سبع	علوم حاسوب	ذكاء اصطناعي	ذكاء اصطناعي	ملاك	-
م.م. شهد محمود مصطفى محمود	علوم الحاسوب	علوم الحاسوب	الذكاء الاصطناعي	ملاك	-
م.م. قبس عامر حميد محمد	علوم حاسوب	علوم حاسوب	الواقع المعزز	ملاك	-
م.م. محمد باسم عمر صالح	علم الحاسوب	علم الحاسوب	الذكاء الاصطناعي	ملاك	-
م.م. مصطفى عبدالفتاح حبيب رجب	علوم الحاسوب	ITتكنلوجيا المعلومات	الذكاء الاصطناعي	ملاك	-
م.م. همام نعمة حسين درويش	هندسة كهرباء وحاسبات	هندسة شبكات الاتصالات	شبكات الاتصالات	ملاك	-
م. سيف مهند ماهر جاسم	علوم حاسوب	هندسة البرامجيات	هندسة برامجيات	ملاك	-
م.م. حذيفة محمد كنوش حمد	علوم الحاسوب	علوم الحاسوب	الذكاء الاصطناعي	ملاك	-
م.م. احمد فايق صابر عجوان	اللغة الانكليزية	لغة انكليزية	طرائق تدريس اللغة الانكليزية	ملاك	-
م. ريا نزار اسماعيل عبد الله	علوم حاسوب	ذكاء اصطناعي	الذكاء الاصطناعي	ملاك	-
م.م. نور سعود عبد عزيز	علوم الحاسوب	تعلم الالة	الذكاء الاصطناعي	ملاك	-
م.م. مهند حاتم رمضان قدوري	علوم الحاسوب	علوم الحاسوب	علوم الحاسوب	ملاك	-
م.م. مصطفى لطيف فاضل لطيف	علوم الحاسوب	علوم الحاسوب	قواعد بيانات، تحليل بيانات	ملاك	-
م.م. اسراء رافع عبد القادر محمد	علوم حاسوب	الذكاء الاصطناعي	الذكاء الاصطناعي	ملاك	-
م.م. صبا علاء عبد الوهاب عبد الرزاق	علوم حاسوب	امنية حاسوب	امنية حاسوب	ملاك	-

Academic Staff

Harith Abdullah Hussein | MS.C. in Computer Science | Professor
Email: Harith_abd1981@tu.edu.iq
Mobile no.: 07740890318

Mahmoud Maher Salih | Ph.D. in Computer Science | Assistant Prof.
Email: mahmood.maher@tu.edu.iq
Mobile no.:07701849377

Mohamed Aktham Ahmed | Ph.D. in Computer Science | Assistant Prof.
Email: mohamed.aktham@tu.edu.iq
Mobile no.: 07705137200

Zaidoun Tariq Abdel-Wahhab | Ph.D. in Computer Science | Assistant Prof.
Email: ziadoontareq@tu.edu.iq
Mobile no.: 07701849377

Saadi Hamad snow Ahmed | Ph.D. in Computer Science | Assistant Prof.
Email: saadi.alluhaibi@tu.edu.iq
Mobile no.: 07701230860

Muhannad Diab Mahdi | Ph.D. in Computer Science | Assistant Prof.
Email: muhaned@tu.edu.iq
Mobile no.: 07737736042

Meshary Ayed Askar | Ph.D. in Computer Science | Assistant Prof.
Email: Dr.mshary.alshmmry@tu.edu.iq
Mobile no.: 07700884407

Maytham Mustafa Hammoud | Ph.D. in Computer Science | Assistant Prof.
Email: Maythamhammood@tu.edu.iq
Mobile no.: 7722415385

Nayef Mohammed Hamash | MS.C. in Computer Science | Assistant Prof.
Email: nayifit81@tu.edu.iq
Mobile no.: 07727696595

Marwa Adeeb Muhammad | MS.C. in Computer Science | Assistant Prof.
Email: Marwa.A.Aljawaherry@tu.edu.iq
Mobile no.: 07705136406

Salwa Khaled Abdel | MS.C. in Computer Science | Assistant Prof.
Email: khalid.salwa@tu.edu.iq
Mobile no.: 7702007940

Ahmed Saadi Abdullah | MS.C. in Computer Science | Assistant Prof.
Email: ahmedalbasha@tu.edu.iq
Mobile no.: 07825254938

Muhammad Walid Nofan | MS.C. in Computer Science | Assistant Prof.
Email: moha.wn@tu.edu.iq
Mobile no.: 07717171590

Wissam Daoud Abdullah | MS.C. in Computer Science | Assistant Prof.
Email: wisamdawood@tu.edu.iq

Mobile no.: 07716656567

Iyad Ahmed Yas | Ph.D. in Computer Science | Lecturer
Email: ayad@alzwea.com
Mobile no.: 07500449091

Moceheb Lazam Shwandi | Ph.D. in Computer Science | Lecturer
Email: moceheb@tu.edu.iq
Mobile no.: 7732558541

Khaled Khalis Ibrahim | Ph.D. in Computer Science | Lecturer
Email: khalid.kh.ibrahim@tu.edu.iq
Mobile no.: 07722987251

Armanisa Numan Hassoun | MS.C. in Computer Science | Assistant Prof.
Email: armaneesa@tu.edu.iq
Mobile no.: 07710791059

Laith Rafi Hazem | MS.C. in Computer Science | Lecturer
Email: layth.r.hazim@tu.edu.iq
Mobile no.: 7703773050

Rawan Adel Fawzi | MS.C. in Computer Science | Lecturer
Email: Rawan_adel@tu.edu.iq
Mobile no.: 07701715536

Muhannad Daoud Salman | MS.C. in Computer Science | Lecturer
Email: mohanaddawoodalroomi@tu.edu.iq
Mobile no.: 7727696585

Majid Hamed Ali | MS.C. in Computer Science | Assistant Prof.
Email: Majid.hamid@tu.edu.iq
Mobile no.: 07730598525

Louay Ibrahim Khalaf | MS.C. in Computer Science | Assistant Lect.
Email: Luay.I.khalaf@tu.edu.iq
Mobile no.: 07701351845

Ammar Farouk Abbas | MS.C. in Computer Science | Assistant Lect.
Email: ammar.abbas@tu.edu.iq
Mobile no.: 07702786369

Oqba Salem Attia | MS.C. in Computer Science | Assistant Lect.
Email: akabasalim4@gmail.com
Mobile no.: 07703582947

Yahya Laith Khalil | MS.C. in Computer Science | Assistant Lect.
Email: yahya@tu.edu.iq
Mobile no.: 7703235197

Muhammad Tahir Ahmed Muhammad | MS.C. in Computer Science | Assistant Lect.
Email: mohammed.t.a@tu.edu.iq
Mobile no.: 07714719810

Reem Diaa Ismail | MS.C. in Computer Science | Assistant Lect.
Email: reem.dh.ismail@tu.edu.iq
Mobile no.: 07811797206

Tahsin Ali Naim | MS.C. in Computer Science | Assistant Lect.
Email: tahsin.a.naeem@tu.edu.iq
Mobile no.: 07732058949

Hassan Naji Ali | MS.C. in Computer Science | Assistant Lect.
Email: hasan@tu.edu.iq
Mobile no.: 7718440458

Mahmoud Mustafa Mahmoud | MS.C. in Computer Science | Assistant Lect.
Email: Shahad.M.Mustafa@tu.edu.iq
Mobile no.: 07702347775

Qabas Amer Hamid | MS.C. in Computer Science | Assistant Lect.
Email: qabas.a.hameed@tu.edu.iq
Mobile no.: 07718078472

Muhammad Bassem Omar | MS.C. in Computer Science | Assistant Lect.
Email: mohammed.b@tu.edu.iq
Mobile no.: 07732087451

Mostafa Abdel Fattah Habib | MS.C. in Computer Science | Assistant Lect.
Email: Mustafa@tu.edu.iq
Mobile no.: 07706692441

Hammam Nima Hussein | MS.C. in Computer Science | Assistant Lect.
Email: humam.n.hussein@tu.edu.iq

Mobile no.: 07713360666

Saif Muhannad Maher | MS.C. in Computer Science | Lecturer
Email: saif_muhannad1985@tu.edu.iq
Mobile no.: 07700507119

Hudhaifa Muhammad Knoush | MS.C. in Computer Science | Assistant Lect.
Email: huthife@tu.edu.iq
Mobile no.: 07701720731

Raya Nizar Ismail | MS.C. in Computer Science | Assistant Lect.
Email: raya_computer@tu.edu.iq
Mobile no.: 07701874000

Noor Saud Abdel | MS.C. in Computer Science | Assistant Lect.
Email: noorsaud033@gmail.com
Mobile no.: 07734167627

Muhannad Hatem Ramadan | MS.C. in Computer Science | Assistant Lect.
Email: Mohanad.H.Ramadhan@tu.edu.iq
Mobile no.: 7703791541

Mustafa Latif Fadel | MS.C. in Computer Science | Assistant Lect.
Email: mustafalatif9@gmail.com
Mobile no.: 07733762075

Esraa Rafi Abdul Qadir | MS.C. in Computer Science | Assistant Lect.
Email: israa.r.abdalkader@tu.edu.iq
Mobile no.: 7733601366

Professional Development

Orientation of New The teaching staff

For visiting faculty, a specific process can be designed to guide and support them during their time at the college. The following is a brief description of the process used to guide visiting faculty:

1- Welcome and Orientation: Visiting faculty are warmly welcomed and provided with an orientation about the college's culture, policies, and resources. They receive an introduction to the department they will be working with and are introduced to key staff and support personnel.

2- Integration: Visiting faculty are integrated into their department during their visit. They meet with department chairs, fellows, and administrative staff who brief them on departmental expectations, teaching assignments, research opportunities, and departmental initiatives.

3- Evaluation and Feedback: The department conducts periodic evaluations of faculty performance, providing constructive feedback to support their professional development. This evaluation process may include classroom observations, evaluation of research outputs, and discussions with the department chair and department chair.

4- Continuous Professional Development: Professional development ensures that visiting faculty members have continuous professional development opportunities. They are introduced to relevant workshops, training programs and conferences that match their areas of interest and expertise.

Professional development for teaching staff members

The Professional Development Department plays an important role in facilitating the academic and professional development of faculty members within the college or department. The following is a brief description of the plan and arrangements for faculty development in areas such as teaching and learning strategies, assessment of learning outcomes, and professional development:

Teaching and Learning Strategies:

1- The department organizes workshops, seminars, and training courses that focus on effective teaching and learning strategies.

2- The department encourages the use of technology in the classroom and provides support for integrating educational technologies into teaching practices.

Assessment of Learning Outcomes:

1- The department provides guidance and resources on designing assessments that align with learning outcomes and program objectives.

2- Faculty members receive training on various assessment methods.

Professional Development:

1- The department facilitates continuing professional development opportunities for faculty members.

2- The department organizes conferences, seminars, and guest lectures featuring lecturers in various academic disciplines.

3- Faculty members are encouraged to attend external professional development events.

Mentoring and Peer Collaboration:

1- The department promotes a culture of mentoring and collaboration among faculty members.

2- The department facilitates mentoring programs where experienced faculty members guide and support their colleagues, especially new faculty members.

12. Acceptance criteria.

For morning study:
Students are accepted through the central admission system and according to the instructions of the Iraqi Ministry of Higher Education and Scientific Research.

For evening study:
Students are accepted by applying to the college by submitting some required documents, and students are accepted based on the instructions and decisions of the Iraqi Ministry of Higher Education and Scientific Research.

13. The most important sources of information about the program

1- Methodological and support books
2- Lectures of the instructor
3- Internet for the purpose of obtaining research.

14. Program Development Plan:

Academic program development in colleges involves a systematic and collaborative approach. The following outlines an appropriate program development plan:

Needs Assessment:
1- Conduct a comprehensive needs assessment to identify gaps and opportunities in the academic landscape.
2- Analyze market trends, recruitment requirements, and industry requirements to align the program with current and future needs.

Faculty Development:
1- Provide faculty members with professional development opportunities, including workshops, conferences, and training programs.
2- Support faculty members in keeping up with developments in the field through research and scholarly activities.

Resource and Infrastructure Allocation:
1- Identify resource needs to implement the program effectively.
2- Allocate sufficient financial, technological, and human resources to support the program.
3- Ensure that the program has access to the necessary infrastructure, including classrooms, laboratories, equipment, and specialized facilities.

Cooperation and Partnerships:
1- Establish partnerships for internships, cooperative education programs, and research projects.
2- Leverage external expertise and resources to enhance the quality and relevance of the program.

Monitoring and Evaluation:
1- Develop a comprehensive evaluation plan to assess student learning outcomes and program effectiveness.
2- Use evaluation data to identify areas for improvement and make data-informed decisions to enhance the program.

Values				Skills				Knowledge				Course Name	Course Code	Year /Level
ج	ج	ج	ج	ب	ب	ب	ب	أ4	أ3	أ2	أ1			

4	3	2	1	4	3	2	1							
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Programming Basics	TU010101107	المرحلة الاولى الفصل الأول -2024 2025
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Principles of Computer Structure	TU010101109	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Discrete Structures 1	TU010101108	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Differential and Integral Calculus 1	TU010101110	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	English Language 1	TU010101101	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Democracy and Human Rights	TU010101102	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Advanced Programming	TU010101113	المرحلة الاولى الفصل الثاني -2024 2025
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Website Programming	TU010101115	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Algorithm Design	TU010101111	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Logic Design	TU010101112	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Computational Theory	TU010101114	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Arabic Language	UOT001	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Computer Architecture	TU010102002	المرحلة الثانية الفصل الأول -2024 2025
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Systems Analysis and Design	TU010102003	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Numerical Analysis	TU010102004	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Data Structures 1	TU010102005	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Computational	TU010102006	

												Theory 1		
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Entity Programming	TU010102007	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Baath Party Crimes in Iraq	TU010102008	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Research Methodology	TU010102009	المرحلة الثانية الفصل الثاني -2024 2025
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Probability and Random Variables	TU010102010	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Internet of Things	TU010102011	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Data Structures 2	TU010102012	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Microprocessors	TU010102013	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Visual Programming	TU010102014	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Encryption	TU010103001	المرحلة الثالثة الفصل الأول -2024 2025
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Compiler 1	TU010103002	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Operations Research	TU010103003	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Principles of Management	TU010103004	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Databases 1	TU010103005	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Software Engineering	TU010103006	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Artificial Intelligence	TU010103007	المرحلة الثالثة الفصل الثاني -2024 2025
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Databases 2	TU010103008	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Mobile Computing	TU010103009	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Operating Systems 2	TU010103010	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	English Language 2	TU010103011	

✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	compiler 2	TU010103012	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Digital Signal Processing	TU010103013	المرحلة الرابعة الفصل الأول -2024 2025
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Digital Image Processing	TU010104001	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Operating Systems 2	TU010104002	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Computer Security	TU010104003	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Computer Networks	TU010104004	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Modeling and Simulation	TU010104005	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Graduation Project	TU010104012	المرحلة الرابعة الفصل الثاني -2024 2025
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Distributed Systems	TU010104007	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Computer Networks 2	TU010104008	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Multimedia and Network Security	TU010104009	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Neural Networks	TU010104010	
✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	Data Mining	TU010104011	

*Please tick the boxes corresponding to the individual learning outcomes of the programmer being assessed.

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Advanced Programming**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code	TUCS				
ECTS Credits	8				
SWL (hr/sem)	**200**				
Module Level		1	**Semester of Delivery**		2nd
Administering Department		Computer Science	**College**	CCSM	
Module Leader	Mohanad Hatem Ramadhan		**e-mail**	Mohanad.H.Ramadhan@tu.edu.iq	
Module Leader's Acad. Title		Assistant Lecturer	**Module Leader's Qualification**		master
Module Tutor	Yahya Laith Khalil		**e-mail**		
Peer Reviewer Name		Mahammed Aktham	**e-mail**		
Scientific Committee Approval Date		07/06/2023	**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	Programming Fundamentals	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	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.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Demonstrate an in-depth understanding of arrays, strings, pointers, and file handling concepts in C++. 2. Apply advanced array operations, such as searching and sorting algorithms, and multidimensional arrays to solve programming problems. 3. Manipulate strings effectively, including concatenation, substring extraction, searching, and sorting. 4. Utilize pointers proficiently for data manipulation, including memory addresses, and pointer arithmetic 5. Read from and write to files, perform input/output operations, and handle file-related errors using file streams in C++. 6. Manage and organize files effectively, including creating, modifying, deleting, and organizing file directories. 7. Apply efficient programming techniques, optimize code, and adhere to best practices for writing clean and readable code.

	8. Demonstrate problem-solving skills by applying the learned concepts to solve real-world programming challenges. 9. Work collaboratively in teams, communicate effectively, and share knowledge and ideas related to advanced programming concepts. 10. Adapt to new programming concepts and technologies beyond the scope of the course, building a foundation for lifelong learning in programming. These learning outcomes reflect the knowledge, skills, and competencies that students will acquire upon completing the advanced programming course. The outcomes emphasize both theoretical understanding and practical application, preparing students for real-world programming challenges and further studies in the field of computer science.
Indicative Contents المحتويات الإرشادية	1. Review of Basic Programming Concepts: - Recap of fundamental programming concepts, including variables, data types, control structures, and functions. 2. Arrays: - Multidimensional arrays - Array manipulation techniques - Searching and sorting algorithms 3. Strings: - String manipulation and operations - String handling functions 4. Pointers: - Introduction to pointers and their usage - Memory addresses and pointer arithmetic - Pointers to arrays 5. Files: - File handling concepts - Reading from and writing to files - File organization and management

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain 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 the learning process. 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 that students have the necessary resources to complete their lab exercises and assignments effectively.

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 provide support to students who need additional help or guidance in understanding programming concepts or completing

	assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	60	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	140	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	9.4
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	200		

Module Evaluation

تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	
	Assignments	4	20% (20)	7, 12	
	Projects	2	20% (20)	5-14	
	Report				
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Recap of fundamental programming concepts, including variables, data types, control structures, and functions.
Week 2	Introduction to Arrays (Linear arrays)
Week 3	Searching and Sorting Linear Arrays
Week 4	Multidimensional Arrays and Square Arrays
Week 5	Multiplication of Two Arrays and Re-write TicTacToe game with Arrays
Week 6	Introduction to String and Its Operations
Week 7	More Examples on String
Week 8	Introduction to Pointers
Week 9	Pointer to Array and Pointer Arthmetic

Week 10	First Project Due (Reviewing and Comments)
Week 11	Introduction to Files and Directories
Week 12	Working with Text Files (Read, Write)
Week 13	Working with Binary Files
Week 14	Second Project Due (Students Presentations part1)
Week 15	Second Project Due (Students Presentations part1)

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	**Material Covered**
Week 1	**Getting used to CLI Interfaces and practicing some commands on PowerShell**
Week 2	**Running Examples on Array**
Week 3	**Practicing Arrays further (Searching)**
Week 4	Practicing Arrays further (Sorting)
Week 5	**Running Examples on 2D and Square Arrays**
Week 6	**Running Examples on Strings**
Week 7	**Searching in String**
Week 8	**Running Characters Frequency Example**
Week 9	**Running Examples on Pointers**
Week 10	**Running More Examples on Pointers**
Week 11	**Running Examples on Directories and Files**
Week 12	**Running More Examples on Files**
Week 13	**Running More Advanced Programs on Files**
Week 14	**Wrapping up**
Week 15	**Answering Students Questions and Extra Advising on Real World Application Programming**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**

Required Texts	Stroustrup, Bjarne - Programming_ principles and practice using C++-Addison-Wesley (2015)	Yes
Recommended Texts	Olsson, Mikael - C++20 Quick syntax reference: a pocket guide to the language, apis, and library	No
Websites		

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**algorithm design**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☒ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	TUCS110				
ECTS Credits	6				
SWL (hr/sem)	**200**				
Module Level	1		**Semester of Delivery**	2nd	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Mohamed Taheer Ahmed		**e-mail**	Mohanad.H.Ramadhan@tu.edu.iq	
Module Leader's Acad. Title	Assistant Lecturer		**Module Leader's Qualification**	master	
Module Tutor			**e-mail**		
Peer Reviewer Name	Mahammed Aktham		**e-mail**		
Scientific Committee Approval Date	07/06/2023		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	The aim of this module is to introduce students to the fundamental concepts of algorithms, algorithm design, and problem-solving techniques. The module will cover various algorithmic paradigms, data structures, and analysis methods to equip students with the skills necessary for designing and analyzing algorithms effectively.

Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Understand the importance of algorithms in computer science and the significance of algorithmic problem-solving. 2. Design algorithms using flowcharts and pseudocode, and implement them using programming constructs such as flow control statements and loops. 3. Analyze the time and space complexity of algorithms using Big O notation and asymptotic analysis. 4. Implement and utilize basic data structures such as arrays, strings, stacks, and queues for algorithmic problem-solving. 5. Apply various sorting and searching algorithms, including bubble sort, selection sort, insertion sort, quicksort, mergesort, heapsort, linear search, binary search, depth-first search, and breadth-first search. 6. Utilize string algorithms for pattern matching and string manipulation tasks. 7. Demonstrate the ability to review and evaluate projects related to algorithm design and implementation.
Indicative Contents المحتويات الإرشادية	1. Introduction to algorithms: Understanding the role and significance of algorithms in computer science. 2. Algorithmic problem-solving: Exploring strategies and techniques for solving computational problems effectively. 3. Algorithm design: Drawing flowcharts and writing pseudocode to represent algorithmic solutions. 4. Flow control: Implementing flow control statements (if-else, switch-case) for decision-making in algorithms. 5. Loops: Utilizing loops for repetitive tasks, including counter and cumulative variables, and nested loops. 6. Complexity analysis: Analyzing the time and space complexity of algorithms using Big O notation and asymptotic analysis. 7. Basic data structures: Introduction to arrays, strings, stacks, and queues for storing and manipulating data. 8. Sorting algorithms: Implementing and analyzing sorting algorithms such as bubble sort, selection sort, insertion sort, quicksort, mergesort, and heapsort. 9. Searching algorithms: Implementing and analyzing searching algorithms such as linear search, binary search, depth-first search, and breadth-first search. 10. String algorithms: Exploring algorithms for pattern matching and string manipulation tasks. 11. Reviewing students' projects: Providing feedback and evaluation on projects related to algorithm design and implementation.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	Lectures: Traditional lectures can be used to introduce key concepts, theories, and principles related to algorithms. Lectures should be interactive, incorporating examples, demonstrations, and real-world applications to

	illustrate abstract concepts effectively. Group Discussions: Group discussions encourage collaborative learning and critical thinking. Students can discuss challenging topics, share insights, and work together to solve algorithmic problems. Group discussions also promote communication skills and teamwork. Problem-Solving Sessions: Dedicated problem-solving sessions allow students to practice applying algorithmic techniques to solve a variety of problems. These sessions can involve solving algorithmic puzzles, coding challenges, and algorithm design exercises individually or in groups. Practical Coding Assignments: Assigning practical coding assignments allows students to implement algorithms and data structures in programming languages of their choice. Through coding assignments, students gain hands-on experience with algorithm implementation, debugging, and optimization. Case Studies: Case studies provide real-world examples of how algorithms are used to solve practical problems in various domains, such as finance, healthcare, and engineering. Analyzing case studies helps students understand the relevance and applicability of algorithms in different contexts.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ أسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	92	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	6.13
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	108	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	7.2
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	200		

Module Evaluation

تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	#LO 1-3, #LO 5-8
	Assignments	2	10% (10)	7, 12	#LO 3-5, #LO 5-8
	Projects	1	10% (10)	continuous	
	Report	1	10% (10)	14	#LO 1-8
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	#LO 1-7
	Final Exam	2 hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
Week No.	**Material Covered**
Week 1	Importance of algorithms in computer science
Week 2	Importance of algorithmic problem-solving
Week 3	Algorithms Design Drawing Flowchart and Writing pseudocode
Week 4	Flow Control (if-else), (switch – case)
Week 5	Loops (counter and cumulative variables), Nested Loops
Week 6	Time complexity analysis (Big O notation), Space complexity analysis and Asymptotic analysis
Week 7	Midterm exam
Week 8	Basic Data Structures: Arrays, Strings, Stacks, Queues.
Week 9	Sorting Algorithms: Bubble sort, selection sort, insertion sort
Week 10	Sorting Algorithms: Quicksort, mergesort, heapsort
Week 11	Searching Algorithms: Linear search, binary search
Week 12	Searching Algorithms: Depth-first search, breadth-first search
Week 13	String Algorithms: Pattern matching algorithms
Week 14	String Algorithms: String manipulation techniques
Week 15	Reviewing Students' Projects

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	Material Covered
Week 1	Introduction to Algorithm Design Overview of the course objectives and expectations Introduction to algorithm design methodologies Hands-on activity: Drawing flowcharts for simple algorithms Assignment: Practice drawing flowcharts for algorithmic problems
Week 2	Review of pseudocode and its importance in algorithm design

	Introduction to flow control statements (if-else, switch-case) Hands-on activity: Writing pseudocode for algorithmic problems Assignment: Implementing algorithms using flow control in a programming language
Week 3	Understanding loop structures and their importance in algorithms Hands-on activity: Implementing loops for counter and cumulative variables Introduction to nested loops Assignment: Solving algorithmic problems using nested loops
Week 4	Time Complexity Analysis Introduction to time complexity analysis using Big O notation Understanding the concept of asymptotic analysis Hands-on activity: Analyzing the time complexity of algorithms Assignment: Analyzing the time complexity of sorting algorithms
Week 5	Space Complexity Analysis Introduction to space complexity analysis Hands-on activity: Analyzing the space complexity of algorithms Assignment: Analyzing the space complexity of searching algorithms
Week 6	Basic Data Structures Introduction to arrays, strings, stacks, and queues Hands-on activity: Implementing basic data structures in a programming language Assignment: Implementing algorithms using basic data structures
Week 7	Sorting Algorithms Introduction to sorting algorithms: bubble sort, selection sort, insertion sort Hands-on activity: Implementing sorting algorithms Assignment: Comparing the performance of different sorting algorithms
Week 8	Sorting Algorithms (continued) Introduction to more advanced sorting algorithms: quicksort, mergesort, heapsort Hands-on activity: Implementing advanced sorting algorithms Assignment: Optimizing sorting algorithms for different datasets
Week 9	Searching Algorithms Introduction to searching algorithms: linear search, binary search Hands-on activity: Implementing searching algorithms Assignment: Analyzing the performance of searching algorithms

Week 10	Graph Algorithms Introduction to graph algorithms: depth-first search, breadth-first search Hands-on activity: Implementing graph traversal algorithms Assignment: Solving graph-related problems using depth-first search and breadth-first search
Week 11	String Algorithms Introduction to string matching algorithms Hands-on activity: Implementing pattern matching algorithms Assignment: Applying string manipulation techniques to solve algorithmic problems
Week 12	Review and Project Work
Week 13	Project Work and Consultation Project work: Students continue working on their projects Individual consultations with the instructor for project guidance and feedback
Week 14	Project Presentation Preparation Preparation for project presentations Practice sessions for project presentations Final touches on project implementations and documentation
Week 15	Project Presentations

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	Introduction to Algorithms, Third Edition By Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein	No
Recommended Texts	Introduction to Algorithmic Design and Analysis	No
Websites		

Grading Scheme
مخطط الدرجات

Group	Grade	التقدير	Marks (%)	Definition
Success Group	**A** - Excellent	امتياز	90 - 100	Outstanding Performance

(50 - 100)	**B** - Very Good	جيد جدا	80 - 89	Above average with some errors
	C - Good	جيد	70 - 79	Sound work with notable errors
	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings
	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required

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

Calculus I

تفاضل وتكامل 1

Module Information معلومات المادة الدراسية					
Module Title	**Calculus I**			**Module Delivery**	
Module Type	Supportive			☒ **Theory** ☒ **Lecture** ☐ **Lab** ☒ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code	TUCS112				
ECTS Credits	**6**				
SWL (hr/sem)	**150**				
Module Level		1	**Semester of Delivery**		1st
Administering Department		Computer sciences	**College**	CCSM	
Module Leader	Oqba Salim		**e-mail**	akabasalim4@gmail.com	
Module Leader's Acad. Title		Assistant Lecturer	**Module Leader's Qualification**		Master
Module Tutor			**e-mail**		
Peer Reviewer Name		Mohammed Aktham	**e-mail**		
Scientific Committee Approval Date		07/06/2023	**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	Calculus II, Advanced Calculus	**Semester**	3

