

**Ministry of Higher Education and Scientific Research
Scientific Supervision and Evaluation Authority
Directorate of Quality Assurance and Academic
Accreditation
Accreditation Department**



Academic Program Guide and Course Descriptions

2024

Academic Program Description Form

University Name: Diyala University

Faculty/Institute: College of Education for Pure Sciences

Scientific Department: Computer Department

Academic or Professional Program Name: Bachelor.

Final Certificate Name: Bachelor. in Computer Science

Academic System: Annual

Description Prepared by: 20/11/2023

File completion date: 20/2/2024

Head of Department Name:
Prof. Nihad Mahmoud Nasser

Scientific Associate Name:
Prof. Dr. Khansa Salman Farman

The file is checked by:

Department of Quality Assurance and University Performance/ Noor Hassan Hasoon
Director of Quality Assurance and University Performance Department:

Approval Dean's
Prof. Ghaleb Idris Attia

1. Program Vision

The program's vision for the Operating Systems course aims to develop students' understanding and ability to:

- **Understand key concepts:** This includes understanding the basic principles that govern the interaction between hardware, software, processes, memory management, file systems, security, and concurrency control.
- **Application of theoretical knowledge:** Students should be able to apply the concepts acquired to solve practical problems in the design and configuration of the operating system.

By realizing this program vision, students will gain a solid foundation for working with and potentially contributing to the operating systems that power today's technology.

2. Program Mission

The program vision describes the desired long-term outcome of the Operating Systems course, focusing on the knowledge and skills that students will possess after completion.

However, the mission of the program will be a more specific statement outlining the main steps the course takes to achieve this vision. Here's how the program's mission can be derived from the vision:

Program Mission:

- Provide students with a comprehensive understanding of operating system concepts through theoretical foundations and practical application.
- To enhance critical thinking skills that allow students to analyze and compare operating system designs.
- To provide a strong foundation for further study and careers in computer science related to operating systems.

The mission of the program translates the broad goals of the vision into actionable steps by highlighting the methods used (e.g., theoretical foundations and practical application) to achieve the desired outcomes for students.

3. Program Objectives

Based on the program's vision for the operating systems cycle, here are the program's potential goals:

- Explain the basic functions and services provided by the operating system.
- Analyze the core components of the operating system, including processes, memory management, file systems, and security mechanisms.
- Evaluate different scheduling algorithms and their impact on system performance.
- Design and implement solutions for process synchronization and communication between processes.
- Apply virtual memory knowledge to analyze memory management techniques.
- Explain the structure and organization of file systems, including ways to access files and directory structures.
- Discuss the security principles and mechanisms used by operating systems.
- Compare and compare different types of operating systems (for example, batch, multiple

programming, real-time).

These goals translate the overall goals of the program's vision into more specific and measurable learning outcomes. By achieving these goals, students will be on their way to achieving the program's vision.

4. Program Accreditation

No

5. Other external influences

No

6. Program Structure

Program Structure	Number of cycles	Credit Hours	Percentage	Reviews*
of the Requirements institution	8	17	9.3%	
College Requirements	9	36	19.7%	
Department Requirements	20	129	69.2%	
Summer Training				
End				

* This can include feedback on whether the course is basic or elective.

7. Program Description

Year/Level	Course Code	Course Name	Credit Hours	
			theoretical	practical
year	Logical design	CSLD101	2	2
year	ctural Stru Programming	CSSP102	3	2
year	Computer Technologies and Installation	CSCO 103	2	2
year	Discontinuous structures	CSDS104	3	
year	Math	CSMA105	3	
year	Educational Psychology	CSES106	2	
year	Foundations of education	CSFE107	2	
year	Language Arabic	CSAL108	1	
year	English Language	TH109	1	
year	Human Rights and Democracy	CSHD110	2	
year	Data structures and algorithms	CSDA201	3	2

