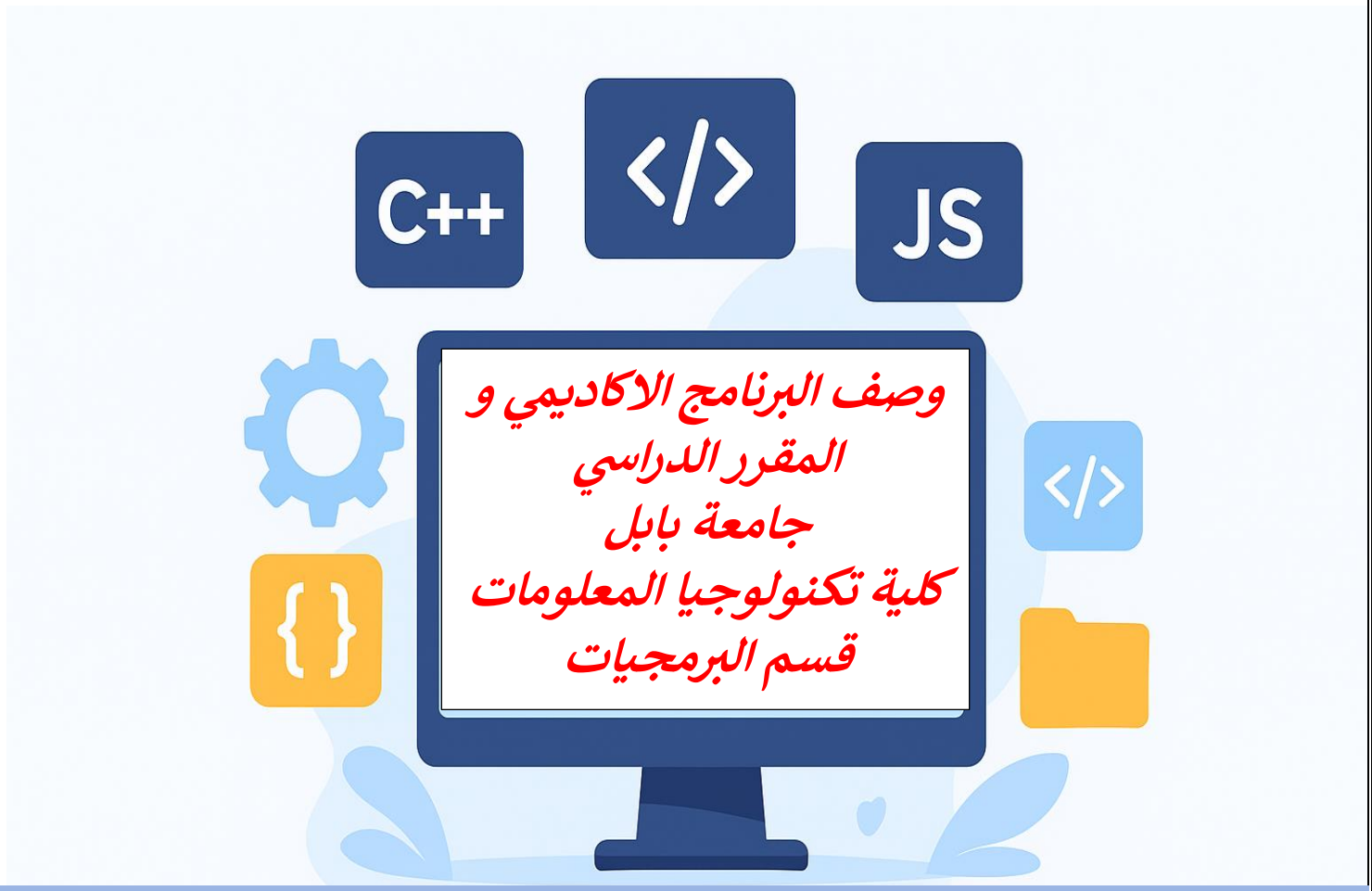
Ministry of Higher Education and Scientific Research
University of Babylon
College of Information Technology
Software Department

وزارة التعليم العالي والبحث العلمي
جامعة بابل
كلية تكنولوجيا المعلومات
قسم البرمجيات



Academic Programme and Module Description

University of Babylon
College of Information Technology
Software Department

2024-2025

Ministry of Higher Education and Scientific Research
University of Babylon
College of Information Technology
Software Department

وزارة التعليم العالي والبحث العلمي
جامعة بابل
كلية تكنولوجيا المعلومات
قسم البرمجيات

وصف البرنامج الاكاديمي
و المقرر الدراسي
جامعة بابل
كلية تكنولوجيا المعلومات
قسم البرمجيات

Academic Programme and Module Description
University of Babylon
College of Information Technology
Software Department
2024-2025

وصف البرنامج الاكاديمي
كلية تكنولوجيا المعلومات- قسم البرمجيات

University:	University of Babylon
College:	College of Information Technology
Department:	Department of Software
Name of Academic programme:	Bachelor Degree in Information Technology, Department of Software
Final Degree:	Bachelor Degree in Information Technology, Department of Software/ Courses
Date of Planning the Description:	1-3-2025
Date of Filling the Form:	1-3-2025

Head of the Department of Quality assurance and Academic Accreditation:
Nawras Nasrullah Khudhair

Signature:
Head of Department:
Asst. Prof. Dr.Sura Zaki Naji
Date: / / 2025

Signature:
Assistant Dean for Scientific Affairs
Prof. Dr. Eman Salih Al-Shammary
Date: 24 / 5 / 2025

Approval of the Dean
Prof. Wesam S. Bhaya
Date: Mar. / 2025

University of Babylon
College Of Information Technology
Dean Office

Introduction

The Software Department at the College of Information Technology – University of Babylon is one of the pioneering academic departments, established in 2009 to serve as a foundation for preparing distinguished professionals in the fields of software analysis, design, development, and maintenance. Over time, the department has expanded to include a comprehensive programme spanning four undergraduate stages (Bachelor's level), in addition to postgraduate programmes covering Master's and Doctorate degrees, thereby ensuring integration between foundational education and advanced scientific research. The department strives to create a balanced academic environment that combines theoretical and practical aspects through modern curricula focusing on advanced programming principles, algorithms, system requirements analysis and management, software project management, and performance testing. Special attention is given to developing students' skills in designing and implementing software solutions that meet the demands of both the public and private sectors. The department also promotes scientific research by providing suitable infrastructure and encouraging faculty members and students to engage in research projects, scientific publications, and academic activities at both the local and international levels. Furthermore, it maintains close academic and administrative collaboration with the Departments of Information Networks and Information Security within the College, fostering interdisciplinary integration and enabling students to appreciate the interconnected nature of various IT specializations. Since its establishment, the department has contributed to graduating competent professionals who have demonstrated their presence in both academic and professional fields through active participation in conferences, seminars, workshops, and specialized exhibitions. In doing so, the department affirms its position as an accredited academic institution that advances education, scientific research, and community service, while responding to the needs of local, regional, and international labour markets in the field of software engineering and development.

وصف البرنامج الاكاديمي
كلية تكنولوجيا المعلومات - قسم البرمجيات

1. Programme Vision

The Software Program aspires to be a leader in academia and research in the field of software development and maintenance, distinguished by preparing graduates who are capable of innovation and competitiveness in the local, regional, and international labor markets.

2. Programme Mission

The Software Programme aims to achieve the following:

1. Graduating specialized professionals with a solid knowledge base and advanced practical skills in software analysis, design, development, and maintenance, capable of keeping pace with technological advancements in the field.
2. Preparing graduates for the labour market by equipping them with the ability to apply modern methodologies and effective software tools in solving real-world problems and developing integrated technological solutions.
3. Strengthening graduates' leadership and teamwork abilities through training in software project management and the application of contemporary methodologies in implementation and evaluation.

3. Programme Objectives

The Software Programme seeks to prepare graduates equipped with both theoretical knowledge and practical skills in software analysis, design, development, and maintenance, enabling them to produce innovative software solutions that meet labour market demands and serve the community.

4. Software Accreditation

There is no Software Accreditation. It is been working on.