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	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.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Deep understanding of concepts: Students are expected to gain a deep understanding of basic concepts in differential calculus. Students should be able to understand the concept of a derivative and its meaning as the rate of change of a function, as well as the concept of inverse differential and integration of functions. 2. Numerical and Application Skills: Students should acquire strong skills in calculating and using derivatives in solving applied calculus problems. They should be able to compute the derivative of a variety of functions and apply it in analyzing the behavior of functions, identifying critical points, and estimating absolute values and variable ratios. 3. Analytical Thinking: By studying differential calculus, students are expected to develop abilities in analytical thinking and mathematical reasoning. They should be able to analyze mathematical problems and draw conclusions based on learned mathematical concepts and tools. 4. Applications in Other Fields: Students should have the ability to apply the concepts of differential calculus in other fields such as science, engineering, and economics. They learn how to represent real phenomena by functions and use differential calculus to analyze these phenomena and derive practical results. 5. Use of Technology: Students should have the ability to use appropriate technology such as mathematical calculation programs and
Indicative Contents المحتويات الإرشادية	Indicative content includes the following: 1. Introduction to calculus: includes the definition of a derivative and the concept of a derivative as the instantaneous rate of change of a function. Students are exposed to the basic rules of calculus and related concepts. 2. The basic rules of differentiation: It includes studying the basic rules of differentiation such as the rule of differentiation rules, the rule of differentiation of constants, the rule of differentiation of forces, and other rules of differentiation of known functions. 3. Higher Derivatives: Students learn how to calculate higher derivatives, how to work with recursive differentiation, and how to use differential rules related to it. 4. Applications in differential calculus: Students explore the practical applications of calculus in different fields such as physics, engineering, and computer science. Practical examples of solving various differential problems are presented. 5. Relative and Total Differential: Students learn the concept of relative

	differential and total differentiation and how to calculate them. They are exposed to its applications in analyzing the behavior of functions and estimating variable changes. 6. Practical Applications of Calculus: Students are introduced to the use of differential calculus in solving problems in mathematical modeling, economic analysis, statistics, and other fields.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	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. 4. 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

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	77	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	5.13
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	73	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	4.86
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	150		

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

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	Introduction to function, domain, range Invers functions, even and odd function
Week 2	. Graph the functions
Week 3	Limits and continuous
Week 4	Exponential Functions , Logarithm Functions, Trigonometric functions.
Week 5	Derivatives, Rules of differentiation, Applications of Derivatives.
Week 6	The mean value theorem
Week 7	**Mid-term exam**
Week 8	The derivative and extrema
Week 9	Derivatives of Exponential Functions , Logarithm Functions
Week 10	Derivatives of Trigonometric functions, Derivatives of inverse functions
Week 11	Integration
Week 12	The mean value theorem for integrals
Week 13	basic application of integration
Week 14	Area , volume
Week 15	Arc length
Week 16	**Preparatory week before the final exam.**

Delivery Plan (Weekly Lab. Syllabus): There is no Lab activities المنهاج الاسبوعي للمختبر:لا توجد فعاليات مختبرية	
	Material Covered

Week 1	
Week 2	
Week 3	
Week 4	
Week 5	
Week 6	
Week 7	

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	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).	No
Recommended Texts	Grossman, Stanley I. *Calculus*. Academic Press, 2014.	No
Websites	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	

Grading Scheme
مخطط الدرجات

Group	Grade	التقدير	Marks (%)	Definition
Success Group (50 - 100)	**A** - Excellent	امتياز	90 - 100	Outstanding Performance
	B - Very Good	جيد جدا	80 - 89	Above average with some errors

	C - Good	جيد	70 - 79	Sound work with notable errors
	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings
	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Computation theory**			**Module Delivery**	
Module Type	Supportive			☒ **Theory** ☒ **Lecture** ☐ **Lab** ☒ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	TUCS112				
ECTS Credits	**6**				
SWL (hr/sem)	**150**				
Module Level	1		**Semester of Delivery**	2	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Marwa Adeeb Mohammud		**e-mail**	Marwa.a.aljawaherry@tu.edu.iq	
Module Leader's Acad. Title	Assistant Professor		**Module Leader's Qualification**	master	
Module Tutor			**e-mail**		
Peer Reviewer Name			**e-mail**		
Scientific Committee Approval Date	01/01/2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	Compiler	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	Theory of computation is the theoretical study of capabilities and limitations of Computers (Theoretical models of computation). Providing students with: 1. Understand of basic concepts in the theory of computation through simple models of computational devices. 2. Apply models in practice to solving problems in diverse areas such as

	string searching, pattern matching, cryptography, and language design. 3. Understand the limitations of computing, the relative power of formal languages and the inherent complexity of many computational problems. 4. Be familiar with standard tools and notation for formal reasoning about machines and programs.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Upon successful completion of this module, students should be able to: Cognitive objectives: 1. Enabling the student to know and understand the theoretical principles of the programming process. 2. Enabling the student to know the intellectual framework of computer basics. 3. Enabling the student to know the practical applications of the programming foundation that the subject has. 4. Enabling the student to explain the steps of program implementation. kills objectives for the course: 1. Designing basic diagrams for computational theoretical methods. 2. Writing algorithms. 3. Writing and implementing methods. 4. Following up on the implementation and planning of algorithms and discovering errors. Teaching and learning methods, theoretical lectures, using projectors for clarification with illustrative examples, drawing diagrams that facilitate the process of understanding and conveying the idea to the student.
Indicative Contents المحتويات الإرشادية	Focuses interest on studying how calculations and reasoning are performed in the context of computer systems. 1. Computing Paradigms: Paradigms such as the Turing machine, quantum computing are explored and used to analyze computational capabilities and limits. 2. Description Languages: Computational theory studies the languages and models that are used to describe and analyze computational operations, such as neural language and conditional language. 3. Computational complexity: It is concerned with analyzing computer complexity and classifying problems according to their computational difficulty and the ability of computer systems to solve them. 4. Mental Machines: Computational theory studies the relationship between computing and the human mind, and whether the mind can be

	represented and simulated by computational machines. 5. Artificial Intelligence: Artificial intelligence is an important application of computational theory, where computational models are used to develop artificial intelligence systems capable of thinking and making decisions.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain 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 the learning process. Discussions can include reviewing code examples, discussing programming best practices, and exploring real-world applications of programming concepts. 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++. 6. Office Hours and Individual Support: The instructor should be available for individual consultations and provide support to students who need additional help or guidance in understanding programming concepts or completing assignments.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ أسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	77	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	5.1
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	73	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	4.8

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

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	LO #1-4 , LO# 5-7
	Assignments	1	10% (10)	7, 12	LO #1-4 , LO# 5-7
	Projects	1	10% (10)	continuous	LO #1-7
	Report	1	5% (10)	14	LO # 4 - 7
Summative assessment	**Midterm Exam**	2 hr	15% (10)	11	LO # 1-6
	Final Exam	2 hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Set, string, alphabet and language
Week 2	Regular expression
Week 3	Finite state automata, deterministic and nondeterministic finite state automata.
Week 4	Equivalence between deterministic and nondeterministic finite state automata
Week 5	Finite state automata with empty move.
Week 6	Equivalence of NFA with and without ϵ move
Week 7	The equivalence between Moore and Mealy machine
Week 8	Grammar, Chomsky hierarchy of languages.
Week 9	- The regular grammars and regular languages.
Week 10	- Closure properties of regular sets (union, concatenation and kleen closure).
Week 11	Regular expression, closure properties of regular languages (intersection, complementation and substitution)
Week 12	Decision procedures for regular sets(emptiness, finiteness, containment and equivalence).
Week 13	Context- free grammars and languages with their Properties.
Week 14	Context-free grammar without empty string production (λ-free grammar).
Week 15	Derivation trees.

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Introduction to Computer Theory 3 rd Edition michael sipser, USA, 2013. ISBN-13: 978-1-133-18779-0.	Yes
Recommended Texts	Introduction to Automata Theory, Languages, and Computation, 2/E, John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman, Addison-Wesley 2001. ISBN 0-201-44124-1.	No
Websites		

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Principles of Computer Organization**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☒ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	TUCS111				
ECTS Credits	**6**				
SWL (hr/sem)	**150**				
Module Level	1		**Semester of Delivery**	1st	
Administering Department	Computer science		**College**	CCSM	
Module Leader	Nayif Mohamed		**e-mail**		
Module Leader's Acad. Title	**Asst. Prof.**		**Module Leader's Qualification**	Master	
Module Tutor	Ammar Farooq Abbas		**e-mail**	ammar.abbas@tu.edu.iq	
Peer Reviewer Name	Mohamed Aktham		**e-mail**		
Scientific Committee Approval Date	07/06/2023		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	1. Understanding the Basic Components: The module aims to familiarize students with the fundamental components of computer systems, including the CPU (Central Processing Unit), memory, input/output devices, and storage devices. Students should gain an understanding of the purpose and functionality of each component. 2. Learning about Instruction Set Architecture: Students will learn about

	different instruction set architectures, such as the von Neumann architecture, Harvard architecture, and reduced instruction set computing (RISC) architecture. The module aims to provide an understanding of the design principles behind instruction sets and their impact on computer performance. 3. Understanding Processor Design: The module aims to delve into the design and operation of processors, including topics such as control unit design, arithmetic logic units (ALUs), pipelining, and parallel processing. Students should gain an understanding of how processors execute instructions and the factors affecting their performance. 4. Studying Memory Systems: This module aims to cover different types of memory systems, including cache memory, main memory (RAM), and secondary storage (hard drives, solid-state drives). Students should understand the organization, hierarchy, and access methods of these memory systems, as well as their impact on system performance. 5. Exploring Input/Output Systems: The module aims to provide an understanding of input/output (I/O) systems and their interface with the CPU and memory. Topics covered may include I/O devices, interrupt handling, I/O techniques (e.g., programmed I/O, interrupt-driven I/O, DMA), and I/O performance considerations. 6. Discussing Parallel Processing and Multiprocessing: Students may learn about the concepts of parallel processing and multiprocessing, including techniques such as parallel algorithms, multi-core processors, and parallel architectures. The module aims to explore the benefits and challenges of parallel computing. 7. Analyzing Performance Evaluation: The module may introduce techniques for evaluating the performance of computer systems, including benchmarking, measurement metrics, and performance analysis. Students should gain the ability to assess and compare the performance of different computer systems. 8. Understanding Operating Systems: The module aims to provide students with an understanding of operating systems, including their purpose, functionalities, and components. Topics covered may include process management, memory management, file systems, device management, and scheduling algorithms. Students should gain knowledge of the key concepts and mechanisms used in operating systems. 9. Discussing System Maintenance and Administration: The module may cover system maintenance and administration principles, including software updates, patch management, system monitoring, troubleshooting, and system backup and recovery. Students should gain knowledge of the practices and tools used in maintaining and administering system software. 10. Understanding Software Functionality: The module aims to familiarize students with the functionality and features of application software, specifically focusing on tools and applications within productivity suites like Microsoft Office. This includes word processing, spreadsheet analysis, presentation creation, and database management.
Module Learning Outcomes	1. Understand the basic components of computer systems, including the CPU, memory, input/output devices, and storage devices. 2. Demonstrate knowledge of different system architectures and their impact on

مخرجات التعلم للمادة الدراسية	computer performance. 3. Understand instruction set architectures and their design principles. 4. Explain the design and operation of processors, including control unit design, ALUs, pipelining, and parallel processing. 5. Understand memory systems, including cache memory, main memory, and secondary storage, and their impact on system performance. 6. Describe input/output systems and their interface with the CPU and memory, including I/O devices, interrupt handling, and I/O techniques. 7. Discuss parallel processing and multiprocessing concepts, including parallel algorithms and architectures. 8. Analyze the performance of computer systems using benchmarking and performance evaluation techniques. 9. Understand the purpose, functionalities, and components of operating systems. 10. Demonstrate knowledge of system programming, including system calls, libraries, drivers, and low-level programming. 11. Understand virtualization and emulation principles and their applications. 12. Demonstrate knowledge of system security concepts and practices. 13. Understand system maintenance and administration principles, including software updates, system monitoring, and troubleshooting. 14. Demonstrate proficiency in using application software tools and features within the productivity suite (e.g., Microsoft Word, Excel, PowerPoint): a. Create and edit documents, spreadsheets, presentations, and databases effectively and efficiently. b. Utilize formatting options, templates, and styles to enhance the visual appeal and professionalism of documents. c. Use formulas, functions, and data analysis tools to manipulate and analyze data in spreadsheets. d. Design visually appealing and engaging presentations using multimedia elements and slide transition effects. 15. Apply best practices for efficient software usage: a. Navigate the user interface and menu options to locate and utilize software features effectively. b. Utilize keyboard shortcuts and automation features to enhance productivity and streamline tasks. c. Optimize workflow and utilize time-saving techniques within the software. 16. Communicate effectively through the creation of professional-looking documents and presentations: a. Create clear, concise, and well-organized documents with appropriate formatting, headings, and sections. b. Design visually appealing slides with effective use of text, images, charts, and other multimedia elements. c. Utilize collaboration and sharing features to facilitate teamwork and effective communication. 17. Analyze and visualize data using spreadsheet software: a. Organize and manipulate data effectively using tables, filters, and sorting features. b. Utilize formulas, functions, and data analysis tools to perform

	calculations and derive insights from data. c. Create informative charts, graphs, and pivot tables to visualize data and present findings effectively. 18. Demonstrate problem-solving skills and troubleshoot common software issues: d. Identify and resolve common software-related challenges and errors. a. Seek out appropriate resources and support channels to resolve problems independently. b. Apply critical thinking and problem-solving strategies when encountering software-related issues. 19. Manage information effectively and maintain data integrity and security: a. Organize and store files and data in a structured and easily retrievable manner. b. Manage document versions, track changes, and utilize collaboration features to ensure data integrity. c. Apply security measures to protect sensitive information and maintain confidentiality. 20. Demonstrate an ability and willingness to learn and adapt to new software features and advancements: a. Stay updated with software updates and new features within the application software. b. Explore online resources, tutorials, and communities to expand knowledge and skills. c. Display a mindset of continuous learning and adaptability in using application software.
Indicative Contents المحتويات الإرشادية	Indicative content includes the following: 1. Basic Components of Computer Systems: [6 hours] • CPU (Central Processing Unit) • Memory (Primary and Secondary) • Input/Output Devices • Storage Devices 2. System Architecture: [6 hours] • Instruction Sets • Data Representation • Memory Hierarchy • Caches and Cache Organization 3. Processor Design: [6 hours] • Control Unit Design • Arithmetic Logic Units (ALUs) • Pipelining and its Challenges • Parallel Processing and Multi-core Processors 4. Memory Systems: [6 hours] • Cache Memory • Main Memory (RAM)

- Secondary Storage (Hard Drives, Solid-State Drives)
- Virtual Memory

5. Input/Output Systems: [6 hours]
    - I/O Devices
    - Interrupt Handling
    - Programmed I/O, Interrupt-Driven I/O, DMA
    - I/O Performance Considerations
6. Parallel Processing and Multiprocessing: [1.5 hours]
    - Parallel Algorithms
    - Multi-core Processors
    - Parallel Architectures
    - Performance Considerations and Challenges
7. Performance Evaluation: [1.5 hours]
    - Benchmarking
    - Measurement Metrics
    - Performance Analysis Techniques
    - Comparing and Assessing Computer System Performance

Indicative Contents for System Software:

1. Operating Systems: [6 hours]
    - Purpose and Functions of Operating Systems
    - Process Management
    - Memory Management
    - File Systems
    - Device Management
    - Scheduling Algorithms
2. System Maintenance and Administration: [3 hours]
    - Software Updates and Patch Management
    - System Monitoring and Troubleshooting
    - System Backup and Recovery
    - Performance Tuning
3. System Maintenance and Administration: [3 hours]
    - Software Updates and Patch Management
    - System Monitoring and Troubleshooting
    - System Backup and Recovery
    - Performance Tuning

Indicative Contents for Studying Application Software (e.g., Microsoft Office):

1. Introduction to Productivity Suites: [3 hours]
- Overview of productivity suites and their significance in various professional

	contexts. - Introduction to the features and components of popular productivity suites like Microsoft Office. 2. Word Processing Software (e.g., Microsoft Word): [3 hours] - Creating, editing, and formatting documents. - Working with text, paragraphs, and styles. - Inserting and formatting images, tables, and other graphical elements. - Utilizing document templates and mail merge functionality. - Collaboration and reviewing features. 3. Spreadsheet Software (e.g., Microsoft Excel): [3 hours] - Creating, formatting, and managing spreadsheets. - Working with formulas, functions, and calculations. - Data manipulation and analysis techniques. - Creating charts, graphs, and pivot tables. - Spreadsheet automation and advanced features. 4. Presentation Software (e.g., Microsoft PowerPoint): [3 hours] - Creating and formatting slides. - Inserting and formatting text, images, shapes, and multimedia elements. - Utilizing slide layouts, transitions, and animations. - Designing effective presentations for different audiences and purposes. - Collaboration and presentation delivery techniques. 5. Document Security and Protection: [3 hours] - Applying document security measures, password protection, and encryption. - Managing document permissions and access controls. - Ensuring data integrity and confidentiality within productivity suites.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Active Learning: • Actively engage with the material by reading, taking notes, and asking questions. • Participate in discussions, group activities, and practical exercises. • Seek opportunities for hands-on practice and application of concepts through exercises, projects, and real-world examples. 2. Practical Application: • Apply the learned concepts in practical scenarios, such as building and analyzing computer systems, designing software solutions, or creating documents and presentations using application software. • Seek out real-world examples and case studies to understand how the concepts are applied in practice.

- Undertake practical projects or assignments to gain hands-on experience.

3. Experimentation and Exploration:
    - Explore and experiment with different computer architectures, operating systems, and application software to gain a deeper understanding.
    - Set up virtual environments or use online simulators to experiment with various software configurations and settings.
    - Explore additional features and functionalities beyond the basics, going beyond the prescribed curriculum to expand knowledge.
4. Collaborative Learning:
    - Engage in group discussions, study groups, or online forums to share knowledge, exchange ideas, and discuss complex topics.
    - Collaborate with peers on projects or assignments to enhance learning through teamwork and shared problem-solving.
5. Continuous Practice and Review:
    - Regularly practice the learned concepts through exercises, assignments, or self-assessment quizzes.
    - Review and reinforce knowledge through periodic revision of key topics.
    - Seek feedback from instructors or mentors to identify areas for improvement and further learning.
6. Utilize Resources:
    - Make use of textbooks, lecture notes, online tutorials, and supplementary resources to enhance understanding and clarify concepts.
    - Explore online resources, forums, and communities related to the subject to access additional learning materials, discussion threads, and expert guidance.
7. Stay Updated:
    - Keep up-to-date with the latest advancements, trends, and updates in the field of computer organization, system software, and application software.
    - Follow relevant industry news, blogs, and publications to stay informed about emerging technologies and best practices.
8. Time Management and Regular Study:
    - Allocate dedicated study time for each subject to ensure consistent progress and understanding.
    - Break down the learning material into manageable chunks and create a study schedule.
    - Avoid procrastination and maintain discipline in adhering to the study

	plan. 9. Seek Support: • Seek guidance and support from instructors, teaching assistants, or mentors when encountering difficulties or complex concepts. • Participate in office hours or seek clarification during lectures or tutorials. 10. Reflect and Apply Learning: • Regularly reflect on the learning process, strengths, and areas for improvement. • Apply the acquired knowledge in practical situations to reinforce understanding and develop practical skills. • Connect the learned concepts to real-world applications and scenarios

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	92	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	6.13
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	58	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	3.86
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	150		

Module Evaluation

تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 10	LO #1-5 , LO# 10-14
	Assignments	2	10% (10)	2, 12	LO #10-14, LO#14-16
	Seminar	1	10% (10)	11	LO # 5-11
	Report	1	10% (10)	12	LO # 5-11
Summative assessment	**Midterm Exam**	2 hr	10% (10)	7	LO # 1-17
	Final Exam	2hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

	Material Covered
Week 1	Introduction: Computer Overview
Week 2	Computer Functions
Week 3	Introduction of Computer System Components
Week 4	Computer System Components: Processor Components
Week 5	Computer System Components: Internal Memory (RAM and ROM)
Week 6	Computer System Components: Storage Devices (secondary storage)
Week 7	Internal Hardware Computer Components: Power supply, Computer case, Internal cables
Week 8	Introduction of External Hardware Computer Components - Input Devices
Week 9	External Hardware Computer Components - Output Devices
Week 10	Introduction of System Software
Week 11	Computer Software - Operating Systems
Week 12	Computer Software - Utility Programs
Week 13	Application Software – M.S. Word
Week 14	Application Software – M.S. Excel
Week 15	Application Software – M.S. PowerPoint
Week 16	Preparatory week before the final exam.

Delivery Plan (Weekly Lab. Syllabus):

المنهاج الاسبوعي للمختبر:

	Material Covered
Week 1	Lab1_ Introduction of Computer Organization to understand the internal workings of the CPU and memory.
Week 2	Lab2_ Computer architecture simulations: Use simulation tools to explore different computer architectures, such as the von Neumann architecture or pipelined processors.
Week 3	Lab3_ Memory hierarchy experiments: Measure and analyze the performance of different levels of cache memory and understand their impact on system performance.
Week 4	Lab4_ System Software Lab: Operating system installation and configuration: Install and configure different operating systems (e.g., Windows) on virtual machines or physical hardware.
Week 5	Lab5_ Document processing and formatting using M.S. Word: Create and format documents using word processing software, applying different styles, headers, footers, and page layouts.
Week 6	Lab6_ Spreadsheet analysis and data manipulation using M.S. Excel: Perform data analysis tasks, such as sorting, filtering, and creating formulas and functions in spreadsheet software.
Week 7	Lab7_ Presentation design and delivery using M.S. PowerPoint: Create visually appealing presentations with multimedia elements and practice delivering effective presentations.

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	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. 4. "Operating System Concepts" by Abraham Silberschatz, Peter B. Galvin, and Greg Gagne.	Yes
Recommended Texts	1. "Structured Computer Organization" by Andrew S. Tanenbaum and Todd Austin. 2. "Operating Systems: Internals and Design Principles" by William Stallings. 3. "Exploring Microsoft Office 2019" by Mary Anne Poatsy, Keith Mulbery, Cynthia Krebs, and Lynn Hogan.	No
Websites	https://ccms.tu.edu.iq/csd/electronic-lectures/409-stage1-8.html	

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Discrete Structure 1			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☐ **Lab** ☒ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code	UoT11103				
ECTS Credits	**6**				
SWL (hr/sem)	**150**				
Module Level	1		**Semester of Delivery**	1	
Administering Department	Computer		**College**	CCSM	
Module Leader	Salwa Khalid Abdulateef		**e-mail**	Khalid.salwa@tu.edu.iq	
Module Leader's Acad. Title	Assistant Professor		**Module Leader's Qualification**	MSc.	
Module Tutor	None		**e-mail**		
Peer Reviewer Name	None		**e-mail**		
Scientific Committee Approval Date	20/01/2025		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	1- Discrete structures provides the mathematical foundations for many courses including data structures and algorithm, Compilers, Automata theory and formal languages, operating systems theory and many other subjects. 2- Discrete structures are the abstract mathematical structures used to represent discrete object and relations lies between those objects. 3- Students' realization of the basic concepts of discrete structure, such as

	mathematics logic, graphs. 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 Giving students the necessary experience to solve some of the functions as linear and factorials.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	**1** 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. **2** Thinking skills :Giving the learner the skill to use logical hypotheses in building accurate software. **3** Giving the learner the skills to have the ability to build relationships between components, models and theoretical structures with algorithms and computer programs. **4** Enabling students to continue self-development after graduation. **5** Making the learner well acquainted with all types of logical deductive proof and types of proof by other methods. **6** Build basic causal skills in creating and validating algorithms and programs. - **7** Building skills to analyze and solve some important issues and the approximate time to solve them. - **8** Build skills on how to choose the appropriate solutions for some issues and designate the best algorithms to solve them
Indicative Contents المحتويات الإرشادية	Indicative content includes the following: <u>Part A – Mathematics Logic: Foundations of Logic:</u> Overview Proposition, compound proposition, How to Built a Truth Table, Logical operators, Bit operators and Translation English Sentences into Propositional Logic and Vice Versa with some examples. [16 hours] <u>Part B- Set theory</u> Definition of sets, Sets of numbers such as standard numerical , power set, cardinality, cartesian products of set, Venn Diagram, set operations, Algebra of sets with some examples [16 hours]

	Part C- Relations Definition of Relation, Graphical Representation of Relation, Properties of relations such as reflexive, symmetric and transitive with examples. [12 hours] Part D- Functions Definition of Function and examples, Types of function as one to one, onto, representation of function , application of functions with exampes. [16 hours)

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	• 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. 4- Quarterly exams for the academic curriculum in addition to the mid-year exam and the final exam.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	63	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	87	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	5.8
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	150		

Module Evaluation تقييم المادة الدراسية				
	Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**

Formative assessment	**Quizzes**	2	15% (15)	4, 10,12	LO #1-5 , LO# 9 - 12
	Assignments	2	10% (10)	2, 12	LO # 3, 4, LO#8 -10
	Projects / Lab.				
	Report	1	15% (15)	12	LO # 8, 10 and 12
Summative assessment	**Midterm Exam**	2 hr	10% (10)	7	LO # 1-8
	Final Exam	2hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	Mathematical logic
Week 2	Logical operators
Week 3	Logical operators
Week 4	Logical Equivalences, Compound Propositions Classification:
Week 5	Examples, Set of theory, Properties of set
Week 6	Sets of Number, Sets and elements, subsets
Week 7	**Mid Exam**
Week 8	Set's Algebra,
Week 9	Set's Algebra with examples
Week 10	relations
Week 11	Properties of relations
Week 12	Properties of relations with examples
Week 13	Review of Functions
Week 14	Types of Functions
Week 15	mathematics functions
Week 16	**Preparatory week before the final exam.**

Delivery Plan (Weekly Lab. Syllabus): There is no Lab activities المنهاج الاسبوعي للمختبر:لا توجد فعاليات مختبرية	
	Material Covered
Week 1	
Week 2	
Week 3	

Week 4	
Week 5	
Week 6	
Week 7	

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	Theory and problems of Discrete mathematics, by Seymour Lipschutz & Marc Lars Lipson, Schaum's Outline Series, third edition 2007 Discrete Mathematics and Its Applications, Seventh Edition, Kenneth H. Rosen, AT&T Laboratories, 2012	Yes
Recommended Texts	• 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	No
Websites	• http://en.wikibooks.org/wiki/Discrete_mathematics/Set_theory	

Grading Scheme
مخطط الدرجات

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**English Language**			**Module Delivery**	
Module Type	Elective			☒ **Theory** ☒ **Lecture** ☐ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code	TUCS115				
ECTS Credits	2				
SWL (hr/sem)	**50**				
Module Level	1	**Semester of Delivery**	1		
Administering Department	Computer	**College**	CCSM		
Module Leader	Ahmed fayeq saber	**e-mail**	Ahmed.f.saber@tu.edu.iq		
Module Leader's Acad. Title	Assistant Lecturer	**Module Leader's Qualification**	M.Sc.		
Module Tutor	None	**e-mail**	------		
Peer Reviewer Name	Ayham Mahmood	**e-mail**			
Scientific Committee Approval Date	7/06/2023	**Version Number**	1.0		

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	0
Co-requisites module	None	**Semester**	0

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	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.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Demonstrate through face-to-face conversations comprehension of simple words and phrases used in common everyday context. 2.. Knowing the basic rules of the English language. 3- Familiarity with the four English language skills: listening, reading, speaking and writing. 4. Determine the negative effects of the mother tongue on the English language. 5. Interpreting the texts in different contexts. 6- Generate simple sentences containing learned vocabulary and using appropriate grammatical structures 7- Express awareness of social and environmental issues. 8- Acquire varied range of vocabulary; understand increased complexity of sentence structures both in reading and writing. 9- 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. Enable students to communicate in English more confidently and effectively in their work or study environment.
Indicative Contents المحتويات الإرشادية	**Indicative content** - Teaching students how to communicate each other in English by using the four skills: speaking, listening, reading and writing. The use of different examples from the everyday life, dialogues, conversations and the weekly compositions are useful for more development. [50 hours]

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	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.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	32	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	18	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	1.2
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	50		