year	Entity Programming	CSOP202	3	2
year	Microprocessors	CSMP203	2	2
year	Systems Analysis	CSSA204	2	2
year	tational Compu theory	CSCT205	2	
year	Numerical Analysis	CSNA206	2	2
year	English Language	TH207	1	
year	Scientific Research Methodology	CSSR208	2	
year	Developmental Psychology	CSDP209	2	
year	Educational Administration	CSEM210	2	
year	Artificial genceIntelli	CSAI301	2	2
year	Computer Fees	CSCG302	2	2
year	Vogue Basic	CSVB303	2	2
year	Compilers	CSCO 304	2	2
year	Computer Architecture	CSCA305	2	
year	Software Engineering	CSSE306	2	
year	Database Design	CSDB307	1	2
year	Curricula and teaching methods	CSCT308	2	
year	Counseling and mental health	PIN309	2	
year	Data Security	CSDS401	2	2
year	Communications & Computer Networks	CSCN402	2	2
year	Operating Systems	CSOS403	2	2
year	Image processing	CHIP404	2	2
year	Web Design	CSWD405	2	2
year	on Graduati Research Project	CSP406	2	
year	Measurement and evaluation	CSME407	2	
year	Practical Education	CSV408	1	3

8. Expected learning outcomes of the program

Knowledge

1. Providing students with integrated scientific knowledge in the field of computer science in a high-quality manner.
2. The ability to apply the scientific knowledge acquired in the field of computer, mathematics and other sciences to ensure the achievement of cognitive communication between them and benefit from modern developments.
3. Preparing highly qualified professionals and specialists in the field of computer science in order to contribute effectively to the establishment of a knowledge society and achieve national

development goals by creating an optimal academic environment for knowledge development and transfer of skills in research and innovation in the field of computers.

4. Achieving exposure to the applied reality within public and private educational institutions by employing techniques, skills, technical tools and modern technology necessary to practice the teaching profession.

Skills

1. The student should master the basic and advanced programming skills required to enrich his intellectual and technical talents in this field of computer science and its various applications.

2. Master the skills required to manage information systems, databases and web design with high efficiency.

3. To master the preparation of scientific research taking into account an integrated scientific methodology.

4. Mastering the correct educational and psychological method in dealing within educational institutions.

5. Possess the professional skills required in the field of software development and projects that make them confident in developing high-quality software solutions in different application areas under different realistic constraints.

ethics

1. Appreciating the greatness of the Creator Almighty in creating the human mind and making it capable. On creativity in various fields

2. Participate and succeed in their careers through teamwork, proper ethical behavior and effective communication among them.

3. Acquire and understand the importance of lifelong learning through professional development and practical training.

4. Appreciating the efforts of scientists in developing software and delivering scientific knowledge in the field of computer science to the level of progress and development it has reached.

9. Teaching and Learning Strategies

- Traditional lectures and discussion method.
- Lab activities and additional exercises as tasks
- Scientific books.
- Daily and monthly exams

10. Evaluation methods

- Theory tests
- Practical tests
- Reports and projects

11.College

Faculty Members

Academic Rank	Specialization	Special Requirements/Skills (if applicable)	Number of faculty members
---------------	----------------	---	---------------------------

	year	special			crew	lecturer
Professor	Calculators					
Professor	Physics	Computer Modeling				
Professor	Educational and psychological sciences	Solid state				
Assistant Professor	Computer Science	Philosophy of Education			1	
Assistant Professor	Computer Science	Databases			1	
Assistant Professor	Computer Science	Image processing			1	
Assistant Professor	Computer Science	Systems & Informatics			1	
Assistant Professor	Computer Science	Networking & Telecommunications			1	
Assistant Professor	Computer Science	Computational Mathematics			1	
Assistant Professor	Computer Science	Algebraic topology			1	
Assistant Professor	Mathematical Sciences	Data Security			1	
Assistant Professor	Computer Science	Network Security			1	
Assistant Professor	Networks	Information Technology			1	
Assistant Professor	Computer Science	Computer Modeling			1	
Assistant Professor	Computer Science	Electronic Engineering			1	
teacher	Computer Science	Connections			1	
teacher	Electronic Engineering	Artificial Intelligence			1	
teacher	Computer Science	Machine Learning			1	
teacher	Computer Science	Computer Engineering			1	
teacher	Electrical Engineering	Information Technology			3	1
Assistant Lecturer	Computer Engineering	Computer Science				
Assistant Professor	Computer Science	Mathematics Pure				