5. Other External Influences

Summer training within governmental institutions

6. Programme Structure

Programme Structure	No. of Modules	Credits	Percentage	Other*
Institution Requirements	5	9	%8.2568	
College Requirements	13	42	38.533%	
Department Requirements	20	58	53.211%	
Summer Training	1			
Other				

***It could include whether te module is core or elective**

Level/Year	Course or Module Code	Course or Module Title	Credits	
			Theoretical	Practical
First Stage	**First Course**			
	ItSwCa100303(3+0)	Computer skills I	2	-
	ItSWCO100101(2+2)	Programming Fundamental I	2	2
	ItSwCa100404(3+0)	Calculus I	3	-
	ItSw100202(2+3)	Discrete Structure I	3	2
	ItSwCa100505(2+2)	Digital Logic	2	2
	ItSwCa100606(1+0)	Human Right	1	-
	ItSwCa100606(1+0)	Arabic Language I	1	-
	Second Course			
	ItSwCa101111(3+0)	Computer skills II	2	2
	ItSWCO100808(2+2)	Programming Fundamental II	2	2
	ItSwCa101212(2+2)	Calculus II	3	-
	ItSw100909(2+2)	Discrete Structure II	3	-
	ItSwCa101010(3+0)	Computer Organization	2	2
	ItSwCa101313(1+0)	Arabic Language II	1	-
	ItSwCa100606(1+0)	English Language	1	2

Level/Year	Course or Module Code	Course or Module Title	Credits	
			Theoretical	Practical
Second Stage	**First Course**			
	ItSwCt201501(3+0)	Computation Theory I	2	-
	ItSwDb201602(2+2)	Concepts of Data Base I	2	2
	ItSwOo201703(2+2)	Object Oriented Programming I	3	-
	ItSwNm201804(2+2)	Linear Algebra	3	2
	ItSwCs201905(1+0)	Communication skills	2	2
	ItSwMi202107(2+2)	Microprocessors	1	-
	ItSwEl202108(2+0)	Baath Party Crimes	1	-
	ItSwEl202209(2+0)	English II		
	Second Course			
	ItSwCt202308(3+0)	Computation Theory II	2	2
	ItSwDb202409(2+2)	Concepts of Data Base II	2	2
	ItSwOo202510(2+2)	Object Oriented Programming II	3	-
	ItSwDs202611(2+2)	Data Structure	3	-
	ItSwPs202712(2+2)	Probability and statistic	2	2
	ItSwCg302813(2+2)	Computer Graphics	1	-
	ItSwEl202915(2+0)	Freedom and Democracy	1	2

Level/Year	Course or Module Code	Course or Module Title	Credits	
			Theoretical	Practical
Third Stage	**First Course**			
	ItSwCo302901(2+2)	Compilers I	3	-
	ItSwCa303002(3+0)	Computer Architecture	3	-
	ItSwAd303103(2+2	Algorithm Design and Analysis	2	2
	ItSwSe303204(3+0)	Software Engineering	3	-
	ItSwAi303305(2+2)	Artificial Intelligence	2	2
	ItSwWd303406(2+2)	Web Design and Programming	1	2
	Second Course			
	ItSwCo303507(2+2)	Compilers II	2	2
	ItSwWa303608(2+2)	Web Design and Programming	2	2
	ItSwIp303709(2+2)	Image Processing	2	2
	ItSwHc303810(3+0)	Human Computer Interaction	3	.
	ItSwMl303911(2+2)	Machine Learning	2	2
	ItSwSe304012(2+2)	Software Engineering with Team Project	2	2
	ItSwEl304113(2+0)	English III	2	.

Level/Year	Course or Module Code	Course or Module Title	Credits	
			Theoretical	Practical
Fourth Stage	**First Course**			
	ItSwOs404201(2+2)	Concepts of Operating Systems I	2	2
	ItSwCn404302(2+2)	Computer Networks	2	2
	ItSwCs404403(2+2)	Computing Security I	2	2
	ItSwMc404504(3+2)	Mobile Computation Programming	2	2
	ItSwOr404605(3+0)	Operation research	3	-
	ItSwPr404706(0+4)	Project	-	4
	Second Course			
	ItSwOs404807(2+2)	Concepts of Operating Systems II	2	2
	ItSwGd404908(2+2)	Game Design and programming	2	2
	ItSwIa405009(2+2)	Cloud computing	2	2
	ItSwNs405110(2+2)	Computing Security II	2	2
	ItSwWms405211(3+0)	Modeling and simulation	3	.
	ItSwE1405312(2+0)	English IV	2	.
	ItSwPr405413(0+4)	Project	.	4

8. Learning Outcomes

A. Knowledge	
A1	Know the nature of information technology
A2	Evaluate and build programming systems depending on the bases of software engineering
A3	Know computer network, cloud computing, artificial intelligence, machine learning
A4	Use the computer in most applications
A5	Find scientific solutions to social problems by means of software

B. Skills	
B1	Writes and implements various software projects in accordance with professional and academic standards.
B2	Thinks critically and analyzes problems that arise during code implementation, proposing appropriate solutions.
B3	Keeps up with developments in information technology and modern programming methodologies.
B4	Works effectively within software development teams, engaging with team members to exchange knowledge and expertise.

C. Values	
C1	Commits to academic integrity and professional ethics in all educational and research practices.
C2	Employs programming skills and knowledge in serving the community and embracing social responsibility.
C3	Promotes a culture of teamwork and collaboration across diverse professional and academic contexts.

9. Teaching and Learning Strategies

A. Active learning: Encouraging students to participate in educational activities that include group discussions, puzzle solving, and practical projects that promote critical thinking and practical application of programming concepts.

B. Cooperative learning: Organizing classrooms so that students can cooperate with each other in solving problems and completing software projects, which enhances social interaction and learning from other experiences.

C. Problem-based learning: Providing programming challenges and applied problems to students, and encouraging them to search, analyze problems, and devise effective programming solutions.

D. Diagnostic evaluation: Using various evaluation methods, such as written tests, practical projects, and peer evaluation, to estimate students' progress and understand the extent to which they have achieved the learning objectives.

E. Strengthening the link between theory and practice: organizing lessons so that students can understand theories and programming concepts and apply them in practical projects and solve programming puzzles.

F. Reinforcing interest and participation: Providing stimulating and interesting activities, such as programming challenges and competitions, to increase student engagement and enhance their desire to learn

10. Assessment Methods

A. Quizzes
B. Homework assignments.
C. Semester and final exams for theoretical and practical subjects.
D. Small projects within the lesson.
E. Interaction within the lecture.
F. Reports.
G. Following up and discussing graduation projects.
H. Reports on completion of the summer training programme.
I. Commitment to the specified deadline for submitting the assignments and research papers required of the student.

11.Faculty

	Faculty Members						
No.	Name	Academic title	Specialization		Prerequisites/ Skills (when available)	Faculty members	
			Major	minor		Staff	Non - staff
1	Dr. Tawfeeq Abdulkhaliq Abdulridha Al- Asadi	Prof.	Computer sciences	Image processing		Staff	
2	Dr. Israa Hadi Ali Hassan Al- Shammari	Prof.	Computer sciences	Multi Media and Data Mining		Staff	

3	Dr. Nidaa Abdulmuhsin Abass Abd Hassan Al-Atwani	Prof.	Computer sciences	Artificial Intelligence		Staff	
4	Dr. Asaad Sabah Hadi Abass Al-Jubouri	Prof.	Computer sciences	Artificial intelligence and Data Mining		Staff	
5	Dr. Ahmed Saleem Abass Al-Saffar	Prof.	Computer Engineering	Software and Network Engineering		Staff	
6	Dr. Wafaa Muhammad Saeed Hamza Al- Hameed	Prof.	Computer sciences	Artificial Intelligence and Image Processing		Staff	
7	Dr. Ali Hadi Hassan Abbas Al-Najar	Prof.	Computer sciences	Artificial Intelligence and Developing Techniques		Staff	
8	Dr. Khawla Kadhim Muhammad Radhi Al-Maamouri	Asst. Prof.	Law	Special Law		Staff	
9.	Dr. Ahmed Habib Saeed Al- Azzawi	Asst. Prof.	Computer sciences	Artificial Intelligence and Bio Informaic		Staff	
10	Dr. Sura Zaki Naji Alwan Al- Rashid	Asst. Prof.	Computer sciences	Artificial Intelligence and Bio Informatics		Staff	
11	Dr. Haider Kadhim Zughair Aswad	Asst. Prof.	Mathematics	Dynamic Systems and Data Encryption		Staff	
12	Dr. May Abdulmunim Salih	Asst. Prof.	Computer sciences	Multi Media and Information Security		Staff	
13	Dr. Safa Saad Abbas Ali Al- Murib	Asst. Prof.	Computer sciences	Multi Media and Data Security		Staff	
14	Dr. Muhannad Muhammad Jassim Al- Yasiri	Lect.	Computer sciences	Artificial Intelligence and Bio Informatics		Staff	