Module Evaluation

تقييم المادة الدراسية

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

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

	Material Covered
Week 1	**An introduction to English language.**
Week 2	**Unit (1) Hello am/ are/ is, my/your · This is … • How are you? • Good morning! What's this in English? • Numbers • Plurals**
Week 3	**Unit (2) Your World Countries • he/she/ they, his/her • Where's he from? fantastic/ awful/ beautiful · Numbers 11-30**

Week 4	**Unit (3) All about You Jobs • am/are/ is • Negatives and questions • Personal information • Social expressions**
Week 5	**Unit (4) Family and Friends our/their • Possessive 's • The family • has/have • The alphabet.**
Week 6	**Unit (5) The Way I live Sports/ Food/ Drinks • Present Simple - I/you/ we/ they • a/an Languages and nationalities • Numbers and prices.**
Week 7	**Unit (6) Every day the time • Present Simple-he/she · always/sometimes/never Words that go together • Days of week.**
Week 8	**Unit (7) My favorites Question words • me/him/us/them • this/that Adjectives • Can I.?**
Week 9	**Unit (8) Where I live Rooms and furniture • There is/ are • Prepositions • Directions**
Week 10	**Unit (9) Times past Saying years • was/were born • Past Simple - irregular verbs • have/do/go • When's your birthday?**
Week 11	**Unit (10) We had a great time! Past Simple - regular and irregular • Questions and negatives • Sport and leisure • Going sightseeing.**
Week 12	**Unit (11) I can do that! can/can't · Adverbs • Adjective + noun • Everyday problems.**
Week 13	**Unit (12) Please and thank you I'd like - some/any • In a restaurant • Signs all around.**
Week 14	**Unit (13) Here and now Colors and clothes • Present Continuous • Opposite verbs • What's the matter?**
Week 15	**Unit (14) It's time to go! Future plans • Grammar revision Vocabulary revision • Social expressions.**
Week 16	**Preparatory week before the final exam.**

Delivery Plan (Weekly Lab. Syllabus): There is no Lab activities المنهاج الاسبوعي للمختبر: لا توجد فعاليات مختبرية	
	Material Covered
Week 1	
Week 2	
Week 3	
Week 4	
Week 5	
Week 6	
Week 7	

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	1- Oxford Headway plus for Beginners. 2- New Headway English Course (2002) by Julia Starr Keddle.	Yes
Recommended Texts	English for Everyone	No
Websites	https://www.unionlearn.org.uk/english-and-maths-learning-resources-and-tools	

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Logic Design**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code					
ECTS Credits					
SWL (hr/sem)	**148**				
Module Level	1		**Semester of Delivery**	2nd	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Ahmed Saadi Abdullah		**e-mail**	ahmedalbasha@tu.edu.iq	
Module Leader's Acad. Title	**Asst. prof.Dr**		**Module Leader's Qualification**		
Module Tutor			**e-mail**		
Peer Reviewer Name	Mohamed Aktham		**e-mail**		
Scientific Committee Approval Date	22/01/2025		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	Logic design	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	• **Understand the fundamentals of digital logic design**: Provide foundational knowledge on digital systems and how to design logic circuits using basic logic tools. • **Develop logical and analytical thinking skills**: Enable students to analyze complex problems and choose appropriate methods for designing and implementing digital systems. • **Use Boolean algebra in circuit design**: Train students to use Boolean algebra to simplify circuits and develop efficient solutions. • **Explore advanced design techniques**: Study advanced methods for designing logic circuits, including combinational circuits, sequential circuits,

	and memory-based systems
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	• **Analyze and design basic logic circuits**: Using gates like AND, OR, NOT, XOR. • **Apply Boolean algebra**: Simplifying and designing digital circuits using Boolean algebra. • **Convert between different number systems**: Such as binary, decimal, and hexadecimal, and understand their relevance in logic design. • **Design and implement arithmetic circuits**: Like adders and subtractors using digital components. • **Work with memory systems**: Understand and apply flip-flops (e.g., SR, D, JK) and registers in digital designs. • **Simplify and design complex logic circuits**: Using Karnaugh maps (K-map) or Boolean algebraic methods. • **Simulate and design digital systems using specialized software tools**: Such as VHDL or Verilog for digital design simulation.
Indicative Contents المحتويات الإرشادية	1. **Introduction to Logic Design**: o Digital systems: Basic concepts and types. o Number systems representation. o Binary representation and conversion between number systems. 2. **Boolean Algebra**: o Definition of logical values (true/false). o Basic operations: AND, OR, NOT. o Boolean laws and simplification techniques. 3. **Logic Gates**: o Design and usage of AND, OR, NOT, XOR gates. o Combinational logic circuits using these gates. 4. **Simplification of Logic Circuits**: o Simplifying circuits using Boolean algebra. o Karnaugh maps (K-map) for circuit simplification. o Algebraic methods of simplification. 5. **Arithmetic Circuits**: o Half Adder and Full Adder circuits. o Subtractor circuits. o Multipliers and Dividers. 6. **Memory and Storage**: o Flip-flops (SR, D, JK): Types and applications. o Registers and shift registers. o Sequential circuits and their design. 7. **Design of Integrated Systems**: o Designing complex systems using logical components. o Techniques for optimizing circuits and achieving efficiency. 8. **Digital System Design and Simulation Tools**: o Introduction to software tools for designing digital systems. o Simulation and design of logic circuits using tools like VHDL or Verilog. 9. **Review and Case Studies**: o Case studies of real-world logic circuit applications like control units, embedded systems.

Learning and Teaching Strategies
استراتيجيات التعلم والتعليم

Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain logic design concepts, 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 the learning process. 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 how can building logic circutes. b. Guided Practice: Lab instructors or teaching assistants will be available to provide guidance, assistance, and immediate feedback on how can using the simulation software. 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. 4. Office Hours and Individual Support: The instructor should be available for individual consultations and provide support to students who need additional help or guidance in understanding programming concepts or completing assignments.

Student Workload (SWL)
الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	56	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3.7
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	148		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	2	10% (10)	5, 11	
	Assignments	4	20% (20)	7, 12	
	Projects	1	20% (20)	5-14	
	Report	1			
Summative assessment	Midterm Exam	2 hr	10% (10)	11	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Introduction to Digital Logic Design
Week 2	Logic Gates
Week 3	Boolean Algebra Basics
Week 4	Simplification Using Boolean Algebra
Week 5	Canonical Forms.
Week 6	Karnaugh Maps (K-map) for Simplification
Week 7	**First exam.**
Week 8	Combinational Logic Circuits
Week 9	Multiplexers,
Week 10	Demultiplexers, and Decoders
Week 11	Sequential Logic Circuits
Week 12	Flip-flops: SR, D
Week 13	Flip-flops: JK, and T
Week 14	Registers and Counters
Week 15	Design and simulate flip-flop circuits

Delivery Plan (Weekly Lab. Syllabus):
المنهاج الاسبوعي للمختبر:

Week No.	Material Covered
Week 1	**Getting to Know the Crocodile Interface**

Week 2	Building Circuits with Basic Logic Gates[1]
Week 3	Building Circuits with Basic Logic Gates[2]
Week 4	Designing Circuits with Advanced Logic Gates[1]
Week 5	Designing Circuits with Advanced Logic Gates[2]
Week 6	Designing a Half Adder and Half subtraction
Week 7	Designing a Full Adder and full subtraction
Week 8	Designing a D and SR Flip-Flop
Week 9	Designing aT and jk Flip-Flop
Week 10	Designing a Register
Week 11	Converting Between Number Systems[1]
Week 12	Converting Between Number Systems[2]
Week 13	Designing a Shift Register
Week 14	Designing a Sequential Counter
Week 15	Review and Final Project Design

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	• Digital Design" by M. Morris Mano • "Digital Logic Design" by Guy V. Steele Jr.	Yes
Recommended Texts	• "Digital Logic Design and Computer Architecture" by David L. Tarnoff	No
Websites		

Grading Scheme
مخطط الدرجات

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

(0 – 49)	**F –** Fail	راسب	(0-44)	Considerable amount of work required

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Probability and Statistics**			**Module Delivery**	
Module Type				☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code					
ECTS Credits	**6**				
SWL (hr/sem)	**150**				
Module Level	**UG I**		**Semester of Delivery**	**2**	
Administering Department	**Computer Science**		**College**	**CSM**	
Module Leader	Kahlan Faiq Hasan		**e-mail**	kahlan.aljuburi@tu.edu.iq	
Module Leader's Acad. Title	Assistant Lecturer		**Module Leader's Qualification**	**Ms**	
Module Tutor	**Rafal Salih**		**e-mail**	**Rafalsaleh@tu.edu.iq**	
Peer Reviewer Name	**Mohammed Aktham**		**e-mail**		
Scientific Committee Approval Date	**1/1/2025**		**Version Number**	**1.0**	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	1. Mathematics for Computer Science 2. Calculus 3. Linear Algebra 4. Discrete Mathematics 5. Introduction to Programming	**Semester**	
Co-requisites module	1. **Artificial Intelligence and Machine Learning** 2. **Data Science and Big Data** 3. Algorithms and Optimization 4. Computer Vision and Image Processing	**Semester**	

	5. Cryptography and Network Security 6. Computer Networks and Distributed Systems 7. Wireless Communication and Signal Processing 8. Queuing Theory in Network Design 9. Network Simulations 10. Internet of Things (IoT) and Sensor Networks		

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Objectives أهداف المادة الدراسية	1. Clarification of the fundamental concepts of probability. 2. Understanding Bayes' theorem and conditional probability. 3. Learning about discrete distributions and how to find expectation and variance. 4. Understanding continuous distributions and how to find expectation and variance, as well as studying the associated theorems. 5. Studying probability mass function (PMF), probability density function (PDF), cumulative distribution function (CDF).
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. The student should state the basic definitions. 2. Introduction and basic definitions of the topic of probability. 3. Distinguish between the probability mass function (PMF), probability density function (PDF), cumulative distribution function (CDF) with various examples. 4. Understand the joint random variables and dependent and independent random variables. 5. Recognize discrete distributions, their types, and study examples with theorems on expectation and variance. 6. Student can compute conditional probabilities based on the definition of conditional probability 7. Understand the continuous distributions and their types. 8. To be able to distinguish between discrete distributions and continuous distributions.

	9. To be able to distinguish between discrete random variables and continuous random variables.
Indicative Contents المحتويات الإرشادية	1. Introduction and basic definitions of the topic of probability. 2. Study of Bayes' theorem and conditional probability. 3. Understanding discrete distributions and their types. 4. Studying examples of discrete distributions with theorems on expectation and variance. 5. Understanding continuous distributions, with focus on the normal and standard normal distributions, accompanied by examples.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	The primary strategy to be adopted in delivering this unit is to encourage students to engage in exercises while simultaneously enhancing and expanding their critical thinking skills. This will be achieved through interactive classes and tutorials, as well as by considering types of simple experiments that involve some sampling activities of interest to students.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ أسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	78	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	5.2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	72	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	4.8
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	**150**		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment التقييم التكويني	**Quizzes**	10	20% (20)	3-14	
	Assignments	2	10% (10)	6 , 12	
	Projects / Lab.				
	Report				
Summative assessment التقييم التلخيصي	**First Midterm Exam**	2hr	10% (10)	6	
	Second Midterm	2hr	10% (10)	12	
	Final Exam	3hr	50% (50)	16	
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	**Module 1** - Introduction to statistics and probability - Importance of statistics in computer science. - Fundamental concepts: population, sample, parameters. - Mean, Median, Mode. - Application examples in computer science.
Week 2	**Module 1** - Probability space - Kolmogorov's axioms - Conditional probability
Week 3	**Module 1** - Independence - The law of total probability - Bayes' rule
Week 4	**Module 1** - Random variables (discrete, continuous, mixed) - Probability Mass Function (pmf)
Week 5	**Module 1** - Cumulative Distribution Function (cdf) - Probability Distribution Function (pdf)
Week 6	***First MIDTERM EXAMINATION***

Delivery Plan (Weekly Lab. Syllabus)

المنهاج الاسبوعي للمختبر

Week 7	**Module 1** - Expected value, moments - Variance, and standard deviation - Conditional Distributions and Moments
Week 8	**Module 1** Important distributions: Definitions, Parameters, and Properties. - Discrete Uniform - Bernoulli - Binomial
Week 9	**Module 1** Important distributions: Definitions, Parameters, and Properties. - Geometric - Poisson - Continuous Uniform
Week 10	**Module 1** Important distributions: Definitions, Parameters, and Properties. - Exponential - Gaussian
Week 11	***Second MIDTERM EXAMINATION***
Week 12	**Module 2** Bounding Properties - Markov inequality - Chebyshev inequality - Chernoff bounds
Week 13	**Module 2** Random Vectors - Joint pmf - Marginal pmf - Conditional Distributions - Independence of random variables
Week 14	**Module 2** - Joint moments of random vectors - Correlation, Covariance
Week 15	**Module 2** - Correlation Coefficient - Correlation and Covariance matrices
Week 16	***Final Examination***

	Material Covered		
Week 1	**Introduction to MATLAB for Probability and Statistics** • Setting up the environment • Basic operations and commands	**Week 8**	**EXAM**
Week 2	**Introduction to MATLAB for Probability and Statistics** • Setting up the environment • Basic operations and commands	**Week 9**	**Random Variables and Simulations** • Defining and working with discrete and continuous random variables • Simulating random experiments
Week 3	**Descriptive Statistics with MATLAB** • Calculating mean, median, and mode • Visualizing data distributions and their functions	**Week 10**	**Random Variables and Simulations** • Defining and working with discrete and continuous random variables • Simulating random experiments
Week 4	**Descriptive Statistics with MATLAB** • Calculating mean, median, and mode • Visualizing data distributions and their functions	**Week 11**	**Random Variables and Simulations** • Defining and working with discrete and continuous random variables • Simulating random experiments
Week 5	**Probability Distributions in MATLAB** • Generating and plotting common probability distributions (e.g., POISSON, EXPONENTIAL, NORMAL) • Calculating probabilities pdf and cdf	**Week 12**	**Random Variables and Simulations** • Defining and working with discrete and continuous random variables • Simulating random experiments
Week 6	**Probability Distributions in MATLAB** • Generating and plotting common probability distributions (e.g., POISSON, EXPONENTIAL, NORMAL) • Calculating probabilities pdf and cdf	**Week 13**	A comprehensive review
Week 7	**Probability Distributions in MATLAB** • Generating and plotting common probability distributions (e.g., POISSON, EXPONENTIAL, NORMAL)	**Week 14**	EXAM

	• Calculating probabilities pdf and cdf		
		Week 15	A comprehensive review

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts النصوص المطلوبة	- Probability, Statistics, and Random Processes for Electrical Engineering Alberto Leon-Garcia Pearson, 3rd Edition	NO
Recommended Texts	- Introduction to probability models, Sheldon M. Ross, 9th Edition - Intuitive Probability and Random Processes using MATLAB, Steven Kay	No
Websites	https://ocw.mit.edu/courses/18-05-introduction-to-probability-and-statistics-spring-2022/	

Grading Scheme مخطط الدرجات				
Group	**Grade**	التقدير	**Marks %**	**Definition**
Success Group (50 - 100)	**A -** Excellent	امتياز	90 - 100	Outstanding Performance أداء مذهل
	B - Very Good	جيد جدا	80 - 89	Above average with some errors فوق المتوسط مع بعض الأخطاء
	C - Good	جيد	70 - 79	Sound work with notable errors العمل السليم مع أخطاء ملحوظة
	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings عادل ولكن مع نواقص كبيرة
	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria العمل يلبي الحد الأدنى من المعايير
Fail Group (0 – 49)	**FX –** Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded مطلوب المزيد من العمل ولكن الائتمان الممنوح

	F – Fail	راسب	(0-44)	Considerable amount of work required قدر كبير من العمل المطلوب

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Programming Fundamentals**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	TUCS				
ECTS Credits	8				
SWL (hr/sem)	**200**				
Module Level		1	**Semester of Delivery**		1st
Administering Department		Computer Science	**College**	CCSM	
Module Leader	Mohanad Hatem Ramadhan		**e-mail**	Mohanad.H.Ramadhan@tu.edu.iq	
Module Leader's Acad. Title		Assistant Lecturer	**Module Leader's Qualification**		master
Module Tutor	Yahya Laith Khalil		**e-mail**		
Peer Reviewer Name		Mahammed Aktham	**e-mail**		
Scientific Committee Approval Date		07/09/2023	**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	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.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Upon successful completion of this module, students should be able to: 1. Demonstrate a solid understanding of the fundamental principles and concepts of programming. 2. Utilize the syntax and structure of the C++ programming language to write well-structured and efficient code. 3. Apply problem-solving skills and algorithmic thinking to develop solutions for a variety of computational problems. 4. Design, implement, and test programs using C++ to solve specific tasks and challenges. 5. Utilize modular programming techniques to create reusable and maintainable code. 6. Debug and troubleshoot programs effectively using appropriate debugging

	techniques and tools. 7. Collaborate and work effectively in teams to complete programming projects. 8. Communicate programming concepts, solutions, and ideas clearly and effectively, both orally and in written form. 9. Demonstrate a readiness to progress to more advanced programming courses or pursue a career in software development.
Indicative Contents المحتويات الإرشادية	1. Introduction to Computer Science: - Overview of computer science as a discipline - Key concepts and principles in computer science - Role of programming in computer science 2. Introduction to Computers, Binary System, and Information Representation: - Basics of computer architecture and components - Understanding the binary system and its significance in computing - Conversion between binary and decimal. - Representation of different data types in computers - ASCII and Unicode for character encoding 3. Algorithm Design and Problem Solving: - Understanding algorithms and problem-solving strategies - Analyzing problem requirements and designing algorithmic solutions - Time and space complexity analysis - Representing algorithms with Pseudocode and Flowcharts: - Using pseudocode as a high-level representation of algorithms - Writing pseudocode to describe the logic and steps of an algorithm - Understanding flowcharts as visual representations of algorithms - Basic flowchart symbols and their meanings - Creating flowcharts to represent the flow of control in algorithms 4. Introduction to C++: - History and features of the C++ programming language - Setting up a C++ development environment - Basic syntax and structure of C++ programs 5. Variables and Data Types: - Declaring and initializing variables - Fundamental data types (integers, floating-point numbers, characters) - Working with constants and literals 6. Operators and Expressions: - Arithmetic operators - Assignment operators

	- Comparison operators - Logical operators 7. Control Structures: - Decision-making with if-else statements - Switch statements for multiple choices - Repetition with loops (while, do-while, for) - Handling user input and validation 8. Functions: - Function declaration and definition - Parameters and argument passing - Return values and function overloading - Scope and lifetime of variables

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain 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 the learning process. 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 that students have the necessary resources to complete their lab exercises and assignments effectively. 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 provide support to students who need additional help or guidance in understanding programming concepts or completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	60	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	140	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	9.4
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	200		

Module Evaluation

تقييم المادة الدراسية

	Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome

Formative assessment	**Quizzes**	2	10% (10)	5, 11	
	Assignments	2	10% (10)	7, 12	
	Projects	1	10% (10)	5-14	
	Report				
Summative assessment	**Midterm Exam**	2 hr	20% (20)	11	
	Final Exam	2hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
Week No.	**Material Covered**
Week 1	Introduction to Computer Science, Computers Components, Binary and Info Representation
Week 2	Algorithms Design and Writing pseudocode
Week 3	Algorithms design and Drawing Flowchart
Week 4	Introduction to Programming Languages (History, Categories, Main Differences) and, Introduction to Programming in C++ (Program Structure and Coding Environment)
Week 5	Variables, Datatypes, Output, and Input
Week 6	Operations (Arithmetic and Assignment) and Math Functions
Week 7	Operations (Comparison and Logical)
Week 8	Flow Control (if – else)
Week 9	Flow Control (switch – case)
Week 10	Loops (counter and cumulative variables)
Week 11	Uncountable Loops
Week 12	Nested Loops
Week 13	Functions
Week 14	building a TikTacToe Game
Week 15	Reviewing Students' Projects

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	**Material Covered**
Week 1	**Using Operating System, Creating Files and Folders, Writing Text)**
Week 2	**Difference among (Text Editor, Word Processor, Code Editor and IDE)**

Week 3	**Drawing (Darg and drop) Flowcharts**
Week 4	**Installing C++ coding environment and running Hello World program**
Week 5	**Running Examples on Variables, Datatypes, Output, and Input**
Week 6	**Running Examples on Operations (Arithmetic and Assignment) and Math Functions**
Week 7	**Running Examples on Operations (Comparison and Logical)**
Week 8	**Running Examples on Flow Control (if – else)**
Week 9	**Running Examples on Flow Control (switch – case)**
Week 10	**Running Examples on Loops (counter and cumulative variables)**
Week 11	**Running Examples on Uncountable Loops**
Week 12	**Running Examples on Nested Loops**
Week 13	**Running Examples on Functions**
Week 14	**Fixing problems in students' projects**
Week 15	**Applying instructor's feedback on students' projects**

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	Stroustrup, Bjarne - Programming_ principles and practice using C++-Addison-Wesley (2015)	Yes
Recommended Texts	Olsson, Mikael - C++20 Quick syntax reference: a pocket guide to the language, apis, and library	No
Websites		

Grading Scheme
مخطط الدرجات

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

(0 – 49)	**F** – Fail	راسب	(0-44)	Considerable amount of work required

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information
معلومات المادة الدراسية

Module Title	**Web programming**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☒ **Practical** ☐ **Seminar**	
Module Code	TUCS				
ECTS Credits	6				
SWL (hr/sem)	**150**				
Module Level		1	**Semester of Delivery**		2nd
Administering Department		Computer Sciences	**College**	CCSM	
Module Leader	Yahya Layth Khaleel		**e-mail**	yahya@tu.edu.iq	
Module Leader's Acad. Title		Assistant Lecturer	**Module Leader's Qualification**		master
Module Tutor			**e-mail**		
Peer Reviewer Name			**e-mail**		
Scientific Committee Approval Date		2024-2025	**Version Number**	1.0	

Relation with other Modules
العلاقة مع المواد الدراسية الأخرى

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

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	1- Understand the concept of internet, protocols, how we can browse the internet? 2- Understand the principles of creating an effective web page, including an in-depth consideration of information architecture. 3- Develop skills in analyzing the usability of a website. 4- Understand how to plan and conduct user research related to web usability. 5- Learn the core web technologies and programming languages that power the modern web. Starting with HTML and CSS. 6- Exploring how to create static website. 7- Learn some concepts of server-side programming.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Upon successful completion of this module, students should be able to: 1. Understand the fundamental concepts of web development: Gain a solid understanding of how the web works, including client-server architecture, HTTP protocols, and the role of web browsers. 2. Develop proficiency in HTML and CSS: Acquire the skills to create well-structured web pages using HTML markup, apply CSS styles for layout and design, and create responsive and visually appealing websites. 3. Build static website. 4. Problem-solving and debugging skills: Develop the ability to identify and fix common issues in web development through effective troubleshooting and debugging techniques. 5. Implement security measures: Understand common web security vulnerabilities and learn techniques to protect web applications from attacks.

Indicative Contents المحتويات الإرشادية	1- Introduction to Internet Technology and websites • Internet working concepts, Internet and Intranet • WWW and URL parts • Introduction to web programming, Classifying of websites • Static and dynamic web sites 2- Introduction to Web Development • Overview of web technologies and standards • Client-server architecture and HTTP protocols • Understanding web browsers and their rendering engines • HTML Fundamentals 3- Structure of HTML • HTML tags and elements • Working with text, images, links, and lists • Creating forms for user input • Semantic HTML and accessibility best practices 4- CSS Styling • CSS syntax and selectors • Box model and layout techniques • Applying styles to text, colors, backgrounds, and borders • CSS positioning and responsive design • CSS frameworks and libraries

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Hands-on Practice: This Encourage students to actively engage in hands-on coding exercises and projects. Providing them with opportunities to apply theoretical concepts in practical scenarios, allowing them to gain proficiency through practice. 2. Project-Based Learning: Assigning projects that simulate real-world web development scenarios. This approach allows students to apply their knowledge and skills to create fully functional web applications, reinforcing their understanding and problem-solving abilities. 3. Code Review and Feedback: Incorporating code review sessions where students can share their code and receive constructive feedback. This

process helps students identify areas for improvement, learn best practices, and enhance their coding style and techniques.

4. Collaborative Learning: Foster a collaborative learning environment where students can work together on group projects or problem-solving tasks. Encourage peer-to-peer discussions, code sharing, and knowledge exchange, as this can enhance understanding and expose students to diverse perspectives and solutions.

5. Online Resources and Documentation: Introduce students to reputable online resources, documentation, and tutorials related to web development. Teaching students how to effectively search for solutions, read and understand documentation, and leverage online communities and forums for support and learning.

6. Practical Examples and Case Studies: Provide practical examples and case studies that demonstrate the application of web programming concepts in real-world scenarios. This helps students relate theoretical concepts to practical use cases, enhancing their understanding and problem-solving abilities.

7. Regular Assessments and Feedback: Conducting regular assessments, quizzes, and coding challenges to evaluate students' progress and understanding. Providing timely feedback to help students identify their strengths and areas that require improvement.

8. Continuous Learning and Exploration: Encouragement students to stay updated with the latest trends, tools, and technologies in web development.

9. Office Hours and Individual Support: The instructor should be available for individual consultations and provide support to students who need additional help or guidance in understanding programming concepts or completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	77	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	73	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	4.8
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	150		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	2	10% (10)	5, 11	
	Assignments	2	10% (10)	3, 12	
	Projects	1	10% (10)	8-14	
	Report				
Summative assessment	Midterm Exam	2 hr	20% (20)	11	
	Final Exam	2hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Introduction to Internet Technology
Week 2	Internet working concepts, Internet and Intranet
Week 3	WWW and URL parts
Week 4	Introduction to web programming, Classifying of websites
Week 5	Static and dynamic web sites
Week 6	Introduction to HTML, tags, attributes, Lists, Images
Week 7	HTML Tables and forms
Week 8	Introduction to CSS
Week 9	CSS box model, Internal and external CSS
Week 10	Applications and more examples in the CSS
Week 11	Developing static website 1
Week 12	Developing static website 2
Week 13	Developing static website 3
Week 14	Client-side scripting and website hosting
Week 15	AI web services

Delivery Plan (Weekly Lab. Syllabus):
المنهاج الاسبوعي للمختبر:

Week No.	Material Covered
Week 1	Introduction to Internet Technology, Websites, URL
Week 2	Install Notepad++, Introduction to HTML
Week 3	HTML – Titles, headings and Lists
Week 4	HTML Images
Week 5	HTML – Tables
Week 6	HTML Frame and Form
Week 7	Internal CSS
Week 8	External CSS
Week 9	Developing static website 1
Week 10	Developing static website 2
Week 11	Developing static website 3
Week 12	AI web services
Week 13	Client-side scripting and website hosting
Week 14	Content Management Systems (CMS): Install WAMP server, install Joomla
Week 15	Create dynamic website

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	None	
Recommended Texts	Jon, Duckett. "HTML and CSS: Design and Build Websites." (2016).	
Websites		

Grading Scheme
مخطط الدرجات

Group	Grade	التقدير	Marks (%)	Definition
Success Group (50 - 100)	**A** - Excellent	امتياز	90 - 100	Outstanding Performance
	B - Very Good	جيد جدا	80 - 89	Above average with some errors

	C - Good	جيد	70 - 79	Sound work with notable errors
	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings
	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Artificial Intelligence**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code	TU010101113				
ECTS Credits	6				
SWL (hr/sem)	**148**				
Module Level	2		**Semester of Delivery**	2nd	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Armaneesa Naaman Hasoon		**e-mail**	armaneesa@tu.edu.iq	
Module Leader's Acad. Title	Assistant Professor		**Module Leader's Qualification**	Master	
Module Tutor			**e-mail**		
Peer Reviewer Name	Dr. Mohamed Aktham		**e-mail**		
Scientific Committee Approval Date	13/01/2025		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية

Module Aims أهداف المادة الدراسية	• Understanding the concept of AI and its applications • Understanding the problem state space and problem-solving. • Implementing and employing intelligent search methods to • solve problems that are not solved with traditional methods. • Using Heuristics in games. • Understanding the difference between blind search and heuristics search • To enable the learner to understand the basics of machine learning.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	• Understand the definition and application of AI • Understand the basic syntax and semantics of Python programming language. • Analyze and solve problems using Python programming. • Develop the student's thinking skills and the ability to analyze problems using logical programming principles. • Develop skills in debugging and testing AI search strategies in Python programming language. • • Providing the learner with the necessary skills to be able to build a learning model.
Indicative Contents المحتويات الإرشادية	• Fundamentals of AI • Problem in AI (Problem state space, Search space, and Problem-solving) • AI search taxonomy • Blind search algorithms • Heuristic search algorithms • the basics of machine learning

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain the definition of Artificial intelligence (AI), its applications, problem-solving techniques, types of search algorithms in AI, and the basic principles related to machine learning. 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 AI search algorithm examples, discussing practically

	how to program these algorithms, and exploring real-world applications of AI 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 AI projects. 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 AI 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 that students have the necessary resources to complete their lab exercises and assignments effectively. 4. Programming Assignments: Assignments will be given to students to reinforce their understanding of how to program search strategies of AI and encourage independent problem-solving. These assignments may involve implementing algorithms, designing software systems, or developing small-scale projects using Python. 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 provide support to students who need additional help or guidance in understanding completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	56	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3.7

Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	148		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	
	Assignments	4	20% (20)	7, 12	
	Projects	1	20% (20)	9-14	
	Report	1			
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	AI Fundamental definitions, some techniques used today in AI, Some Applications of A
Week 2	Problem spaces; problem-solving by search
Week 3	Brute-force search (depth-first, breadth-first)
Week 4	Iterative deep DFS, Uniform cost first search
Week 5	Heuristic search (hill-climbing, best first search)
Week 6	A* algorithm
Week 7	Dijkstra algorithm
Week 8	Two-player games (minimax search, alpha-beta pruning)
Week 9	ɔduction to Machine Learning
Week 10	Main Steps of ML
Week 11	**Midterm Exam**
Week 12	Supervised learning, Unsupervised learning, Reinforcement learning
Week 13	Methods for evaluating machine learning algorithms
Week 14	Project Due (Students Presentations part1)
Week 15	Project Due (Students Presentations part2)

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	Material Covered
Week 1	Introduction to the basics of Python programming Language
Week 2	Introduction to the basics of Python programming Language
Week 3	Introduction to the basics of Python programming Language
Week 4	Running Examples of depth-first search
Week 5	Running Examples of breadth-first search
Week 6	Running examples of Uniform cost first search
Week 7	Running Examples of Best First Search
Week 8	Running Examples of A* search algorithm
Week 9	Running Examples of the Dijkstra Algorithm
Week 10	Exploring Python Libraries for ML: Discuss libraries like Numpy
Week 11	Exploring Python Libraries for ML: pandas
Week 12	Exploring Python Libraries for ML: matplotlib
Week 13	Exploring Python Libraries for ML: SciPy
Week 14	Exploring Python Libraries for ML: scikit-learn
Week 15	Answering students' questions and reviewing previous topics.