	Computer Science					
	Mathematical Sciences					

Professional Development
Mentoring new faculty members
<ol style="list-style-type: none"> 1. Identify the vision, strategic plan and the role of the faculty member in achieving them. 2. Modern teaching methods in the field of learning (including the integration of technology) 3. Introducing methods of dealing with distinguished and creative students (developing students with talents and abilities) 4. Introduction to Job Performance Appraisal 5. Introducing the diversity of distance learning methods, uploading files and electronic exams 6. Clarify quality requirements, course descriptions and training plan 7. Know the rights and duties of teachers

Professional development of faculty members
<ol style="list-style-type: none"> 1. Self-methods: Conversations with colleagues, reading and writing about teaching, attending workshops and panel discussions, and participating in other activities provide opportunities to reflect on teaching and seek new teaching methods through in-person or peer education. 2. Professional methods: Educational conferences, workshops, distance learning and some other professional development methods: Vocational qualification, cooperation programs between colleagues, scholarships (scholarships), practical training

12. Acceptance Criterion
Central Admission

13. The most important sources of information about the program
College and University Locations/ University Guide/ College and Department Guidebook/ Books and Resources in the Department/

14. Program Development Level
Develop and establish specialized scientific laboratories such as Arduino so that students can learn about the latest modern technologies

	Scientific Research Methodology	208CSSR	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Developmental Psychology	209CSDP	fundamental	X	X	X	X	X	X	X		X	X	X	X
	Educational Administration	210CSEM	fundamental	X	X	X	X	X	X	X		X	X	X	X
Third stage	Artificial Intelligence	301CSAI	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Computer Fees	302CSCG	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Vogue Basic	303CSVB	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Compilers	304 Cisco	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	er Comput Architecture	305CSCA	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Software Engineering	306CSSE	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Database Design	307CSDB	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Curricula and teaching methods	308CSCT	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Counseling and mental health	309PIN	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
Fourth stage	Data Security	401CSDS	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Communications Computer & Networks	402CSCN	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Operating Systems	403CSOS	fundamental	X	X	X		X	X	X		X	X	X	
	Image processing	404CHIP	fundamental	X	X			X	X	X		X	X	X	
	Web Design	405CSWD	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Graduation Research Project	406CSP	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Measurement and ionevaluat	407CSME	fundamental	X	X	X	X	X	X	X	X	X	X	X	X
	Practical Education	408CSV	fundamental	X	X	X	X	X	X	X	X	X	X	X	X

Course Description Form

1. Course Name:	
Advanced databases	
:Code Course .2	
403CSOS	
:Year /Semester .3	
2023/2024	
:Description Preparation date .4	
20/3/2024	
5. Available attendance forms:	
Physical attendance	
6. Number of Credit Hours (Total) / Number of Units (Total)	
4 hours / 6 units	
7. Course administrator name (list all, if more than one)	
Name: Assoc. Prof. Qasim Radam Mahmoud Email: pure.comp.qasim.radam@uodiyala.edu.iq	
8. Course Objective	
Course Objective	<ul style="list-style-type: none"> • Explain the basic functions and services provided by the database management system. • Analysis of the basic components of a database management system • Types of database management systems. • Introduction to structured query language. • Explain how to create tables and add data to them by writing their own queries. • Explain the basic operations that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc). • Create relationships between tables by querying. • Explain the queries for comparing literal strings. • Explain aggregate operations on digital fields. • Explanation of scaling functions in the query language and how to write them • Explain the orders to arrange data in ascending or descending order. • Main query and subquery. • Use commands (IN, Exists) in subqueries. • Use commands (Any, All) with subqueries. • Explain operations on groups (Union, Intersect, difference). • Explain joint queries or joins queries. • Explanation of System Data queries.
9. Teaching and Learning Strategies	
strategy	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Lab activities and additional exercises as tasks - Scientific books. - Daily and monthly exams
10. Course structure	

Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2	Understand what a database management system is	Introduction to Database Management System	Diction + examples	Oral exam
2	2	Understand what a database management system is	Introduction to Database Management System	Diction + Examples	Oral exam
3	2	Know the basic components of a database system	Analysis of the basic components of a database management system (entity, properties, relationships)	Diction + Examples	Oral exam
4	2	The student should know the types of database management systems	Types of database management systems.	Diction + Examples	Oral exam
5	2	The student should know the types of database management systems	Types of database management systems.	Diction + Examples	exam Oral
6	2	The student should know the types of database management systems	Types of database management systems.	Diction + Examples	Oral exam
7	2		examination		
8	2	Learn about structured query language	Introduction to structured query language.	Diction + Examples + Laboratory	Oral exam
9	2	The ability to create tables within a specific database by querying.	Explain how to create tables and add data to them by writing their own queries.	Diction + Examples + Laboratory	Oral and practical exam
10	2	The ability to write queries for operations that can be performed on data within tables.	Explain the basic operations that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc).	Diction + Examples + Laboratory	Oral and practical exam
12	2	The ability to write queries for operations that can be performed on data within tables.	Explain the basic operations that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc).	Diction + Examples + Laboratory	Oral exam and homework
13	2	The ability to	Explain the basic operations	Diction +	Oral and

		write queries for operations that can be performed on data within tables.	that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc).	Examples + Laboratory	practical exam
14	2	The ability to write queries for operations that can be performed on data within tables.	Explain the basic operations that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc).	Diction + Examples + Laboratory	Oral and practical exam
15	2		examination		
13	2	The student should be able to establish relationships between tables	Create relationships between tables by querying.	Diction + Examples + Laboratory	Oral and practical exam
14	2	The student should be able to write queries for comparison between literal strings.	Explain the queries for comparing literal strings.	Diction + Examples + Laboratory	Oral and al practical exam
15	2	The student should be able to write queries for comparison between literal strings.	Explain the queries for comparing literal strings.	Diction + Examples + Laboratory	Oral and practical exam
16	2	The student should be able to write queries to perform aggregate operations on table data	Explain aggregate operations on digital fields.	Diction + Examples + Laboratory	Oral and practical exam
17	2	The student should be able to write queries to perform aggregate operations on table data	Explain aggregate operations on digital fields.	Diction + Examples + Laboratory	Oral and practical exam
18	2		examination		
19	2	The student should be able to write sizing queries	Explanation of scaling functions in the query language and how to write them	Diction + Examples + Laboratory	Oral and practical exam
20	2	The student should be able to write sizing queries	Explanation of scaling functions in the query language and how to write them	Diction + Examples + Laboratory	Oral and practical exam
21	2	The student should be able to write queries in	Explain the orders to arrange data in ascending or descending order.	Diction + Examples +	Oral and practical exam

		ascending and descending order.		Laboratory	
22	2	The student should be able to write queries that link the external and internal query	Main query and subquery.	Diction + Examples + Laboratory	Oral and practical exam
23	2	The student should be able to write queries that link the external and internal query	Main query and subquery.	Diction + Examples + Laboratory	Oral and practical exam
24	2		examination		
25		The student should be able to write queries that include commands (In, Exists)	Use commands (IN, Exists) in subqueries.	Diction + Examples + Laboratory	Oral and practical exam
26	2	The student should be able to write queries that include commands (Any, All)	Use commands (Any, All) in subqueries.	Diction + Examples + Laboratory	Oral and practical exam
27	2	The student should be able to write queries for operations on groups.	Explain operations on groups (Union, Intersect, difference).	Diction + Examples + Laboratory	Oral and practical exam
28	2	The student should be able to write queries for linking data from more than one table and show them.	Explain joint queries or joins queries.	Diction + Examples + Laboratory	Oral and practical exam
29	2	The student should be able to write queries for linking data from more than one table and show them.	Explain joint queries or joins queries.	Diction + Examples + Laboratory	Oral and practical exam
30		The student should be able to write queries for system data	Explanation of System Data queries.	Diction + Examples + Laboratory	Oral and practical exam

11.Course Evaluation

Monthly exam (30) / Practical exam (10) / Final exam (60)

urcesLearning and Teaching Reso.12

- Hector Georrica, Jeffrey D. Ullman and Jennifer Widom, "Database System The Complet Book", Pearson Prentice Hall,2002-2009.

