University of Babylon

College of Science for Women

Department of Computer Science

Academic Program Description for Undergraduate Studies Department of Computer science for the Year 2024-2025

Academic Program Description

University Name: University of Babylon

College/Institute: College of Science for Women

Name of the academic: Bachelor's in Computer Science

Name of final degree: Bachelor's in Computer Science

Study system: ***Undergraduate Studies***\bologna track

Description preparation date: 10/10/2024

Date of filling out the file: 16/10/2024

Signature:

Name of Department Head

Dr. Saif Mahmoud Khalaf

Date: 16/ 10 / 2024

Signature:

Name of Scientific Assistant

Dr. Kawther Mohammed Ali

Date: 16 / 10 / 2024

The file is checked by

Department of Quality Assurance and University Performance

Director of the Quality Assurance and University Performance Department:

Mohammed J.Jader

Date: 16 / 10 / 2024

Signature:

Approval of the Dean

Introduction – Academic Program Description

The vision of the Bologna track Program in Computer Science aims to provide academic education characterized by high quality, global competitiveness, and innovation, while qualifying graduates to contribute to achieving digital transformation and building an advanced and sustainable technological future. The Department of Computer Science is one of the main departments in the Faculties of Science, as it includes a number of different scientific specializations, including: communications, computer networks, artificial intelligence, and cybersecurity. The Department of Computer Science is of great importance due to its influential role in providing the labor market with programmers and software developers, and therefore it has the highest priority in keeping pace with the latest developments in the field of information technology. The development of software has opened up broad horizons for the employment of information technology in all scientific and applied fields, as the use of information technology has increased the efficiency and accuracy of work in addition to reducing human effort by relying on computer software.

1. Program Vision

To make the Department of Computer Science a regional and Arab leader in the field of applied research, as the use of information technology has increased the efficiency and accuracy of work, in addition to reducing human effort by relying on computer software.

2. Program Mission

The mission of the Bologna track Programme of the Department of Computer Science is to prepare graduates with high technical and scientific skills, capable of innovation and solving advanced digital problems, with a commitment to applying European quality standards and promoting scientific research and international cooperation. Contributing to the development of the digital society by providing graduates capable of contributing to the digital transformation of society through their skills in software development, data analysis, and IT solutions, and also supporting students to be leaders in the digital knowledge society and contribute to building a sustainable future based on technology.

3. Program Objectives

The objectives of the Bologna track Program for the Department of Computer Science aim to provide students with the knowledge and skills necessary to excel in the fields of computer science and information technology. These objectives are based on developing academic education, enhancing graduates' capabilities, and supporting scientific research. The Department of Computer Science at the College of Science for Girls aims to provide the labor market with what it needs of programmers and software and application developers in various applied levels, in addition to the constant endeavor to keep pace with scientific and technological developments that are reflected in increasing the skills of the department's graduates. In addition to contributing to the development of this important field of knowledge (computer science) by having its researchers present scientific research at international and local conferences, in addition to publishing scientific research in various international journals, as well as patents, workshops, seminars, and training courses that drive the wheel of progress in this field.

4. Programmatic Accreditation

Nothing

5. Other External Influences

Training courses for students to develop scientific skills in advanced technology and programs + field visits

6. Program Structure

Distribution of units across study stages and the percentages of departmental requirements (mandatory or elective), college, and university requirements were as follows:

Stage	Number of Units	Core (Mandatory)	Elective	Supported
First	30	19	0	11
Second	41	24	15	2
Third	39	27	9	3
Fourth	35	19	14	2
Total	145	89	38	18
Percentage		61.4%	26.2%	12.4%

7. Detailed Description of Courses.

This guide covers the courses offered by the Computer Science program for the Bachelor of Science degree. The program offers (48) courses with (6000) total student load hours and 240 total European units. The delivery of courses is based on the Bologna process.

7. *Program Description*				
Year/level	***course code***	***Name of the course***	***Credit hours***	
			Theoretical	***Practical***
The first stage, Course (1), according to the Bologna system	**UOBAB06040101**	**Programming Fundamentals**	**2**	**2**
	UOBAB06040102	**Digital Logic**	**2**	**2**
		foundation of discrete structure	**2**	**-**
	UOBAB06040103	**Computer Organization**	**2**	**2**
		Linear algebra	**2**	**-**
		English language	**2**	**-**
The first stage, Course (2), according to the Bologna system	**UOBAB06040201**	**Structured programming**	2	2
	UOBAB06040202	**Computer Skills**	2	2
		Communication Skills	2	-
		Discrete Structures	2	-
		Probability and Statistics	2	1
		Arabic Language	2	-

Second stage Course (1)	**C8**	**Object Oriented Programming (1)**	2	2
	C11	**Computation Theory (1)**	3	-
	C16	**Database (1)**	2	2
	E57	**Linux Fundamentals**	2	2
	E55	**Microprocessor and Assembly Languages**	2	2
	C9	**Data Structures**	2	2
	S4	**English Language (2)**	2	-
Second stage Course (2)	**E1**	**Object Oriented Programming (2)**	2	2
	C12	**Computation Theory (2)**	3	-
	C17	**Database (2)**	2	2
	C4	**Computer Architecture**	3	-
	E45	**Web Design & Programming**	2	2
	E4	**Operational Researches**	3	-
	C27	**Probability and Statistics**	3	-

8.The expected learning outcomes of the program

Knowledge	
Knowledge and Understanding	1. The student learns about the nature of computer science. 2. The student learns about preparing scientific research in the field of computers. 3. The student is able to use computers in most applications. 4. The student is able to analyze and solve problems that may occur in the field of computer science. 5. The ability to find scientific solutions to community problems programmatically. 6. The ability to analyze and evaluate software systems before starting to design the system 7. Providing the student with some basic rules in evaluating and building software systems based on the basics of Software Engineering
Skills	
Subject-Specific Skills	B1 Theoretical - B2 Practical - B3 Summer Training - B4 Graduation Research
Thinking Skills	1. Let's Think about Thinking Ability: The goal of this skill is for the student to believe in what is tangible (student's abilities) and understand when, what and how he should think and work on improving the ability to think reasonably.

	2. High Thinking Skill: The goal of this skill is to teach thinking well before making a decision that determines the student's life, for example if the student wants to make a good decision, it is important to think well before making the decision and if he decides without thinking or if he cannot think well or if he cannot decide or perhaps will not decide, this means he does not have a high thinking skill 3. Critical Thinking Strategy in Learning (Critical Thanking): It is a term that symbolizes the highest levels of thinking that aims to pose a problem and then analyze it logically to reach the required solution. 4. Brainstorming
Ethics	
Evaluation methods	1- Through the regular exam (paper). 2- Through writing computer programs (practical application). 3- Through the method of expression with faces. 4- Preparing reports by students. 5-Relying on scheduled and unscheduled hours. By conducting the exam on the Moodle system using the E-learning technology

9. Teaching and Learning Strategies

Learning strategies

1- Thinking strategy according to the student's ability (example: if the student can learn the correct management concept, he will acquire the skill of managing and organizing his personal life).

2- High thinking skill strategy (example: if the student wants to make a good decision, it is important to think well before making the decision, and if he decides without thinking or if he cannot think well or if he cannot decide or perhaps will not decide, this means he does not have a high thinking skill).

3- Critical thinking strategy in learning (Critical Thanking) (is a term that symbolizes the highest levels of thinking that aims to pose a problem and then analyze it logically to reach the required solution).

4- Brainstorming.

Methods of teaching and learning

1- Method of delivering lectures.

2- Student Center

3- (Student groups Team Project)

4- (Work shop workshops)

5- (Scientific trips to follow up on the environmental reality)

6- (Learning Technologies on Campus)

7- (experiential learning)

8-(Application Learning)

10. Evaluation methods : : The assessment methods in the Bologna Pathway programme rely on a combination of different assessment methods that aim to measure students' performance and their deep understanding of scientific and applied concepts, in addition to their practical skills and their ability to innovate and solve problems. These methods include:

1- Exams

2-Matrix (Learning Matrix)

3-Which Face (Method of Expression by Faces)

4-CAT (Feedback from Students)

5-Learning Triangle (Learning Triangle)

6-Preparing reports.

7-Relying on scheduled and unscheduled hours.

8- Additional duties inside and outside the college.