15	Dr. Wadhah Rzouqi Aboud Hassan Baai	Lect.	Computer sciences	Information Systems and Multi Media		Staff	
16	Dr. Hazim Jalil Hassan Mhesin Abu Rageef	Lect.	Computer sciences	Network and Artificial Intelligence and		Staff	
17	Dr. Iman Kadhim Abbood	Lect.	Computer sciences	Multimedia		Staff	
18	Dr. Ruaa Safaa Hussein Muhammad Shubbar	Lect.	Computer Engineering	Networks		Staff	
19	Mazin Kadhim Hameed Ali Al-Mansouri	Lect.	Computer sciences	Wireless Sensor Network		Staff	
20	Dr. Hiba Muhammad Jaafar	Lect.	Computer sciences	Information Security		Staff	
21	Firyal Jasim Abd Razzaq Al-Hmeadawi	Lect.	Computer sciences	Data Security and Data Mining		Staff	
22	Dr. Muhammad Jawad Kadhim Al-Janabi	Lect.	Computer sciences	Data Security		Staff	
23	Dr. Hawraa Sharif Hamzah Hassan Al-Murshidi	Lect.	Computer sciences	Machine Learning		Staff	
24	Dr, Sura Jasim Muhammad Abdulameer Alhajjaj	Lect.	Information Technology	Data Mining		Staff	
25	Dr. Hussein Ali Ismael	Lect.	Information Technology	Artificial Intelligence		Staff	
26	Dr. Ameer Al Haq Adil Sahib Ali	Lect.	Information Technology	Artificial Intelligence		Staff	
27	Dr. Raed Ghazi Hameed Al-Azzawi	Lect.	Computer sciences	Data Mining		Staff	
28	Hawraa Abduikadhim Hassan Al-Hay	Lect.	Computer sciences	Information Networks		Staff	
29	Zahraa Adnan Fadhil Al- Murib	Lect.	English	Linguistics		Staff	
30	Zainab Abdullah Jasim Muhammad	Asst. Lect.	Computer Sciences	Data Security		Staff	

	Jaafar						
31	Duaa Aed Muhammad	Asst. Lect.	Information Technology	Networks		Staff	
32	Sara Abdulridha Abd Muhajhaj	Asst. Lect.	Information Technology	Multi Media		Staff	
33	Zahraa Mazin Bahlul	Asst. Lect.	Information Technology	Information Networks		Staff	
34	Abrar Saad Khadhim	Asst. Lect.	Information Technology	Information Networks		Staff	
35	Rajaa Mahmoud Kareem	Asst. Lect.	Information Technology	Information Networks		Staff	
Total :35							

Professional Development
Guidance for New Faculty Members
a. Ensure a thorough understanding of the curricula and materials you will be teaching, and familiarize yourself with the available curricula and educational resources. b. Build positive relationships with students. c. Communicate with faculty members by establishing respectful and cooperative relationships, and be prepared to exchange ideas and experiences. d. Develop teaching skills and use diverse and effective teaching methods to capture students' attention and enhance their engagement and understanding. e. Prepare lessons thoroughly; strive to be well-prepared for each class through careful planning, and ensure that all necessary resources are available to students. f. Keep up with technological advancements. g. Ensure adherence to academic ethics and university regulations, and promote academic integrity among students. h. Engage in continuous research and development by updating your knowledge and skills through reading, research, and participation in professional development programs. i. Be open to receiving feedback and evaluation from students, colleagues, and supervisors, and use it as an opportunity for improvement and continuous growth.

Professional Development for Faculty Members
The professional development of faculty members in the Software Department, College of Information Technology, University of Babylon, is considered an essential part of the department's strategy to enhance the quality of education, research, and community service. The following are some key aspects of faculty professional development in the Software Department: a. **Training and Workshops:** The college organizes workshops and training courses for faculty members to update their skills and knowledge with the latest technologies and software development tools. b. **Participation in Conferences and Seminars:** Faculty members are encouraged to participate in local and international scientific conferences and seminars to exchange experiences and ideas, and to expand their academic and professional networks. c. **Scientific Research:** Providing support and encouragement for faculty members to engage in scientific research and publish their findings in peer-reviewed journals. d. **Periodic Evaluation:** Conducting regular evaluations of faculty performance and providing feedback and guidance to improve performance and develop skills. e. **Mentorship and Guidance:** Offering mentoring programs and individualized support to assist faculty members in achieving their professional and academic goals. f. **Active and Interactive Learning:** Promoting active and interactive teaching methods that involve student participation in learning processes and encourage inquiry and research. g. **Technological Development:** Providing technical and technological support for faculty members to learn and utilize modern technologies in teaching and research.

13. Admission Standard

Morning study admission (central admission):

Students are accepted according to the criteria mentioned in the Student Affairs Procedures Manual and admission controls and conditions for the academic year (2024-2025).

Evening study acceptance:

1. Acceptance of all high school graduates in biology, applied sciences, and occupational studies (computers).
2. A Grade Point Average of no less than 62.
3. Tuition Fee: one million and a half is paid in installments.
4. Applying through the application link and fill out the required form before the deadline.

14. Sources of Information about the Programme

Ministry of Higher Education and Scientific Research

College of Science Councils

Website of the College of Information Technology

15. Programme Development Plan

a. Conduct a comprehensive evaluation of the current program, including an analysis of curricula, course materials, and supplementary resources. Collect feedback from students, graduates, and faculty members regarding strengths, weaknesses, and challenges. Assess labour market needs and technological advancements to update curricula and course materials.

b. Redesign the curricula to align with labour market requirements and modern technological developments. Incorporate specialized courses that reflect emerging technologies and tools in programming and software development. Diversify and enhance teaching methods to encourage student interaction and active participation.

c. Offer training programs and workshops to strengthen students' skills in programming and software development. Provide opportunities for applied learning through practical projects and hands-on experiences. Enhance academic advising and student support.

d. Establish mechanisms for evaluating the programme's sustainability and its achievement of defined objectives. Conduct student satisfaction surveys and alumni feedback surveys to assess the quality of the academic experience. Introduce mechanisms for continuous feedback and ongoing programme adjustments in response to market demands and student expectations.

e. Carefully implement the plan and initiatives according to the specified timeline. Regularly monitor and evaluate programme performance to ensure desired outcomes. Adapt and continuously improve the program in accordance with changes in the academic environment, labour market, and student aspirations.

Curriculum Skills Map

Required Learning Outcomes of the Programme

Year	Course Code	Course Title	Core/Optional Core	Knowledge			objectives					Values			
				A1	A2	A3	A4	A5	B1	B2	B3	B4	C1	C2	C3
First Year – First and Second Course	ItSWCO100101 (2+2)	Computer skills I	Core	*			*					*		*	
	ItSw100202(2+3)	Programming Fundamental I	Core		*			*	*	*			*		*
	ItSwCa100303(3+0)	Calculus I	Core		*			*		*			*		
	ItSwCa100404(3+0)	Discrete Structure I	Core		*					*			*		
	ItSwCa100505(2+2)	Digital Logic	Core				*			*			*		
	ItSwCa100606(1+0)	Arabic Language	Core										*		*
	ItSwCa100707(1+0)	Human Right	Core										*		
	ItSWCO100808 (2+2)	Computer skills II	Core	*			*					*		*	*
	ItSw100909(2+3)	Programming Fundamental II	Core		*			*	*	*				*	
	ItSwCa101010(3+0)	Calculus II	Core		*			*		*			*		
	ItSwCa101111(3+0)	Discrete Structure II	Core		*					*			*		
	ItSwCa101212(2+2)	Computer Organization	Core	*			*				*			*	
	ItSwCa101313(1+0)	Arabic Language II	Core												*
	ItSwCa101414(2+0)	English I													*