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	"Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig (2020, 4th Edition) 2. "Artificial Intelligence: Structures and Strategies for Complex Problem Solving" by George F. Luger (2021, 7th Edition)	No

Recommended Texts	Dimitris Varkas and Ioannis Pl. Vlashavos, " Artificial Intelligence for Advanced Problem-Solving Technique", published in the USA by Information science reference (an imprint of "IGI" Global),2008.	No
Websites		

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Data Structure**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code					
ECTS Credits					
SWL (hr/sem)	**148**				
Module Level		2	**Semester of Delivery**		1nd
Administering Department		Computer Science	**College**	CCSM	
Module Leader	**Mohammed Basim Omar**		**e-mail**	**mohammed.b@tu.edu.iq**	
Module Leader's Acad. Title		**Asst. lect.**	**Module Leader's Qualification**		
Module Tutor			**e-mail**		

Peer Reviewer Name	Mohamed Aktham	**e-mail**	
Scientific Committee Approval Date	26/08/2024	**Version Number**	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	Data Structure	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	1. Learning different data structures 2. Learning how to choose the best data structure for your algorithm. 3. learn how to deal with your problem, building its algorithm and fitting the best data structures to it. 4. Building the student scientifically and qualifying him to work in the field of computer science. 5. Building and preparing the student psychologically to play his role as a reliable programmer in this field. 6. A general definition of the subject of data structures and how to store data in memory. 7. The difference between linear and non-linear data structures. 8. A detailed explanation of algorithms and how to program them. 9. Teaching the student to calculate the complexity time of the algorithm. 10. Encouraging the student to be creative and think about the specialization projects and keep pace with the development taking place in this field.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Knowledge of the basic concepts of data structures. 2. The student learns about the importance of general concepts of algorithms. 3. Identify the most important algorithms used to organize data in memory. 4. Study the most important factors that affect the speed of algorithm execution
Indicative Contents	1. Review of Basic Programming Concepts:

المحتويات الإرشادية	- Recap of fundamental programming concepts, including variables, data types, control structures, and functions. 2. Arrays: - Multidimensional arrays - Array manipulation techniques - Searching and sorting algorithms 3. Strings: - String manipulation and operations - String handling functions 4. Pointers: - Introduction to pointers and their usage - Memory addresses and pointer arithmetic - Pointers to arrays 5. Tree: - tree handling concepts - Searching from and writing to tree 6. Graph - Graph handling concepts - explain types of graphs - Graph Representation

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain data structure concepts, 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 the learning process. 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. 4. 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. 5. Office Hours and Individual Support: The instructor should be available for individual consultations and provide support to students who need additional help or guidance in understanding programming concepts or completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	56	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3.7
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	148		

Module Evaluation

تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	
	Assignments	4	20% (20)	7, 12	
	Projects	1	20% (20)	5-14	
	Report	1			

Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
Week No.	**Material Covered**
Week 1	Introduction to Data Structure.
Week 2	Algorithms and Complexity.
Week 3	Array in Data Structure & Represent arrays in memory.
Week 4	Pointer.
Week 5	Stack in Data structure.
Week 6	Expression Parsing in Data structure.
Week 7	**First exam.**
Week 8	Queue in Data structure.
Week 9	Circular queue in Data structure.
Week 10	LinkedList in Data structure.
Week 11	Introduction Tree.
Week 12	binary tree & binary search tree.
Week 13	Graph.
Week 14	Sorting.
Week 15	Hashing.

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	**Material Covered**
Week 1	**Re-explaining the basics of programming to the student in C++.**
Week 2	**Introduction on Array programming**
Week 3	**Explain how stack types work in programming [1].**
Week 4	**Explain how stack types work in programming [2].**
Week 5	**Explain how queue types work in programming [1].**
Week 6	**Explain how queue types work in programming [2].**
Week 7	**Start Of Linked List programming.**

Week 8	**Add Node & Delete Node from Linked list [1]**
Week 9	**Add Node & Delete Node from Linked list [2]**
Week 10	**How Work Binary Search Programming.**
Week 11	**Binary Search with Iterative Programming.**
Week 12	**Binary Search with Recursive Programming.**
Week 13	**Selection Sort programming.**
Week 14	**Bubble Sort Programming.**
Week 15	**Comprehensive review in Programming.**

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	• Vinu V Das, *Principles of Data Structures Using C and C++* • Introduction to Algorithm, third Edition, Thomas H. Cormen Algorithms, fourth edition, Robert Sedgewick and Kevin Wayne.	Yes
Recommended Texts	• Ramesh Vasappanavara, Anand Vasappanavara, *Data Structures using C by practice*, • D.S. MALIK, *Data Structures Using C++*, 2nd Edition, • Robert L. Kruse, Alexander J. Ryba, *Data Structures and Program Design in C++*,	No
Websites		

Grading Scheme
مخطط الدرجات

Group	Grade	التقدير	Marks (%)	Definition
Success Group (50 - 100)	**A** - Excellent	امتياز	90 - 100	Outstanding Performance
	B - Very Good	جيد جدا	80 - 89	Above average with some errors
	C - Good	جيد	70 - 79	Sound work with notable errors
	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings

	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information
معلومات المادة الدراسية

Module Title	**Embedded Systems**	**Module Delivery**	
Module Type	Core	☒ **Theory** ☐ **Lecture** **Lab** ☐ **Tutorial** ☒ **Practical** ☐ **Seminar**	
Module Code			
ECTS Credits			
SWL (hr/sem)	**60**		
Module Level	2	**Semester of Delivery**	2nd
Administering Department	Computer Science	**College**	CCSM
Module Leader	Ammar Farooq Abbas	**e-mail**	ammar.abbas@tu.edu.iq
Module Leader's Acad. Title	Lecturer	**Module Leader's Qualification**	Master
Module Tutor		**e-mail**	
Peer Reviewer Name	Mahammed Aktham	**e-mail**	
Scientific Committee Approval Date	09/01/2025	**Version Number**	1.0

Relation with other Modules
العلاقة مع المواد الدراسية الأخرى

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

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	1. Provide students with a fundamental understanding of **embedded systems**, their components, and their applications in real-world scenarios. 2. Develop students' ability to **design and program embedded systems** using microcontrollers and sensors. 3. Equip students with knowledge of key communication protocols like **UART, SPI, and I2C**. 4. Enhance problem-solving skills through the **implementation of practical projects** combining hardware and software. 5. Introduce students to modern advancements in **wireless embedded systems** and power management techniques. 6. Prepare students for future careers in fields like **IoT, robotics, and automation** by understanding embedded system design principles.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	By the end of this module, students will be able to: 1. Understand the fundamental **concepts of embedded systems**, including their architecture and components. 2. Demonstrate the ability to **program microcontrollers** to interact with sensors and actuators. 3. Apply knowledge of **communication protocols** (UART, SPI, I2C) to facilitate communication between devices. 4. Design and implement **embedded systems projects** by integrating hardware and software components. 5. Analyze and solve technical challenges related to **real-time system constraints** like power, memory, and timing. 6. Explore and utilize **wireless technologies** such as Wi-Fi and Bluetooth in embedded systems. 7. Present and document their projects effectively, showcasing their understanding of embedded systems.
Indicative Contents المحتويات الإرشادية	The module covers the following topics: 1. **Introduction to Embedded Systems** o Definition, importance, and applications in daily life. 2. **Embedded System Components** o Microcontrollers, sensors, actuators, and I/O interfaces. 3. **Microcontroller Architecture**

	○ Processor basics, memory types (RAM, ROM, Flash), and internal peripherals (GPIO, timers). 4. **Programming for Embedded Systems** ○ Introduction to C/C++ for embedded systems. ○ Basics of low-level programming and debugging. 5. **Communication Protocols** ○ UART, SPI, and I2C protocols. ○ Data exchange between components. 6. **System Design and Development** ○ Analyzing requirements. ○ Combining hardware and software components. 7. **Real-Time Operating Systems (RTOS)** ○ Basic concepts and importance in embedded systems. 8. **Power Management** ○ Techniques to optimize power usage. 9. **Wireless Communication in Embedded Systems** ○ Wi-Fi and Bluetooth modules for data transfer. 10. **Practical Projects and Case Studies** ○ Designing a basic embedded system project integrating sensors, actuators, and communication.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	- Interactive Lectures - Dialogue and discussion - Brainstorming - Solving problems - Projects, tasks and costs projects - Self-learning - Learning Cooperative - Exchanging experiences among colleagues.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ أسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل		**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	

Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل		**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	60		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	3	15% (10)	5, 11	#LO 1-3, #LO 5-8
	Assignments	5	15% (10)	7, 12	#LO 3-5, #LO 5-8
	Projects	2	10% (10)	continuous	
	Report		10% (10)	14	#LO 1-8
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	#LO 1-7
	Final Exam	2 hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	**Introduction to Embedded Systems**
Week 2	**Components of Embedded Systems**
Week 3	**Microcontroller Architecture**
Week 4	**Programming for Embedded Systems**
Week 5	**Communication Protocols1**
Week 6	**Communication Protocols2**
Week 7	**Embedded System Design1**
Week 8	**Embedded System Design2**
Week 9	**Real-Time Operating Systems (RTOS)**
Week 10	**Power Management in Embedded Systems**
Week 11	**Wireless Embedded Systems**
Week 12	**Challenges in Embedded Systems**
Week 13	**Applications of Embedded Systems1**
Week 14	**Applications of Embedded Systems2**
Week 15	**Comprehensive Review**

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
1.	**Setting Up the Development Environment**
2.	**Writing Your First Program**
3.	**Reading Sensor Inputs**
4.	**Controlling Actuators**
5.	**Using an LCD Display**
6.	**UART Communication**
7.	**Mini Project with Sensors**
8.	**Working with I2C Protocol**
9.	**Working with SPI Protocol**
10.	**Wireless Communication**
11.	**Final Project**
12.	**Project Review and Evaluation**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Embedded Systems: Introduction to ARM Cortex-M Microcontrollers	No
Recommended Texts	**1-** ***كتاب تعلم برمجة النظم المدمجة اللاسلكية WiFi باستخدامESP8266*** **2-** ***اردوينو ببساطة***	No
Websites		

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

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

Course Description Form

1. Course Name:
English Language
2. Course Code:
English Language 3
3. Semester / Year:
2023-2024
4. Description Preparation Date:
21/3/2024
5. Available Attendance Forms:
Attendance
6. Number of Credit Hours (Total) / Number of Units (Total)
Total (30) hours and (2) hours of each week
7. Course administrator's name (mention all, if more than one name)
Name: Ahmed fayeq saber Email: ahmed.f.saber@tu.edu.iq ....
8. Course Objectives

Course Objectives	1- Enabling students to know contemporary strategies for learning English 2- The course aims to introduce students to the principles of the English language. 3- Developing speaking, reading, writing and listening skills. 4- Enabling students to learn the language of dialogue. 5- Enabling students to learn about the cultures of some English-speaking countries
9. Teaching and Learning Strategies	
Strategy	The main strategy that will be adopted in developing the four skills: 1. The skill of speaking 2. The skill of reading

	3. The skill of writing 4. The skill of listening Also, enable the students for the use of grammar correctly.

10. Course Structure

Week	Hours	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
1	**2**	**Introduction: Review about Study materials.**	**Unit 1**	Lectures	Midterm exam +quizzes
2	**2**	**Quantities, wh- questions and answers.**	**Unit 2**	Lectures	
3	**2**	**Articles, reading (about shopping).**	**Unit 3**	Lectures	
4	**2**	**Vocabulary: Shopping, prices.**	**Unit 4**	Lectures	
5	**2**	**Grammar: Verb patterns, making questions.**	**Unit 5**	Lectures	
6	**2**	**Future: Going to/will, practices, reading (Hollywood kids).**	**Unit 6**	Lectures	

7	**2**	**Model and related verbs, phrasal verbs, polite requests**	**Unit 7**	Lectures	
8	**2**	**Grammar: hot verbs, listening, everyday English: how do you feel?**	**Unit 8**	Lectures	
9	**2**	**Grammar: What ….. like? , Comparative and superlative adjectives big, bigger, practices.**	**Unit 9**	Lectures	
10	**2**	**Vocabulary: Synonyms and antonyms.**	**Unit 10**	Lectures	
11	**2**	**everyday English about directions, listening and reading, practices.**	**Unit 11**	Lectures	
12	**2**	**Present tense, simple present, present continuous, practices.**	**Unit 12**	Lectures	
13	**2**	**Grammar: for/ since, practices, questions.**	**Unit 13**	Lectures	

14	**2**	**Adverbs, word pairs, practices.**	**Unit 14**	Lectures	
15	**2**	**Everyday English about short answers (Question tags).**	**Unit 15**	Lectures	

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, report.

12 .Learning and Teaching Resources	
Required textbooks (circular, if any)	
Main Reverences (sources)	New Headway intermediate 4th edition
Recommended books and references (scientific journals, reports...)	New Headway pronunciation course
Electronic Reverences, Websites	https://7esl.com/

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Object Oriented Programming**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code	TU010101113				
ECTS Credits	6				
SWL (hr/sem)	**148**				
Module Level	2		**Semester of Delivery**	1st	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Israa Rafaa Abdulkader		**e-mail**	Israa.R.Abdalkader@tu.edu.iq	
Module Leader's Acad. Title	lecture		**Module Leader's Qualification**	Master	
Module Tutor			**e-mail**		

Peer Reviewer Name	Mohamed Aktham	**e-mail**	
Scientific Committee Approval Date	30/08/2024	**Version Number**	1.0

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	1. Abstraction: hiding the implementation details and showing only functionality to the user. 2. Encapsulation: data hiding. 3. Polymorphism: the ability to take more than one form. 4. Inheritance: objects of one class acquire the properties of objects of another class.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Codes basic programs in java programming language 2. Introduction (Object Oriented Programming Characteristics, OOP Definition, OOP Concepts, Differences from Procedure Oriented Programming (POP)). 3. Classes and Objects (methods, properties). 4. Constructors and Destructors. 5. Inheritance (Single Inheritance, Multilevel Inheritance) and Polymorphism (Function Overloading, Operator Overloading, Virtual Function). 6. Abstract Classes and Methods, Sealed Classes and Interfaces, Generic types and methods and collection class.
Indicative Contents المحتويات الإرشادية	Basics of java Characteristics of Object-Oriented Programming Classes and Objects Working with Constructors Data Members Using Static Variables & Understanding Scope Overloading

	Inheritance Exceptions and Errors

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain programming concepts, java 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 the learning process. 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 that students have the necessary resources to complete their lab exercises and assignments effectively. 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 java . 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 provide support to students who need additional help or guidance in understanding programming concepts or completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	56	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3.7
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	148		

Module Evaluation

تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	
	Assignments	4	20% (20)	7, 12	
	Projects	1	20% (20)	9-14	
	Report	1			
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Recap of fundamental programming concepts, including variables, data types, control structures Using Java language

Week 2	Introduction to method(function)
Week 3	Mathematical Functions and Class Character
Week 4	Introduction to String and methods
Week 5	Introduction to Arrays (Linear arrays) and Searching and Sorting Linear Arrays
Week 6	Multidimensional Arrays and Square Arrays
Week 7	Introduction (Object Oriented Programming Characteristics, OOP Definition, OOP Concepts, Differences from Procedure Oriented Programming (POP)).
Week 8	Classes and Objects (methods, properties).
Week 9	Polymorphism (Overloading and Overriding Methods) & Constructors.
Week 10	Inheritance (Single Inheritance, Multilevel Inheritance).
Week 11	**Midterm Exam**
Week 12	Abstract Classes and Methods, Interfaces, Generic types and methods..
Week 13	Array List Class
Week 14	Project Due (Students Presentations part1)
Week 15	Project Due (Students Presentations part2)

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	Material Covered
Week 1	Getting used to java netbeans Interfaces and practicing some commands
Week 2	Running Examples on method
Week 3	Running examples on Mathematical Functions and Class Character
Week 4	Running Examples on Strings
Week 5	Practicing Arrays further (Searching and Sorting)
Week 6	Running Examples on 2D and Square Arrays
Week 7	Running Examples on Object Oriented Programming Characteristics
Week 8	Running Examples Classes and Objects (methods, properties).
Week 9	Running Examples on Overloading , Overriding Methods and Constructors.
Week 10	Running More Examples on Inheritance (Single Inheritance, Multilevel Inheritance).
Week 11	Running Examples on classes
Week 12	Running More Examples on Abstract Classes and Methods, Interfaces, Generic types and

	methods.
Week 13	Running examples on Array List Class
Week 14	Running programs of advanced of class
Week 15	Answering students' questions and reviewing previous topics.

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Herbert Scildt, Java The complete reference Eleventh edition	No
Recommended Texts		No
Websites		

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية

Module Title	Relational databases and structured query language	Module Delivery
Module Type	Core	☐ Theory ☒ Lecture ☒ Lab ☐ Tutorial ☐ Practical ☐ Seminar
Module Code	TU010101113	
ECTS Credits	6	
SWL (hr/sem)	148	

Module Level	2	Semester of Delivery	2nd
Administering Department	Computer Science	College	CCSM
Module Leader	Mustafa Lateef Fadhil	e-mail	Mustafa.l.fadhil@tu.edu.iq
Module Leader's Acad. Title	Assistant Lecturer	Module Leader's Qualification	Master
Module Tutor		e-mail	
Peer Reviewer Name	Mohamed Aktham	e-mail	
Scientific Committee Approval Date	00/00/2024	Version Number	1.0

Relation with other Modules
العلاقة مع المواد الدراسية الأخرى

Prerequisite module	Relational databases and structured query language	Semester	
Co-requisites module	None	Semester	

Module Aims, Learning Outcomes and Indicative Contents
أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية

Module Aims أهداف المادة الدراسية	1. Understanding Relational databases and structured query language: The module aims to provide students with a deep understanding of relational databases, structured query language, and their manipulation techniques. Students will learn about data modeling using the entity–relationship (ER) model. 2. Mastery of Entity: The module aims to develop students' proficiency in using

	entity diagram in Microsoft SQL Server. Students will learn the concepts of Entity Types, Entity Sets, Attributes, and Keys. 3. Data Handling and Structured SQL Query Operations: The module aims to introduce students to file handling concepts and techniques in relational database. Students will learn how to Relationship Types, Relationship Sets, Roles, and Structural Constraints. 4. Data Management and Organization: The module aims to teach students how to manage and organize data effectively in database. They will learn to create, update, and delete data, organize data directories, and handle data-related operations. Students will understand the importance of data management in real-world database scenarios. 5. Practical Application and Problem-Solving: Throughout the module, students will be exposed to practical The Enhanced Entity–Relationship (EER) Model 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 the relational algebra, relational calculus and data handling in Microsoft SQL server, this advanced 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 , handle data, and write efficient and reliable code.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Demonstrate an in-depth understanding of Relational Database Design by ER- and EER-to-Relational Mapping and data handling concepts in SQL. 2. Apply advanced operations, such as Update Operations, Transactions, and Dealing with Constraint Violations. 3. Manipulate Relational Calculus like Unary Relational Operations, Binary Relational Operations and Relational Algebra Operations from Set Theory. 4. Utilize pointers proficiently for data manipulation, including memory addresses, and pointer arithmetic 5. Manage and organize data effectively, including creating, updating, deleting, and organizing data directories. 6. Apply efficient programming techniques, optimize code, and adhere to best practices for writing clean and readable code. 7. Demonstrate problem-solving skills by applying the learned concepts to solve real-world programming challenges. 8. Work collaboratively in teams, communicate effectively, and share knowledge and ideas related to advance programming concepts. 9. Adapt to new programming concepts and technologies beyond the scope of the course, building a foundation for lifelong learning in programming.

	These learning outcomes reflect the knowledge, skills, and competencies that students will acquire upon completing the advanced programming course. The outcomes emphasize both theoretical understanding and practical application, preparing students for real-world programming challenges and further studies in the field of computer science.
Indicative Contents المحتويات الإرشادية	1. Review of Basic Database Concepts: - Recap of fundamental database concepts, including variables, data types, control structures, and functions. 2. Data Modeling Using the Entity–Relationship (ER) Model: - Using High-Level Conceptual Data Models for Database Design - A Sample Database Application - Entity Types, Entity Sets, Attributes, and Keys - Relationship Types, Relationship Sets, Roles, and Structural Constraints - Weak Entity Types 3. The Enhanced Entity–Relationship (EER) Model: - Subclasses, Super classes, and Inheritance - Specialization and Generalization 4. Relational Model Concepts: - Domains, Attributes, Tuples, and Relations - Characteristics of Relations

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain database concepts, SQL syntax, and 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 the learning process. Discussions can include reviewing code examples, discussing database best practices, and exploring real-world applications of database concepts.

	3. Laboratory Sessions: Laboratory sessions are dedicated practical sessions where students apply the concepts learned in lectures to hands-on database exercises. Key strategies for the laboratory sessions include: a. Programming Exercises: Students will work on database 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 that students have the necessary resources to complete their lab exercises and assignments effectively. 4. Programming Assignments: Assignments will be given to students to reinforce their understanding of database programming concepts and encourage independent problem-solving. These assignments may involve implementing sql query, designing database systems and tables. 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 provide support to students who need additional help or guidance in understanding database programming concepts or completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	56	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3.7
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.13
Total SWL (h/sem)	148		

الحمل الدراسي الكلي للطالب خلال الفصل	

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	2	10% (10)	5, 11	
	Assignments	4	20% (20)	7, 12	
	Projects	1	20% (20)	5-14	
	Report	1			
Summative assessment	Midterm Exam	2 hr	10% (10)	11	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Recap of fundamental database concepts, including variables, data types, control structures, and functions.
Week 2	Data Modeling Using the Entity–Relationship (ER) Model
Week 3	Using High-Level Conceptual Data Models for Database Design
Week 4	A Sample Database Application
Week 5	Relationship Types, Relationship Sets, Roles, and Structural Constraints
Week 6	Weak Entity Types
Week 7	The Enhanced Entity–Relationship (EER) Model
Week 8	ER Diagrams, Naming Conventions, and Design Issues
Week 9	Example of Other Notation: UML Class Diagrams
Week 10	The Enhanced Entity–Relationship (EER) Model
Week 11	Constraints and Characteristics of Specialization and Generalization Hierarchies
Week 12	Modeling of UNION Types Using Categories
Week 13	The Relational Data Model and Relational Database Constraints
Week 14	Second Project Due (Students Presentations part1)
Week 15	Second Project Due (Students Presentations part1)

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	**Material Covered**
Week 1	**Modeling of UNION Types Using Categories**
Week 2	**Running Examples on Unary Relational Operations:** **SELECT and PROJECT**
Week 3	**Relational Algebra Operations from Set Theory:** **The UNION, INTERSECTION, and MINUS Operations**
Week 4	**Binary Relational Operations:** **JOIN and DIVISION**
Week 5	**Running Examples on unary and binary**
Week 6	**Running Examples on set theory**
Week 7	**Additional Relational Operations**
Week 8	**Examples of Queries in Relational Algebra**
Week 9	**The Tuple Relational Calculus**
Week 10	**Running More Examples on Tuple Variables and Range Relations**
Week 11	**Running Examples on Database and tables**
Week 12	**Running More Examples on database**
Week 13	**Running More Advanced Programs on data**
Week 14	**Wrapping up**
Week 15	**Answering Students Questions and Extra Advising on Real World Application Programming**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Elmasri, R. (2021). Fundamentals of database systems seventh edition.	Yes

Recommended Texts		No
Websites		

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Visual Programming**			**Module Delivery**	
Module Type				☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code					
ECTS Credits					
SWL (hr/sem)	**148**				
Module Level	2		**Semester of Delivery**	2nd	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Israa Rafaa Abdulkader		**e-mail**	Israa.R.Abdalkader@tu.edu.iq	
Module Leader's Acad. Title	lecture		**Module Leader's Qualification**	Master	
Module Tutor			**e-mail**		

Peer Reviewer Name	Mohamed Aktham	**e-mail**	
Scientific Committee Approval Date	30/08/2024	**Version Number**	1.0

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	5. Be familiar with the general characteristics of the GUI. 6. Understand the important conceptual level issues associated with good interface design. 7. Understand the purpose of toolkits and high-level class libraries, and be able to write significant programs using the java netbeans.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Codes basic programs in java (GUI). 2. Introduction JOptionPane Class and uses Methods of the JOptionPane Class. 3. Explains and uses (jButton , jTextfield , jLabel)Components. 4. Explains and uses (jRadioButton , jCheckBox)Components. 5. Explains and uses (jComboBox ,jListBox, jTable, jTextArea)Components. 6. Explains and uses JFileChooser , file load , FileWriter class
Indicative Contents المحتويات الإرشادية	java (GUI) Overview Methods of the JOptionPane Class GUI components JFileChooser file load FileWriter class

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain programming concepts, java 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 the learning process. 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 that students have the necessary resources to complete their lab exercises and assignments effectively. 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 java . 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 provide support to students who need additional help or guidance in understanding programming concepts or completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	56	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3.7
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	148		

Module Evaluation

تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	3	10% (10)	4, 8,11	
	Assignments	3	20% (20)	4, 9,13	
	Projects	1	20% (20)	9-14	
	Report	1			
Summative assessment	**Midterm Exam**	2 hr	10% (10)	7	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	java netbeans Introduction
Week 2	Methods of the JOptionPane Class
Week 3	Methods of the JOptionPane Class
Week 4	GUI components (jButton + jTextField + jLabel)
Week 5	GUI components (RadioButton + CheckBox)
Week 6	GUI components (ListBox + ComboBox)
Week 7	**Midterm Exam**

Week 8	GUI components (ListBox + ComboBox)
Week 9	jTable Component
Week 10	JFileChooser
Week 11	jTextArea Component
Week 12	file load
Week 13	FileWriter class and methods
Week 14	Project Due (Students Presentations part1)
Week 15	Project Due (Students Presentations part2)

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	Material Covered
Week 1	Experiment (Write the first program in java)
Week 2	Running Examples(Methods of the JOptionPane Class)
Week 3	Running Examples(Methods of the JOptionPane Class)
Week 4	Running Examples on (GUI components (jButton + jTextField + jLabel))
Week 5	Running Examples on (GUI components (RadioButton + CheckBox))
Week 6	Running Examples on (GUI components (ListBox + ComboBox))
Week 7	Running program contain on all of the above
Week 8	Running Examples jTable Component
Week 9	Running Examples JFileChooser.
Week 10	Running More Examples on jTextArea Component
Week 11	Running Examples on File Load
Week 12	Running More Examples on FileWriter class and methods
Week 13	Running More Examples on File Load FileWriter class and methods
Week 14	Running programs of advanced of programs
Week 15	Answering students' questions and reviewing previous topics.