11. Faculty

Faculty Members

Academic Rank	*Instructor's name*	*Specialization*		*Special Requirements/skills (it applicable)*	*Number of the teaching staff*	
		General	*Special*		*staff*	*lecturer*
Professor	Dr. Hussein Attia Lafta	Computer	Artificial Intelligence		√	
Professor	Dr. Suhad Ahmed Ali	Computer	Artificial Intelligence		√	
Professor	Dr. Majid Jabbar Jawad	computer	security and information processing		√	
Professor	Dr . Samaher Hussein Ali	Computer	Artificial Intelligence		√	

Professor	Dr. Samah Abdel Hadi Abbas,	Mathematics	Mathematics		√	
Professor	Dr Muhammad Abdullah Nasser	Computer	security and information processing		√	
Professor	Dr. Sahar Adel Kazem	Computer	Security and information		√	
Professor	Dr. Israa Hadi Obaid,	Computer	Translators and Computational Theory		√	
Assistant Professor	Dr. Saif Mahmoud,	Computer	computer networks		√	
Professor	Dr. Mahdi Abdel Salman	Computer	Distributed Systems		√	
Professor	Dr. Muhammad Obaid	Computer	Information Technology/Software		√	
Assistant	Dr. Salah Mahdi Saleh	Computer	Timbers Patterns		√	
Professor	Dr. Ahmed Badri Muslim,	Computer	Parallel Algorithms		√	
Assistant Professor	Dr.Ali Yaqoub Youssef	Computer	artificial intelligence		√	
Assistant Professor	Dr. Ali Kazem Muhammad	Computer	Information Technology/Software		√	
Teacher	Dr . Farah Muhammad Hassan,	Computer	information security		√	
Teacher	Dr. wed Kazem Aliwi	Computer	Computer		√	

Assistant Professor	Dr. Ahmed Mohamed Hussein,	Computer	Computer		√	
Assistant Professor	Dr.. Zainab Abdel Moneim Abdel Hadi,	Mathematics	Functional Approximation Theory		√	
Teacher	Zainab Falah Hassan	Computer	Computer		√	
Assistant Professor	Dr. Elaf Ali Abboud	Computer	Computer		√	
Teacher	Noor Kazem Ayoub	Computer	Computer		√	
Assistant Professor	Asraa Abdullah Hussein	Computer	Computer		√	
Teacher	Russell Muhammad Nimah	Computer	Computer		√	
Teacher	Dr. Hossam Jawad Kazem,	Computer	communications systems		√	
assistant teacher	Nada Fadel Muhammad	Computer	Computer		√	
assistant teacher	Ishraq Abdel Amir Yahya	Computer	Computer		√	
teacher	Hadeel Qasim Ghani	Computer	Computer		√	
teacher	Zahraa Jabbar Hussein	Computer	Computer		√	
assistant teacher	Zahraa Abdel Mohamed	Computer	Computer		√	
assistant teacher	Jinan Ali Abd	Computer	Computer		√	
assistant teacher	Shaima Abdel Kazem Hadi	Computer	Computer		√	

assistant teacher	Zahraa Aboud Ahmed	Computer	Computer		√	
assistant teacher	Rafif Mazhar Katran	Computer	Computer		√	

Professional Development

Mentoring new faculty members

Teaching, like any other art, can be acquired through practicing and following its methods and principles, provided that there is a sincere desire to practice the teaching profession. The method in education means taking interconnected steps to reach a specific goal that is hoped to be achieved. Therefore, the basic principles of good teaching must be followed, which are:

1- Guiding and directing learners by creating educational situations that lead to desired activities.

2- Providing an atmosphere of love, affection and cooperation between the teacher and learners and between the learners themselves through his love for his students without discrimination and not overdoing feminization.

3- Adopting democratic leadership through the sensory relationship between the teacher and his students, which leads them to control based on mutual respect and creating a cooperative atmosphere between the students and between the teacher and his students.

Professional development for faculty members

1- Thinking strategy according to the student's ability (example: if the student can learn the correct concept of management, he will acquire the skill of managing and organizing his personal life). And the strategy of high thinking skill (example: if the student wants to make a good decision, it is important to think well before making the decision and if he decides without thinking or if he cannot think well or if he cannot decide or perhaps will not decide, this means he does not have high thinking skill).

2- General and transferable skills (other skills related to employability and personal development).

3- Verbal communication.

4- Teamwork.

5- Analysis and investigation (collecting information systematically and scientifically to establish facts and principles to solve the problem). Initiative (motivation to work and the ability to take initiative, identify opportunities and put forward ideas and solutions.

12. Acceptance criterion

Central acceptance and parallel acceptance

13. The most important sources of information about the program

5- The website of the college and university.

https://csg.uobabylon.edu.iq/
https://csg.uobabylon.edu.iq/department/?cdid=4

https://csg.uobabylon.edu.iq/department/dep_lectures.aspx?cdid=4

2- The electronic system of the Bologna Process.

3-University guide .https://systems.uobabylon.edu.iq /

4-The most important books and resources in the college library.

14. Program development plan

The Bologna process was applied to first-year students and work is underway to apply it to future stages, with workshops and seminars being held to familiarize faculty members with the

requirements of the Bologna process and how to work with it, discuss the negatives and obstacles, and find solutions for them. The electronic system was applied in the education process.

Program skills Outline

Year/Level	*Course Code*	*Course Name*	*Basic or optional*	*Required program learning outcomes*															
				Knowledge				*Skills*				*Ethics*				*Other skills related to employability and personal development*			
				A_1	A_2	A_3	A_4	B_1	B_2	B_3	B_4	C_1	C_2	C_3	C_4	D_1	D_2	D_3	D_4
The first stage, Course (1), according to the Bologna system	UOBAB06040101	**Programming Fundamentals**	C	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	UOBAB06040102	**Digital Logic**	S	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
		foundation of discrete structure	S	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	UOBAB06040103	**Computer Organization**	B	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
		Linear algebra	S	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
		English language	B	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Program skills Outline																			
				Required program learning outcomes															
Year/Level	*Course Code*	*Course Name*	*Basic or optional*	*Knowledge*				*Skills*				*Ethics*				*Other skills related to employability and personal development*			
				A_1	A_2	A_3	A_4	B_1	B_2	B_3	B_4	C_1	C_2	C_3	C_4	D_1	D_2	D_3	D_4
The first stage, Course (2), according to the Bologna system	UOBAB06040201	**Structured programming**	C	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	UOBAB06040202	**Computer Skills**	S	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
		Communication Skills	B	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

		Discrete Structures	S	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
		Probability and Statistics	S	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
		Arabic Language	B	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Program skills Outline

				Required program learning outcomes															
Year/Level	***Course Code***	***Course Name***	***Basic or optional***	***Knowledge***				***Skills***				***Ethics***				***Other skills related to employability and personal development***			
				A_1	A_2	A_3	A_4	B_1	B_2	B_3	B_4	C_1	C_2	C_3	C_4	D_1	D_2	D_3	D_4
Second stage Course (1)	**C8**	**Object Oriented Programming (1)**	**Core**	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	C11	**Computation Theory (1)**	**Core**	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	C16	**Database (1)**	**Core**	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	E57	**Linux Fundamentals**	**Elective**	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	E55	**Microprocessor and Assembly**	**Elective**	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

		Languages																	
	C9	Data Structures	Core	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	S4	English Language (2)	Supported	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

Program skills Outline

				Required program learning outcomes															
Year /Level	*Course Code*	*Course Name*	*Basic or optional*	*Knowledge*				*Skills*				*Ethics*				*Other skills related to employability and personal development*			
				A_1	A_2	A_3	A_4	B_1	B_2	B_3	B_4	C_1	C_2	C_3	C_4	D_1	D_2	D_3	D_4
Second stage Course (2)	E1	Object Oriented Programming (2)	Core	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	C12	Computation Theory (2)	Core	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
	C17	Database (2)	Core	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*

C4	Computer Architecture	Core	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*		*
E45	Web Design & Programming	Elective	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*		*
E4	Operational Researches	Elective	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*		*
C27	Probability and Statistics	Core	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*		*

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Linux Fundamentals			**Module Delivery**	
Module Type	BASIC			**Theory** **Lecture** **Seminar**	
Module Code	COM23010				
ECTS Credits	5				
SWL (hr/sem)					
Module Level	2		**Semester of Delivery**	1	
Administering Department	Computer Science		**College**	College of Science for Women	
Module Leader	Ahmed Badri Muslim		**e-mail**	ahmed.fanfakh@uobabylon.edu.iq	
Module Leader's Acad. Title	Prof.		**Module Leader's Qualification**	PhD	
Module Tutor	None		**e-mail**	None	
Peer Reviewer Name			**e-mail**		
Review Committee Approval			**Version Number**		

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- التعرف على بنية نظام التشغيل والهيكلة الخاصة به. 2- مقارنة نظام التشغيل لنكس مع الانظمة الاخرى. 3- فهم مميزات النظام واسباب اهميته للمبرمجين. 4- التعرف على مبادئ النظام المفتوح المصدر . 5- دراسة ايعازات النظام عن طريق الامثلة الخاصة بادارة البيانات. 6- دراسة طرق ربط ايعازات النظام مع لغة البرمجة لتوليد برامج لها قابلية اكبر من الايعازات الموجودة بالنظام لادراة الملفات والبيانات كحزمة واحدة.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. فهم طريقة عمل النظام والتعامل مع ايعازاته.1 2. يستطيع الطالب ان يفهم نظام التشغيل المفتوح المصدر. 3. التعرف على ادوات النظام التي تعمل كوسيلة ممتازة لتظيم الملفات بطرق عدة. 4. التعرف على طرق تنصيب وازالة البرامج الخدمية من والى النظام . 5. تعلم طرق برمجة انظمة التشفيل عن طريق لغات البرمجة مثل لغة Linux Batch script
Indicative Contents المحتويات الإرشادية	