Course Description Form

13.Course Name:	
Artificial Intelligence	
:Code Course.14	
301CSAI	
:Year /Semester.15	
2023/2024	
:Description Preparation date.16	
20/3/2024	
17.Available attendance forms:	
Physical attendance	
18.Number of Credit Hours (Total) / Number of Units (Total)	
4 hours / 6 units	
19. Course administrator name (list all, if more than one)	
Spectra Ismail Karim .Assoc. Prof :Name	
الإلكتروني البريد: dr.atyaf.ismaeel@uodiyala.edu.iq	
20.Course Objective	
Course Objective	<ul style="list-style-type: none"> • ATo enable the student to have an understanding of the subject of intelligence • Knowledge of ways to organize and represent AI programs • Understanding AI (its advantages and disadvantages) • - Enable the student to learn about different ways of artificial intelligence • Ability to use language - programming prolog to represent artificial intelligence programs • The student is proficient in representing what he has learned theoretically in a practical way in the laboratory
21.Teaching and Learning Strategies	
strategy	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Lab activities and additional exercises as tasks - Scientific books. - Daily and monthly exams

22. Course structure					
Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2	Introduction to Artificial Intelligence	Introduction to the Prolock language and its properties and used	Reading Lab +	
2	2	Properties, applications and architecture Languages and branches of artificial intelligence	Programming Language Components (Questions and Rules))Facts	Reading + Lab	Oral exam
3	2	Define the problem solving domain (First lecture)	Types of variables in Prolock	Reading + Lab	Oral exam
4	2	Characteristics of the problem solving field (Second lecture)	Binding and reverse return (and, or)	Reading + Lab	Oral exam
5	2	Systems Protection	Data types and structuring Program	Reading + Lab	Contest
6	2	Examples of problem solving field such as 8-puzzle , monkey and banana.....etc	Input and output statements	Reading + Lab	Oral exam
7	2	Types of Depth First Search	Illustrative examples of phrases Input and output	Reading + Lab	Oral exam
8	2	Types of research (exploration research - first lecture) hill climbing	Mathematical and logical operations In Prolock language	Reading + Lab	Homework
9	2	Types of research (exploration research - second lecture) best first search	Comprehensive vocabulary programs All previous ones	Reading + Lab	Applied algorithm in LAB
10	2	Expert Systems	Comprehensive vocabulary programs All previous ones	Reading + Lab	Applied algorithm in LAB
11	2	Genetic Algorithm (Introduction)	Comprehensive vocabulary programs All previous ones	Reading + Lab	Oral exam
12	2	Genetic algorithm (Part Two _Example)	The mechanism of the subprograms and their interference during the call	Reading + Lab	Oral exam
13	2	Genetic algorithm	Similar to the mechanism of	Reading	Oral exam

		(Part Two _Applications)	the subprograms and their overlap during the call	+ Lab	
14	2	Biological Neurosystems Introduction_ Artificial Neural Network ANN	Similar to the mechanism of the subprograms and their overlap during the call	Reading + Lab	Contest
15	2	Biological Neurosystems _Introduction (Cell training)	Similar to the mechanism of the subprograms and their overlap during the call	Reading + Lab	Homework
16	2	Biological Neurological Systems (Applications)	Cut, stop, and negation sfunction		
17	2	Fuzzy Logic Fuzzy Logic	Cut, stop, and negation functions	Reading + Lab	Oral exam
18	2	ROBOTICS TYPES OF ROBOTS AND THEIR BASIC PARTS	Cut, stop, and negation functions	Reading + Lab	Oral exam
19	2	Technology Resolution	Final self-recall and the infinite	Reading + Lab	Oral exam
20	2	Decision Strategy	Cut, stop, and negation functions	Reading + Lab	Oral exam
21		Control strategy	Additional examples and programs on Final self-recall and the infinite		
22		Representation of knowledge (mathematical and cognitive logic)	The menu is in Prolock language		
23		Representation of the event in the form of the	Syntactic formula of the list tail and the list and its		
24		Review and link to the previous topics with an explanation of the latest technologies with a brief summary and prospects for future development in the field of intelligence	Programs about the list		
25			General Review		
26			Exam Includes		
27					Oral exam
28					Oral exam