Learning and Teaching Resources مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	java: graphical user interface Introduction to java programming , David Etheridge	No
Recommended Texts	GUI(GraphicalUser Interface), bilalAmjad	No
Websites		

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

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

نموذج وصف المقرر)منهج بحث(

وصف المقرر

يقدم وصف المقرر لمحة شاملة عن أهم ميزاته وأهدافه التعليمية المتوقعة من الطالب. حيث يركز على تقييم مدى استفادة الطالب من فرص التعلم المتاحة، وذلك من خلال ربطه بوصف البرنامج الأكاديمي ككل. ويسُاعد هذا الربط على فهم كيفية تكَامُل المقرر مع بقية مقررات البرنامج، مما يعُزّز قدرة الطالب على تحقيق أهدافه التعليمية بشكل عام.

1 المؤسسة التعليمية .	جامعة تكريت /كلية علوم الحاسوب والرياضيات
2 القسم العلمي / المركز.	علوم الحاسوب
3 اسم / رمز المقرر.	منهج بح ث
4 أشكال الحضور المتاحة.	حضوري

5 الفصل / السنة.	2024-2023
6 عدد الساعات الدراسية)الكلي(.	ساعة نظري 30
7 تاريخ إعداد هذا الوصف .	2024/1/28

8 اهداف المقرر.

1. تعليم الطالب ا لمهارات المستمدة من مادة منهج بحث.
2. تعميق معرفة الطالب الجامعي بخطوات البحث العلمي.
3. تأهيل الطلبة لتدريس هذه المادة لطلبة المدارس المتوسطة والاعدادية.
4. تأهيل الطلبة ل لمساهمة في تقدم العلم من كافة النواحي، وذلك من خلال الاكتشافات الحديثة ،وتطوير الاكتشافات القديمة.
5. تعريف الطالب بمفهوم البحث والطريقة العلمية في البحث.
6. التعريف بـ خطوات البحث العلمي، مشاريع التخرج.
7. إكساب الطالب المعرفة في كيفية تحديد الافتراضات ومشكلات البحث.
8. تعريف الطالب بتصنيفات البحوث من حيث أهدافها ومناهجه ا.
9. فهم أسلوب النقد والتحليل.
10. التعريف بأهمية المكتبة الالكترونية و الانترنت ودورهما في البحث العلمي.
11. التعريف بأهمية الجانب الاخلاقي في البحث العلمي.

8 مخرجات المقرر وطرائق التعليم والتعلم والتقييم .

أ- الاهداف المعرفية

أ 1 - المعرفة بالمفاهيم الأساسية لمادة منهج بحث

أ 2 - يتعرف الطالب على اهمية المفاهيم العامة للبحث

أ 3 - يتعرف الطالب على أهمية مفردات وأدوات البحث العلمي العلمي

أ 4 - تعريف الطالب بمفهوم أدوات البحث ودورها في جمع البيانات والشروط التي يجب توفرها فيها من حيث صدقها وثباته ا

ب - الاهداف المهاراتية الخاصة بالمقرر

- تعلم خطوات كتابة البحث العلمي، مشروع التخرج، التقرير العلمي وامتلاك المقدرة على 1ب إعدادها.
- تعويد الطالب على إتباع الاساليب والقواعد العلمية المعتمدة في كتابة 2ب
- تعويد الطالب على استخدام الوثائق و الكتب ومصادر المعلومات 3البحوث ب والربط بينها
- التعرف على اهم قواعد البيانات والمكتبات الالكترونية للمساعدة في الحصول على أحدث 4ب
- التعرف على طريقة البحث وجمع المصادر العلمية الموثوقة واستخدامها في الكتابة 5المصادر ب

طرائق التعليم والتعلم

- محاضرات تقليدية واسلوب المناقشة والحوار
- الانشطة ا لصفية واعداد التقارير
- محاضرات متطورة)عرض تقديم ي)
- استخدام المناقشة عبر الصف الالكتروني (Classroom Google(
- استخدام برامج كتابة المصادر بالصيغة الالكترونية
- الاستعانة ببعض الصور التوضيحية والمخططات والامثلة الرسومية في توضيح المادة العلمية

طرائق التقييم

اجراء الاختبارات النظرية والعملية/
يومية/فصليه/نهائية التقارير
إجراء الاختبارات الشفوية

ج- الاهداف الوجدانية والقيمية

- استخدام المهارات العلمية والمعرفية من خلال اسلوب الحوار في المواضيع الاكاديمية. 1ج
- تنمية قدرات الطالب في مجال البحث العلمي، وتنمية قدرته ع لى إدراك وفهم واستيعاب المعايير 2ج
- ترسيخ روح المشاركة بين الطلبة3التي يستند إليها من خلال المناقشات والأنشطة الصفية ج والعمل بروح الفريق.
- ترسيخ وزرع مبدأ ان الوقت عامل ضروري ومهم في العمل البحثي 4ج

طرائق التعليم والتعلم

- الواجبات الصفية والبيتية
- الانشطة العملية
- المناقشة والمشاركة الايجابية عبر الصف الالكتروني

طرائق التقييم

- جراء الاختبارات النظرية والعملية .
- اعداد التقارير.

د - المهارات العامة والتأهيلية)المنقولة(المهارات الاخرى المتعلقة بقابلية التوظيف والتطور الشخصي.

د1 - التفكير الايجابي وتوظيف المعرفة التي تلقاها.

د2 - القدرة على التخاطب مع الجهات خارج الجامعة والتدريب معهم.

د3 - تمكن الطالب من تدريس المادة التي تعلمها في حال توظيفه في إحد ى المؤسسات التربوية .

د4 - قدرة الطالب في حالة رغبته في تطوير نفسه علميا بالتقديم للدراسات العليا أن يكون مهيئاً للعمل البحثي.

9 بنية المقرر.

الأسبوع	الساعات	مخرجات التعلم المطلوبة	اسم الوحدة / أو الموضوع	طريقة التعليم	طريقة التقييم
1	2	ما معنى البحث العلمي	Meaning of the Scientific Research	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
2	2	الأهداف الرئيسية للبحث العلمي	The main objectives of the Scientific Research	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
3	2	اهم اشكال البحث العلمي	The main form of the Scientific Research	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
4	2	اهم أنواع البحث العلمي	Types of Scientific Research	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
5	2	خصائص البحث العلمي	Research Characteristics	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
6	2	عناصر البحث العلمي	The Elements of Scientific Research	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
7	2	مناهج البحث العلمي	Research Methodology	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
8	2	تصنيف مناهج البحث العلمي	Classification of scientific research methods	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
9	2	المنهج العلمي	Scientific Method	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات

10	2	خطوات البحث العلمي	Steps of Scientific Research	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
11	2	تحديد مشكلة البحث واختيار موضو ع البحث وجمع المادة العلمي ة	Defining the research problem, choosing the research topic, and collecting scientific material	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
12	2	شروط الكتابة الجيدة وخطواتها	Good writing conditions and steps	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
13	2	طريقة كتابة المصادر والاقتباس	How to write references and citations	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
14	2	إخراج البحث العلمي	Producing Scientific Research	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات
15	2	هيكلية مشروع التخرج	Graduation Project Structure	محاضرات تقليدية واسلوب المناقشة عرض تقديمي	المناقشة والاختبارات

10 . البنية التحتية	
1 - الكتب المقررة المطلوبة	- Kumar, Ranjit - Research methodology_ a step-by-step guide for beginners-SAGE Publications (2012). - C.R. Kothari - Research Methodology_ Methods and Techniques-New Age Publications (Academic))2009(.
2 - المراجع الرئيسية)المصادر(	- أساسيات البحث العلمي لطلاب الحاسوب وتقنية المعلومات
أ(الكتب والمراجع التي يوصى بها)المجلات العلمية، التقارير، .....(	
ب(المراجع الالكترونية، مواقع الانترنت. ....،	https://www.youtube.com/watch?v=sE7IbPxJz gw

11 خطة تطوير المقرر الدراسي.

% اعتمادا على المصادر الحديثة. 10تغيير المفردات سنويا بنسبة

Course Description Form

1. Course Name:
System Analysis and Design
2. Course Code:
3. Semester / Year:
2024\ first semester
4. Description Preparation Date:
March 2024
5. Available Attendance Forms:
In presence
6. Number of Credit Hours (Total) / Number of Units (Total)
4 hours weekly\
7. Course administrator's name (mention all, if more than one name)
Name: Raya Nazar Ismail ..... Email: raya_computer@tu.edu.iq
8. Course Objectives

Course Objectives	To make students familiar with the basic concepts of Analysis and Design of information system . ▪ To explain the basic applications of Analysis and Design of information system . ▪ To explore the principles and practice of Design of information system models. ▪ To highlight the technical and social issues of Design of information system.
9. Teaching and Learning Strategies	

Strategy	The main strategy that will be adopted in delivering this module are: 1. Power point presentation (Data show). 2. Explanation on the white board using different color markers. 3. Discussions with the student during teaching. 4. Interaction with students through daily problems practice through lecture. 5. Solve different problems with more exercises. 6. Submit assignment that develop student learning.

10. Course Structure

Week	Hours	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
Week 1	4	Introduction to system analysis & design. System development life cycle	Lecture 1 Lecture 2	Theoretical	Quiz
Week 2	4	Drawing physical data flow diagram . Drawing logical data flow diagram	Lecture 3 Lecture 4	Theoretical	Assignment
Week 3	4	Process specification. Object oriented analysis.	Lecture 5 Lecture 6	Theoretical	Quiz
Week 4	4	Use case modeling. Class diagram.	Lecture 7 Lecture 8	Theoretical	Assignment
Week 5	4	Entity modeling. Use case samples. Software development & testing.	Lecture 9 Lecture 10	Theoretical	quiz

Week 6	4	Design real system as an example to system analysis and design.	Lecture 11	Theoretical	project

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)	System analysis & design,V.,G.&Hoffer,5th
Recommended books and references (scientific journals, reports…)	
Electronic References, Websites	

وصف مادة

اللغة العربية

Module Information معلومات المادة الدراسية					
Module Title	**اللغة العربية**			**Module Delivery**	
Module Type	**غير أساسية**			☒ **Theory** ☒ **Lecture** ☐ **Lab** ☒ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code					
ECTS Credits	**2**				
SWL (hr/sem)	**50**				
Module Level	2		**Semester of Delivery**	1	
Administering Department	**Computer Science**		**College**	**CCSM**	
Module Leader	**احمد محمد الشريف**		**e-mail**	**Bushra.a.salih@st.tu.edu.iq**	
Module Leader's Acad. Title	**Assistant Lecturer**		**Module Leader's Qualification**	**Master**	
Module Tutor	None		**e-mail**	/	
Peer Reviewer Name	**محمد اكثم احمد**		**e-mail**	/	
Scientific Committee Approval Date	6/11/2024		**Version Number**		

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

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	**1_ تقوية القدرة اللغوية لدى الطلاب .** **2_ اكتساب الطالب المعرفة الكاملة لأسس اللغة العربية وإكسابهم مهارة التعبير الصحيح.** **3_ توضيح اهمية القواعد النحوية للغة** **4_ تقوية ملكة الطلاب اللغوية ومعرفة الأخطاء الشائعة في اللغة.** **5_ معرفة القواعد الأساسية والقدرة على استخدامها وتطبيقها.**
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	**1- تعريف الطالب بالمصطلحات ذات الصلة بمفهوم القواعد الإملائية والنحوية .** **2- تعريف الطالب على القواعد النحوية والقدرة على استخدامها.** **3- تعريف الطالب على أهمية الدقة في الملاحظة والتمييز بين الصواب والخطأ فيما يسمعون أو يقرؤون مما يساعدهم على فهم معاني الجمل والأساليب.** **4_ تعريف الطالب بالكلام وما يتألف منه ، ومعرفته بالتمييز بين الفعل والاسم والحرف من خلال عرض علامات كل قسم من أقسام الكلام.** **5_ تعريف الطالب بالممنوع من الصرف.**
Indicative Contents المحتويات الإرشادية	

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	وُضِعت استراتيجية التعلم والتعليم من أجل ان يحصل الطالب على معلومات كاملة تغطي المنهج الدراسي المعد للمادة ولكي تتحقق الغاية الاساسية للمنهج الذي ينصب نحو المام وادراك الطالب بالمفاهيم الاساسية لمادة اللغة العربية ,إذ يتميز هذا المساق بحقيقة أنه يحتاج إلى نهج خاص يعتمد بشكل أساسي على تنمية قدرات الطالب على فهم القواعد النحوية والإملائية وكيفية تطبيقها على النصوص القرآنية والنصوص الشعرية ، وعدم الوقوع في اللحن.

Student Workload (SWL) الحمل الدراسي للطالب			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	32	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	2.2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	18	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	1.1

Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	50

Module Evaluation
تقييم المادة الدراسية

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

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

	Material Covered
Week 1	الكلام وما يتألف منه
Week 2	أقسام الفعل
Week 3	علامات الاسم والفعل
Week 4	التاء المربوطة والتاء المبسوطة
Week 5	الاسم المقصور وتثنيته وجمعه، والاسم الممدود وتثنيته وجمعه
Week 6	كان وأخواتها
Week 7	أقسام كان وأخواتها
Week 8	الأحرف المشبهة بالفعل ومعانيها
Week 9	امتحان نصف الفصل
Week 10	مواضع كسر همزة إن وفتحها
Week 11	الفاعل وأحواله
Week 12	المعرب والمبني
Week 13	الأفعال الخمسة وإعرابها
Week 14	الممنوع من الصرف
Week 15	الأخطاء الشائعة في العربية
Week 16	امتحان نهاية الفصل

Delivery Plan (Weekly Lab. Syllabus): There is no Lab activities المنهاج الاسبوعي للمختبر:	
	Material Covered
Week 1	
Week 2	
Week 3	
Week 4	
Week 5	
Week 6	
Week 7	

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	اللغة العربية العامة للأقسام غير الاختصاص / جمع واعداد ، م.م. بشرى عادل صالح / جامعة تكريت / علوم الحاسوب والرياضيات	No
Recommended Texts	شرح ابن عقيل ، و قطر الندى	No
Websites		

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Compilers1** مترجمات1			**Module Delivery**	
Module Type	**Core**			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code					
ECTS Credits					
SWL (hr/sem)	**125**				
Module Level	3		**Semester of Delivery**	1	
Administering Department	Computer Science		**College**	Computer Science and Mathematics	
Module Leader	**Lecturer. Mohanad Dawood Salman**		**e-mail**	mohanaddawoodalroomi@tu.edu.iq	
Module Leader's Acad. Title	**Lecturer.**		**Module Leader's Qualification**	M.S.C	
Module Tutor	**Assistant Lec.: Luay Ibrahim Klalif**		**e-mail**	luay.i.khalaf@tu.edu.iq	
Peer Reviewer Name	Mohamed Aktham		**e-mail**		
Scientific Committee Approval Date	20/09/2024		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	Basic and advanced programming language Computational theory Computer architecture	**Semester**	1 2 2
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims	8. Student be able to Identify principles, techniques and tools for

أهداف المادة الدراسية	compilers of programming languages. 9. Student be able to design a compiler for a (simplified) (programming) language. 10. Student know how to use compiler construction tools, such as generators of scanners and parsing. 11. Student be familiar with assembly code and virtual machines. 12. Be able to define LL(1), LR(1), and SLR(1) grammars 13. Student be familiar with compiler analysis and optimization techniques.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	**After completing this module, a student should be able to:** 1. Understand the principles guiding all aspects of the compilation process. 2. Recognize the function of each of the fundamental components of a conventional compiler. 3. Demonstrate familiarity with compiler front-end concepts, such as lexical analysis, syntactic analysis, semantic analysis, type checking, symbol tables, and so on. 4. Identify the type of compiler errors that appear during programming and analyze the errors. 5. Review and practical application of the C++ programming language. 6. Finally, the student will build the front-end parts of his or her own compiler using a higher language.
Indicative Contents المحتويات الإرشادية	Indicative content includes the following. <u>Part A - Front End</u> - Compiler phases and overall working, Scanners (Regular Expressions, NFA/DFA, Scanner Generators). [10 hrs] - Parsers (Grammars, Ambiguity Removal, LL, LR, and other deterministic parsing).[10 hrs] - Semantic Analysis (Symbol Tables, Syntax Driven Analysis, Type Systems, Attribute Grammars). [10 hrs] Revision problem classes [5 hrs] <u>Part C –Middle End</u> - Intermediate Forms, Syntax Directed Translation, Translation of: Expressions, Control Structures and Back-patching, Function and Procedure Calls, Runtime Organization. [10 hrs] Revision problem classes [5 hrs] <u>Part B - Back-end</u> - Code optimization: code is made shorter and simpler by combining steps or removing unnecessary steps which lead to the generation of optimized code. . [10 hrs] - Code generation: coverts the intermediate code or the optimized code into the target language. Usually, the target language is the machine code. [5 hrs]

	- Selected, allocated all available memory and registers, then run to process inputs and provide results. [5 hrs]

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	• Using active learning: This type of learning includes engaging learners in active and interactive learning processes, such as discussions, practical experiments, educational games, and knowing the students' scientific level by evaluating their test results. • Software project: Emphasize design patterns, teamwork, and programming methodology by constructing a compiler to meet assigned specifications. • Technology-based learning: includes the use of technology in learning and teaching processes, such as the use of multimedia, educational applications, and online educational platforms. Technology can help improve access to knowledge and enhance interaction and engagement. • Application of theory: Emphasize the role of theory to enable automation of compiler tasks, and illustrate the limitations of that theory.

Student Workload (SWL) الحمل الدراسي للطالب			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	75	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	5
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	50	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	3.3
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	125		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	3	10% (10)	5, 9,12	LO #1-4
	Assignments	5	10% (10)	4,5,8,9,12	LO # 2- 5
	Projects / Lab.	1	10% (10)	Continuous	
	Report	0			
Summative assessment	**Midterm Exam**	2 hr	10% (10)	7	LO # 1-7
	Final Exam	3 hr	60% (60)	16	All

Total assessment	100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

	Material Covered
Week 1	▪ Programing languages. Introduction to compiler.
Week 2	▪ Language-processing system (Compilation). The Structure of Compiler.
Week 3	▪ The phases of compiler design.
Week 4	▪ Lexical analysis phase. Type of errors for Lexical analysis phase
Week 5	▪ Syntax definition. Context free grammar.
Week 6	▪ Parsing tree & Leftmost and Rightmost derivations.
Week 7	▪ Mid-term Exam
Week 8	▪ syntax analysis phase.
Week 9	▪ Problems of compiler for syntax analysis phase. ▪ regular expression and Regular Definitions
Week 10	▪ Top –down parsing .
Week 11	▪ bottom up parsing.
Week 12	▪ Ambiguous Grammars
Week 13	▪ Parser generators.
Week 14	▪ Exam Parser generators.
Week 15	▪ Semantic analysis phase. ▪ Type of errors for semantic analysis.
Week 16	▪ Preparatory week before the final Exam

Delivery Plan (Weekly Lab. Syllabus)

المنهاج الاسبوعي للمختبر

	Material Covered
Week 1 , 2	Lab 1: Array characters in C++ and related functions. String characters in C++ and related functions.
Week 3 , 4	Lab 2: program Lexical analysis phase of second tasks.
Week 5 , 6	Lab 3: program to delete comment in one line. program to delete to comment in more lines. program of compaction of consecutive white spaces into one.
Week 6	Lab 4: program to build the Lexemes table.
Week 7	Mid-term Exam
Week 8 , 9	Lab 5: Programs of transition diagram
Week 10 , 11 , 12 , 13	Lab 6: Program checks keywords. Program checks identifiers. Program check the number. Program checks mathematical operations. Program check the number. Program checks mathematical operations.

Week 14 , 15	**Type Checking**
Week 16	**Preparatory week before the final Exam.**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Basics of Compiler Design, Torben E. Mogensen, 2009.	Yes
Recommended Texts	Compilers , principles , Techniques and tools by Aho,Lam, Sethi and Ullman, 2nd Ed. Addison – Wesely , 2006. من البداية إلى البرمجة الكيانية ، الدكتور المهندس .نضال خضير العبادي C++ .(2011)	Yes
Websites	• http://lectures.iugaza.edu.ps/playlist/145/%D8%AA%D8%B5%D9%85%D9%8A%D9%85-%D8%A7%D9%84%D9%85%D8%AA%D8%B1%D8%AC%D9%85%D8%A7%D8%AA-compiler-design	

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

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

MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Compilers2** مترجمات2			**Module Delivery**	
Module Type	**Core**			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code					
ECTS Credits					
SWL (hr/sem)	**125**				
Module Level		3	**Semester of Delivery**		2
Administering Department		Computer Science	**College**		Computer Science and Mathematics
Module Leader	**Lecturer. Mohanad Dawood Salman**		**e-mail**		mohanaddawoodalroomi@tu.edu.iq
Module Leader's Acad. Title		**Lecturer.**	**Module Leader's Qualification**		M.S.C
Module Tutor	**Assistant Lec.: Luay Ibrahim Khalaf** **Assistant Lec.: Israa Mohammed Ahmed**		**e-mail**		luay.i.khalaf@tu.edu.iq asraa.m.ahmed@tu.edu.iq
Peer Reviewer Name		Mohamed Aktham	**e-mail**		
Scientific Committee Approval Date		20/01/2025	**Version Number**		1.0

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	Basic and advanced programming language Computational theory Computer architecture	**Semester**	1 2 2
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims	14. Student be able to Identify principles, techniques and tools for

أهداف المادة الدراسية	compilers of programming languages. 15. Student be able to design a compiler for a (simplified) (programming) language. 16. Student know how to use compiler construction tools, such as generators of scanners and parsing. 17. Student be familiar with assembly code and virtual machines. 18. Be able to define LL(1), LR(1), and SLR(1) grammars 19. Student be familiar with compiler analysis and optimization techniques.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	**After completing this module, a student should be able to:** 7. Understand the principles guiding all aspects of the compilation process. 8. Recognize the function of each of the fundamental components of a conventional compiler. 9. Demonstrate familiarity with compiler front-end concepts, such as lexical analysis, syntactic analysis, semantic analysis, type checking, symbol tables, and so on. 10. Identify the type of compiler errors that appear during programming and analyze the errors. 11. Review and practical application of the C++ programming language. 12. Finally, the student will build the front-end parts of his or her own compiler using a higher language.
Indicative Contents المحتويات الإرشادية	Indicative content includes the following. <u>Part A - Front End</u> - Compiler phases and overall working, Scanners (Regular Expressions, NFA/DFA, Scanner Generators). [10 hrs] - Parsers (Grammars, Ambiguity Removal, LL, LR, and other deterministic parsing).[10 hrs] - Semantic Analysis (Symbol Tables, Syntax Driven Analysis, Type Systems, Attribute Grammars). [10 hrs] Revision problem classes [5 hrs] <u>Part C –Middle End</u> - Intermediate Forms, Syntax Directed Translation, Translation of: Expressions, Control Structures and Back-patching, Function and Procedure Calls, Runtime Organization. [10 hrs] Revision problem classes [5 hrs] <u>Part B - Back-end</u> - Code optimization: code is made shorter and simpler by combining steps or removing unnecessary steps which lead to the generation of optimized code. . [10 hrs] - Code generation: coverts the intermediate code or the optimized code into the target language. Usually, the target language is the machine code. [5 hrs]

	- Selected, allocated all available memory and registers, then run to process inputs and provide results. [5 hrs]

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	• Using active learning: This type of learning includes engaging learners in active and interactive learning processes, such as discussions, practical experiments, educational games, and knowing the students' scientific level by evaluating their test results. • Software project: Emphasize design patterns, teamwork, and programming methodology by constructing a compiler to meet assigned specifications. • Technology-based learning: includes the use of technology in learning and teaching processes, such as the use of multimedia, educational applications, and online educational platforms. Technology can help improve access to knowledge and enhance interaction and engagement. • Application of theory: Emphasize the role of theory to enable automation of compiler tasks, and illustrate the limitations of that theory.

Student Workload (SWL) الحمل الدراسي للطالب			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	75	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	5
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	50	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	3.3
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	125		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	3	10% (10)	5, 9,12	LO #1-4
	Assignments	5	10% (10)	4,5,8,9,12	LO # 2- 5
	Projects / Lab.	1	10% (10)	Continuous	

	Report	0			
Summative assessment	**Midterm Exam**	2 hr	10% (10)	7	LO # 1-7
	Final Exam	3 hr	60% (60)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

	Material Covered
Week 1	▪ **Intermediate code Generation.**
Week 2	▪ **Three-address code**
Week 3	▪ **Types and declarations, Translation of expressions.**
Week 4	▪ **Type Checking, Control Flow.**
Week 5	▪ **Backpatching. Switch-Statements.**
Week 6	▪ **Intermediate Code for Procedures.**
Week 7	▪ **Mid-term Exam.**
Week 8	▪ **Code Optimization. Examples of code Optimization.**
Week 9	▪ **Gode generation. The target language.**
Week 10	▪ **Addresses in the Target code.**
Week 11	▪ **Basic Blocks and flow Graphs.**
Week 12	▪ **A Simple Code Generator.**
Week 13	▪ **Register Allocation and Assignment.**
Week 14	▪ **Machine – Independent Optimizations.**
Week 15	▪ **Example Code generation.**
Week 16	▪ **Code generation. Examples of code generation.**

Delivery Plan (Weekly Lab. Syllabus)

المنهاج الاسبوعي للمختبر

	Material Covered
Week 1	**Lab 1: Program of intermediate code Generation in c++ language.**
Week 2	**Program of Three-address code in c++ language.**
Week 3	**Structures (struct_array) in c++ language.**
Week 4	**Lexemes table program, Token table program in c++ language.**
Week 5	**Symbol table program in c++ language.**
Week 6	**Increasing and decreasing program in c++ language.**
Week 7	**Mid-term Exam**

Week 8	**Code Optimization program1 in c++ language.**
Week 9	**Code Optimization program2 in c++ language.**
Week 10	**Control Flow program in c++ language.**
Week 11	**Three address code program of loop in c++ language.**
Week 12	**Registration booking program in c++ language.**
Week 13	**Code generation program1 in c++ language.**
Week 15	**Code generation program2 in c++ language.**
Week 16	**Preparatory week before the final Exam.**

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	Basics of Compiler Design, Torben E. Mogensen, 2009.	Yes
Recommended Texts	Compilers , principles , Techniques and tools by Aho,Lam, Sethi and Ullman, 2nd Ed. Addison – Wesely , 2006. من البداية إلى البرمجة الكيانية ، الدكتور المهندس .نضال خضير العبادي C++ .(2011)	Yes
Websites	• http://lectures.iugaza.edu.ps/playlist/145/%D8%AA%D8%B5%D9%85%D9%8A%D9%85-%D8%A7%D9%84%D9%85%D8%AA%D8%B1%D8%AC%D9%85%D8%A7%D8%AA-compiler-design	

Grading Scheme
مخطط الدرجات

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

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

Digital signal processing

Course description

The course description provides a comprehensive overview of its most important features and educational objectives expected of the student. It focuses on evaluating the extent to which the student benefits from the available learning opportunities, by linking it to a description of the academic program as a whole.This link helps understand how the course integrates with the rest of the program's courses, which enhances the student's ability to achieve his educational goals in general.

مدرس المادة: د.م.أ: زيدون القيسي

Organization	Tikrit University / College of Computer Science and Mathematics
Department	Department of Computer Science
Topic	Digital signal processing
Attendance	Attendance weekly
Course	Second Term
No of Hours	30 Theory+30 Practical
Date of preparation	28/1/2024
Course objectives	
The aim of this course is to teach the student the basic topics of signal processing in the time and frequency domains and their uses in processing audio and video signals, in addition to the use of digital filters.	
10 .Course outcomes and teaching, learning and evaluation methods	

A- Cognitive objectives

If the student successfully completes this course, he will be able to:

A1- Understanding and classifying digital signal processing systems.