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	طرائق التعليم والتعلم 1. استخدام الشاشات الذكية بالاضافة الى السبورات العادية 2. عرض المحاضرة بموقع الجامعة وكذلك في موقع الكلية والقسم 3. التركيز على الحلقات النقاشية بين الاستاذ والطالبات 4. تشجيع التعلم الذاتي ومساعدة الطالبات على الاستنتاج 5. التاكيد على المنافسة بين الطالبات

Student Workload (SWL) الحمل الدراسي للطالب			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	100	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	2
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	100		

Module Evaluation
تقييم المادة الدراسية

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

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

Week	Material Covered
Week 1	General definition of Linux
Week 2	General remarks on the operating system UNIX/Linux
Week 3	Linux history and distributions
Week 4	Components of Linux System and Architecture
Week 5	Linux command line Man pages
Week 6	Working with directories
Week 7	Working with directories
Week 8	Working with files: File command, touch command, remove file
Week 9	Working with files: copy files, copy to another directory, copy multiple files to directory, moving files
Week 10	Working with file contents: Head command, tail, cat, concatenate, create files with cat, copy files using cat, Tac, more and less commands
Week 11	Installing and uninstalling packages
Week 12	Command and arguments
Week 13	Linux filters
Week 14	Script programming: input, output primitives and control
Week 15	Script programing: loops and other useful Bach commands
Week 16	**Final Exam**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Working with Linux terminals
Week 2	Working with MAN pages of Linux
Week 3	Working with standard Linux command: Change Directory, Absolute and relative paths
Week 4	Path Completion, List Contents, Make directory, remove dir.
Week 5	Working with File command, touch command, remove file
Week 6	copy files, copy to another directory
Week 7	copy multiple files to directory, moving files
Week8	Head, tai and cat commands
Week9	Create files with cat, copy files using cat, Tac, more and less commands
Week10	Quiz
Week11	dpkg, APT, install and update the system software
Week 12	Control operators
Week13	Project
Week 14	Writing program in Bach script language
Week 15	Writing program that perform some operating system services

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Richard Petersen, **Linux: The Complete Reference**, Sixth Edition, 2008.	
Recommended Texts	Paul Cobbaut, **Linux Fundamentals**, Netsec BVBA, 2015.	
Websites	There are a lot of information over internets	

APPENDIX:

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

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

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Computer Organization**			**Module Delivery**	
Module Type	CORE			**Theory** **Lecture** **Seminar**	
Module Code	UOBAB0604012				
ECTS Credits	8				
SWL (hr/sem)	200				
Module Level	1		**Semester of Delivery**	1	
Administering Department	Computer Science		**College**	College of Science for Women	
Module Leader	Ahmed Mohammed Hussein		**e-mail**	wsci.ahmed.mohammed@uobabylon.edu.iq	
Module Leader's Acad. Title	Assist. Prof.		**Module Leader's Qualification**	PhD	
Module Tutor	None		**e-mail**	None	
Peer Reviewer Name			**e-mail**		
Review Committee Approval	2023-11-05		**Version Number**		

Relation With Other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	None	**Semester**	
Co-requisites module	None	**Semester**	1
Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية			
Module Aims أهداف المادة الدراسية	تهدف هذه المادة على تطوير الفهم الاساسي لتنظيم وتشغيل جهاز الحاسوب المكتبي بما في ذلك معمارية وحدة المعالجة المركزية , الذاكرة و اجهزة الادخال والاخراج . الطالب سوف يكون قادرا على مناقشة مبادئ تمثيل المعلومات وقادرا على استخدام تمثيل اعداد متنوعة والتحويل بينهم. ايضا , سوف يكتسب الطالب فهم اساسي للخصائص المعمارية لأنظمة الحاسوب الحديثة , بما في ذلك مسارات النقل وتنظيم الذاكرة الحديثة . كما يوفر مقدمة لتنظيم وتشغيل برنامج نظام التشغيل .		
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. اجزاء الحاسوب الاساسية واجيال الحاسوب. 2. تمثيل البيانات (اعداد واحرف). تمثيل البيانات. 3. اهم العمليات التي يقوم بتنفيذها الحاسوب مثل الجمع والطرح والضرب والقسمة. 4. تخزين البيانات وتوضيح اجهزة الخزن الرئيسية والثانوية.		
Indicative Contents المحتويات الإرشادية	-1 قراءات، تعلم ذاتي، حلقات نقاش -2 التدريبات والانشطة في قاعة الدرس -3 ارشاد الطلاب الى بعض المصادر والموقع الالكترونية للاستفادة منها.		
Learning and Teaching Strategies استراتيجيات التعلم والتعليم			
Strategies	-1 طريقة القاء المحاضرات. -2 Student Center. -3 (المجاميع الطلابية Team Project). -4 (Work shopورش العمل). -5 (Learning Technologies on Campus) (التعلم الالكتروني داخل الحرم الجامعي). -6 (experiential learning التعلم التجريبي) من خلال كتابة برامج حاسوب وتطبيقها ومعرفة مخرجاتها.		

Student Workload (SWL) الحمل الدراسي للطالب			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	100	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	2
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	100		

Module Evaluation
تقييم المادة الدراسية

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

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

	Material Covered
Week 1	مقدمة عن تنظيم الحاسوب, اجيال الحاسوب , فئات الحاسوب. Introduction to Computer Organization, Computer Generations and Computer Categories.
Week 2	اجزاء الحاسوب الاساسية Main computer parts
Week 3	تمثيل البيانات Data Representation
Week 4	عمليات الحاسوب الحسابية Arithmetic Computer Operations
Week 5	تخزين البيانات Data Storage
Week 6	ذاكرة الوصول العشوائي وانواعها . RAM
Week 7	ذاكرة القراءة فقظ وانواعها . ROM
Week 8	الأنظمة الرقمية
Week 9	التحويل بين الأنظمة الرقمية
Week 10	كيفية تعامل الحاسوب مع الأنظمة الرقمية
Week 11	التعرف نظام البايوز
Week 12	التعرف على كيفية أدارة العمليات في الحاسوب
Week 13	محاضرة على طريقة فرمتت الحاسوب الشخصي
Week 14	**مراجعة عامة للمادة**
Week 15	**Final Exam**

Delivery Plan (Weekly Lab. Syllabus)
المنهاج الاسبوعي للمختبر

	Material Covered
Week 1	مقدمة عن نظام التشغيل
Week 2	التعرف على سطح المكتب واجزاءه
Week 3	التعرف على أجزاء نظام التشغيل
Week 4	مقدمة عن برنامج مايكروسوفت اوفيس
Week 5	التعرف على الوورد
Week 6	تطبيق عملي على نظام الوورد
Week 7	عمل الجداول في نظام الوورد
Week 8	التعرف على النصوص في الوورد
Week 9	و تنسيق النص واضافة تنسقات اضافية
Week 10	طباعة الملف وتنسيقات الطباعة
Week 11	اختبار تطبيق ادراج الصور والجداول
Week 12	تنسيق الترقيم في الوورد
Week 13	تنسيق التصاميم الجاهزة في الوورد
Week 14	اختبارات عمل وطباعة الكتب الرسمية وتنسيقها
Week 15	Mid Exam

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text	Available in the Library?
Required Texts	See nested table below	
Recommended Texts		
Websites		

الرقم المعياري ISBN	اسماء المؤلفين	سنة الاصدار	دار النشر	عنوان المؤلف
,9781718500662 ,9781718500679 ,2020024168 ,2020024169 1718500661	Matthew Justice	2020	No Starch Press	How Computers Really Work
,9780134997193 ,1292420103 ,9781292420103 9781292420080	William Stallings	2021	Global Edition- Pearson	Computer Organizatio n and Architecture
,9781284259445 ,2022062125 9781284259438	Linda Null	2023	Jones & Bartlett Learning	The Essentials of Computer Organizatio n and Architecture , Sixth Edition

APPENDIX:

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

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

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Computer Skills**			**Module Delivery**	
Module Type	CORE			**Theory** **Lecture** **Seminar**	
Module Code	UOBAB0604022				
ECTS Credits	6				
SWL (hr/sem)	150				
Module Level	1		**Semester of Delivery**	2	
Administering Department	Computer Science		**College**	College of Science for Women	
Module Leader	Ahmed Mohammed Hussein		**e-mail**	wsci.ahmed.mohammed@uobabylon.edu.iq	
Module Leader's Acad. Title	Assist. Prof.		**Module Leader's Qualification**	PhD	
Module Tutor	None		**e-mail**	None	
Peer Reviewer Name			**e-mail**		
Review Committee Approval	2024-02-29		**Version Number**		

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	تهدف هذه المادة على تطوير الفهم الاساسي لتنظيم وتشغيل جهاز الحاسوب المكتبي بما في ذلك معمارية وحدة المعالجة المركزية , الذاكرة و اجهزة الادخال والاخراج . الطالب سوف يكون قادرا على مناقشة مبادئ تمثيل المعلومات وقادرا على استخدام تمثيل اعداد متنوعة والتحويل بينهم. ايضا , سوف يكتسب الطالب فهم اساسي للخصائص المعمارية لأنظمة الحاسوب الحديثة , بما في ذلك مسارات النقل وتنظيم الذاكرة الحديثة . كما يوفر مقدمة لتنظيم وتشغيل برنامج نظام التشغيل .		
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. ذاكرة الوصول العشوائي وانواعها. 2. ذاكرة القراءة فقط وانواعها. الذاكرة المؤقتة ووظيفتها. 3. الذاكرة الافتراضية ووظيفتها. 4. اللوحة الام واجهزة الادخال والاخراج.		
Indicative Contents المحتويات الإرشادية	-1 قراءات، تعلم ذاتي، حلقات نقاش -2 التدريبات والانشطة في قاعة الدرس -3 ارشاد الطلاب الى بعض المصادر والموقع الالكترونية للاستفادة منها.		
Learning and Teaching Strategies استراتيجيات التعلم والتعليم			
Strategies	-1 طريقة القاء المحاضرات. -2 Student Center. -3 (المجاميع الطلابية Team Project). -4 (Work shopورش العمل). -5 (Learning Technologies on Campus) (التعلم الالكتروني داخل الحرم الجامعي). -6 (experiential learning التعلم التجريبي) من خلال كتابة برامج حاسوب وتطبيقها ومعرفة مخرجاتها.		