Year	Course Code	Course Title	Core/Optional Core	Knowledge objectives								Values				
				A1	A2	A3	A4	A5	B1	B2	B3	B4	C1	C2	C3
Second Year – First and Second Course	ItSwCt201501(3+0)	Computation Theory I	Core		*					*				*	
	ItSwDb201602(2+2)	Concepts of Data base I	Core		*			*	*	*				*	*
	ItSwOo201703(2+2)	Object Oriented Programming I	Core		*			*	*	*				*	
	ItSwNm201804(2+2)	Linear Algebra	Core		*					*					
	ItSwCs201905(1+0)	Communication skills	Core												
	ItSwMi202007(2+2)	Microprocessors	Core		*						*			*	*
	ItSwEl202108(2+0)	Baath Regime Crimes	Core										*		
	ItSwEl202209(2+0)	English Language	Core												*
	ItSwCt202308(3+0)	Computation Theory II	Core		*					*				*	
	ItSwDb202409(2+2)	Concepts of Data Base II	Core		*			*	*	*				*	
	ItSwOo202510(2+2)	Object Oriented Programming II	Core		*			*	*	*				*	
	ItSwDs202611(2+2)	Data Structure	Core		*				*	*				*	
	ItSwPs202712(2+2)	Probability and statistics	Core		*					*				*	*
	ItSwCg302813(2+2)	Computer Graphics	Core		*							*		*	*
	ItSwEl202915(2+0)	Freedom and Democracy	Core										*		

Year	Course Code	Course Title	Core/Optional Core	Knowledge			objectives					Values				
				A1	A2	A3	A4	A5	B1	B2	B3	B4	C1	C2	C3	
Third Year – First and Second Course	ItSwCo302901(2+2)	Compilers I	Core	*				*		*			*			
	ItSwCa303002(3+0)	Computer Architecture	Core	*			*					*	*			
	ItSwAd303103(2+2	Algorithm Design and Analysis	Core		*			*		*				*		
	ItSwSe303204(3+0)	Software Engineering	Core	*	*		*					*		*	*	
	ItSwAi303305(2+2)	Artificial Intelligence	Core			*	*				*			*		
	ItSwWd303406(2+2)	Web Design and Programming	Core			*						*	*			
	ItSwCo303507(2+2)	Compilers II	Core	*				*		*			*			
	ItSwWa303608(2+2)	Web Design and Programming	Core			*						*	*			
	ItSwIp303709(2+2)	Image Processing	Core	*				*			*		*			
	ItSwHc303810(3+0)	Human Computer Interaction	Core	*			*				*	*		*	*	
	ItSwMl303911(2+2)	Machine Learning	Core		*	*					*			*		
	ItSwSe304012(2+2)	Software Engineering with Team Project	Core		*			*	*			*	*	*	*	
	ItSwEl304113(2+0)	English III	Core												*	

Year	Course Code	Course Title	Core/Optional Core	Knowledge			objectives						Values		
				A1	A2	A3	A4	A5	B1	B2	B3	B4	C1	C2	C3
Fourth Year – First and Second Course	ItSwOs404201(2+2)	Concepts of Operating Systems I	Core	*	*					*				*	
	ItSwCn404302(2+2)	Computer Networks	Core			*					*			*	
	ItSwCs404403(2+2)	Computing Security	Core	*	*					*			*		
	ItSwMc404504(2+2)	Mobile Computation Programming	Core		*		*	*			*				*
	ItSwOr404605(3+0)	Operation research	Core				*	*		*				*	
	ItSwPr404706(0+4)	Project	Core				*	*	*	*	*		*	*	
	ItSwOs404807(2+2)	Concepts of Operating Systems II	Core	*	*					*				*	
	ItSwGd404908(2+2)	Game Programming and Design	Core		*		*	*			*	*	*		*
	ItSwIa405009(2+2)	Cloud Computing	Core		*		*				*			*	
	ItSwNs405110(2+2)	Computing Security II	Core	*	*					*			*		
	ItSwWms405211(3+0)	Modeling and Simulation	Core	*			*			*			*		
	ItSwE1405312(2+0)	English IV	Core												*
	ItSwPr405413(0+4)	Project	Core				*	*	*	*	*		*	*	

Course Description

Department of Software / College of Information Technology

This course introduces the fundamental concepts of structured programming and provides a comprehensive introduction to programming for computer science and technology majors. Topics include one dimensional arrays, two dimensional arrays, methods, and pointers in Java.

1. The Educational Institution	University of Babylon
2. University Department / Center	Department of Software
3. Course Name / Code	Programming Fundamentals II
4. Name of Academic programme:	Bachelor
5. Available Attendance Modes	Weekly
6. Semester / Year	Semester / Year: Course
7. Total Credit Hours	60
8. Date of Preparing this Description	29/6/2022

9. Course Objectives

This course aims to:

1. Demonstrate the use of one-dimensional arrays.
2. Implement several different ways of dealing with one-dimensional arrays.
3. Demonstrate the use of two-dimensional arrays.
4. Implement several different ways of dealing with two-dimensional arrays.
5. Introduce the construction of methods in Java.
6. Discuss the ways of calling methods in Java and deal with methods parameters.
7. Build a list in Java and learn how to deal with different concepts of pointers in Java

10. Learning Outcomes and Methods of Teaching, Learning, and Assessment

Upon completion of this course, the student should be able to:

1 -Define the key terms of arrays.

2 -Classify the ways of dealing with arrays.

3 -Learn building one and two-dimensional arrays.

4 -Learn different concepts of methods in Java.

5 -Learn how to build methods in Java and how to deal with parameters.

6 -Learn the concept of lists and pointers in Java.

B. Subject-Specific Skills

This course will provide the ability for Computer Science students to analyze and convert every problem into diagram, algorithm, and then into Java code program.

12. Assessment Methods

1. Active participation,
2. Online participation
3. Homework assignments,
4. Attendance, quizzes ,
5. Seminars
6. Middle Exams.
7. Practical Exams.
8. Final exam.

13. Course Structure

First Week	Intended Learning Outcomes	Intended Learning Outcomes	Name of the Unit / Course or Topic	Teaching Method	Assessment Methods
Second Week	4	Key concepts	One dimensional arrays I	Theoretical and Practical	Quiz and Practical projects
Third Week	4	Implementing one dimensional arrays	One dimensional arrays II	Theoretical and Practical	Quiz and Practical projects
Fourth Week	4	Examples of different	One dimensional	Theoretical and Practical	Quiz and Practical projects

		programs	arrays III		
Fifth Week	4	Key concepts	Two dimensional arrays I	Theoretical and Practical	Quiz and Practical projects
Sixth Week	4	Implementing one dimensional arrays	Two dimensional arrays II	Theoretical and Practical	Quiz and Practical projects
Seventh Week	4	Exam I	Exam I	Theoretical and Practical	Quiz and Practical projects
Eighth Week	4	Examples of different programs	Two dimensional arrays III	Theoretical and Practical	Quiz and Practical projects
Ninth Week	4	Key concepts	Methods I	Theoretical and Practical	Quiz and Practical projects
Tenth Week	4	Implementing methods	Methods III	Theoretical and Practical	Quiz and Practical projects
Eleventh Week	4	Examples of different methods	Methods III	Theoretical and Practical	Quiz and Practical projects
Twelfth Week	4	Key concepts	Pointers I	Theoretical and Practical	Quiz and Practical projects
Thirteenth Week	4	Examples of pointers	Pointers III	Theoretical and Practical	Quiz and Practical projects
Fourteenth Week	4	Course review	Course review	Theoretical and Practical	Quiz and Practical projects
Fifteenth Week	4	Examples of pointers	Pointers III	Theoretical and Practical	Quiz and Practical projects

10. Infrastructure	
Required Readings: • **Textbook** • **Internet** • **Follow the College Website**	• Core Java Volume I – Fundamentals (11th Edition), Author – Cay S. Horstmann, Publisher – Prentice Hall • Java: A Beginner's Guide, Author – Herbert Schildt, Edition – 8th, Publisher – McGraw-Hill Education • www.w3school.com
Special Requirements (including, for example, workshops, journals, software, and websites)	• Internet Courses • YouTube Lessons • Cpp.sh page • Stackoverflow site
Social Services (including, for example, guest lectures, professional training, and field studies)	

Course

Department of Software / College of Information Technology

Human rights are fundamental standards without which people cannot live with dignity as human beings. They are the basis of freedom, justice, and equality, and respecting these principles leads to the full development of the individual and society. The term “human” refers to every individual of the human species (male or female). Therefore, God’s message in most holy books and philosophies is directed to all people regarding the respect for human life and dignity. The importance of studying human rights can be summarized as follows:

1. The human is the universe, and the mind is its core.
2. Establishing principles of dignity, humanity, freedom, and peace.
3. Building humane societies with free, independent administrations, away from tyranny, injustice, and oppression.
4. Allowing individuals to exercise their basic rights within a framework of justice.
5. Authority derives from the people; thus, one cannot deny the rights of the true owner, which is the people.
6. Studying and educating about human rights clarifies them, leaving no room for political tyranny and dictatorships, while promoting transparency, openness to civilizational development, and engagement with all human-centered studies.