A2- Understand how to convert an analogue signal to digital.

A3-Understanding the pulse and frequency analysis of intermittent signals.

A4- Design digital filters and study their response.

B - The skills objectives of the course.

B1 - The student should be able to apply engineering-mathematical analyses.

B2 - The ability to identify, formulate and solve engineering problems.

B3- Mastery of the mathematical, basic, and engineering sciences necessary to conduct the analysis and design of electrical engineering systems.

B4- The ability to use systems simulation programs.

Teaching and learning methods

1. Theoretical lectures
2. Practical application in the laboratory of curriculum vocabulary.
3. Using some general engineering principles, which are related to the analysis and design of the engineering problem, in addition to using the laws and rules for signal processing in order to identify the source of the problem and solve it.

Evaluation methods

1. Periodic and quarterly theoretical exams
2. Periodic and quarterly practical exams
3. Quizzes
4. Duties (Homeworks)

C- Emotional and value goals

C1- Realizing the requirements of the engineering profession and ethical responsibility.

C2- Understanding the impact of engineering solutions on economic and environmental activities and the societal context.

C3- Recognizing the need for lifelong learning and the ability to engage in

it.

Teaching and learning methods

- Theoretical lectures
- Practical lectures and practical application in the laboratory
- Group discussions
- Case Study

Evaluation methods Present the results in class to be discussed and the rest of the students can participate in the discussion.

D - Transferable general and qualifying skills (other skills related to employability and personal development.

D1- The ability to identify, formulate and solve engineering problems.

D2- The ability to design and conduct experiments and analyze and interpret data.

D3- The ability to use modern engineering techniques, skills, and tools necessary to practice the engineering profession.

11. The structure of the academic material

Evaluation	Learning Method	Topic	learning outcomes	Hours	WEEK
quiz	Theory+Practical	Introduction to digital signal processing	The student understands the topic	2 Theory 2 Practical	First
quiz	Theory+Practical	Basic elements of DSP, DSP vs. ASP, application of DSP,	The student understands the topic	2 Theory 2 Practical	Second

quiz	Theory+Practical	Continues time signals vs. discrete time signals	The student understands the topic	2 Theory 2 Practical	Third
quiz	Theory+Practical	Discrete time signals and sequences	The student understands the topic	2 Theory 2 Practical	Fourth
quiz	Theory+Practical	Discrete time signals and sequences	The student understands the topic	2 Theory 2 Practical	Fifth
quiz	Theory+Practical	Discrete time signals and sequences	The student understands the topic	2 Theory 2 Practical	Sixth
quiz	Theory+Practical	Standard of discrete time signals (sequences)	The student understands the topic	2 Theory 2 Practical	Seventh
quiz	Theory+Practical	Unit sample sequence, Unit step sequence,	The student understands the topic	2 Theory 2 Practical	Eighth
quiz	Theory+Practical	Unit ramp sequence Exponential sequence.	The student understands the topic	2 Theory 2 Practical	Ninth
quiz	Theory+Practical	(classification of discrete time signals) system properties	The student understands the topic	2 Theory 2 Practical	Tenth

quiz	Theory+Practical	Static and dynamic system, shift invariant and shift variant system,	The student understands the topic	2 Theory 2 Practical	Eleventh
quiz	Theory+Practical	Causal and non-causal system, linear and nonlinear system, stable and unstable	The student understands the topic	2 Theory 2 Practical	Twelvth

References	*Hwei P. Hsu, "Schaum's Outlines of Theory and Problems of Signals and Systems ," McGraw- Hill Companies. *Monson H. Hayes," Schaum's Outline of Theory and Problems of Digital Signal Processing ," McGraw- Hill Companies. *John G. Proakis, Dimitris G. Manolakis ", Digital Signal Processing", 3rd Edition. *Pall A. lynn," Digital signal processing with computer applications", 2nd edition. *John W. Leis,"Digital Signal Processing Using Matlab for Students And Researchers." *Vinay K. Ingle,John G. Proakis," Digital Signal Processing Using MATLAB."
References & Websites	***"Signals and systems Introduction"***, Tutorials Point website, http://www.tutorialspoint.com/dip/signals_and_system_introduction.htm

12. Course development plan
- 1. Familiarity with everything new and innovative in teaching and learning strategies. 2. Providing seminars and student projects regarding digital signal processing

systems

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**The principles of Encryption**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code					
ECTS Credits	6				
SWL (hr/sem)	**148**				
Module Level	3		**Semester of Delivery**	1st	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Khalid khalis ibrahim		**e-mail**	khalid.kh.ibrahim@tu.edu.iq	
Module Leader's Acad. Title	Lecturer		**Module Leader's Qualification**	doctor	
Module Tutor	Humam Neamah Hussein Oqbah Salim Atiyah		**e-mail**	humam.n.hussein@tu.edu.iq oqbah_salim@tu.edu.iq	
Peer Reviewer Name	Dr. Mohamed Aktham		**e-mail**		
Scientific Committee Approval Date	13/09/2025		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	• Understanding the concept of Encryption Algorithms and its applications • Understanding the problem of Encryption Algorithms and problem-solving. • Implementing and employing of Encryption Algorithms to solve problems that are not solved with traditional methods. • Using of Encryption Algorithms (PKE) for solving the complex problems. • Understanding the difference between Encryption Algorithms (PKE) and Encryption Algorithms (DES) • To enable the learner to understand the basics of Encryption Algorithms .
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	• Understand the definition and application of Encryption Algorithms. • Understand the basic syntax and semantics of Python programming language. • Analyze and solve problems using Python programming. • Develop the student's thinking skills and the ability to analyze problems using logical programming principles. • Develop skills in debugging and testing of Encryption Algorithms in Python programming language. • • Providing the learner with the necessary skills to be able to build a learning model.
Indicative Contents المحتويات الإرشادية	• Fundamentals of Encryption Algorithms. • Problems Solving by of Encryption Algorithms. • Classification of Research in Encryption Algorithms • Classical Encryption Algorithms. • Public Key Algorithms(RSA). • Basics of Encryption Algorithms Standard Data(DES)

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain the definition of Encryption Algorithms, its applications, problem-solving techniques, types of Encryption Algorithms in data security, and the basic principles related to Encryption Algorithms. 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 Encryption Algorithms examples, discussing practically how to program these algorithms. 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 Encryption Algorithms projects. 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 AI 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 that students have the necessary resources to complete their lab exercises and assignments effectively. 4. Programming Assignments: Assignments will be given to students to reinforce their understanding of how to program Encryption Algorithms. These assignments may involve implementing algorithms, designing software systems, or developing small-scale projects using Python.

	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 provide support to students who need additional help or guidance in understanding completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	56	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3.7
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	148		

Module Evaluation

تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	
	Assignments	4	20% (20)	7, 12	
	Projects	1	20% (20)	9-14	
	Report	1			
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Basic Terminology Encryption Algorithms
Week 2	Definition of cryptographic terminology
Week 3	Cryptanalysis and Attacks on Cryptosystems
Week 4	Brute-force attack

Week 5	Transposition Ciphers Encryption Algorithms
Week 6	Columnar Transposition Encryption Algorithms
Week 7	Substitution Ciphers Encryption Algorithms
Week 8	Symmetric Encryption Algorithms
Week 9	nditional vs. Computational Security in Encryption Algorithms
Week 10	Types of ciphers in Encryption Algorithms
Week 11	**Midterm Exam**
Week 12	Private Key Encryption (PKE)
Week 13	Public key cryptosystems/ciphers
Week 14	Project Due (Students Presentations part1)
Week 15	Project Due (Students Presentations part2)

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	Material Covered
Week 1	Introduction to the basics of Python programming Language
Week 2	Introduction to the basics of Python programming Language
Week 3	Introduction to the basics of Python programming Language
Week 4	Running Examples of Caesar Cipher
Week 5	Running Examples of Play fair Cipher
Week 6	Running examples of polyalphabetic substitution cipher
Week 7	Running Examples of Rail Fence cipher
Week 8	Running Examples of Private Key Encryption (PKE)
Week 9	Running Examples of the Public key cryptosystems/ciphers
Week 10	Exploring Python Libraries for encryption algorithms
Week 11	Exploring Python Libraries for encryption algorithms
Week 12	Exploring Python Libraries for encryption algorithms
Week 13	Exploring Python Libraries for encryption algorithms
Week 14	Exploring Python Libraries for encryption algorithms
Week 15	Answering students' questions and reviewing previous topics.

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	William Stallings, "Cryptography and Network Security Principles and Practice", fifth edition, Prentice Hall	yes
Recommended Texts	[1]"Cryptographic Algorithms for Protection of Computer Data During Transmission and Dormant Storage," Federal Register 38, No. 93 (May 15, 1973). [2] Data Encryption Standard, Federal Information Processing Standard (FIPS) Publication 46, National Bureau of Standards, U.S. Department of Commerce, Washington D.C. (January 1977). [3] Bruce Schneier, Applied Cryptography, Second Edition, John Wiley & Sons, New York, 1996. [4] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone, Handbook of Applied Cryptography, CRC Press, Boca Raton, 1997	No
Websites		

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information
معلومات المادة الدراسية

Module Title	**Mobile Computing**	**Module Delivery**	
Module Type	Core	☒ **Theory** ☒ **Lecture** **Lab** ☒ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code			
ECTS Credits			
SWL (hr/sem)			
Module Level	3	**Semester of Delivery**	2nd
Administering Department	Computer Science	**College**	CCSM
Module Leader	Qabas Amer Hameed	**e-mail**	Qabas.a.hameed@tu.edu.iq
Module Leader's Acad. Title	Asst.Lecturer	**Module Leader's Qualification**	master
Module Tutor	Mustafa Fadil Latef Suaad Mohammed Shakur	**e-mail**	Mustafa.l.fadhil@tu.edu.iq suaad.shakur@tu.edu.iq
Peer Reviewer Name	Mahammed Aktham	**e-mail**	
Scientific Committee Approval Date		**Version Number**	1.0

Relation with other Modules
العلاقة مع المواد الدراسية الأخرى

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

Module Aims, Learning Outcomes and Indicative Contents
أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية

Module Aims أهداف المادة الدراسية	This course aims to introduce the principles and core concepts of mobile application development. It begins with the fundamentals of mobile platforms, development environments, and user interface design, providing a comprehensive foundation for creating functional and intuitive mobile applications. This course will equip students with the necessary skills to develop and deploy applications for modern mobile operating systems. The curriculum is structured around key topics in mobile development, including mobile architecture, responsive design, app lifecycle management, and integration with cloud services and APIs. By the end of the course, students will know to design, develop, and optimize robust mobile applications tailored to user needs a strong foundation for students to understand the modern eras of computer architecture. The course is structured around different main subject of computer architecture. Those subjects include different parts of computer such as memory, CPU and input output devices.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	**A1 - Understand Mobile Platforms** Explain the architecture, features, and differences between major mobile operating systems (e.g., Android, iOS). **A2- Develop User Interfaces** Design and implement intuitive, responsive, and user-friendly interfaces for mobile applications using industry-standard tools and frameworks. **B1- Master App Lifecycle Management** Demonstrate an understanding of the mobile application lifecycle and effectively manage activities, processes, and transitions within the app. **B2 - Implement Core Mobile Functionality** Learn how to add basic features to mobile apps, like using the camera or GPS, to make them more useful and interactive **C1 -Work with APIs and Cloud Services** Integrate mobile applications with external APIs and cloud services to provide dynamic functionality and data management. **C2 – Design and develop three mobile application projects** Design, develop, and implement three fully functional mobile application projects, demonstrating proficiency in mobile app development tools, frameworks, and best practices while addressing real-world user needs and challenges.
Indicative Contents المحتويات الإرشادية	• Introduction to Mobile Application Development Overview of mobile platforms (e.g., Android, iOS). Differences between native, hybrid, and web applications.

	Setting up the development environment. • User Interface (UI) Design Basics of UI/UX principles for mobile apps. Designing responsive and user-friendly layouts. Handling navigation and input forms. App Lifecycle and Architecture • Understanding the app lifecycle and states. Activity/fragment management (Android) or view controllers (iOS). Introduction to MVC, MVVM, and other architectural patterns. Working with Device Features • Accessing device hardware (e.g., camera, GPS, accelerometer). Managing permissions and security for device features. Notifications and background processes. Data Storage and Management • Using local storage options (e.g., shared preferences, SQLite, Realm). Managing files and caching data. Introduction to cloud storage and databases. • Networking and APIs Fetching data from APIs (e.g., RESTful services, JSON). Sending and receiving data over the internet. Basics of authentication and API security. Cross-Platform Development (Optional) • Introduction to cross-platform frameworks like Flutter or React Native. Advantages and limitations of cross-platform solutions. • Testing and Debugging

	Using debugging tools and simulators. Writing and executing unit tests. Ensuring app stability and performance. Deployment and Publishing • Planning and developing mobile applications. Collaborative app development with version control systems (e.g., Git). Presentation and evaluation of completed projects.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	- Interactive Lectures - Dialogue and discussion - Brainstorming - Solving problems - Projects, tasks and costs projects - Self-learning - Learning Cooperative - Exchanging experiences among colleagues.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ أسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل		**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل		**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	60		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	2	2	5, 11	#LO 1-3, #LO 5-8
	Assignments	1	2	7, 12	#LO 3-5, #LO 5-8
	Projects	1	2	continuous	
	Report	1	2	14	#LO 1-8
Summative assessment	Midterm Exam	2 hr	30	11	#LO 1-7
	Final Exam	2 hr	60	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Introduction to Flutter & Dart Basics
Week 2	Widgets & Basic UI
Week 3	Advanced Widgets & UI
Week 4	User Interaction & Navigation
Week 5	State Management Basics
Week 6	Exam
Week 7	Project 1: To-Do List
Week 8	Project 1: To-Do List
Week 9	Project 2: E-commerce
Week 10	Project 2: E-commerce
Week 11	Networking & APIs
Week 12	REST API with Flutter
Week 13	Exam
Week 14	Project 3: Fetching data with API
Week 15	Reviewing Students' Projects

Delivery Plan (Weekly Lab. Syllabus):
المنهاج الاسبوعي للمختبر:

Week 1	**Write simple Dart programs to solidify syntax**
Week 2	**Create a simple app with text, buttons, and images.**
Week 3	**Navigating between screens using Navigator**
Week 4	**Loading dynamic content (e.g., lists of items)**
Week 5	**Build a scrolling list of items with images and text.**
Week 6	**Using ListView and GridView**
Week 7	**Exam**
Week 8	**Project 1: To-do list basic implantation**
Week 9	**Project 1: To-do list full implantation**
Week 10	**Exam**
Week 11	**Project 2: E-commerce basic implantation**
Week 12	**Project 2: E-commerce full implementation**
Week 13	**Build an app that fetches and displays data from a public API.**
Week 14	**Project 3: Fetching data with API basic implantation**
Week 15	**Project 3: Fetching data with API Full implantation**

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	**Beginning Flutter: A Hands On Guide to App Development**	No
Recommended Texts	**Learn Google Flutter Fast 65 Example Apps**	No
Websites		

Grading Scheme
مخطط الدرجات

Group	Grade	التقدير	Marks (%)	Definition
Success Group (50 - 100)	**A** - Excellent	امتياز	90 - 100	Outstanding Performance
	B - Very Good	جيد جدا	80 - 89	Above average with some errors

	C - Good	جيد	70 - 79	Sound work with notable errors
	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings
	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information
معلومات المادة الدراسية

Module Title	**Operating System 1**	**Module Delivery**	
Module Type	Core	☐ **Theory** ☒ **Lecture** ☒ **Lab** ☒ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	TU010101107		
ECTS Credits	8		
SWL (hr/sem)	**200**		

Module Level	3	**Semester of Delivery**	2st
Administering Department	Computer Science	**College**	CCSM
Module Leader	Ayad Ahmed Yass	**e-mail**	ayad.a.yass@tu.edu.iq
Module Leader's Acad. Title	Lecturer	**Module Leader's Qualification**	Phd
Module Tutor		**e-mail**	
Peer Reviewer Name	Mahammed Aktham	**e-mail**	
Scientific Committee Approval Date	00/00/	**Version Number**	1.0

Relation with other Modules
العلاقة مع المواد الدراسية الأخرى

Prerequisite module	None	**Semester**	
Co-requisites module	Understanding basics of Operating system	**Semester**	2

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	1. To describe the basic organization of computer systems 2. To describe the services an operating system provides to users, processes, and other systems. 3. To develop students' understanding skills for various operating system 4. To help students to deal with computer components. 5. To provide students with hands-on experience in Linux command line 6. To advance the students skills to working with scripting to deal with operating system. 7. To enhance students' ability to troubleshooting operating system problems. 8. To level up the students' skills so they be able to enter the market places since the OS consider the most required in most organizations
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Upon successful completion of this module, students should be able to: 1. Demonstrate a solid understanding of the fundamental principles and concepts of operating system. 2. to help the student to understand the components and security of the operating system. 3. At this point, the student will be able to deal with different operating system 4. Enhance the operating system components and avoid the security issues 5. Student able to work with Linux command as the Linux is the most required in the market nowadays 6. Less work and increase productivity by learning the scripting 7. fixing and troubleshooting all OS related issues 8. leveling up the skills means now the students can enter any market place without hesitating
Indicative Contents	

المحتويات الإرشادية	1- To describe the basic organization of computer systems. 2- To describe the services an operating system provides to users, processes, and other systems 3- To discuss the various ways of structuring an operating system 4- To introduce the notion of a process and a thread 5- To introduce CPU scheduling, which is the basis for multi programmed operating systems 6- To develop a description of deadlocks

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain operating concepts, like processes, threads, memory management, file systems, and system calls. 2. Interactive Discussions: Engaging students in interactive discussions allows them to ask questions, seek clarifications, and participate actively in the learning process. Discussions can include reviewing code examples, discussing database best practices, and exploring real-world applications of operating system 3. Laboratory Sessions: Laboratory sessions are dedicated practical sessions where students apply the concepts learned in lectures to hands-on database exercises. Key strategies for the laboratory sessions include: a. Linux Exercises: Students will work on linux exercises and projects in the laboratory, providing them with practical experience in command line and problem-solving. b. Guided Practice: Lab instructors or teaching assistants will be available to provide guidance, assistance, and immediate feedback on students' skills They can help students debug their scripting, identify errors, and improve their command line skills. c. Collaboration and Peer Learning: Students can collaborate with their peers in the laboratory, fostering teamwork and enabling knowledge sharing. Working together on Linux 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 that students have the necessary resources to

	complete their lab exercises and assignments effectively. 4. command line Assignments: Assignments will be given to students to reinforce their understanding of command line concepts and encourage independent problem-solving. These assignments may involve implementing scripting. 5. Reviews and Feedback: The instructor will provide feedback on students' scripting, 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 provide support to students who need additional help or guidance in understanding command concepts or completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ أسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	93	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	6.2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	107	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	7.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	200		

Module Evaluation

تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	#LO 1-3, #LO 5-8
	Assignments	2	10% (10)	7, 12	#LO 3-5, #LO 5-8
	Projects	1	10% (10)	continuous	
	Report	1	10% (10)	14	#LO 1-8
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	#LO 1-7
	Final Exam	2 hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)

المنهاج الاسبوعي النظري	
Week No.	**Material Covered**
Week 1	Introduction to operating system
Week 2	Computer-System Architecture Multiprocessing Architecture
Week 3	Multiprocessing Architecture
Week 4	Operating system services
Week 5	System Program
Week 6	Process 1
Week 7	Process 2
Week 8	Threads 1
Week 9	Threads 2
Week 10	CPU Scheduling 1
Week 11	CPU Scheduling 2
Week 12	Scheduling algorithms
Week 13	Scheduling Algorithms tutorials 1
Week 14	Scheduling Algorithms tutorials 2
Week 15	Reviewing Students' Projects

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	**Material Covered**
Week 1	**Introduction about the OS to be used in the lab and why we are using it**
Week 2	**Differences between Linux and other operating system**
Week 3	**Explain about the App that using Linux**
Week 4	**Installing Microsoft SQL Server Management Studio**
Week 5	**Installing Linux**
Week 6	**How to use Linux remotely**
Week 7	**Linux basic command line1**
Week 8	**Linux command line 2**
Week 9	**Linux command line 3**
Week 10	**Installing software in Linux**

Week 11	**Understanding Linux logs**
Week 12	**Identifying issues from the logs**
Week 13	**Troubleshooting issues 1**
Week 14	**Troubleshooting issues 2**
Week 15	**Applying instructor's feedback on students' projects**

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	*Abraham* Silberschatz (2022). Operating System Concept nine edition.	Yes
Recommended Texts		
Websites		

Grading Scheme
مخطط الدرجات

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

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

MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Web programming (2)**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☒ **Practical** ☐ **Seminar**	
Module Code	TUCS				
ECTS Credits	6				
SWL (hr/sem)	**150**				
Module Level	3		**Semester of Delivery**	1st	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Yahya Layth Khaleel		**e-mail**	yahya@tu.edu.iq	
Module Leader's Acad. Title	Assistant Lecturer		**Module Leader's Qualification**	master	
Module Tutor			**e-mail**		
Peer Reviewer Name	Harith Abdullah		**e-mail**		
Scientific Committee Approval Date	2024-2025		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims	8- Understand the principles of creating an effective web page, including an

أهداف المادة الدراسية	in-depth consideration of information architecture. 9- Develop skills in analyzing the usability of a web site. 10- Understand how to plan and conduct user research related to web usability. 11- Learn the core web technologies and programming languages that power the modern web. Starting with HTML and CSS and Javascript. 12- Learn some concepts in server-side programming using (PHP), along with MySQL database. 13- - Exploring how to create dynamic web applications that can process user data, store information, using content management systems (CMS).
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Upon successful completion of this module, students should be able to: 6. Understand the fundamental concepts of web development: Gain a solid understanding of how the web works, including client-server architecture, HTTP protocols, and the role of web browsers. 7. Develop proficiency in HTML and CSS: Acquire the skills to create well-structured web pages using HTML markup, apply CSS styles for layout and design, and create responsive and visually appealing websites. 8. Understand JavaScript programming: Learn the essentials of JavaScript programming, including variables, data types, and operators. 9. Build dynamic web applications: Explore server-side programming using PHP. Understand how to process user data, connect to databases, and generate dynamic content. 10. Work with databases: Gain familiarity with database management systems like MySQL. Learn how to design and create database schemas. 11. Implement security measures: Understand common web security vulnerabilities and learn techniques to protect web applications from attacks. 12. Problem-solving and debugging skills: Develop the ability to identify and fix common issues in web development through effective troubleshooting and debugging techniques.
Indicative Contents	

المحتويات الإرشادية	1- Introduction to Web Development • Overview of web technologies and standards • Client-server architecture and HTTP protocols • Understanding web browsers and their rendering engines • HTML Fundamentals 2- Structure of HTML • HTML tags and elements • Working with text, images, links, and lists • Creating forms for user input • Semantic HTML and accessibility best practices 3- CSS Styling • CSS syntax and selectors • Box model and layout techniques • Applying styles to text, colors, backgrounds, and borders • CSS positioning and responsive design • CSS frameworks and libraries 4- JavaScript Basics • Introduction to JavaScript and its role in web development • Variables, data types, and operators 5- Server-Side Programming • Introduction to server-side programming languages (PHP) • Handling user input and form data • Working with databases (MySQL) • Templates for dynamic content • Session management and user authentication 6-Web Security • Common web security vulnerabilities • Guidelines and directions to protect the website

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	10. Hands-on Practice: This Encourage students to actively engage in hands-on coding exercises and projects. Providing them with opportunities to apply theoretical concepts in practical scenarios, allowing them to gain proficiency through practice.

11. Project-Based Learning: Assigning projects that simulate real-world web development scenarios. This approach allows students to apply their knowledge and skills to create fully functional web applications, reinforcing their understanding and problem-solving abilities.

12. Code Review and Feedback: Incorporating code review sessions where students can share their code and receive constructive feedback. This process helps students identify areas for improvement, learn best practices, and enhance their coding style and techniques.

13. Collaborative Learning: Foster a collaborative learning environment where students can work together on group projects or problem-solving tasks. Encourage peer-to-peer discussions, code sharing, and knowledge exchange, as this can enhance understanding and expose students to diverse perspectives and solutions.

14. Online Resources and Documentation: Introduce students to reputable online resources, documentation, and tutorials related to web development. Teaching students how to effectively search for solutions, read and understand documentation, and leverage online communities and forums for support and learning.

15. Practical Examples and Case Studies: Provide practical examples and case studies that demonstrate the application of web programming concepts in real-world scenarios. This helps students relate theoretical concepts to practical use cases, enhancing their understanding and problem-solving abilities.

16. Regular Assessments and Feedback: Conducting regular assessments, quizzes, and coding challenges to evaluate students' progress and understanding. Providing timely feedback to help students identify their strengths and areas that require improvement.

17. Continuous Learning and Exploration: Encouragement students to stay updated with the latest trends, tools, and technologies in web development.

18. Office Hours and Individual Support: The instructor should be available for individual consultations and provide support to students who need additional help or guidance in understanding programming concepts or completing assignments.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	77	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	4
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	73	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	4.8
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	150		

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	2	10% (10)	5, 11	
	Assignments	2	10% (10)	3, 12	
	Projects	1	10% (10)	8-14	
	Report				
Summative assessment	**Midterm Exam**	2 hr	20% (20)	11	
	Final Exam	2hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
Week No.	**Material Covered**
Week 1	Introduction web programming
Week 2	Internet and Intranet, Web application ,web page, website , Classifying websites
Week 3	Client side script and server side scrip, Introduction to HTML
Week 4	HTML tags and attributes
Week 5	HTML - Titles and headings
Week 6	HTML – Lists
Week 7	HTML Images
Week 8	HTML – Tables
Week 9	HTML Frame and Form
Week 10	Introduction to CSS

Week 11	Internal and external CSS
Week 12	Introduction to JavaScript
Week 13	JavaScript Variables, data types, and operators
Week 14	Server-Side Programming (PHP with MySQL)
Week 15	Content Management Systems (CMS), Security vulnerabilities

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	**Material Covered**
Week 1	**HTML tags and attributes**
Week 2	**HTML – Titles, headings and Lists**
Week 3	**HTML Images**
Week 4	**HTML – Tables**
Week 5	**HTML Frame and Form**
Week 6	**Internal CSS**
Week 7	**External CSS**
Week 8	**Creating web page structure using HTML tags**
Week 9	**JavaScript variables definitions, data types**
Week 10	**JavaScript operators**
Week 11	**Implementing interactivity and dynamic content on web pages**
Week 12	**Install WAMP server**
Week 13	**Work with MySQL database**
Week 14	**Content Management Systems (CMS): install Joomla**
Week 15	**Create dynamic website**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	None	

Recommended Texts	Jon, Duckett. "HTML and CSS: Design and Build Websites." (2016).	
Websites		

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

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

Course Description Form .....