Student Workload (SWL) الحمل الدراسي للطالب			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	100	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	2
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	100		

Module Evaluation
تقييم المادة الدراسية

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

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

	Material Covered
Week 1	الذاكرة المؤقتة Memory Cache
Week 2	الذاكرة المؤقتة Memory Cache
Week 3	الذاكرة الافتراضية Virtual Memory
Week 4	الذاكرة الافتراضية Virtual Memory
Week 5	اللوحة الام Motherboard
Week 6	أجهزة الادخال Input Devices
Week 7	اجهزة الاخراج Output Devices
Week 8	اختبارات عن المحاضرات السابقة
Week 9	الشاشة والطابعات Monitors and Printers
Week 10	مقدمة عن نظام التشغيل Introduction to Operating System
Week 11	ASCII Code
Week 12	المعالجات وانواعها Microprocessor
Week 13	محاضرة عن صيانة الحاسوب الجزء الثاني
Week 14	Class Assignment
Week 15	**Mid Exam**

Delivery Plan (Weekly Lab. Syllabus)
المنهاج الاسبوعي للمختبر

	Material Covered
Week 1	مقدمة عن الاكسل
Week 2	التعرف على الواجهة الرئيسية للاكسل
Week 3	التعرف على الدوال الخاصة بالاكسل
Week 4	التعرف على كيفية التعامل مع الجداول في نظام الاكسل
Week 5	تطبيق عملي على نظام الاكسل
Week 6	تنسيق الرسوم وادراجها
Week 7	عمل الاحصائيات العامة الضرورية
Week 8	اختبارات عن تطبيق الاكسل
Week 9	مقدمة عن البوربوينت
Week 10	التعرف على كيفية انشاء السلايد في نظام البوربوينت
Week 11	معرفة الحركات وأجزاء البوربوينت الرئيسية
Week 12	تطبيق عملي على نظام البوربوينت
Week 13	كيفية تحويل محاضرة الى عرض تقدمي في البوربوينت
Week 14	اختبارات عن تطبيق البوربوينت
Week 15	Mid Exam

Learning and Teaching Resources
مصادر التعلم والتدريس

	Text					Available in the Library?
Required Texts	الرقم المعياري ISBN	اسماء المؤلفين	سنة الاصدار	دار النشر	عنوان المؤلف	
	,9811656614 9789811656 613	Shuangb ao Paul Wang	2021	Springer	Computer Architecture and Organization: Fundamentals and Architecture Security	
	,0128203315 9780128203 316	David A. Patterson , John L. Henness y	2020	Morgan Kaufmann	Computer Organization and Design RISC-V Edition: The Hardware Software Interface, Second Edition	

	,1498772714 9781498772 ,716 ,1498772722 9781498772 723	Joseph D. Dumas II	2016	CRC Press	Computer architecture: fundamentals and principles of computer design	
Recommended Texts						
Websites						

APPENDIX:

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

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

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Microprocessor and Assembly Language			**Module Delivery**	
Module Type	CORE			**Theory** **Lecture** **Seminar**	
Module Code					
ECTS Credits					
SWL (hr/sem)	60 ساعة				
Module Level	1		**Semester of Delivery**	1	
Administering Department	Computer		**College**	Computer Science for Women	
Module Leader	Hussein A. Lafta		**e-mail**	wsci.husein.attia@uobabylon.edu.iq	
Module Leader's Acad. Title	Prof. Dr.		**Module Leader's Qualification**	Phd	
Module Tutor	Hussein A. Lafta				
Peer Reviewer Name			**e-mail**		
Review Committee Approval			**Version Number**		

Relation With Other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module	NONE	**Semester**	1
Co-requisites module	NONE	**Semester**	1
Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية			
Module Aims أهداف المادة الدراسية	This unit guide is intended to provide a general idea of the teaching content and assessment criteria for the unit entitled Microprocessor. General aims are to provide an understanding of the operation of microprocessors and their interfacing components, and to offer essential design considerations in Microprocessor and Computer Interfacing applications. Microprocessors and Interfacing deals with the general principles of microprocessor design and interfacing by looking at the Intel 8086 microprocessor and its associated peripheral interface chips. Programming the microprocessor is done using the TASM assembly language on the PC. This is done to emphasis the sequence of operations of software code and their implications on the hardware. The unit deals with microprocessor architecture, operation of registers and data manipulation as well a program control		
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	By the end of the course, students will be able to: 1- Understand components of the computers, microprocessors. 2- Know how to approach and undertake microprocessor development. 3- Learning role of CPU, registers, buses. 4- Know how interface memory and peripheral devices to a microprocessor. 5- Learning addressing modes (Immediate, direct, extended, indexed, indexed-indirect, and relative addressing modes). 6- Know the architecture of the 80x86-type microprocessor. Its capabilities and limitation and how it fits in with modern computers. 7- Understanding the function of each pin in 8086 microprocessors. 8- Learning interrupt vectors, interrupt process, interrupt priorities, external and advanced interrupts 9- Learning how to write program in assembly language using TASM. .		
Indicative Contents المحتويات الإرشادية	This unit guide is intended to provide a general idea of the teaching content and assessment criteria for the unit entitled Microprocessor. General aims are to provide an understanding of the operation of microprocessors and their interfacing components, and to offer essential design considerations in Microprocessor and Computer Interfacing applications. Microprocessors and Interfacing deals with the general principles of microprocessor design and interfacing by looking at the Intel 8086 microprocessor and its associated peripheral interface chips. Programming the microprocessor is done using the TASM assembly language on the PC. This is done to emphasis the sequence of operations of software code and their implications on the hardware. The unit deals with microprocessor architecture, operation of registers and data manipulation as well a program control.		

Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	1- Use smart screens in addition to regular whiteboards. 2- Display the lecture on the university website, as well as on the college and department websites. 3- Focus on discussion groups between professors and students. 4- Encourage self-learning and help students draw conclusions. 5- Emphasize competition among students.

Student Workload (SWL) الحمل الدراسي للطالب			
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			

Module Evaluation تقييم المادة الدراسية					
		Time/Number	**Weight (Marks)**	**Week Due**	**Relevant Learning Outcome**
Formative assessment	**Quizzes**	3	10	3,6,8	
	Assignments	3	10	2,4,7	
	Projects / Lab.	1	10	10	
	Report	1	10	3	
Summative assessment	**Midterm Exam**	2hr	10	7	
	Final Exam	3h	50	16	
Total assessment			100		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	Introduction and History of Microprocessors; Basic Block Diagram of a computer; Organization of Microprocessor Based System; Bus Organization.
Week 2	Stored program Concept and Von Neumann Machine; Processing Cycle of a Stored Program Computer
Week 3	Microinstructions and Hardwired/Microprogrammed Control Unit ; Introduction to Register Transfer Language
Week 4	Internal Architecture and Features of 8086 Microprocessor ; BIU and Components; EU and Components
Week 5	EU and BIU Operations; Segment and EU and BIU Operations; Segment and Offset Address
Week 6	Move,XChange,Push,Pup
Week 7	ADD,SUB Instructions
Week 8	First Exam
Week 9	AND, OR, XOR, NOT Instructions
Week 10	shift and rotate instructions
Week 11	Review
Week 12	Simple Programs for Arithmetic, Logical, String Input/Output
Week 13	Design and implement (Simple Project
Week 14	Design and implement (Simple Project)
Week 15	**Second Examination**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Setting up the emu8086 simulation
Week 2	the concept of Assembly Language
Week 3	Practical basic on assembly language
Week 4	learn to build a code using emu8086 simulation
Week 5	Learn to create code for data transfer instruction set
Week 6	Learn to create code for data transfer instruction set
Week 7	Learn to convert from Assembly language to machine language

Week 8	Learn to convert from Assembly language to machine language
Week 9	Learn to create code for arithmetic and logical instruction set
Week 10	Learn to create code for arithmetic and logical instruction set
Week 11	Learn to deal with variable and array in emu8086 simulation
Week 12	Learn to deal with variable and array in emu8086 simulation
Week 13	Learn to create code for rotate and shift instruction set
Week 14	Learn to create code for rotate and shift instruction set
Week 15	implemented a code for preparing to the final exam