29	2			Applied algorithm in LAB
30		Exam		

23. Course Evaluation

Monthly exam (30) / Practical exam (10) / Final exam (60)

Learning and Teaching Resources.24

. Introduction to Artificial Intelligence by Bojana Dalbelo Basic and Jan Snajder 2019/2020

- Peter Norvege "Artificial intelligence Modern Approach", 2019
- www.tutorialspoint.com
- [file?utm_source=22-write-and-read-to-program-https://www.codepoc.io/blog/prolog/4990/prolog](https://www.codepoc.io/blog/prolog/4990/prolog) 2020

Course Description Form

25. Course Title:

Computational theory
Theory of Computation

:Code Course.26

205CSCT

:Year /Semester.27

2023/2024

:Description Preparation date.28

20/3/2024

29. Available attendance forms:

Physical attendance

30. Number of Credit Hours (Total) / Number of Units (Total)

4 hours

31. Course administrator name (list all, if more than one)

Name: Dr. Mohamed Sami Mohamed
Email dr.mohammed.sami@uodiyala.edu.iq

32. Course Objectives

Course Objectives

- Preparing the student and giving him the basics necessary to understand and study the calculation material in the second grade
- Understand the languages used in computational to solve general programming and mathematical problems
- understand the meaning of FA , NFS and TG in the computational material and differentiate between them
- Identify the rules and laws that represent the general drawing of these methods
- Understand the way of linguistic expression and transitions from

	<p>one state to another from beginning to end</p> <ul style="list-style-type: none"> • Definition of computational theory • Introducing students to the importance of computational theory • Distinguish between types of automation systems and the method of finding standard languages • Understand the concepts of standard languages and generate standard grammar • Material awareness and future employability • Dealing with the computer and knowing how to make it process data better through knowledge of the stages of converting data into digital data understandable to the calculator
33. Teaching and Learning Strategies	
strategy	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Lab activities and additional exercises as tasks - Scientific books. - Daily and monthly exams

34. Course Structure					
Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2	Study of Computational Theory Introduction	Set theory	Reading and reviewing lectures	
2	2	Types of rules in calculation	Set theory and grammar types	Reading and reviewing lectures	Oral exam
3	2	Types of rules in calculation	Grammar types	Reading and reviewing lectures	Oral exam
4	2	Logical and general idioms	Regular expression	Reading and reviewing lectures	Oral exam
5	2	Logical and general idioms	Regular expression	Reading and reviewing lectures	Contest
6	2	FA Learn the method	Finite automata	Reading and reviewing lectures	Oral exam
7	2	Learn transitional drawing	Transition graph	Reading and reviewing lectures	Oral exam
8	2	Types FA	Convert from NDFA- ϵ to NDFA without ϵ	Reading	Homework

				and reviewing lectures	
9	2	Interspecific conversion	Convert from NDF to DFA	Reading and reviewing lectures	Applied algorithm in LAB
10	2	Interspecific conversion	Convert RE to NDFA	Reading and reviewing lectures	Applied algorithm in LAB
11	2	Interspecific conversion	Compare FA,TG,RE	Reading and reviewing lectures	Oral exam
12	2	Introduction to Clen Theory	Introduction to Kleene theory	Reading and reviewing lectures	Oral exam
13	2	Theory overview	Review	Reading and reviewing lectures	Oral exam
14	2	First specific proof of the theory	Kleene theory review & prove first section	Reading and reviewing lectures	Racing
15	2	A second specific proof of the theory	Prove second section with EX	Reading and reviewing lectures	Homework
16	2	A third specific proof of the theory	EX on second and prove third section	Reading and reviewing lectures	
17	2	A specific third and fourth proof of the theory	EX on third section and prove fourth section	Reading and reviewing lectures	Oral exam
18	2	Introduction to CFG	Ex on fourth section and introduction of CFG	Reading and reviewing lectures	Oral exam
19	2	CFG Learn about	CFG	Reading and reviewing lectures	Oral exam
20	2	toCFG Convert CNF	Convert CFG to CNF	Reading and reviewing lectures	Oral exam