1. Course Name: Management principles
2. Course Code: Management principles
3. Semester / Year: The first / 2023-2024
4. Description Preparation Date: 31/3/2024
5. Available Attendance Forms: Blended learning (in-person + electronic)
6. Number of Credit Hours (Total) / Number of Units (Total)
30 hours / 2 units
7. Course administrator's name (mention all, if more than one name)

Lecturer Kamal Majeed Wahab / Lecturer Mahmoud Hussein Alwan
Email: kamal.m.wahhab@tu.edu.iq
Mahmood3333@tu.edu.iq

8. Course Objectives

Course Objectives	
- Learn management functions and... - Access to ancient and modern management schools and their theories - Studying the organization's general and specific external environment	

9. Teaching and Learning Strategies

Strategy	

10. Course Structure

Week	Hours	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
1	2	Learn about the concept of management and the manager and their functions	The concept of management and the manager and their functions	Explanation, dialogue and discussion	Direct questions and daily and semester exams
2	2	Identify the types of managers and their levels	Types of managers and their levels	Explanation, dialogue and discussion	Direct questions and daily and semester exams
3	2	Identify the manager's administrative skills, their sources and roles	The manager's administrative skills, their sources and roles	Explanation, dialogue and discussion	Direct questions and daily and semester exams

4	2	Getting to know management schools and their theories	Management schools and their theories	Explanation, dialogue and discussion	Direct questions and daily and semester exams
5	2	Learn about the organization's environment	Organization environment	Explanation, dialogue and discussion	Direct questions and daily and semester exams
6	2	Learn about planning and decision making	Planning and decision making	Explanation, dialogue and discussion	Direct questions and daily and semester exams
7	2	Identify the types of plans and their obstacles	Types of plans and their obstacles	Explanation, dialogue and discussion	Direct questions and daily and semester exams
8	2	Recognizing information and making decisions	Information and decision making	Explanation, dialogue and discussion	Direct questions and daily and semester exams
9	2	Learn about the administrative organization	Administrative regulation	Explanation, dialogue and discussion	Direct questions and daily and semester exams
10	2	Learn about leadership, motivation and job satisfaction	Leadership, motivation and job satisfaction	Explanation, dialogue and discussion	Direct questions and daily and semester exams

11	2	Identify communications, their types, methods, and channels	Communications, their types, methods and channels	Explanation, dialogue and discussion	Direct questions and daily and semester exams
12	2	Learn about monitoring and evaluation	Monitoring and evaluation	Explanation, dialogue and discussion	Direct questions and daily and semester exams
13	2	Learn about the social responsibility of a business organization	Business organization ethics	Explanation, dialogue and discussion	Direct questions and daily and semester exams

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)	
Recommended books and references (scientific journals, reports…)	
Electronic References, Websites	

نموذج وصف المقرر

وصف المقرر لمادة قواعد البيانات 1

أ.م. د. سعدي حمد ثلج /قسم علوم الحاسوب

يوفر وصف المقرر هذا إيجازاً مقتضياً لأهم خصائص المقرر من تمارين وتطبيقات عملية ومفاهيم اساسية ولابد من الربط بينها وبين وصف .. ومشاريع عملية في نطاق تراكيب البيانات والبرمجةdatabase1في البرنامج.

2. المؤسسة التعليمية	جامعة تكريت
3. القسم العلمي / المركز	علوم الحاسوب
4. اسم / رمز المقرر	قواعد بيانات 1
5. أشكال الحضور المتاحة	محاضرات ،مختبر
6. الفصل / السنة	الفصل الدراسي الاول 2023-2024
7. عدد الساعات الدراسية (الكلي)	4* ساعات = 60 كفصل دراسي 15
8. تاريخ إعداد هذا الوصف	2023/9

9. أهداف المقرر
اطلاع الطالب مفاهيم قواعد البيانات واستخدامها لمعالجة واسترجاع البيانات باستخدام لغات برمجية تدعىSQL Server
تدريب الطالب على استخدام وتصميم قواعد البيانات في حل المسائل
معرفة طرق تجسيد قواعد البيانات (الاستعلام والجمل البرمجية المستخدمة لبناء النظام)
الالمام باساسيات البيانات وقواعد البيانات

10. مخرجات المقرر وطرائق التعليم والتعلم والتقييم
أ- الاهداف المعرفية : أ.1- وصف البيانات. أ.2- تسجيد واستخدام قواعد البيانات المبينة وتلك التي يعرفها المستخدم أ.3- مقارنة طرق المختلفة لقواعد البيانات من حيث كفاءة الاداء للنظام أ.4- تجسيد مفاهيم البنى الاساسية لادارة قواعد البيانات
ب - الأهداف المهاراتية الخاصة بالمقرر. ب.1- مهارات ادائية عن طريق اشراك الطالب بالمحاضرة ب.2- كيفية التعامل مع فكرة بناء النظام واعطاء امثلة توضيحية عملية ب.3- تطبيق المحاضرة من قبل الطلبة
طرائق التعليم والتعلم
• التطبيقات البرمجية في المختبر على بيئة لغة SQL Server • طريقة المحاضرة • تكليف الطالب بحل اسئلة اسبوعية • عرض الموضوع مدعم بالامثلة التفصيلية
طرائق التقييم
• الاختبارات العملية • الاختبارات النظرية • الواجبات • الامتحانات اليومية المفاجئة واختبار فهم الطالب اثناء المحاضره

ج- الأهداف الوجدانية والقيمية ج.1- استثمار ماتعلمه الطالب في المرحلة السابقة عن اسلوب تحليل النظام ج.2- تفهم طريقة حفظ وتغيير وبحث واسترجاع البيانات في قاعدة البيانات
طرائق التعليم والتعلم
• المحاضرات والتمارين والمناقشات • التطبيقات العملية المختبرية
طرائق التقييم
1. الاختبارات العملية 2. الاختبارات النظرية 3. الواجبات
د - المهارات العامة والتأهيلية المنقولة (المهارات الأخرى المتعلقة بقابلية التوظيف والتطور الشخصي). د.أ- قابلية استخدام الحواسيب في استرجاع البيانات د.ب- فهم ومتابعة البرمجيات المستخدمة

11. بنية المقرر

الأسبوع	الساعات	مخرجات التعلم المطلوبة	اسم الوحدة / أو الموضوع	طريقة التعليم	طريقة التقييم
الاول	4	Introduction to database and database systems:	Introduction to dat and database and Types of Data in database	شرح وتوضيح	مناقشة
الثاني	4	Architecture of Database System	Structure of System and Levels of System	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
الثالث	4	Process of System and Query	Process data in system and query strategy	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
الرابع	4	Data Models, Schemas, and Instances	Models of database and instances	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
الخامس	4	Three-Schema Architecture and Data Independence	Schemas database and independence of data	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	حل امثلة
السادس	4	The Database System Environment	System Environment	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
السابع	4	Centralized and Client/Server Architectures for DBMSs and Classification of Database Management Systems	Distributed data in system	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
الثامن	4	The Relational Data Model and Relational Database Constraints	Type of database system model relational database and using in SQL Server	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
التاسع	4	Relational Model Constraints and Relational Database	Using constraints in tables	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات

			Schemas		
العاشر	4	Update Operations, Transactions, and Dealing with Constraint Violations	Statement in DDL operatin and transactions types	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
الحادي عشر+ الثاني عشر	4	Unary Relational Operations: SELECT and PROJECT	Using relational algebra DML Languages and statements DML	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
الثالث عشر+ الرابع عشر	4	Relational Algebra Operations from Set Theory	Relational Algebra commands	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
الخامش عشر	4	Examples in relational algebra	Three examples unary statements and other commands	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات وحل امثله

12. البنية التحتية	
1- الكتب المقررة المطلوبة	Fundamentals_of_Database_Systems,_6th_Edition
2- المراجع الرئيسية (المصادر)	Basic Sql Statements

ا- الكتب والمراجع التي يوصى بها (المجلات العلمية , التقارير ,....)	
ب - المراجع الالكترونية, مواقع الانترنيت	https://www.w3schools.com/

13. خطة تطوير المقرر الدراسي
استبدال واضافة نسبة من التمارين والتطبيقات والنظري سنويا .

نموذج وصف المقرر

وصف المقرر لمادة قواعد البيانات2

أ.م. د. سعدي حمد ثلج /قسم علوم الحاسوب

يوفر وصف المقرر هذا إيجازاً مقتضياً لأهم خصائص المقرر من تمارين وتطبيقات عملية ومفاهيم اساسية ولابد من الربط بينها وبين . . ومشاريع عملية في نطاق تراكيب البيانات والبرمجةdatabase2في وصف البرنامج.

10. المؤسسة التعليمية	جامعة تكريت
11. القسم العلمي / المركز	علوم الحاسوب
12. اسم / رمز المقرر	قواعد البيانات 2
13. أشكال الحضور المتاحة	محاضرات ،مختبر
14. الفصل / السنة	الفصل الدراسي الثاني 2023-2024

15. عدد الساعات الدراسية (الكلي)	*4 ساعات = 60 كفصل دراسي 15
16. تاريخ إعداد هذا الوصف	2024/1/3
17. أهداف المقرر	
اطلاع الطالب مفاهيم قواعد البيانات وتنظيم البيانات وحوسبتها وكيفية التعامل معها وبناءها ووطريقة استرجاعها وتحديثها وتوزيعها على عدة مواقعة لسهولة التعامل معها والاستفادة منها باكثر من مكان وهذا كلة بتم باستخدام لغات برمجية عدة ومنها اللغة التي نستخدمها في منهجنا و تدعى SQL Server	
تدريب الطالب على طريقة بناء قاعدة البيانات واستخدام وتصميم قواعد البيانات في حل المسائل	
معرفة طرق تجسيد قواعد البيانات (الاستعلام والجمل البرمجية المستخدمة لبناء النظام)	
الالمام باساسيات البيانات وقواعد البيانات	

14. مخرجات المقرر وطرائق التعليم والتعلم والتقييم
ب- الاهداف المعرفية : أ.1- وصف البيانات. أ.2- تسجيد واستخدام قواعد البيانات المبينة وتلك التي يعرفها المستخدم أ.3- مقارنة طرق المختلفة لقواعد البيانات من حيث كفاءة الاداء للنظام أ.4- تجسيد مفاهيم البنى الاساسية لادارة قواعد البيانات
ب - الأهداف المهاراتية الخاصة بالمقرر. ب.1- مهارات ادائية عن طريق اشراك الطالب بالمحاضرة ب.2- كيفية التعامل مع فكرة بناء النظام واعطاء امثلة توضيحية عملية ب.3- تطبيق المحاضرة من قبل الطلبة
طرائق التعليم والتعلم
• التطبيقات البرمجية في المختبر على بيئة لغة SQL Server • طريقة المحاضرة • تكليف الطالب بحل اسئلة اسبوعية • عرض الموضوع مدعم بالامثلة التفصيلية
طرائق التقييم

- الاختبارات العملية
- الاختبارات النظرية
- الواجبات
- الامتحانات اليومية المفاجئة واختبار فهم الطالب اثناء المحاضره

ج- الأهداف الوجدانية والقيمية

ج.1- استثمار ماتعلمه الطالب في المرحلة السابقة عن اسلوب تحليل النظام

ج.2- تفهم طريقة حفظ وتغيير وبحث واسترجاع البيانات في قاعدة البيانات

طرائق التعليم والتعلم

- المحاضرات والتمارين والمناقشات
- التطبيقات العملية المختبرية

طرائق التقييم

4. الاختبارات العملية
5. الاختبارات النظرية
6. الواجبات

د - المهارات العامة والتأهيلية المنقولة (المهارات الأخرى المتعلقة بقابلية التوظيف والتطور الشخصي).

د.أ- قابلية استخدام الحواسيب في استرجاع البيانات

د.ب- فهم ومتابعة البرمجيات المستخدمة

15. بنية المقرر

الأسبوع	الساعات	مخرجات التعلم المطلوبة	اسم الوحدة / أو الموضوع	طريقة التعليم	طريقة التقييم
الاول	4	Introduction to relational databa	Introduction totypes of databases	شرح وتوضيح	مناقشة
الثاني	4	Unary Relational Operations: SELECT and PROJECT	SELECT Statement in Relational Algebra	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
الثالث	4	Unary Relational Operations: SELECT and PROJECT	PROJECT Statement in Relational Algebra	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
الرابع		THE ENHANCED ENTITY-RELATIONSHIP (EER) MODEL	تحسين عملية الترابط بين الكيانات في الجداول	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
الخامس		RELATIONAL DATABASE DESIGN BY ER- AND EER-TO-RELATIONAL MAPPING	التصميم العلائقي لقواعد البيانات	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
السادس		SQL-99: SCHEMA DEFINITION, CONSTRAINTS, QUERIES, AND VIEWS	شرح الاستعلام وشكل الاستعلام وتهيئة الطالب للدخول في فهم الجمل البرمجية للغة SQL SERVER	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
السابع		INTRODUCTION TO SQL PROGRAMMING TECHNIQUES	مقدمة عن لغة برمجة قواعد البيانات SQL SERVER	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
الثامن		FUNCTIONAL DEPENDENCIES AND NORMALIZATION FOR RELATIONAL DATABASES	شرحة عمليات التطبيع في قواعد البيانات واعتمادية الطبقات في بنية قواعد البيانات على بعضها	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
التاسع		:RELATIONAL DATABASE DESIGN ALGORITHMS AND FURTHER DEPENDENCIES	مفهوم تصميم قواعد البيانات العلائقية	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
العاشر	4	Relational Algebra Operations from Set Theory	UNION Statement in relational algebra	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
الحادي عشر	4	Relational Algebra Operations from Set Theory	INTERSECTION Statement in relational algebra	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية

الثاني عشر	4	Relational Algebra Operations from Set Theory	MINUS Statement in relational algebra	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
الثالث عشر	4	Binary Relational Operations: JOIN and DIVISION	JOIN Statement in relational algebra	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
الرابع عشر	4	The Relational Data Model andhgjhsu uav Relational Database Constraints	DIVISION Statement in relational algebra	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات حل تمارين الكتاب للجمل البرمجية
الخامس عشر	4	Additional Relational Operations	Generalized projection and Aggregate functions and grouping	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
السادس عشر	4	OUTER JOIN Operations Operations, Transactions, and Dealing with Constraint Violations	Statement operation and transactions	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
السابع عشر	4	Unary Relational Operations: SELECT and PROJECT	Using relational algebra DML Languages and statements DML	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات

الثامن عشر	4	Relational Algebra Operations from Set Theory	Relational Algebra commands	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات
التاسع عشر	4	Examples of Queries in Relational Algebra	examples statements and other commands	شرح وعرض الموضوع مدعم بالامثلة التفصيلية	مناقشة وواجبات وحل امثله من الكتاب

16. البنية التحتية	
1- الكتب المقررة المطلوبة	Fundamentals_of_Database_Systems,_6th_Edition
2- المراجع الرئيسية (المصادر)	Basic Sql Statements
ا- الكتب والمراجع التي يوصى بها (المجلات العلمية , التقارير ,....)	
ب - المراجع الالكترونية, مواقع الانترنيت	https://www.w3schools.com/

17. خطة تطوير المقرر الدراسي
استبدال واضافة نسبة من التمارين والتطبيقات والنظري سنويا .

Course Description

Computer security is one of the susceptible topics because of the importance of this device, as it contains private information or provides access to people for spying, etc. Therefore, this topic is one of the matters closely related to the ABCs of the student's specialization as a programmer. Therefore, it has become part of the curriculum scheduled for this stage. The fourth in the Department of Computer Science is an explicit curriculum because it contains the basics of important computer security and software applications that develop students' mental abilities and give them many basic concepts that enrich their cognitive and programming requirements.

Subject teacher: Lecturer . Dr. Moceheb Lazam Shuwandy

. Educational Institution/ college	Tikrit University/College of Computer Science and Mathematics
2. Scientific Department/Center:	Computer Science Department
3. Course name/code:	Computer Security
4. Available attendance formats:	Weekly
5. Course /year:	Second Course 2023/2024
6. Number of study hours (total):	30 theoretical hours and 30 practical hours

18. Course objectives

The course aims to teach the student the different ways to defend computer security from various attacks such as viruses and worms, in addition to Learn about different software protection methods, such as:

- Firewall
- Prevent Intrusion System
- Detection Intrusion System .and others

In addition to some protocols to ensure the reliability and credibility of both the sender and the recipient.

19. Course Outcomes and Teaching, Learning and Evaluation Methods

A- Knowledge objectives

A1- Know the basics of computer security work.
A2- Knowledge and understanding of computer security functions.
A3- Knowledge of computer security programming Data protection.
A4- Knowing the vulnerabilities, some attack methods, and how to deal with them.

B - The skills objectives of the course

The students responded clearly to the subject through the course teacher's evaluation as a result of the students' interaction during the explanation of the topics to be taught and through their effective contribution in using computer security programs and the ability to apply them.

B 1 - Enabling the student to choose the best ways to protect the computer.
B2 - Teaching students about the dangers that exist on the Internet.

Teaching and learning methods

Theoretical Lectures

Practical Lectures

Evaluation methods

1 . Direct questions during the theoretical lecture
2 .Daily exams in each lecture on the subject of the previous lecture
3 .Homework assignments and reports
4. Monthly exams

C- Expressive and Value Objectives

C-1 Asking them in the lecture and assigning them to search for the answer by searching on the Internet.
C-2 Do not transfer solutions between all groups of students by changing tasks from one group to another.
C-3 Pushing the student to commit to attending theoretical lectures by taking daily exam.

20. Course structure

Number of teaching hours		Syllabus Vocabulary	Week
Practical	Theoretical		
2	2	What is computer security? Introduction to CIA security objectives and computer security challenges and a review of the practical aspect	.1
2	2	Concepts of threats, vulnerabilities, and attacks, and conducting practical experiments on the computer	.2
2	2	How to secure access to resources - Two-step authentication: identification of the access requester and authorization to grant or deny access. And apply it programmatically	.3
2	2	How to secure communications over a computer network - in three steps:	.4
2	2	First: Confidentiality by preventing intercepted communications from being understood. Explain this practically	.5
2	2	Second: Authentication by creating an identification of the sender's identity. Explaining physical security and authentication models. Learn about PKI and encryption protocols.	.6
2	2	Third: Clarifying the integrity of the information by proving that communications have not been tampered with.	.7
2	2	General access control techniques: something you have, something you know, something you are. Access control methods and models and how to apply them in practice.	.8
2	2	Advantages and disadvantages of passwords. Programming examples implemented in the laboratory.	.9
2	2	ACL and C-list identification declaration	.10
2	2	Explain the concepts of: spyware, advertising-supported software, malware (viruses and worms), Trojan horses, logic bombs... etc.	.11
2	2	Defensive measures: firewall and intrusion detection system. And methods of applying it	.12
2	2	Methods for checking and removing viruses	.13
2	2	Methods for checking and removing malware	.14
2	2	Security applications: e-commerce security, SSL/TLS, virtual private networks (VPN), web security	.15

21. Infrastructure

1- Required prescribed books	Nothing
2- Main references (sources)	- Network Security Essentials: Applications & Standards, William S., Pearson Education Asia. - CompTIA security+ - David L. Prowse, Pearson USA 4th Edition, 2019.

3- Recommended books and references (scientific journals, reports …..)	- Database Security Mechanisms for Computer Network-Sead Muftic, John Wiles . - Designing Security Architecture Solutions -Jay Ramachandran, Wiley dream tech - Security in Computer Operating System - G. O.Shea , NCC Blackwell Manchester Oxford

Course Description Form

1. Course Name:
Digital Image processing
2. Course Code:
3. Semester / Year:
Fourth/first semester
4. Description Preparation Date:
1/4/2024
5. Available Attendance Forms:
Full attendance
6. Number of Credit Hours (Total) / Number of Units (Total)
4 hours 3 units .....
7. Course administrator's name (mention all, if more than one name)
Name: ahmed saadi abdullah Email: ahmedalbasha@tu.edu.iq
8. Course Objectives

Course Objectives	• •Familiarity with the digital image processing system. • Understand the basics of digital image. - Learn about transformations and techniques for improving the digital image. - Restore the image and how to fragment it. • The image is adequately represented and described by a computer

9. Teaching and Learning Strategies

Strategy	

10. Course Structure

Week	Hours	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
1	4	Learn about the fields of image processing	Introduction to image processing	Explanation and practical application	Direct questions and daily and monthly exams
2	4	Identify the types of images and why they exist in more than one type	Types of digital image	Explanation and practical application	Direct questions and daily and monthly exams
3	4	Identify the structure by which images are organized	Formats of digital image	Explanation and practical application	Direct questions and daily and monthly exams
4	4	Learn how to use mathematical laws on images	Image algebra	Explanation and practical application	Direct questions and daily and monthly exams
5	4	Identify the transformations that occur in images	Geometry transformation	Explanation and practical application	Direct questions and daily and monthly exams
6	4	Learn about wave transforms	Wavelet transform	Explanation and practical application	Direct questions and daily and monthly exams
7	4	First half exam	First exam	Explanation and practical application	Direct questions and daily and monthly exams
8	4	Recognize the benefit of the image quantization process	Image quantization	Explanation and practical application	Direct questions and daily and monthly exams
9	4	Learn about image enlargement techniques	Zoom algorithm	Explanation and practical application	Direct questions and daily and monthly exams
10	4	Learn about image analysis processes	Image Analysis	Explanation and practical application	Direct questions and daily and monthly exams
11	4	Learn about image enhancement techniques	Image enhancement	Explanation and practical application	Direct questions and daily and monthly exams
12	4	Learn about edge detection techniques	Edge detection	Explanation and practical application	Direct questions and daily and monthly exams
13	4	Learn about image segmentation techniques	Image segmentation	Explanation and practical application	Direct questions and daily and monthly exams
14	4	Learn about image compression techniques	Image compression	Explanation and practical application	Direct questions and daily and monthly exams

15	4	Second half exam	Second exam	Explanation and practical application	Direct questions and daily and monthly exams

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)	
Recommended books and references (scientific journals, reports…)	
Electronic References, Websites	

MODULE DESCRIPTION FORM

نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Machine Learning**			**Module Delivery**	
Module Type	Core			☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code					
ECTS Credits	**4**				
SWL (hr/sem)	**148**				
Module Level	2		**Semester of Delivery**	1st	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Armaneesa Naaman Hasoon		**e-mail**	armaneesa@tu.edu.iq	
Module Leader's Acad. Title	Assistant Professor		**Module Leader's Qualification**	ماجستير	
Module Tutor			**e-mail**		
Peer Reviewer Name	Dr. Mohammed Aktham		**e-mail**		
Scientific Committee Approval Date	13/01/2025		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	Artificial intelligence	**Semester**	
Co-requisites module	Python	**Semester**	

	Calculus, Probability		

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	•Understand the term machine learning and its applications •Understand the steps of machine learning. •Study the types of learning •Solve problems that traditional methods cannot solve. •Understand the methods of obtaining and processing data •Understand the methods of extracting features • Train and evaluate machine learning models
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	•Understand the definition and applications of machine learning •Understand data processing methods. •Develop the student's thinking skills and the ability to build a machine learning model used for prediction or classification •Provide the learner with the skills necessary to be able to evaluate the performance of the machine learning model and apply it in real-time
Indicative Contents المحتويات الإرشادية	• Machine Learning Basics •Problem in AI (Problem State Space, Search Space, Problem Solution) •Search Classification in AI •Blind Search Algorithms •Heuristic Search Algorithms •Machine Learning Basics

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1 .Lectures: The instructor will give lectures to introduce and explain the definition of machine learning methods and their applications. 2 .Interactive discussions: Engaging students in interactive discussions allows them to ask questions, seek clarification, and actively participate in learning. Discussions can include examples of machine learning algorithms, discuss how these algorithms are programmed in practice, and explore real-world applications of AI concepts. 3 .Lab: Lab sessions are dedicated, hands-on sessions where students apply the concepts they have learned in lectures to practical programming exercises. Key strategies for lab sessions include: a. Programming exercises: Students will work on programming exercises and projects in the lab, providing them with hands-on experience in programming machine learning projects. b. Guided training: 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 lab, promoting teamwork and enabling knowledge sharing. Working together on machine learning project programming assignments promotes discussion, problem solving, and peer learning. D. Access to Equipment and Resources: The lab should provide access to computers, necessary software tools, programming references, and relevant online resources. This ensures that students have the resources needed to complete lab exercises and assignments effectively. 4 .Programming Assignments: Students will be given assignments to enhance their understanding of how to program machine learning models to encourage independent problem solving. These assignments may include implementing algorithms, designing software systems, or developing small-scale projects using Python. 5 .Code Reviews and Feedback: The instructor will provide feedback on students' code, review their solutions, and make suggestions for improvement. This feedback will help students improve their coding skills and adhere to best practices. 6 .Office Hours and Individual Support: The instructor should provide individual consultations and support for students who need additional help or guidance in understanding and completing assignments.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	56	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3.7

Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	148		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	
	Assignments	4	20% (20)	7, 12	
	Projects	1	20% (20)	9-14	
	Report	1			
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Introduction to machine learning:
Week 2	Importance of Machine Learning and its Applications
Week 3	Types of Machine Learning techniques and the Differences Between Supervised and Unsupervised Learning
Week 4	Steps of a Machine Learning Model in Detail
Week 5	The performance evaluation metrics for machine learning methods
Week 6	Supervised Learning:
Week 7	Classification: Introduction to Classification, Decision tree, Support Vector Machines, Random Forest
Week 8	Regression: Introduction to Regression, Simple Linear Regression, Multi-linear regression.
Week 9	Exam
Week 10	Unsupervised Learning:
Week 11	Clustering: Introduction to Clustering, K-Means, Hierarchical clustering
Week 12	Dimensionality reduction
Week 13	Reinforcement learning
Week 14	Project Due (Students Presentations part1)

Week 15	Project Due (Students Presentations part2)

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	Material Covered
Week 1	Introduction to the basics of Python programming Language
Week 2	Use libraries like Pandas, Numpy, and Matplotlib to analyze datasets.
Week 3	Build a Simple Linear Regression using Scikit-learn.
Week 4	Build a multiple variables Regression using Scikit-learn.
Week 5	Build a Logistic Linear Regression using Scikit-learn.
Week 6	Build a Decision tree classifier using Scikit-learn.
Week 7	Build a Decision tree classifier using Scikit-learn.
Week 8	Build a Random Forest classifier using Scikit-learn.
Week 9	Build a Support vector machine classifier using Scikit-learn.
Week 10	Build a K-Means clustering method using Scikit-learn.
Week 11	Build a hierarchical clustering using Scikit-learn.
Week 12	Build a PCA using Scikit-learn.
Week 13	Build a PCA using Scikit-learn.
Week 14	Answering students' questions and reviewing previous topics.
Week 15	Answering students' questions and reviewing previous topics.

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems 1st Edition by Aurélien Géron (Author), 2017.	No
Recommended Texts		
Websites		

Grading Scheme مخطط الدرجات				
Group	**Grade**	التقدير	**Marks**	**Definition**

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

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Multimedia and Network Security**			**Module Delivery**	
Module Type	Core			☐ **Theory** ☒ **Lecture** ☒ **Lab** ☒ **Tutorial** ☐ **Practical** ☐ **Seminar**	
Module Code	TU010101107				
ECTS Credits	8				
SWL (hr/sem)	**200**				
Module Level	4		**Semester of Delivery**	2nd	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Humam Neamah Hussein		**e-mail**	Humam.n.hussein@tu.edu.iq	
Module Leader's Acad. Title	Assistant Lecturer		**Module Leader's Qualification**	Master	
Module Tutor	Udai Wasmi Abboud		**e-mail**	udai.wasmi@tu.edu.iq	
Peer Reviewer Name	Mahammed Aktham		**e-mail**		
Scientific Committee Approval Date	15/01/2025		**Version Number**	1.0	

Relation with other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	Computer Security	**Semester**	1

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	1- Provide a strong foundation in multimedia systems. 2- Enabling the student to understand the basics of multimedia security science 3- Protecting multimedia content during its transmission over the Internet.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	**A- Knowledge objectives** A1- Knowing the basics of multimedia and network security. A2- Knowledge and understanding of multimedia security functions. A3- Knowledge of multimedia security programming and network protection A4- Know how to benefit from multimedia in protecting data and networks. **B- The skills objectives of the course** The students responded clearly to the subject through the course teacher's evaluation as a result of the students' interaction during the explanation of the topics to be taught and through their effective contribution to using multimedia security programs and their ability to apply them. B1 - Enabling the student to choose the best way to hide information and protect rights. B2 - Teaching the student about the dangers that exist on the Internet.
Guidance Contents المحتويات الإرشادية	1- Introduction to Multimedia Security and Networks: 2- Overview of multimedia data types and their security challenges in network environments. 3- Understanding Multimedia Network Security Threats: 4- Common threats to multimedia data in network transmission (e.g., interception, tampering, eavesdropping). 5- Principles of Multimedia Encryption:

	6- Explanation of encryption techniques for securing multimedia data during transmission and storage. 7- Multimedia Data Authentication and Integrity: 8- Importance of data authentication and integrity verification to ensure the trustworthiness of multimedia content. 9- Multimedia Watermarking and Copyright Protection: 10- How watermarking techniques can be used to protect multimedia content from unauthorized use. 11- Secure Multimedia Transmission Protocols: 12- Overview of secure protocols for transmitting multimedia data over networks. 13- Multimedia Content Access Control: 14- Methods for controlling access to multimedia content to prevent unauthorized viewing or modification. 15- Multimedia Data Hiding Techniques: 16- Introduction to steganography and other data hiding techniques used to conceal multimedia data. 17- Secure Multimedia Storage Solutions: 18- Best practices for securely storing multimedia data to prevent data loss or unauthorized access. 19- Multimedia Network Forensics: 20- Overview of multimedia forensics techniques used to investigate security incidents involving multimedia data. 21- Securing Multimedia Communication in IoT Networks: 22- Guidelines for securing multimedia communication within Internet of Things (IoT) networks. 23- Multimedia Content Filtering and Monitoring: 24- Importance of content filtering and monitoring tools to prevent the distribution of malicious or inappropriate multimedia content. 25- Multimedia Network Security Best Practices: 26- General best practices and guidelines for ensuring the security of multimedia data in network environments. 27- Multimedia Network Security Tools and Software: 28- Overview of tools and software that can help enhance multimedia network security. 29- Case Studies and Examples: 30- Real-world examples and case studies illustrating the importance of

	multimedia security in network environments.