Learning and Teaching Resources

مصادر التعلم والتدريس

	Text	**Available in the Library?**
Required Texts	1. John Uffenbeck, The 8086Design, Programming and Interfacing. 2012. 2-Barry B. Brey, " The Intel Microprocessors 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, and Pentium Pro Processor rchitecture, Programming, and Interfacing, 6th Edition, Prentic-Hall Inc., 2003.	
Recommended Texts	Complete Notes of Microprocessor with Tutorials and Solutions [1] Published by Raju Dawadi at January 7, 2016 J. T. Streib, Guide to Assembly Language: A Concise Introduction, [2] Springer-Verlag London Limited, 2011	
Websites	• http://www.emu8086.com	

APPENDIX:

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

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

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Logic Design			**Module Delivery**	
Module Type	CORE			**Theory** **Lecture** **Seminar**	
Module Code					
ECTS Credits					
SWL (hr/sem)	60 ساعة				
Module Level	2		**Semester of Delivery**	2	
Administering Department	Computer		**College**	Computer Science for Women	
Module Leader	Hussein A. Lafta		**e-mail**	Wsci.husein.attia@uobabylon.edu.iq	
Module Leader's Acad. Title	Prof. Dr.		**Module Leader's Qualification**	PhD	
Module Tutor	Hussein A. Lafta				
Peer Reviewer Name			**e-mail**		
Review Committee Approval			**Version Number**		

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. Able to perform the conversion among different number systems; Familiar with basic logic gates -- AND, OR & NOT, XOR, XNOR; Independently or work in team to build simple logic circuits using basic. 2. Understand Boolean algebra and basic properties of Boolean algebra; able to simplify simple Boolean functions by using the basic Boolean properties. 3. Able to design simple combinational logics using basic gates. Able to optimize simple logic using Karnaugh maps, understand "don t care". 4. Familiar with basic sequential logic components: SR Latch, D Flip-Flop and their usage and able to analyze sequential logic circuits. 5. Understand finite state machines (FSM) concept and work in team to do sequence circuit design-based FSM and state table using D-FFs. 6. Familiar with basic combinational and sequential components used in the typical data path designs: Register, Adders, Shifters, Comparators; Counters, Multiplier, Arithmetic-Logic Units (ALUs), RAM. Able to do simple register-transfer level (RTL) design. 7. Able to understand and use one high-level hardware description languages (VHDL or Verilog) to design combinational or sequential circuits. 8. Understand that the design process for today s billion-transistor digital systems becomes a more programming-based process than before and programming skills are important.		
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	✓ The student should understand encoder, decoder and multiplexers. ✓ The student should understand flip-flops and how to use them. ✓ The student should understand registers and their types. ✓ The student should understand counters and their types. ✓ The student should understand ROM and PLA implementation.		
Indicative Contents المحتويات الإرشادية	This course covers the logic design advanced concepts. It starts with combinational logic circuit design. From these designs are adder and subtractor. This course also covers the explanation of different circuit such as decoder, encoder and multiplexers. At the end of course, the flip-flop, latches and counter are covered.		
Learning and Teaching Strategies استراتيجيات التعلم والتعليم			
Strategies	✓ The student should use utilities in the lab to apply scientific experiment. ✓ The ability to design a logic circuit.		

Student Workload (SWL) الحمل الدراسي للطالب			
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			

Module Evaluation
تقييم المادة الدراسية

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

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

	Material Covered
Week 1	NUMBERS USED IN DIGITAL ELECTRONICS
Week 2	BASIC LOGIC GATES
Week 3	OTHER LOGIC GATES
Week 4	SIMPLIFYING LOGIC CIRCUITS: MAPPING
Week 5	Offset Address
Week 6	SIMPLIFYING LOGIC CIRCUITS: MAPPING
Week 7	Karnaugh Maps
Week 8	CODE CONVERSION
Week 9	BINARY ARITHMETIC AND ARITHMETIC CIRCUITS
Week 10	FLIP-FLOPS AND OTHER

	MULTMBRATORS
Week 11	COUNTERS
Week 12	Parallel Counters
Week 13	EXAMINATION
Week 14	SHIFT REGISTERS
Week 15	**MICROCOMPUTER MEMORY**

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	number conversation
Week 2	And,OR ,NOT GATES REPRESENTATION
Week 3	NAND,NOR,XOR REPRESENTATION
Week 4	NAND,NOR,XOR REPRESENTATION
Week 5	Karnaugh Maps REPRESANTAION
Week 6	CODE CONVERSION REPRESANTATION
Week 7	BINARY ARITHMETIC AND ARITHMETIC CIRCUITS REPRESANTATION
Week 8	SR FF REPRESANTATION
Week 9	COUNTERS (SERIAL COUNTER) REPRESANTATION
Week 10	Parallel Counters REPRESANTATION
Week 11	EXAMINATION
Week 12	SHIFT REGISTERS REPRESANTAION
Week 13	MICROCOMPUTER MEMORY REPRESANATION
Week 14	Design n-bits Adder Subtractor
Week 15	Design circuit for converting from gray code to binary using XOR Gates.

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	1. John Uffenbeck, The 8086Design, Programming and Interfacing. 2012. 2. Barry B. Brey, " The Intel Microprocessors 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, and Pentium Pro Processor architecture, Programming, and Interfacing, 6th Edition, Prentic-Hall Inc., 2003.	

Recommended Texts	1. Complete Notes of Microprocessor with Tutorials and Solutions Published by Raju Dawadi at January 7, 2016. 2. J. T. Streib, Guide to Assembly Language: A Concise Introduction, Springer-Verlag London Limited, 2011.	
Websites	http://www.emu8086.com	

APPENDIX:

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

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

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	**Data Structures**			**Module Delivery**	
Module Type	**CORE**			▪ **Theory Lecture** ▪ **Particular Lecture** ▪ **Project**	
Module Code	**UOBABCOM21014**				
ECTS Credits	**5**				
SWL (hr/sem)	**125**				
Module Level	2		**Semester of Delivery**	1	
Administering Department	Computer Science		**College**	College of Science for Women	
Module Leader	Prof. Dr. Samaher Al-Janabi		**e-mail**	samaher@uobabylon.edu.iq samaher@itnet.uobabylon.edu.iq	
Module Leader's Acad. Title	Prof.		**Module Leader's Qualification**	Ph.D	
Module Tutor	None		**e-mail**	None	
Peer Reviewer Name			**e-mail**		
Review Committee Approval	15/09/2024		**Version Number**		

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 typical data structures course, which introduces a collection of fundamental data structures and algorithms, can be taught using any of the different programming languages available today. In recent years, more colleges have begun to adopt the Python language for introducing students to programming and problem solving. Python provides several benefits over other languages such as C++ and Java, the most important of which is that Python has a simple syntax that is easier to learn. This book expands upon that use of Python by providing a Python-centric text for the data structures course. The clean syntax and powerful features of the language are used throughout, but the underlying mechanisms of these features are fully explored not only to expose the \magic" but also to study their overall For a number of years, many data structures textbooks have been written to serve a dual role of introducing data structures and providing an in-depth study of object-oriented programming (OOP). In some instances, this dual role may compromise the original purpose of the data structures course by placing more focus on OOP and less on the abstract data types and their underlying data structures. To stress the importance of abstract data types, data structures, and algorithms, we limit the discussion of OOP to the use of base classes for implementing the various abstract data types. We do not use class inheritance or polymorphism in the main part of the text but instead provide a basic introduction as an appendix. This choice was made for several reasons. First, our objective is to provide a \back to basics" approach to learning data structures and algorithms without overwhelming the reader with all of the OOP terminology and concepts, which is especially important when the instructor has no plans to cover such topics. Second, different instructors take different approaches with Python in their first course. 1. focus on the known data structures and algorithms, also designing the examples to allow the introduction of object-oriented programming if so desired. 2. data structures are introduced, with the major details contained in individual sections. 3. Understuend the main principle of Python. • **Prerequisites** This course assumes that the student has completed the standard introduction to programming and problem-solving course using the Python language. Since the contents of the first course can differ from college to college and instructor to instructor, we assume the students are familiar with or can do the following: ✓ Design and implement complete programs in Python, including the use of modules and namespaces		