Characteristics of Human Rights: The main characteristics of human rights are as follows:

1. Human rights cannot be bought, acquired, or inherited; they simply belong to people because they are human and are inherent in every individual.
2. Human rights are equal for all humans, regardless of race, gender, religion, political opinion, or national or social origin. We are all born free and equal in dignity and rights; human rights are universal.
3. Human rights cannot be taken away; no one has the right to deprive another of their rights. They are fixed and inalienable.
4. To live with dignity, all people have the right to freedom, security, and an adequate standard of living; human rights are indivisible.

Concept of Human Rights

When discussing human rights, it is necessary to refer to key documents, especially the Universal Declaration of Human Rights, issued by the United Nations General Assembly on December 10, 1948. This declaration is the main source of ideas about human rights and fundamental freedoms in the modern era. Accordingly, the United Nations defines human rights as: *"Global legal guarantees that protect individuals and groups from actions that interfere with the enjoyment of fundamental freedoms and human dignity."*

1. The Educational Institution	University of Babylon
2. University Department / Center	Software Department
3. Course Name / Code	Freedom and Democracy
4. Name of Academic programme:	Bachelor
5. Available Attendance Modes	Weekly
6. Semester / Year	Semester-based
7. Total Credit Hours	15
8. Date of Preparing this Description	29/6/2022

9. Course Objectives

They are considered guarantees that address every aspect of human life and interaction. Among the rights guaranteed to all human beings, which must be recognized, are the following:

- The right to life**,** which includes the right to exercise certain personal freedoms such as freedom of movement, the use of one's talents, and the organization of one's way of living.
- The right not to be subjected to torture, ill-treatment, cruel, inhuman, or degrading punishment.
- The right to a fair trial.
- The right to enjoy legal protection on an equal basis with others.
- The right to the inviolability of one's private life, family, home, and correspondence.
- The right to health, education, property, and contractual relations.
- The right to form associations, freedom of speech, freedom of opinion, and participation in cultural life.
- The right to vote, contribute, and participate in governance.
- The right of the individual to secure and provide for basic needs such as food, shelter, clothing, and social security.
- The right to development and to benefit from its outcomes.
- The right to freedom of thought, conscience, and religious belief.

10.Learning Outcomes and Methods of Teaching, Learning, and Assessment

A- Knowledge and Understanding

1. The student should be able to identify and understand rights and freedoms.
2. The student should be able to identify and understand the guarantees of human rights.
3. The student should be able to identify and understand the fundamental rights inherent to the human person, which cannot be waived.
4. The student should be able to identify and understand the concept of rights and the meaning of a right.

B - Subject-Specific Skills

1. The student should be able to defend his/her rights.
2. The student should be able to understand the rights that cannot be waived and recognize the guarantees of those rights.

11.Teaching and Learning Methods

1. Using e-learning by creating online classes for students, delivering and discussing lectures through the Meet program, and then uploading the lecture in Word and PDF formats.
2. Assigning students to prepare reports on the course topics.
3. Preparing questions and answers by students related to the lecture.

12.Assessment Methods

1. Discussion
2. Interaction between the instructor and students through questions
3. Quiz (short/quick test)

Course Structure

Week	Hours	Intended Learning Outcomes	Name of the Unit / Course or Topic	Teaching Method	Assessment Method
First Week	1		Meaning of Democracy	Theoretical	
Second Week	1		The Historical Development of the Concept of Democracy	Theoretical	

Third Week	1		Democracy in Western Societies	Theoretical	
Fourth Week	1		Democracy in Eastern Societies	Theoretical	
Fifth Week	1		Definition of Democracy	Theoretical	
Sixth Week	1		The Relationship Between Individual Rights, Public Freedoms, and Democracy	Theoretical	
Seventh Week	1		Forms of Democracy	Theoretical	
Eighth Week	1		Direct Democracy	Theoretical	
Ninth Week	1		Representative Democracy	Theoretical	
Tenth Week	1		Conditions and Elements of a Democratic System	Theoretical	
Eleventh Week	1		Components and Elements of Democracy	Theoretical	
Twelfth Week	1		The Concept of Election	Theoretical	
Thirteenth Week	1		Election Requirements	Theoretical	
Fourteenth Week	1		Legal Adaptation of Elections	Theoretical	
Fifteenth Week	1		Types of Electoral Systems	Theoretical	

Course

Department of Software / College of Information Technology

This is the second course in the introductory programming sequence. Building upon basic programming skills in C++ from fundamentals of programming, this course focuses on the design and analysis of larger, more complex programs using the industry-leading language, Java.

1. The Educational Institution	University of Babylon
2. University Department / Center	Software Department
3. Course Name / Code Object-	oriented programming II- ItSwOo201703(2+3)
4. Programs in Which It Is Included	Bachelor
5. Available Attendance Modes	Weekly
6. Semester / Year	Semester
7. Total Credit Hours	60
8. Date of Preparing this Description	6/6/2022

Course Objectives

The specific goals of this course are:

• To know and use basic Java programming constructs for object-oriented problem solving (e.g., classes, polymorphism, inheritance, interfaces)

• To be able to design and implement a Java program to model a real-world system, and subsequently analyse its behavior.

• To develop programming skills that can serve as a foundation for further study in Software.

11. Learning Outcomes and Methods of Teaching, Learning, and Assessment

A- Knowledge and Understanding

1- Learn the concepts of object-oriented programming
2- Use Java language as an example of languages which applies these concepts.

Apply these concepts to solve real world problems

B. Subject-Specific Skills

1. Develop programming skills that can serve as a foundation for further study in software development.
2. Use Java programming constructs for object-oriented problem solving.

12. Teaching and Learning Methods

1. Lecturing
2. Classroom discussion
3. Demonstrating

13. Assessment Methods

1- Daily quizzes/exercises
2- Programming assignment
3- Two midterm tests
4- Final exam

C- Thinking Skills

1- Problem solving
2- Evaluate ideas and make right decisions
3- Logical thinking

D - General and Transferable Skills (Other Skills Related to Employability and Personal Development)

1- Time management skills
2- Problem solving skills
3- Communication skills
4- Ability to accept and learn from criticism

14. Course Structure					
Week	**Hours**	**Name of the Unit / Course or Topic**	**Intended Learning Outcomes**	**Teaching Method**	**Assessment Method**
First Week	4	Course overview	Remember	Lecturing	Exercises
Second Week	4	Using library classes to implement some more advanced functionality	Understand	Lecturing	Exercises
Third Week	4	Further library classes – Set and map .	Understand & apply	Lecturing and demonstrating	Quizzes and exercises
Fourth Week	4	Improving structure with inheritance	Understand & apply	Lecturing and demonstrating	Quizzes and exercises
Fifth Week	4	More about inheritance - Exploring polymorphism	Understand and apply	Lecturing and demonstrating	Programming assignment
Sixth Week	4	Further abstraction techniques- Abstract classes	Create	Lecturing demonstrating	Quizzes and exercises
Seventh Week	4	1st exam	-------	---------	Written exam
Eighth Week	4	Further abstraction techniques- interfaces	Remember	Lecturing and demonstrating	Quizzes and exercises
Ninth Week	4	Building Graphical User Interfaces 1	Create	Lecturing and demonstrating	Quizzes and exercises
Tenth Week	4	Building Graphical User Interfaces 2	Understand & apply	Lecturing demonstrating	Exercises