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
الاستراتيجيات **Strategies**	**Teaching Strategies:** 1- Interactive Lectures: Incorporate multimedia elements into lectures to demonstrate concepts visually and engage students effectively. 2- Hands-on Labs: Provide hands-on lab exercises where students can practice implementing security measures in multimedia networks. 3- Group Projects: Assign group projects that require students to analyze and propose solutions for multimedia security issues in network environments. 4- Peer Teaching: Encourage students to teach each other about specific topics related to multimedia security and networks. **A- Learning Strategies:** 1- Active Learning: Encourage students to actively engage with the material through discussions, hands-on activities, and problem-solving tasks. 2- Visual Aids: Use multimedia presentations, videos, and diagrams to help students visualize complex concepts. 3- Collaborative Learning: Foster collaboration among students through group projects and discussions on multimedia security topics. 4- Practice Exercises: Offer practice exercises and quizzes to reinforce learning and assess understanding. 5- Real-world Applications: Connect theoretical concepts to real-world applications of multimedia security and networks to demonstrate their relevance. 6- Feedback Mechanisms: Provide timely feedback on assignments and assessments to help students track their progress and improve their understanding. 7- Continuous Assessment: Implement regular assessments to gauge student comprehension and adjust teaching strategies as needed.

Student Workload (SWL) الحمل الدراسي للطالب محسوب لـ ١٥ أسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	93	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	6.2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	107	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	7.13

Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	200

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	#LO 1-3, #LO 5-8
	Assignments	2	10% (10)	7, 12	#LO 3-5, #LO 5-8
	Projects	1	10% (10)	continuous	
	Report	1	10% (10)	14	#LO 1-8
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	#LO 1-7
	Final Exam	2 hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Introduction of Multimedia
Week 2	Multimedia Security Concepts
Week 3	Steganography Concepts
Week 4	Digital Watermarking
Week 5	Digital Watermarking (cont.)
Week 6	Digital Rights Management
Week 7	Digital Watermarking Technologies
Week 8	Digital Watermarking Technologies (cont.)
Week 9	Types of Digital Watermarks
Week 10	Image Watermarking
Week 11	Image Watermarking in the network communication
Week 12	The Concepts of the DCT and IDCT in Watermarking
Week 13	Models of Watermarking-Geometric models
Week 14	Audio Watermarking
Week 15	Video Watermarking

Delivery Plan (Weekly Lab. Syllabus):

المنهاج الاسبوعي للمختبر:

Week No.	Material Covered
Week 1	Introductory Lecture to Matlab
Week 2	How to Import and Export (Audio, Video, Image, txt and excel) to Matlab
Week 3	Introduction to Steganography
Week 4	Steganography (Imstego)
Week 5	Steganography (ImstegoR)
Week 6	Steganography-Embedding Grey Image in color image
Week 7	Steganography-Embedding Grey Image in color image (Cont.)
Week 8	Introduction to Digital Watermarking
Week 9	Digital Watermarking (Same size)
Week 10	Digital Watermarking (Different size)
Week 11	Digital Watermarking (Insert text)
Week 12	The Concepts of the DCT and IDCT in Watermarking
Week 13	The Concepts of the DCT and IDCT in Watermarking (Cont.)
Week 14	Audio Watermarking
Week 15	Video Watermarking

Learning and Teaching Resources

مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	Shih F.Y. (Ed.) - Multimedia Security: Watermarking, Steganography, and Forensics 2012.	Yes
Recommended Texts		
Websites		

Grading Scheme

مخطط الدرجات

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

(0 – 49)	**F** – Fail	راسب	(0-44)	Considerable amount of work required

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

Course Description

The course extensively explores multimedia security, focusing on protecting digital content through steganography, digital watermarking, and digital rights management. It covers the theoretical and practical aspects of watermarking, including the application of DCT and IDCT in watermarking processes, and discusses various watermarking technologies and models.

. Educational Institution/ college	Tikrit University/College of Computer Science and Mathematics
8. Scientific Department/Center:	Computer Science Department
9. Course name/code:	Multimedia security and networks
10. Available attendance formats:	Weekly
11. Course /year:	Second Course 2023/2024
12. Number of study hours (total):	30 theoretical hours and 30 practical hours
22. Course objectives	

4- Provide a strong foundation in multimedia systems.

5- Enabling the student to understand the basics of multimedia security science

6- Protecting multimedia content during its transmission over the Internet.

23. Course Outcomes and Teaching, Learning and Evaluation Methods

A- Knowledge objectives

a1- Knowing the basics of multimedia and network security.
a2- Knowledge and understanding of multimedia security functions.
a3- Knowledge of multimedia security programming and network protection
a4- Know how to benefit from multimedia in protecting data and networks.

B - The skills objectives of the course

The students responded clearly to the subject through the course teacher's evaluation as a result of the students' interaction during the explanation of the topics to be taught and through their effective contribution to using multimedia security programs and their ability to apply them.

b1 - Enabling the student to choose the best way to hide information and protect rights.
b2 - Teaching the student about the dangers that exist on the Internet.

Teaching and learning methods

Theoretical Lectures

Practical Lectures

Evaluation methods

1. Direct questions during the theoretical lecture
2. Daily exams in each lecture on the subject of the previous lecture
3. Homework, assignments and reports
4. Monthly exams

C- Expressive and value objectives

c-1 Asking them in the lecture and assigning them to search for the answer by searching on the Internet.
c-2 Do not transfer solutions between groups of students by changing the tasks

from one group to another.
c-3 Push the student to commit to attending theoretical lectures by taking daily exams.

24. Course structure

Number of teaching hours		Syllabus Vocabulary	Week
Practical	Theoretical		
2	2	Introduction of Multimedia	.16
2	2	Multimedia Security Concepts	.17
2	2	Steganography Concepts	.18
2	2	Digital Watermarking	.19
2	2	Digital Watermarking (cont.)	.20
2	2	Digital Rights Management	.21
2	2	Digital Watermarking Technologies	.22
2	2	Digital Watermarking Technologies (cont.)	.23
2	2	Types of Digital Watermarks	.24
2	2	Image Watermarking	.25
2	2	Image Watermarking in the network communication	.26
2	2	The Concepts of the DCT and IDCT in Watermarking	.27
2	2	Models of Watermarking-Geometric models	.28
2	2	Audio Watermarking	.29
2	2	Video Watermarking	.30

25. Infrastructure

1- Required prescribed books	Nothing
2- Main references (sources)	Shih F.Y. (Ed.) - Multimedia Security: Watermarking, Steganography, and Forensics 2012.

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية		
Module Title	**Operating System 2**	**Module Delivery**

Module Type	Core		☐ **Theory** ☒ **Lecture** ☒ **Lab** ☒ **Tutorial** ☐ **Practical** ☐ **Seminar**		
Module Code					
ECTS Credits	8				
SWL (hr/sem)	**200**				
Module Level	1		**Semester of Delivery**	1st	
Administering Department	Computer Science		**College**	CCSM	
Module Leader	Ayad Ahmed Yass		**e-mail**	ayad.a.yass@tu.edu.iq	
Module Leader's Acad. Title	Lecturer		**Module Leader's Qualification**	Phd	
Module Tutor			**e-mail**		
Peer Reviewer Name	Mahammed Aktham		**e-mail**		
Scientific Committee Approval Date	00/00/		**Version Number**	1.0	

Relation with other Modules

العلاقة مع المواد الدراسية الأخرى

Prerequisite module	None	**Semester**	
Co-requisites module	Understanding basics of Operating system	**Semester**	2

Module Aims, Learning Outcomes and Indicative Contents

أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية

Module Aims أهداف المادة الدراسية	1. To describe the advance of computer systems 2. To describe the services an operating system provides to users, processes, and other systems. 3. To develop students' understanding skills for various operating system 4. To help students to deal with computer components. 5. To provide students with hands-on experience in Linux command line 6. To advance the students skills to working with scripting to deal with

	operating system. 7. To enhance students' ability to troubleshooting operating system problems. 8. To level up the students' skills so they be able to enter the market places since the OS consider the most required in most organizations
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Upon successful completion of this module, students should be able to: 1. Demonstrate a solid understanding of the fundamental principles and concepts of operating system. 2. to help the student to understand the components and security of the operating system. 3. At this point, the student will be able to deal with different operating system 4. Enhance the operating system components and avoid the security issues 5. Student able to work with Linux command as the Linux is the most required in the market nowadays 6. Less work and increase productivity by learning the scripting 7. fixing and troubleshooting all OS related issues 8. leveling up the skills means now the students can enter any market place without hesitating
Indicative Contents المحتويات الإرشادية	7- To describe the basic organization of computer systems. 8- To describe the services an operating system provides to users, processes, and other systems 9- To discuss the various ways of structuring an operating system 10- To introduce the notion of a process and a thread 11- To introduce CPU scheduling, which is the basis for multi programmed operating systems 12- To develop a description of deadlocks

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain operating concepts, like processes, threads, memory management, file systems, and system calls. 2. Interactive Discussions: Engaging students in interactive discussions allows them to ask questions, seek clarifications, and participate actively in the learning process. Discussions can include reviewing code examples, discussing database best practices, and exploring real-world applications of operating system 3. Laboratory Sessions: Laboratory sessions are dedicated practical sessions

	where students apply the concepts learned in lectures to hands-on database exercises. Key strategies for the laboratory sessions include: a. Linux Exercises: Students will work on linux exercises and projects in the laboratory, providing them with practical experience in command line and problem-solving. b. Guided Practice: Lab instructors or teaching assistants will be available to provide guidance, assistance, and immediate feedback on students' skills They can help students debug their scripting, identify errors, and improve their command line skills. c. Collaboration and Peer Learning: Students can collaborate with their peers in the laboratory, fostering teamwork and enabling knowledge sharing. Working together on Linux 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 that students have the necessary resources to complete their lab exercises and assignments effectively. 4. command line Assignments: Assignments will be given to students to reinforce their understanding of command line concepts and encourage independent problem-solving. These assignments may involve implementing scripting. 5. Reviews and Feedback: The instructor will provide feedback on students' scripting, 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 provide support to students who need additional help or guidance in understanding command concepts or completing assignments.

Student Workload (SWL)

الحمل الدراسي للطالب محسوب لـ ١٥ أسبوعا

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	93	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	6.2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	107	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	7.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	200		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	2	10% (10)	5, 11	#LO 1-3, #LO 5-8
	Assignments	2	10% (10)	7, 12	#LO 3-5, #LO 5-8
	Projects	1	10% (10)	continuous	
	Report	1	10% (10)	14	#LO 1-8
Summative assessment	Midterm Exam	2 hr	10% (10)	11	#LO 1-7
	Final Exam	2 hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	Process Synchronization Part1
Week 2	Process Synchronization Part2
Week 3	Critical Section Problems part1
Week 4	Critical Section Problems part2
Week 5	Deadlock
Week 6	Method of Handling Deadlock
Week 7	Deadlock Avoidance part1
Week 8	Deadlock Avoidance part2
Week 9	Deadlock Detection part1
Week 10	Deadlock Detection part2
Week 11	Main Memory Part1
Week 12	Main Memory Part2
Week 13	Segmentation
Week 14	Segmentation and Memory Allocation
Week 15	Reviewing Students' Projects

Delivery Plan (Weekly Lab. Syllabus):
المنهاج الاسبوعي للمختبر:

Week No.	Material Covered
Week 1	Introduction about the advance Linux usage
Week 2	Advance Linux Commands
Week 3	Explain about the App that using Linux
Week 4	Introduction about the bash scripting
Week 5	Linux Shell Scripting
Week 6	How to use shell scripting
Week 7	Advance shell scripting command
Week 8	Linux command line
Week 9	Shell Scripting Excustion
Week 10	Installing software in Linux
Week 11	Understanding Linux logs
Week 12	Identifying issues from the logs
Week 13	Shell scripting comparisons with normal command line
Week 14	Troubleshooting issues
Week 15	Applying instructor's feedback on students' projects

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	*Abraham* Silberschatz (2022). Operating System Concept nine edition.	Yes
Recommended Texts		
Websites		

Grading Scheme
مخطط الدرجات

Group	Grade	التقدير	Marks (%)	Definition
Success Group (50 - 100)	**A** - Excellent	امتياز	90 - 100	Outstanding Performance
	B - Very Good	جيد جدا	80 - 89	Above average with some errors
	C - Good	جيد	70 - 79	Sound work with notable errors

	D - Satisfactory	متوسط	60 - 69	Fair but with major shortcomings
	E - Sufficient	مقبول	50 - 59	Work meets minimum criteria
Fail Group (0 – 49)	**FX** – Fail	راسب (قيد المعالجة)	(45-49)	More work required but credit awarded
	F – Fail	راسب	(0-44)	Considerable amount of work required

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

MODULE DESCRIPTION FORM
نموذج وصف المادة الدراسية

Module Information
معلومات المادة الدراسية

Module Title	Neural Networks	**Module Delivery**	
Module Type	Core	☒ **Theory** ☒ **Lecture** ☒ **Lab** ☐ **Tutorial** ☐ **Practical** ☒ **Seminar**	
Module Code			
ECTS Credits			
SWL (hr/sem)	**148**		
Module Level	4	**Semester of Delivery**	2nd
Administering Department	Computer Science	**College**	CCSM
Module Leader	Salwa Khalid Abdulateef	**e-mail**	Khalid.salwa@tu.edu.iq
Module Leader's Acad. Title	**Asst. Prof.**	**Module Leader's Qualification**	MSc.
Module Tutor		**e-mail**	
Peer Reviewer Name	Mohamed Aktham	**e-mail**	
Scientific Committee Approval Date	20/01/2025	**Version Number**	1.0

Relation with other Modules
العلاقة مع المواد الدراسية الأخرى

Prerequisite module	Machine learning, Artificial intelligent	**Semester**	
Co-requisites module	None	**Semester**	

Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
Module Aims أهداف المادة الدراسية	11. learning the types of neural networks. 12. Knowing how these neural networks work, especially by linking them to practical life. 13. Knowing the problems that appear in neural networks and ways to solve them. 14. The student can be ability to write programs for neural networks. 15. The ability to use neural networks in some research 16. Learning how to choose the best network for your algorithm. 17. A detailed explanation of algorithms and how to program them. 18. Teaching the student to calculate the complexity time of the algorithm. 19. Encouraging the student to be creative and think about the specialization projects and keep pace with the development taking place in this field.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	5. The student is supposed to understand how to start building simple algorithms and know the language of building algorithms (pseudo code). 6. Build basic causal skills in creating and proving the correctness of algorithms and programs. 7. Build skills in analyzing data and acquiring knowledge 8. Build skills in how to choose appropriate solutions for some problems and assign the best algorithms to solve them 9. Enabling students to think and analyze mathematical topics. 10. Enabling students to think and analyze topics related to algorithms. 11. Giving the learner the skills to be able to build a neural network
Indicative Contents المحتويات الإرشادية	1. Review of Basic Programming Concepts: - Cv2, Pandas, Numpy, Keras, and Matplot. 2. overview on Fundamentals of artificial neural networks Theory of Neural Networks (NN) Neural Networks Applications and properties 3. Types of activation functions such as sigmoid, relu and others 4. Models of Neuron 5. Logical Operations: such as AND, OR, and XOR. 6. Types of learning Supervised, unsupervised 7.Network Architectures Signal layer and Multiple Layers of Neurons, 8.Perceptron and Recurrent (Feedback) Networks , Learning algorithm

	9. Explain Backpropagation Neural Networks algorithms and programming. 10. Explain Hopfield Network algorithms mathematical and programming. -

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1. Lectures: The instructor will deliver lectures to introduce and explain neural network concepts, 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 the learning process. 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. 4. 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.
Student Workload (SWL)	

الحمل الدراسي للطالب محسوب لـ ١٥ اسبوعا			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	56	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3.7
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	92	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	6.13
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	148		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	2	10% (10)	5, 11	
	Assignments	4	20% (20)	7, 12	
	Projects	1	20% (20)	5-14	
	Report	1			
Summative assessment	**Midterm Exam**	2 hr	10% (10)	11	
	Final Exam	2hr	40% (40)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week No.	Material Covered
Week 1	**History and Fundamentals of artificial neural networks**
Week 2	**Theory of Neural Networks (NN)**
Week 3	**Neural Networks Applications and properties**
Week 4	**Types of activation functions**
Week 5	**Exam1**
Week 6	**Models of Neuron**
Week 7	**Logical Operations**
Week 8	**Types of learning**
Week 9	**Exam2**
Week 10	**Network Architectures**
Week 11	**Multiple Layers of Neurons,**
Week 12	**Perceptron and Recurrent (Feedback) Networks , Learning algorithm**

Week 13	**Backpropagation Neural Networks**
Week 14	**Backpropagation Neural Networks**
Week 15	**Hopfield Network**

Delivery Plan (Weekly Lab. Syllabus): المنهاج الاسبوعي للمختبر:	
Week No.	**Material Covered**
Week 1	**Explaining the basics of programming in python**
Week 2	**Explaining the libraries in python**
Week 3	**Activation functions**
Week 4	**Activation functions**
Week 5	**Neural Network Toolbox**
Week 6	**Weights**
Week 7	**How can build a simple neural network**
Week 8	**can build a single layer neural network**
Week 9	**can build a multi layers neural network**
Week 10	**perceptron algorithm use to solve a simple classification logical-AND problem.**
Week 11	**perceptron algorithm use to solve a simple classification logical-OR problem.**
Week 12	**perceptron algorithm use to solve a simple classification logical- XOR problem.**
Week 13	**Back propagation learning algorithm**
Week 14	**Hopfield learning algorithm**
Week 15	**Examples**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts		
Recommended Texts	**1** Deep Learning for Vision Systems by MOHAMED ELGENDY: Manning Publications, 2020 .	

	2 Neural Networks: Tricks of the Trade by G. Montavon, G. Orr, K.R. Muller, (Springer, New York, 2012 **3** Neural Networks and Deep Learning: by Charu C. Aggarwal, 2018 textbook **4** An Introduction to Machine Learning, by Miroslav¬Kubat , Springer International Publishing AG 2017, DOI 10.1007/978-3-319-63913-0.	
Websites	• https://www.ibm.com/topics/neural-networks. • https://en.wikipedia.org/wiki/Neural_network	

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

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

Course description form

Course description

This course description provides a summary of the most important characteristics of the course and the learning outcomes that the student is expected to achieve, demonstrating whether he or she has made the most of the learning opportunities available. It must be linked to the program description.

1. Educational institution	Tikrit University / College of Computer Science and Mathematics
2. Scientific department/center	Computer Science
3. Course name/code	Computer networks (2)
4. Available forms of attendance	My presence
5. Semester/year	2024-2025
6. Number of study hours (total)	60 hours practical + theoretical
7. Date this description was prepared	2024/11/3

8. Course objectives

- Teaching the student the theoretical skills of data communications and networks
- Qualifying students to teach this subject to middle and middle school students
- Preparing students to work in the private and governmental telecommunications sector
- Teaching the student theoretical computer networking skills
- Understanding computer networks.
- Understanding the layers of computer networks.
- Understanding the protocols and the packet's journey from source to target.

9. Course outcomes and teaching, learning and evaluation methods

A- Cognitive objectives

1. Knowledge of basic concepts of computer networks
2. The student recognizes the importance of general concepts of computer networks.
3. Identify the most important protocols used to organize the network's work
4. Study the most important factors that affect the work of networks

B - The skills objectives of the course

1. The student learns to use computer networks and their equipment and possesses the ability to set them up.
2. Prepare qualified cadres to teach the networking subject in educational institutions.
3. Learn how to set up network protocols using the simulation program (Packet Tracer)

Teaching and learning methods

- Traditional lectures and discussion style
- Laboratory activities and preparing reports
- Advanced lectures (presentation)
- Using discussion via the electronic classroom (Google Classroom)(
- Using simulation programs prepared for this purpose on computers inside the laboratory
- Use some illustrative images and video clips that explain how network protocols work

Evaluation methods

Conducting theoretical and practical daily/quarterly/final tests
Reports
Conducting oral exams

Emotional and value goals

1. Using scientific and cognitive skills through dialogue on academic topics.
2. The student's ability to determine the type of network appropriate to the spatial and physical conditions through the use of realistic network simulation programs for the purpose of understanding the operation of protocols.
3. Consolidating the spirit of participation among students in solving various technical problems and working in a small team spirit.
4. Consolidating and implanting the principle that time is a necessary and important factor in networks and developing solutions to urgent problems.

Teaching and learning methods

- -Class and homework assignments
- -Practical activities
- -Discussion and positive participation via the electronic class

Evaluation methods

- Conducting theoretical and practical tests.
- Reports preparation.

D - Transferable general and qualifying skills (other skills related to employability and personal development).

1. Positive thinking and utilizing the knowledge you have received.
2. The ability to communicate with parties outside the university and train with them.
3. The student will be able to teach the subject he has learned if he is employed in an educational institution.
4. The ability of the student, if he wishes to develop himself scientifically by applying for postgraduate studies, to pass the test related to the networking subject, as it is one of the subjects included in the competitive examination

for Iraqi universities.

10.Course structure					
Week	**Hours**	**Required learning outcomes**	**Name of the unit/topic**	**Teaching method**	**Evaluation method**
1	4	Definition: Packet Tracer	Tracer Packet	Traditional lectures, discussion style, and presentation	Discussion and tests
2	4	Identify the physical connection of devices	Networks Physical Topologies	Traditional lectures, discussion style, and presentation	Discussion and tests
3	4	To introduce the means of transmission, wired and wireless, and how to link them in the program	Transmission Media Types	Traditional lectures, discussion style, and presentation	Discussion and tests
4	4	Learn about ways to send the signal	Multiplexing, TDM, FDM	Traditional lectures, discussion style, and presentation	Discussion and tests
5	4	Identify the logical connection of devices	Network Logical Topologies	Traditional lectures, discussion style, and presentation	Discussion and tests
6	4	Types of devices in networks	Network Devices	Traditional lectures, discussion style, and presentation	Discussion and tests
7	4	Networking	Network	Traditional	Discussio

		software	Software	lectures, discussion style, and presentation	n and tests
8	4	Introduction to the OSI model	OSI Model	Traditional lectures, discussion style, and presentation	Discussio n and tests
9	4	Introduction to the TCP/IP protocol	TCP/IP Model	Traditional lectures, discussion style, and presentation	Discussio n and tests
10	4	Identify network protocols	Network Protocols	Traditional lectures, discussion style, and presentation	Discussio n and tests
11	4	Learn about Switching Techniques	Switching Techniques	Traditional lectures, discussion style, and presentation	Discussio n and tests
12	4	Learn how to connect different networks	Internetworki ng	Traditional lectures, discussion style, and presentation	Discussio n and tests
13	4	Learn about data transfer	Data Communicati ons	Traditional lectures, discussion style, and presentation	Discussio n and tests
14	4	How to address within the network	IP Addressing	Traditional lectures, discussion style, and presentation	Discussio n and tests
15	4	Learn about the Internet and its applications	Internet and It's Applications	Traditional lectures, discussion style, and presentation	Discussio n and tests

11. Infrastructure	
1- Required prescribed books	1."TCP/ IP Protocol Suites", Behrouz Forouzan, McGraw-Hill, 4th edition, 2010
2- Main references (sources)	2. "Data Communication And Networking", Behrouz a. Forouzan, 4th edition, 2009
A) Recommended books and references (scientific journals, reports, etc.)	
b) Electronic references, Internet sites,...	http://www.youtube.com/playlist?list=PL828D58CF32F123B6

12.Course development plan
Courses are changed annually by 10% based on modern sources.

.....

.....

Course Description Form

1. Course Name:
Data mining
2. Course Code:
3. Semester / Year:
Fourth/ second semester
4. Description Preparation Date:
1/4/2024
5. Available Attendance Forms:
Full attendance
6. Number of Credit Hours (Total) / Number of Units (Total)

4 hours
3 units

7. Course administrator's name (mention all, if more than one name)

Name: ahmed saadi abdullah
Email: ahmedalbasha@tu.edu.iq

8. Course Objectives

Course Objectives	• The course aims to introduce the student to the methods used in data mining and describe their needs with the aim of mechanizing and creating computer systems for them. The course also aims to process data, explore complex types of data, and understand the relationships between inputs to make a specific decision.

9. Teaching and Learning Strategies

Strategy	

10. Course Structure

Week	Hours	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
1	4	Learn about the basic concepts of data mining	Introduction to Data Mining	Explanation and practical application	Direct questions and daily and monthly exams
2	4	Learn how to deal with data	Getting to Know your data	Explanation and practical application	Direct questions and daily and monthly exams
3	4	Learn about initial data processing	Pre-Processing Techniques	Explanation and practical application	Direct questions and daily and monthly exams
4	4	Learn how to choose features	Pre-Processing Techniques – Feature Selection	Explanation and practical application	Direct questions and daily and monthly exams
5	4	Learn about clustering methods	**Clustering techniques**	Explanation and practical application	Direct questions and daily and monthly exams
6	4	Identify one of the clustering methods	k-Means Clustering	Explanation and practical application	Direct questions and daily and monthly exams
7	4	First half exam	Mid Exam	Explanation and practical application	Direct questions and daily and monthly exams
8	4	Learn about classification techniques	**Classifications techniques**	Explanation and practical application	Direct questions and daily and monthly exams

9	4	Identify one of the methods of classification	Decision Trees	Explanation and practical application	Direct questions and daily and monthly exams
10	4	Learn about other techniques in clustering and classification	Learn about other techniques in clustering and classification	Explanation and practical application	Direct questions and daily and monthly exams
11	4	Recursive element mining recognition	Frequent Itemset Mining	Explanation and practical application	Direct questions and daily and monthly exams
12	4	Know the association's rules	Association Rules	Explanation and practical application	Direct questions and daily and monthly exams
13	4	Learn about evaluating what has been learned	**Evaluating what's been learned**	Explanation and practical application	Direct questions and daily and monthly exams
14	4	Learn about evaluating what has been learned	**Evaluating what's been learned2**	Explanation and practical application	Direct questions and daily and monthly exams
15	4	Second half exam	Mid Exam2	Explanation and practical application	Direct questions and daily and monthly exams

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)	
Recommended books and references (scientific journals, reports…)	
Electronic References, Websites	

Course Description Form

1. Course Name:
Distributed Systems
2. Course Code:
3. Semester / Year:
2024\ second semester
4. Description Preparation Date:
March 2024 ....
5. Available Attendance Forms:
In presence
6. Number of Credit Hours (Total) / Number of Units (Total)
4 hours weekly\
7. Course administrator's name (mention all, if more than one name)
Name: Raya Nazar Ismail Email: raya_computer@tu.edu.iq
8. Course Objectives

Course Objectives	• Understanding the characteristics and goals of using distributed systems • and let students describe how could they use these system in our lives .

9. Teaching and Learning Strategies	
Strategy	The main strategy that will be adopted in delivering this module are: 1. Power point presentation (Data show). 2. Explanation on the white board using different color markers. 3. Discussions with the student during teaching. 4. Interaction with students through daily problems practice through lecture. 5. Solve different problems with more exercises. 6. Submit assignment that develop student learning.

10. Course Structure

Week	Hours	Required Learning Outcomes	Unit or subject name	Learning method	Evaluation method
Week 1	4	What is distributed systems? Charectristics of DS. Transparency of DS.	Lecture 1 Lecture 2	Theoretical	Quiz
Week 2	4	Types of DS. Cloud computing services .	Lecture 3 Lecture 4	Theoretical	Assignment
Week 3	4	Architectures. Distributed operating system	Lecture 5 Lecture 6	Theoretical	Quiz
Week 4	4	Clusters for parallel scalable computing. Processes & threads.	Lecture 7 Lecture 8	Theoretical	Assignment
Week 5	4	Communication in DS. Fault tolerance	Lecture 9 Lecture 10	Theoretical	quiz