	✓ Apply the basic data types and constructs, including loops, selection statements, and subprograms (functions) ✓ Create and use the built-in list and dictionary structures ✓ Design and implement basics classes, including the use of helper methods and private attributes
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	**Understanding Fundamental Concepts** ▪ **Define Data**: Explain the concept of data and its significance in computing. ▪ **Define Information**: Distinguish between data and information, emphasizing how data becomes meaningful when processed. ▪ **Define Algorithm**: Describe what an algorithm is and its role in problem-solving within data structures. ▪ **Define Data Structure**: Understand the definition of data structures and their importance in organizing and managing data efficiently. **Arrays** ▪ **One Dimensional Array**: Describe the structure and use cases of one-dimensional arrays. ▪ **Two Dimensional Arrays**: Explain the concept and applications of two-dimensional arrays. ▪ **Three Dimensional Arrays**: Understand three-dimensional arrays and their representation. ▪ **Triangular Matrix**: Define triangular matrices and discuss their applications. ▪ **Representation of Arrays**: Illustrate different methods for representing arrays in memory. **Stack & Notations** ▪ **Stack**: Define stacks, including their properties and operations. ▪ **Main Applications of Stack**: Identify and explain various applications of stacks in computing, such as expression evaluation and backtracking. ▪ **Algorithm of Stack**: Outline algorithms for common stack operations (push, pop, peek). ▪ **Conversion of Infix Expression to Reverse Polish Notation**: Explain the process of converting infix expressions to Reverse Polish Notation using stacks. **Queues** ▪ **Simple Queue**: Define simple queues and their operations. ▪ **Algorithm Insert of Queue**: Describe the algorithm for inserting elements into a queue. ▪ **Algorithm Delete of Queue**: Explain the deletion algorithm for queues. ▪ **Circular Queue**: Define circular queues and their advantages over simple queues. ▪ **Algorithm Insert of Circular Queue**: Outline the insertion algorithm specific to circular queues. ▪ **Algorithm Delete of Circular Queue**: Describe the deletion process for circular queues. **Linked Structures** ▪ **Static Structures**: Define static linked structures and their characteristics. ▪ **Dynamic Structures**: Explain dynamic linked structures and how they differ from static ones. ▪ **Pointers**: Discuss the role of pointers in linked structures. ▪ **Linked List**: Describe linked lists, their types, and uses. ▪ **Algorithm Insert Element to the Start of Linked List**: Outline the algorithm for inserting an element at the beginning of a linked list. ▪ **Algorithm Insert Element to the Middle of Linked List**: Explain how to insert an element in the middle of a linked list. ▪ **Algorithm Insert Element to the End of Linked List**: Describe the process for adding an element at the end of a linked list. **Types of Linked Structures** ▪ **Linked Stack**: Define linked stacks and discuss their implementation. ▪ **Linked Queue**: Explain linked queues and their advantages over simple queues.

	▪ **Circular Linked List**: Describe circular linked lists and their applications. ▪ **Double Linked List**: Discuss double linked lists, including their structure and uses. **Graphics** ▪ **Definition of Graph**: Define what a graph is in computer science terms. ▪ **Types of Graphs**: ✓ *Undirected Graph*: Explain undirected graphs. ✓ *Directed Graph*: Describe directed graphs. ▪ **Graph Representation**: Discuss various methods for representing graphs, including adjacency matrices and adjacency lists. **Types of Edges** ▪ **Primary Path**: Define primary paths within graphs. ▪ **Simple Path**: Explain what constitutes a simple path in graph theory. ▪ **Compound Path**: Discuss compound paths and their characteristics. ▪ **Circular Path**: Define circular paths within graphs. **Trees** ▪ **Tree Types**: Identify different types of trees used in data structures. ▪ **Transformation of a General Tree into Binary Tree**: Explain how to convert general trees into binary trees. ▪ **Tree Traversing Techniques**: ✓ *Level by Level Traversing*: Describe level-order traversal. ✓ *Preorder Traversing*: Explain preorder traversal method. ✓ *In-order Traversing*: Discuss in-order traversal. ✓ *Post-order Traversing*: Outline post-order traversal technique. **Tree Representation** ▪ **General Tree Representation**: ✓ *Max Number of Branches*: Discuss constraints on branches in general trees. ✓ *Two Pointers (Sun, Brother)*: Explain pointer representation for general trees. ✓ *Three-Pointers (Sun, Brother, Father)*: Describe advanced pointer representation techniques. ▪ **Binary Tree Representation**: ✓ *One Dimension Array*: Illustrate binary tree representation using one-dimensional arrays. ✓ *Two Dimensions Array*: Discuss two-dimensional array representations. ✓ *Two Pointers (Left Child, Right Child)*: Explain binary tree node representation using two pointers. ✓ *Three Pointers (Left Child, Right Child, Father)*: Discuss advanced binary tree node representation techniques.
Indicative Contents المحتويات الإرشادية	**1. Fundamental Concepts** ▪ **Data**: Definition and significance in computing. ▪ **Information**: Distinction between data and information. ▪ **Algorithm**: Role and definition of algorithms in problem-solving. ▪ **Data Structure**: Importance and definition of data structures. **2. Arrays** • **One Dimensional Array**: Structure, representation, and use cases. • **Two Dimensional Arrays**: Concept, applications, and representation. • **Three Dimensional Arrays**: Understanding and representation. • **Triangular Matrix**: Definition, properties, and applications. • **Representation of Arrays**: Methods for representing arrays in memory. **3. Stack & Notations** • **Stack**: Definition, properties, and operations. • **Main Applications of Stack**: Use cases in computing (e.g., expression evaluation). • **Algorithm of Stack**: Push, pop, and peek operations. • **Infix to Reverse Polish Notation Conversion**: Process and algorithms involved. 4. **Queues** • **Simple Queue**: Definition, operations, and applications.

- **Insert Algorithm for Queue**: Detailed algorithm for inserting elements.
- **Delete Algorithm for Queue**: Process for removing elements from a queue.
- **Circular Queue**: Definition, advantages, and representation.
- **Insert Algorithm for Circular Queue**: Insertion process specific to circular queues.
- **Delete Algorithm for Circular Queue**: Deletion process for circular queues.

5. Linked Structures

- **Static Structures**: Characteristics and examples.
- **Dynamic Structures**: Differences from static structures.
- **Pointers**: Role of pointers in linked structures.
- **Linked List**: Definition, types, and applications.
  - *Insert Element at Start*: Algorithm for insertion at the beginning.
  - *Insert Element in the Middle*: Algorithm for middle insertion.
  - *Insert Element at End*: Algorithm for end insertion.

6. Types of Linked Structures

- **Linked Stack**: Definition and implementation details.
- **Linked Queue**: Advantages over simple queues and implementation.
- **Circular Linked List**: Structure and applications.
- **Double Linked List**: Characteristics and uses.

7. Graphics

- **Graph Definition**: Basic definition of graphs in computer science.
- **Types of Graphs**:
  - *Undirected Graph*: Characteristics and examples.
  - *Directed Graph*: Characteristics and examples.
- **Graph Representation Methods**:
  - *Adjacency Matrix*: Structure and usage.
  - *Adjacency Lists*: Structure and usage.

8. Types of Edges

- **Primary Path**: Definition and characteristics.
- **Simple Path**: Explanation of simple paths in graphs.
- **Compound Path**: Characteristics of compound paths.
- **Circular Path**: Definition and examples.

9. Trees

- **Tree Types**: Overview of various tree structures (binary trees, AVL trees, etc.).
- **Transformation from General Tree to Binary Tree**: Process of conversion.

10. Tree Traversing Techniques

- *Level by Level Traversing*: Explanation of level-order traversal method.
- *Preorder Traversing*: Description of preorder traversal process.
- *In-order Traversing*: Explanation of in-order traversal method.
- *Post-order Traversing*: Description of post-order traversal process.

11. Tree Representation

- **General Tree Representation Techniques**:
  - *Max Number of Branches*: Constraints on branches in general trees.
  - *Pointer Representation*: Two pointers (Sun, Brother) and three pointers (Sun, Brother, Father).
- **Binary Tree Representation Techniques**:
  - *One-Dimensional Array Representation*: Structure using arrays.
  - *Two-Dimensional Array Representation*: Structure using matrices.
  - *Pointer Representation*: Two pointers (Left Child, Right Child) and three pointers (Left Child, Right Child, Father).

Learning and Teaching Strategies

استراتيجيات التعلم والتعليم

Strategies	**1. Interactive Learning Environments** • **Utilizing Smart Screens**: Incorporate smart screens alongside traditional blackboards to enhance visual learning. This allows for dynamic presentations, interactive demonstrations, and real-time engagement with digital content. **2. Online Resources** • **Lecture Accessibility**: Provide students with access to recorded lectures on the college website and through the learning management system (Moodle). This enables students to review materials at their own pace and reinforces learning through repeated exposure. **3. Collaborative Discussions** • **Discussion Sessions**: Foster an environment that encourages active participation through focused discussion sessions between professors and students. This strategy promotes critical thinking, allows for clarification of concepts, and enhances understanding through peer interaction. **4. Promoting Self-Learning** • **Encouraging Independent Study**: Motivate students to engage in self-directed learning by providing resources and guidance that help them draw their own conclusions. This approach cultivates critical thinking skills and fosters a sense of ownership over their educational journey. **5. Assignments and Activities** • **Graded Activities**: Assign various activities and projects that relate to the course content, allocating a percentage of the overall grade for these tasks. This not only reinforces learning but also encourages students to apply theoretical concepts in practical scenarios, enhancing their understanding of data structures.