Eleventh Week	4	Designing classes - cohesion and coupling	Analysis & evaluate	Lecturing demonstrating	Quizzes and exercises
Twelfth Week	4	2nd exam	-------	---------	Written exam
Thirteenth Week	4	Well-behaved objects	Understand & apply	Lecturing demonstrating	Programming assignment
Fourteenth Week	4	File – based input/output	Analysis & evaluate	Lecturing demonstrating	Exercises
Fifteenth Week	4	Revision week	Remember	Lecturing	Exercises

15. Infrastructure

Required Readings: • **Textbook** • **Internet** **Following the College Website**	• Objects First with Java, 5th edition, David Barnes and Michael Kolling, Prentice Hall, 2012. • Java Concepts, 6th edition, Cay Horstmann, John Wiley & Sons, 2010.
Special Requirements **(including, for example, workshops, journals, software, and websites)**	Using online tools for writing and running java code in a friendly interactive manner.
Social Services (including, for example, guest lectures, professional training, and field studies)	

Course Description

Department of Software / College of Information Technology

The Course Description provides a concise summary of the main features of the course and the expected learning outcomes that the student is anticipated to achieve, demonstrating whether the student has maximized the available learning opportunities. It is essential to link this description with the program description.

The Educational Institution	University of Babylon
University Department / Center	Software Department
Course Name / Code	COTH201
Programs in Which It Is Included	Bachelor
Available Attendance Modes	Weekly
Semester / Year	Semester
Total Credit Hours	48
Date of Preparing this Description	10/12/2014

11. **Course Objectives**

1. Memorize the Introductory to the Computational theory and Automata Languages.
2. Describe the basics of Set Notation.
3. Convert and system model to the BNF- Notation and to the Syntax Chart
4. develop the Equivalent Grammar, Type of Grammars, Context Sensitive Grammar
5. Determine the Context Free Grammar, Useless Symbols
6. Design models in Chomsky Normal Form CNF
7. Design models in Greibach Normal Form GNF
8. Remember Regular Grammars LEFT, RIGHT linear
9. Synthesize Finite State Automata with specific properties.
10. Discuss Non determinism NFA

11. Perform the Equivalent NFA with DFA
12. Perform the Equivalent DFA with Regular Grammar
13. Perform Equivalence Regular Grammar with NFA
14. Convert FA with and without Empty moves
15. Convert Equivalence NFA with and without Empty moves
16. Discuss Regular Expressions.
17. Analyze Equivalence of FA and Regular Expression
18. Convert between Tow Way FA's.
19. Use Turing Machines
20. Design Turing Machines
21. Analyze Turing Machines

10. Learning Outcomes and Methods of Teaching, Learning, and Assessment

A- Knowledge and Understanding

1- The student should understand the fundamental concepts of computational theory.
2- The student should gain an overview of the topics that will be studied in the upcoming weeks.
3- The student should review previously studied material related to computational theory.

B - Subject-Specific Skills

1 – Enable the student to think critically and predict appropriate solutions for any topic during its explanation.
2 – Provide the student with the opportunity to express their opinion on the topic.

11. Methods of Teaching and Learning

1- Lecture Delivery Method
2- Discussion.
3- Team Project
4- Interaction between the lecturer and the students by questions.
5- Modeling exam.
6- Learning Technologies on Campus .
7- Experiential Learning Application Learning.

12. Assessment Methods

1. Asking direct questions to students.
2. Providing immediate feedback to students who participate in answering questions.

C. Thinking Skills

1. Presenting multiple answers to a single question to encourage the student to think and identify the correct answer.
2. Asking a specific question and requiring the student to think through the answer rather than respond immediately.

13. Course Structure

Week	Hours	Intended Learning Outcomes	Name of the Unit / Course or Topic	Teaching Method	Assessment Method
First Week	3	Basic Principles and General Review	INTRODUCTION TO COMPUTATION THEORY & SET THEORY:(set, string, alphabets and language).	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback
Second Week	3	Languages and Regular Grammars	LANGUAGES:(The Chomsky hierarchy of language ,The regular grammars and regular language, pumping lemma on regular languages ,closure properties of regular sets	Explaining lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and giving immediate feedback

Third Week	3	Finite Automata	Finite state Automata	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback
Fourth Week	3	Finite Automata	D. Finite state Automata	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and giving immediate feedback
Fifth Week	3	Non-deterministic Automata	Nondeterministic FSA (The Moore and Meal Machine)	Explaining lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking students questions and providing immediate feedback
Sixth Week	3	Regular Expression	Regular Expressions	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback
Seventh Week	3	Converting a regular expression to a finite automaton and converting a non-deterministic automaton to a deterministic one	Converting Regular expression into FSA and equivalence between D. Finite state Automata and Nondeterministic FSA	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback

Eighth Week	3	Formal Languages	Formal languages, pumping lemma on regular languages	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback
Ninth Week	3	Diagrams and BNF Notation–	Syntax charts BNF notation	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback
Tenth Week	3	Ambiguous and Context-Free Grammars	Grammars and The ambiguous context free grammars.	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback
Eleventh Week	3	Languages in General Chomsky Normal	Chomsky Normal form	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback
Twelfth Week	3	Languages in General Greibach Normal	Greibach Normal form	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback

Thirteenth Week	3	Theory Kleene	Kleene Theorem	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback
Fifteenth Week	3	TURING MACHINE	TURING MACHINE	Delivering lectures on the screen and board, and uploading the lecture to the official college website and the Moodle platform	Asking questions to students and providing immediate feedback

13. Infrastructure	
Required Readings: • **Core Lectures** • **Textbooks** • **Others**	1. Daniel I. A. Cohen. Introduction to Computer Theory. 2nd ed .Wiley. 1996. ISBN-10: 0471137723 . 2. Sipser, Michael. Introduction to the Theory of Computation. 2nd ed. Boston, MA: Course Technology, 2006. ISBN: 0534950973. 3. Martin, John. Introduction to Languages and the Theory of Computation. New York, NY: McGraw Hill, 2002. ISBN: 0072322004. 4. Kozen, Dexter Automata and Computability .New York ,NY:Springer Verlag,1999.ISBN:0387949070.
Special Requirements	
Social Services	

Course

Department of Software / College of Information Technology

In this course, the student will learn how to build and develop a web application using many technologies like CSS, JavaScript, C#, SQL Server and ASP.Net Core.
Farther more, the course will guide the students to find and protect weakness in web applications in addition to the learning of the hosting techniques and remote access using different methods.

1. The Educational Institution	University of Babylon
2. University Department / Center	Software Department
3. Course Name / Code	Web Application
4. .4Programs that include	Bachelor
5. Available Attendance Modes	Weekly
6. Semester / Year	quarterly
7. Total Credit Hours	60
8. Date of Preparing this Description	26/6/2022

9. Course objectives.

1. Design Responsive UI using Html, CSS and Bootstrap.
2. Develop secure web application based on user roles.
3. Develop performance based web applications using caching and Ajax calls.
4. Create REST Service using Web API.
5. Handle Errors in ASP.NET Core.
6. Implement Repository, Unit of Work and Dependency Injection Design Pattern.
7. Write and Debug Unit Test cases using xUnit, MS Test, Moq.

10. Learning Outcomes and Methods of Teaching, Learning, and Assessment

On successful completion of the module, students at will be able to:

a. Know the main idea about the web application
b. Understand how the applications run over the web
c. Learn how to protect data over web.
d. Learn how to host applications
e. Learn how to remote access those applications

11. Teaching and learning methods

1- Electronic Lectures by using MS-PowerPoint.
2- Discussion.
3- Practical training .
4- Interaction between the lecturer and the students by questions.
5- Classroom or Moodle.
6- Quizzes.

12. Assessment Methods

1. Active participation,
2. Homework assignments,
3. Attendance, quizzes ,
4. Seminars
5. Middle Exams.
6. Practical Exams.
7. Final exam.
8. E-Learning using Moodle or Google Classroom .

13. Course Structure					
Week	**Hours**	**Intended Learning Outcomes**	**Name of the Unit / Course or Topic**	**Teaching Method**	**Evaluation method**
First Week	4	Knowing	Introduction	Theoretical and practical	Quiz and Practical projects
Second Week	4	Understanding	CSS	Theoretical and practical	Quiz and Practical projects
Third Week	4	Understanding	JavaScript	Theoretical and practical	Quiz and Practical projects
Fourth Week	4	Understanding	Introduction to ASP.NET Core	Theoretical and practical	Quiz and Practical projects
Fifth Week	4	Understanding	Controllers& Action Methods	Theoretical and practical	Quiz and Practical projects
Sixth Week	4	Skills	Views and Helpers	Theoretical and practical	Quiz and Practical projects
Seventh Week	4	Skills	Model Binding	Theoretical and practical	Quiz and Practical projects
Eighth Week	4	Skills	First Exam	Theoretical and practical	Quiz and Practical projects
Ninth Week	4	Skills	Validations & Data Annotations	Theoretical and practical	Quiz and Practical projects

Tenth Week	4	understanding	State management Techniques	Theoretical and practical	Quiz and Practical projects
Eleventh Week	4	Understanding	Security	Theoretical and practical	Quiz and Practical projects
Twelfth Week	4	Knowing and Understanding and Skills	MVC and Entity Framework Core	Theoretical and practical	Quiz and Practical projects
Thirteenth Week	4	Knowing and Understanding and Skills	ASP.NET Core – Web Caching	Theoretical and practical	Quiz and Practical projects
Fifteenth Week	4	Knowing and Understanding and Skills	Module Development	Theoretical and practical	Quiz and Practical projects
Fifteenth Week	4	skills	Second Exam	Theoretical and practical	Quiz and Practical projects

14. Infrastructure	
1. Creating ASP.NET Core Web Applications: Proven Approaches to Application Design and Development 1st ed. Edition by Dirk Strauss 2. Learning ASP.NET Core 2.0: Build modern web apps with ASP.NET Core 2.0, MVC, and EF Core 2 by Jason De Oliveira, Michel Bruchet	Required readings: • The prescribed textbook • The Internet • Follow the college website Moodle
Special requirements (including, for example, workshops, periodicals, software, and websites)	
١ Social services (including, for example, guest lectures, vocational training, and field studies)	

Course

Department of Software / College of Information Technology

- Machine learning (ML) is a branch of artificial intelligence (AI) that enables computers to self-learn and improve over time without being explicitly programmed. In short, machine learning algorithms are able to detect and learn from patterns in data and make their own predictions.
- While artificial intelligence and machine learning are often used interchangeably, they are two different concepts. AI is the broader concept – machines making decisions, learning new skills, and solving problems in a similar way to humans – whereas machine learning is a subset of AI that enables intelligent systems to autonomously learn new things from data.

1. The Educational Institution	University of Babylon
2. University Department / Center	Software Department
3. Course Name / Code	Machine Learning
4. .Programs that include Week	Bachelor
5. Available Attendance Modes	Weekly
6. Semester / Year	Semester
7. Total Credit Hours	60
8. Date of Preparing this Description	26/6/2022
9. Course objectives.	
Definition and examples of broad variety of machine learning tasks, including :classification • Inductive learning • Simple statistical-based learning, such as Naive Bayesian Classifier, decision trees	

• The over-fitting problem
• Measuring classifier accuracy

10.Learning Outcomes and Methods of Teaching, Learning, and Assessment

On successful completion of the module, students at will be able to:

f. Know the main idea about the machine learning
g. Learn the idea classification
h. Learn the clustering ideas
i. Learn the idea for artificial neural networks
j. Learn the regression idea

B- Subject-specific skills
1- Solving the classification problems
2- Solving the clustering problem
3- Learn the main principles for Artificial Neural Network

11. Teaching and Learning Methods

1. Electronic Lectures by using MS-PowerPoint.
2. Discussion.
3. Practical training .
4. Interaction between the lecturer and the students by questions.
5. Classroom or Moodle.
6. e-Quizzes.

12. Assessment Methods

1. Active participation,
2. Homework assignments,
3. Attendance, quizzes ,
4. Seminars
5. Middle Exams.
6. Practical Exams.
7. Final exam.
8. E-Learning using Moodle or Google Classroom

13. Course Structure

Week	Hours	Intended Learning Outcomes	Name of the Unit / Course or Topic	Teaching Method	Assessment Method
First Week	4	Knowing	Introduction to Machine Learning	Theoretical and Practical	Quiz and Practical projects
Second Week	4	Understanding	Definition and examples of broad variety of machine learning tasks	Theoretical and Practical	Quiz and Practical projects
Third Week	4	Understanding	Supervised learning	Theoretical and Practical	Quiz and Practical projects
Fourth Week	4	Understanding	Learning decision trees	Theoretical and Practical	Quiz and Practical projects
Fifth Week	4	Understanding	introduction to Artificial Neural Networks	Theoretical and Practical	Quiz and Practical projects
Sixth Week	4	Skills	A.N.N classifications, A.N.N Architecture, A.N.N Design	Theoretical and Practical	Quiz and Practical projects
Seventh Week	4	Skills	Activation Functions of A.N.N, Learning Methods of A.N.N	Theoretical and Practical	Quiz and Practical projects
Eighth Week	4	Skills	Learning rules of A.N.N, Examples of Learning rules of A.N.N	Theoretical and Practical	Quiz and Practical projects
Ninth Week	4	Skills	Backpropagation Algorithm	Theoretical and Practical	

Tenth Week	4	Skills	First Exam	Theoretical and Practical	Quiz and Practical projects
Eleventh Week	4	Skills	Classification with K-Nearest Neighbors algorithm	Theoretical and Practical	Quiz and Practical projects
Twelfth Week	4	Knowing and Understanding and Skills	Introduction to Clustering Algorithms	Theoretical and Practical	Quiz and Practical projects
Thirteenth Week	4	Knowing and Understanding and Skills	Clustering with k-means algorithm	Theoretical and Practical	Quiz and Practical projects
Fourteenth Week	4	Knowing and Understanding and Skills	Introduction to Regression Algorithms	Theoretical and Practical	Quiz and Practical projects
Fifteenth Week	4	skills	Second Exam Second Exam	Theoretical and Practical	

14. Infrastructure	
Required Readings: • **Textbook** • **Internet** • **Following the College Moodle Website**	1-Jacek Zurada ,"Introduction to Artifical Neural Network system" , ,USA,2001. 2- Ethem Alpydin , " Introduction to Machine Learning" , 3rd edition , 2014.
Special Requirements (including, for example, workshops, journals, software, and websites)	
Social Services (including, for example, guest lectures, professional training, and field studies)	

Course

Department of Software / College of Information Technology

Operating systems are an essential part of any computer system. Similarly, a course on operating systems is an essential part of any computer-science education.
This module is given for students to support and strengthen their knowledge in programming languages, and especially in Java. Java was chosen as it is considered a platform independent and is widely used in the world. It supports most games, applications, mobile applications, and more.
Our aim is to present these concepts and algorithms about deadlock, memory management and files.

1.The Educational Institution	University of Babylon
2. University Department/Center	Software Department
3. Course Name / Code	Operating Systems II--ItSwOs404207(2+2)
4. . Programs that include	Bachelor
5. Available Attendance Modes	Weekly
6. Semester/Year	Semester
7. Total Credit Hours	60
8. Date of Preparing this description	1/5/2022

9. Course Objectives

1- Developing a description of deadlocks, which prevent sets of concurrent processes from completing their tasks.

2-Presenting a number of different methods for preventing or avoiding deadlocks in a computer system.

3-Providing a detailed description of various ways of organizing memory hardware.

4-Discussing various memory-management techniques, including paging and segmentation.

10. Learning Outcomes and Methods of Teaching, Learning, and Assessment

A. Knowledge and Understanding

1. The student should be able to understand the meaning of **deadlock**.
2. The student should be able to understand how to **avoid deadlocks**.
3. The student should be able to understand how to **prevent deadlocks**.
4. The student should be able to **detect deadlocks**.
5. The student should be able to understand **memory management**.
6. The student should be able to understand how **memory paging** works.
7. The student should be able to understand how **memory segmentation** works.
8. The student should be able to understand **file management and security**.

Subject-Specific Skills

1. The student should be able to **implement deadlock detection, avoidance, and prevention**.
2. The student should be able to **understand and apply the use of programs with multiple threads (multithreading)**.
3. The student should be able to **implement memory division through paging and segmentation**.