Student Workload (SWL)
الحمل الدراسي للطالب

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	65	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	4
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**	2	10% (10)	5, 10	LO #1, 2, 10 and 11
	Assignments	2	10% (10)	2, 12	LO # 3, 4, 6 and 7
	Projects / Lab.	2	10% (10)	Continuous	
	Report	1	10% (10)	13	LO # 5, 8 and 10
Summative assessment	**Midterm Exam**	2 hr	10% (10)	7	LO # 1-7
	Final Exam	2hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus) المنهاج الاسبوعي النظري	
	Material Covered
Week 1	Understanding the concepts of data, information, algorithms, and data structures along with their types.
Week 2	Familiarizing oneself with the types of arrays: one-dimensional, two-dimensional, three-dimensional, and triangular arrays, as well as methods for representing them in memory.
Week 3	Gaining knowledge about stacks, including algorithms for adding and removing elements, their real-world applications, and their use in converting between different notation systems.
Week 4	Exploring various types of queues, such as simple and circular queues, and discussing algorithms for adding and removing elements from different positions (beginning, middle, end).
Week 5	Understanding the fundamental differences between static and dynamic programming.
Week 6	Discussing algorithms for adding and removing elements from various positions within linked structures (beginning, middle, end).
Week 7	Familiarizing oneself with different types of linked structures.
Week 8	Understanding circular and double linked structures, along with their methods for adding and removing elements.
Week 9	Conducting the first practical and theoretical exam.
Week 10	Learning about different types of graphs and methods for representing them in memory and on computers
Week 11	Understanding the various types of edges: primary, simple, compound, and complex edges, as well as how to identify them in any graph.
Week 12	Exploring trees, including how to add elements to them and search for specific elements within them.
Week 13	Learning about different traversal methods for trees.
Week 14	Understanding how to represent general trees and binary trees using dynamic programming.
Week 15	Conducting the second midterm exam.
Week 16	Evaluating projects designed by each student.

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts		
Recommended Texts	1. Problem Solving in Data Structures & Algorithms Using Python, First Edition, By Hemant Jain, 2016 2. Data Structures and Algorithms Using Python, Rance D. Necaise, Department of Computer Science, College of William and Mary, 2011 3. Main Principle of Python and Real Applications in world ; 2023	**yes**
Websites	**https://maxwellacademic.wixsite.com/website**	

APPENDIX:

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

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

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Computer Architecture			**Module Delivery**	
Module Type	CORE			**Theory** **Lecture** **Seminar**	
Module Code	COM2404				
ECTS Credits	6				
SWL (hr/sem)	100				
Module Level	2		**Semester of Delivery**	2	
Administering Department	Computer Science		**College**	College of Science for Women	
Module Leader	Salah Mahdi Saleh		**e-mail**	wsci.salah.alobaidi@uobabylon.edu.iq	
Module Leader's Acad. Title	Lecturer		**Module Leader's Qualification**	PhD	
Module Tutor	None		**e-mail**	None	
Peer Reviewer Name			**e-mail**		
Review Committee Approval	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. Identify some contributors to computer architecture and organization and relate their achievements to the knowledge area. 2. Articulate differences between computer organization and computer architecture. 3. Identify some of the components of a computer. 4. Explain the use of memory hierarchy to reduce the effective memory latency. 5. Explain how interrupts are used to implement I/O control and data transfers. 6. Be able to design an interface to memory 7. Understand how to interface and use peripheral chips 8. Discuss the generation of control signals using hardwired or microprogrammed implementations. 9. Explain basic instruction level parallelism using pipelining and the major hazards that may occur. 10. Discuss the concept of parallel processing beyond the classical von Neumann model		
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	1. Learning the basic concepts of computer architecture. 2. Learn the details of computer architecture types. 3. Understanding the memory addressing modes. 4. The student will be able to learn how the address is calculated. 5. The student will be able to know the memory architecture types and the difference among these types. 6. The student will be able to calculate the performance of processor.		
Indicative Contents المحتويات الإرشادية	1. Introduction to Computer Architecture 2. Basic Computer Organization 3. Data Representation 4. Instruction Set Architecture (ISA) 5. CPU Design and Operation 6. Memory Hierarchy 7. Input/Output Systems 8. Parallel Processing 9. Performance Measurement and Optimization 10. Emerging Trends in Computer Architecture 11. Case Studies		

Learning and Teaching Strategies
استراتيجيات التعلم والتعليم

Strategies	1. Lectures 2. Discussion. 3. Interaction between the lecturer and the students by questions. 4. Google classroom. 5. Reports, Onsite Assignments, Quizzes, and Online Assignments.

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

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	45	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	52	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	1
Total SWL (h/sem) الحمل الدراسي الكلي للطالب خلال الفصل	100		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	**Quizzes**	10	10%(10)	2 to 11	10
	Online Assignment	3	10%(10)	7 and 9	3
	Onsite assignment	2	10%(10)	7 and 8	2
	Report	5	10%(10)	5 to 10	5
Summative assessment	**Midterm Exam**	2hr	10%(10)	12	2hr
	Final Exam	3hr	50% (50)	16	3hr
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

	Material Covered
Week 1	Review of syllabus, Introduction to Computer Architecture. Computer Architecture and Computer Organization.
Week 2	Instruction Set Architecture (CISC and RISC)
Week 3	Classifying Instruction Set Architectures

Week 4	Memory Addressing: Interpreting Memory Addresses, Addressing Modes.
Week 5	Type and Size of Operands
Week 6	Design of CPU Control unit, Microprogrammed vs. Hardwired Control Unit
Week 7	Performance of processor
Week 8	Instruction Pipelining. Arithmetic Pipelining (Integer and Floating point Multiplication).
Week 9	Bus Interface, I/O channels, I/O processor
Week 10	Cache Organization and Operation, Cache references (Direct, Set Associative and Full Associative). Cache performance.
Week 11	Multiprocessor Architecture, Interprocessor Communication Networks
Week 12	**Mid exam**
Week 13	Cache Coherence
Week 14	Associative Memory, Content-Addressable Memories, Arithmetic in Memory
Week 15	Synchronization
Week 16	**Final exam**

Learning and Teaching Resources

مصادر التعلم والتدريس

	Text	**Available in the Library?**
Required Texts	[1] J. L. Hennessy and D. A. Patterson, Computer Architecture : A Quantitative approach, 6th edition, Morgan Kaufmann Publishers Inc., 2019.	No
Recommended Texts	[2] D. M. Harris and S. L. Harris, Digital Design and Computer Architecture", 2nd edition, Elsevier Inc, 2013. [3] W. Stallings, Computer organization and architecture : designing for performance, 10th edition, 2016.	No
Websites		

APPENDIX:

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

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

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Programming Fundamentals			**Module Delivery**	
Module Type	CORE			**Theory** **Lecture** **Seminar**	
Module Code	UOBAB0604011				
ECTS Credits	8				
SWL (hr/sem)	200				
Module Level	1		**Semester of Delivery**	1	
Administering Department	Computer Science		**College**	College of Science for Women	
Module Leader	Hadeel Qasem Gheni		**e-mail**	wsci.hadeel.qasem@uobabylon.edu.iq	
Module Leader's Acad. Title	Lecturer		**Module Leader's Qualification**		
Module Tutor					
Peer Reviewer Name	Majid Jabbar Jawad		**e-mail**		
Review Committee Approval	2023-11-05		**Version Number**		

Relation With Other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module		**Semester**	
Co-requisites module		**Semester**	
Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية			
Module Aims أهداف المادة الدراسية	After completion of this course, the students will be able to understand and explain the principles of the computer programming.		
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Knowledge outcomes 1. The student can describe the algorithm. 2. The student can understand the nature of programming. 3. The student can describe programming languages 4. The student can write a programming code.		
Indicative Contents المحتويات الإرشادية	1. Using smart screens in addition to regular blackboards. 2. Displaying the lecture on the university website as well as on the college and department website. 3. Focusing on discussion sessions between the lecturer and the students. 4. Encouraging self-learning and helping students to draw conclusions. 5. Emphasis on competition among students.		
Learning and Teaching Strategies استراتيجيات التعلم والتعليم			
Strategies	1. Using smart screens in addition to regular blackboards. 2. Displaying the lecture on the university website as well as on the college and department website. 3. Focusing on discussion sessions between the lecturer and the students. 4. Encouraging self-learning and helping students to draw conclusions. 5. Emphasis on competition among students.		

Student Workload (SWL) الحمل الدراسي للطالب			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	3	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3
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	73		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	Continue	10	1	1
	Assignments	Continue	10	1	2
	Projects / Lab.	Continue	10	1	4
	Report	Continue	10	1	3
Summative assessment	Midterm Exam				
	Final Exam				
Total assessment			40		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

	Material Covered
Week 1	Analysis the problem
Week 2	Understanding the algorithms and flowchart
Week 3	Introduction to C++
Week 4	Introduction to C++
Week 5	Understanding the variables declaration
Week 6	Understanding the reading and writing statement
Week 7	Understanding the reading and writing statement
Week 8	Understanding the single if statement
Week 9	Understanding the single if statement
Week 10	Test
Week 11	Understanding the if /else statement
Week 12	Understanding the if /else statement
Week 13	Understanding the if /else statement
Week 14	Understanding the if /else statement
Week 15	Test

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Understanding the operating system
Week 2	Understanding the C++ editor
Week 3	Understanding the C++ editor
Week 4	Understanding the C++ editor
Week 5	Understanding the variables declaration
Week 6	Understanding the reading and writing statement
Week 7	Understanding the reading and writing statement
Week 8	Practical Examples about the single if statement
Week 9	Practical Examples about the single if statement
Week 10	Test
Week 11	Practical Examples about if / else statement
Week 12	Practical Examples about if / else statement
Week 13	Practical Examples about if / else statement
Week 14	Practical Examples about if / else statement
Week 15	Test

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	C++: The Complete Reference Third Edition by Herbert Schildt	
Recommended Texts	C++ Primer (5th Edition) 5th Edition by Stanley Lippman (Author), Josée Lajoie (Author), Barbara Moo (Author)	
Websites	https://www.w3schools.com/cpp/	

APPENDIX:

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

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

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Structured Programming			**Module Delivery**	
Module Type	CORE			**Theory** **Lecture** **Seminar**	
Module Code	UOBAB0604021				
ECTS Credits	8				
SWL (hr/sem)	200				
Module Level	1		**Semester of Delivery**	**2**	
Administering Department	Computer Science		**College**	College of Science for Women	
Module Leader	Hadeel Qasem Gheni		**e-mail**	wsci.hadeel.qasem@uobabylon.edu.iq	
Module Leader's Acad. Title	Assist. Prof.		**Module Leader's Qualification**		
Module Tutor					
Peer Reviewer Name	Majid Jabbar Jawad		**e-mail**		
Review Committee Approval	2024-03-13		**Version Number**		

Relation With Other Modules العلاقة مع المواد الدراسية الأخرى			
Prerequisite module		**Semester**	
Co-requisites module		**Semester**	
Module Aims, Learning Outcomes and Indicative Contents أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية			
Module Aims أهداف المادة الدراسية	After completion of this course, students will be able to explain the basic methods and conclusions of programming language through C++ programming and they will learn other programming languages easily.		
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	Knowledge outcomes. 1. The student can write a computer program perfectly. 2. The student can choose the perfect choice depending on his knowledge. 3. The student can understand the logical relation between individual perfectly. 4. The student can operate with several items such array. 5. The student can build a small project. 6. The student can build user define function according to his requirements.		
Indicative Contents المحتويات الإرشادية	The student can choose the perfect choice depending on his knowledge. 1. The student can understand the logical relation between individual perfectly. 2. The student can operate with several items such array. 3. The student can build a small project.		
Learning and Teaching Strategies استراتيجيات التعلم والتعليم			
Strategies	1. Using smart screens in addition to regular blackboards. 2. Displaying the lecture on the university website as well as on the college and department website. 3. Focusing on discussion sessions between the lecturer and the students. 4. Encouraging self-learning and helping students to draw conclusions. 5. 5. Emphasis on competition among students.		

Student Workload (SWL) الحمل الدراسي للطالب			
Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	3	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	3
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	75		

Module Evaluation
تقييم المادة الدراسية

		Time/Number	Weight (Marks)	Week Due	Relevant Learning Outcome
Formative assessment	Quizzes	Continue	10	1	1
	Assignments	Continue	10	1	1
	Projects / Lab.	Continue	10	1	1
	Report	Continue	10	1	1
Summative assessment	Midterm Exam	1			
	Final Exam	1			
Total assessment			40		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

	Material Covered
Week 1	Understanding the While loop statement
Week 2	Understanding the Do - While loop statement
Week 3	Understanding the For-loop statement
Week 4	Understanding the Nested For loop statement
Week 5	Understanding the One-dimension array
Week 6	Understanding the One-dimension array
Week 7	Understanding the Two dimension array
Week 8	Understanding the Two dimension array
Week 9	Understanding the Two dimension array
Week 10	Understanding the Two dimension array
Week 11	Understanding the Function in C++
Week 12	Understanding the Function in C++
Week 13	Understanding the Function in C++
Week 14	Understanding the Function in C++
Week 15	Understanding the Function in C++

Delivery Plan (Weekly Lab. Syllabus) المنهاج الاسبوعي للمختبر	
	Material Covered
Week 1	Practical Examples about While loop statement
Week 2	Practical Examples about Do - While loop statement
Week 3	Practical Examples about For loop statement
Week 4	Practical Examples about Nested For loop statement
Week 5	Practical Examples about One dimension array
Week 6	Practical Examples about One dimensional array
Week 7	Practical Examples about Two-dimensional array
Week 8	Practical Examples about Two-dimensional array
Week 9	Practical Examples about Two-dimensional array
Week 10	Practical Examples about Two-dimensional array
Week 11	Practical Examples about Function in C++
Week 12	Practical Examples about Function in C++
Week 13	Practical Examples about Function in C++
Week 14	Practical Examples about Function in C++
Week 15	Practical Examples about Function in C++

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	C++: The Complete Reference Third Edition by Herbert Schildt	
Recommended Texts	Programming in CPP	
Websites	https://www.w3schools.com/cpp/	

APPENDIX:

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

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

	Ministry of Higher Education and Scientific Research - Iraq University of Babylon College of Science for Women Computer Science	

MODULE DESCRIPTOR FORM
نموذج وصف المادة الدراسية

Module Information معلومات المادة الدراسية					
Module Title	Computation theory			**Module Delivery**	
Module Type	Core			**Theory** **Lecture** **Seminar**	
Module Code	COM24113				
ECTS Credits	5				
SWL (hr/sem)	125				
Module Level	2		**Semester of Delivery**	4	
Administering Department	Computer science		**College**	Sciw	
Module Leader	Elaf Ali Abbood		**e-mail**	wsci.elaf.ali@uobabylon.edu.iq	
Module Leader's Acad. Title	Lecturer		**Module Leader's Qualification**	Ph. D.	
Module Tutor	None		**e-mail**	None	
Peer Reviewer Name			**e-mail**		
Review Committee Approval	01/06/2023		**Version Number**		

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	On completion of this course, students will be able to explain the basic methods and conclusions of the Theory of Computation. They will be able to apply these methods to problems from different fields and be guided by the results in searching for computational solutions to the problems. Also, this course is offered to undergraduates and introduces basic mathematical models of computation and the finite representation of infinite objects. Topics covered include: finite automata and regular languages, context-free languages, Grammar types, Ambiguous Grammar, Nondeterministic and Deterministic FSA, and Pushdown Automata.
Module Learning Outcomes مخرجات التعلم للمادة الدراسية	• To understand the formal languages and grammars: regular grammar and regular languages, context-free languages and context-free grammar; and introduction to context-sensitive language and context-free grammar, and unrestricted grammar and languages. • To understand the relation between these formal languages, grammars, and machines. • To understand the complexity or difficulty level of problems when solved using these machines. • able to design Finite Automata machines for given problems; • able to analyze a given Finite Automata machine and find out its Language; • able to design Pushdown Automata machine for given CF language(s); • able to generate the strings/sentences of a given context-free languages using its grammar; • able to design Turing machines for given any computational problem.
Indicative Contents المحتويات الإرشادية	
Learning and Teaching Strategies استراتيجيات التعلم والتعليم	
Strategies	Questioning: searching for new information by forming and raising questions. Conclusion: Think beyond the available information to fill in the gaps. Comparison: Noting the similarities and differences between two or more things.

	Classification: putting things into groups according to common characteristics

Student Workload (SWL)
الحمل الدراسي للطالب

Structured SWL (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	47	**Structured SWL (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	2
Unstructured SWL (h/sem) الحمل الدراسي غير المنتظم للطالب خلال الفصل	78	**Unstructured SWL (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	5
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**	2	10% (10)	5, 10	LO #1, 2, 10 and 11
	Assignments	2	10% (10)	2, 12	LO # 3, 4, 6 and 7
	Projects / Lab.	1	10% (10)	Continuous	
	Report	1	10% (10)	13	LO # 5, 8 and 10
Summative assessment	**Midterm Exam**	2 hr	10% (10)	7	LO # 1-7
	Final Exam	2hr	50% (50)	16	All
Total assessment			100% (100 Marks)		

Delivery Plan (Weekly Syllabus)
المنهاج الاسبوعي النظري

	Material Covered
Week 1	Alphabet, String, Formal Language, and Basic Concepts
Week 2	The Grammars: Left Linear Grammar and right Linear Grammar
Week 3	Derivation and Parse Tree
Week 4	Grammar types: Unrestricted, context-sensitive, context-free, regular grammar
Week 5	Ambiguous Grammar
Week 6	Chomosky Normal Form and Greibach Normal Form
Week 7	Regular Expression and Properties of Regular Sets
Week 8	Finite State Automata
Week 9	Nondeterministic FSA and Deterministic FSA

Week 10	Convert NFA into DFA
Week 11	Pushdown Automata
Week 12	Language of Pushdown Automata and examples
Week 13	Turing machine
Week 14	Examples for transitions of Turing machine
Week 15	Mid Exam
Week 16	**Final exam**

Learning and Teaching Resources مصادر التعلم والتدريس		
	Text	**Available in the Library?**
Required Texts	Daniel I. A. Cohen, , Introduction to Computer Theory,2nd, Wiley,1996, ISBN-10: 0471137723	
Recommended Texts	1. Sipser, Michael, Introduction to the Theory of Computation, 3rd ed. Cengage Learning,, 2013 2. Martin, John, Introduction to Languages and the Theory of Computation, New York, NY: McGraw Hill, 2002, ISBN: 0072322004. 3. Kozen, Dexter, Automata Theory , New York, NY: Springer-Verlag, 2016, ISBN: 0387949070.	
Websites		

APPENDIX:

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

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