11. Teaching and Learning Methods

1. Lectures (Present by using MS-PowerPoint) .
2. By using board.
3. Delivering lectures using a projector/screen.
4. Using the whiteboard.

12. Assessment Methods

1. Active participation, Discussion.
2. Interaction between the lecturer and the students by questions.
3. Quiz and Practical projects.

1-

Thinking Skills

1. Develop the student's ability to **design and enhance modern systems and interfaces smoothly, flexibly, and efficiently**, in line with the requirements of the labor market.

13. Course Structure

Week	Hours	Intended Learning Outcomes	Name of the Unit / Course or Topic	Teaching Method	Assessment Methods
First Week	4	RR cpu algorithm	The deadlock problem	Theoretical and Practical	Quiz and Practical projects
Second Week	4	continue	Methods for handling deadlocks	Theoretical and Practical	Quiz and Practical projects
Third Week	4	Multithread	Deadlock prevention	Theoretical and Practical	Quiz and Practical projects
Fourth Week	4	Deadlock avoidance	Deadlock avoidance. 1	Theoretical and Practical	Quiz and Practical projects
Fifth Week	4	Deadlock avoidance	Deadlock detection , Deadlock recovery	Theoretical and Practical	Quiz and Practical projects
Sixth Week	4	Safe state	Address translation: logical versus physical address space	Theoretical and Practical	Quiz and Practical projects
Seventh Week	4	Safe state	Paging	Theoretical and Practical	Quiz and Practical projects
Eighth Week	4	Deadlock detection	segmentation	Theoretical and Practical	Quiz and Practical projects
Ninth Week	4	First Exam	First Exam	Theoretical	Quiz and Practical projects

				and Practical	
Tenth Week	4	Deadlock detection,	Virtual memory	Theoretical and Practical	
Eleventh Week	4	banker's algorithm	Replacement policies for paging and segmentation	Theoretical and Practical	Quiz and Practical projects
Twelfth Week	4	banker's algorithm	Thrashing	Theoretical and Practical	Quiz and Practical projects
Thirteenth Week	4	Second Exam	Second Exam	Theoretical and Practical	Quiz and Practical projects
Fourteenth Week	4	Paging	File system interface and structure	Theoretical and Practical	Quiz and Practical projects
Fifteenth Week	4	Paging	Access and protection	Theoretical and Practical	

14. Infrastructure	
Required Readings: • The prescribed textbook • Online resources • Following the college's Moodle platform	1.Operating System Concepts – 8th Edition, Silberschatz, Galvin and Gagne 2012 2. operating system concepts with java, Silberschatz, Galvin and Gagne 2004
Special Requirements (including, for example, workshops, journals, software, and websites)	
Social Services (including, for example, guest lectures, professional training, and field studies)	

Course

Department of Software / College of Information Technology

This course introduces important concepts of 3D game development and to the use of the 3D Game Engine Unity. Topics for the course might include unity and game programming basics, player movements, cameras, game design, terrain creation, character controllers, path following, steering, and finding, flocking, ethics in games, interfaces, and audio. It provides a strong foundation in software engineering, programming, and the C# language; and to work on all major aspects of developing video games using the Unity engine.

Usually video game development centers on programming and software development, and to be a game developer requires a high level of knowledge in a modern, object-oriented language like C#. Through this course, students will learn programming by working on games, and will learn to write code to run every part of their game, from physics to AI to game servers

1. The Educational Institution	University of Babylon
2. University Department / Center	Software Department
3. Course Name / Code	Game Programming and Design
4. .Programs that include	Bachelor
5. Available Attendance Modes	Weekly
6. Semester / Year	Semester
7. Total Credit Hours	60
8. Date of Preparing this Description	20/4/2022
9. Course Objectives	

will work on nearly all aspects of game design and production: storyboarding, level design, texturing & materials, programming, terrain, physics, animation, 2D & 3D rendering, and user interface, among others. Students will learn the Unity game engine and the C# programming language, and create multiple games for their portfolio. Other topics may be introduced, for instance shader generation, game server development, VR, and other advanced topics.

10. Learning Outcomes and Methods of Teaching, Learning, and Assessment

A. Knowledge and Understanding

1. Think like a game designer
2. Analyze different types of game frameworks
3. Understand Layered Tetras
4. Design game goals by using different models.
5. Game testing via playtesters circles.
6. Guiding the players.
7. Apply theories and develop and test an actual game
8. Use a Version Control Software System (git) to create maintain and distribute a software product.
9. Use the Unity Editor to create interesting game levels
10. Create game play features using Visual C# 2010 and Unity3D
11. Obtain, evaluate and incorporate 3d models or Create 3d models with 3d tools
12. Integrate art and models into a game world
13. Manage a software project using version control software
14. Acquainted with advanced topics such as shaders, physics, AI, and Network based games.
15. Understand the process of game development from idea to beta version

B. Subject-Specific Skills

1. Use the concept of Brainstorming during lectures.
2. Pose direct questions related to the subject and connect them to external events.
3. Use Oral Quizzes.
4. Apply E-learning concepts and link them to the subject matter.
5. Introduce Learning methods rather than relying solely on Teaching.

11. Teaching and Learning Methods

1. Lecture Delivery Method
2. Student Center
3. Team Project
4. E-Learning using Google Class .
5. Experiential Learning
6. Application Learning

12.Assessment Methods

1. Active participation,

2. Homework assignments,
3. Attendance, quizzes
4. Lab Tasks
5. Seminars
6. Moodle exams
7. Middle, Practical and final Exams.

C. **Thinking Skills**

1-Brain storm method

2-Asking smart questions

D. General and Transferable Skills (other skills related to employability and personal development)

1. Develop the concept of teamwork and the students' ability to enhance skills in ideas and communication.
2. Clearly elaborate on game design concepts in a way that aligns with the needs of society in applied digital sciences.
3. Apply and document the acquired skills in solving game design problems.
4. Apply the acquired skills and knowledge for lifelong learning and career advancement, keeping pace with information and communication technologies..

11. Course Structure

Week	HOURS	Intended Learning Outcomes	Unit / Course or Subject Name	Teaching Method	Assessment Methods
First Week	4	Introduction to the Game Programming	Introduction to the Unity 3D	Course Overview	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Second Week	4	Introduction to the Game Design	Unity Setup and User Interface	Game Design Introduction	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Third Week	4	3D Object Basics	Creating 3D objects	Thinking like a Designer	Calculator+ Google Classroom+ Google Meet+ Practical laboratories

Fourth Week	4	Objects Arguments Programming	Define Objects Arguments Programmatically	Game Analysis Frameworks	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Fifth Week	4	Crating 1st Project	Create Apple Picker Game Project	Design Goals	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Sixth Week	4	Game Testing	Gravity and Environments	Game Testing	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Seventh Week	4	Player Guiding Methods	Visual C# and Unity 3D	Guiding the Player	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Eighth Week	4	Controls	Character Movements and Control	1st Exam	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Ninth Week	4	Game Mathss	1st Exam	Math and Game Balance	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Tenth Week	4	New Game Project type	Create Mission Demolition Game Project	Puzzle Design	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Eleventh Week	4	Agile Mentality	Continue Mission Project	The Agile Mentality	Calculator+ Google Classroom+ Google Meet+ Practical laboratories

Twelfth Week	4	Understand Game AI	Camera and UI details	The Game AI I	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Thirteenth Week	4	Understand Game AI	2nd Exam	The Game AI II	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Fourteenth Week	4	Projects Evaluation	Evaluate Students Projects	2nd Exam	Calculator+ Google Classroom+ Google Meet+ Practical laboratories
Fifteenth Week	4	Seminars Evaluation	Evaluate Students Projects	Evaluate Students Seminars	Calculator+ Google Classroom+ Google Meet+ Practical laboratories

12. Infrastructure	
Required Readings: • Core texts • Prescribed textbooks • Other sources	▪ Introduction to Game Design, Prototyping, and Development From Concept to Playable Game with Unity and C#, Jeremy Gibson Bond , 3rd Edition, Addison-Wesley, 2018
Special Requirements (including, for example, workshops, journals, software, and websites)	
Social Services (including, for example, guest lectures, professional training, and field studies)	