21		toCFG Convert CNF	Convert CFG to CNF	Reading and reviewing lectures	
22		Moore machine and Millie's machine	Moore and mealy machine	Reading and reviewing lectures	
23		Switching between Moore and Millie	Convert from Moore to mealy and converse and PDA	Reading and reviewing lectures	
24		Switching between Moore and Millie	Review	Reading and reviewing lectures	
25		Switching between Moore and Millie	PDA and TM	Reading and reviewing lectures	
26		Switching between and Millie Moore	TM and review	Reading and reviewing lectures	
27		Examples and general review	Examples and general review	Reading and reviewing lectures	Oral exam
25		Exam		Reading and reviewing lectures	

35.Course Evaluation

Monthly exam (25) per semester / final exam (50)

Learning and Teaching Resources.36

.1Introduction to Computer Theory 2nd Edition

Daniel I. A. Cohen John Wiley & Sons, Inc 1997. ISBN 0-471-13772-3

.2Introduction to Automata Theory, Languages, and Computation, 2/E•

John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman, Addison-Wesley 2001. ISBN 0-244124-1.

Course Description Form

37.Course Name:	
Structured Programming	
:Code Course.38	
403CSOS	
:Year /Semester.39	
2023/2024	
:Description Preparation date.40	
20/3/2024	
41.Available attendance forms:	
Physical attendance	
42.Number of Credit Hours (Total) / Number of Units (Total)	
5 hours / 6 units	
43. Course administrator name (list all, if more than one)	
Name: Assoc. Prof. Qasim Radam Mahmoud	
Email pure.comp.qasim.mahmood@uodiyala.edu.iq	
44.Course Objective	
Course Objective	<ul style="list-style-type: none"> • Explain how to write algorithms and flowcharts to find solutions to the different problems to be programmed. • Teach the student the general structure of the C++ program and its basic components. • Explain reading and writing commands in C++ . • Explain the conditional sentences of various types and structures in the C++ language. • Explain the repetition sentences of various types and structures in the C++ language. • Explanation of matrices of various types and adapted to dealing with them within the C++ program. • Teaching the student the mechanism of writing and calling sub-programs within the general program of the C++ language. • Enable the student to deal with text files (opening, writing in them, deleting them, closing them) through the program . C++ • Enable the student to be able to analyze various problems and write programs to solve various problems using the C++ language.
45.Teaching and Learning Strategies	
strategy	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Lab activities and additional exercises as tasks - Scientific books. - Daily and monthly exams

46. Course structure					
Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2	Introduction to C++ programming	Introduction to C++ programming	Diction + examples	Oral exam
2	2	Write algorithms	How to write algorithms and	Diction +	Oral exam

		and flowcharts to solve various problems	flowcharts to solve various problems	Examples	
2	2	Write algorithms and flowcharts to solve various problems	How to write algorithms and flowcharts to solve various problems	Diction + Examples	Oral exam
3	2	General structure of the C++ language program	Components of the general structure of the C++ language program	Diction + Examples + Laboratory	and an Oral practical exam
4	2	Ability to define data	Define data types	Diction + Examples + Laboratory	and Oral practical exam
5	2	Use arithmetic and logical operations	Definition of arithmetic and logical operations	Diction + Examples + Laboratory	Oral and practical exam
6	2	Learn how to increase, decrease and equality	Processes of increasing, decreasing, and equalizing	Diction + Examples + Laboratory	Oral and practical exam
7	2	Ability to write comments and prioritize operations	Comment phrases and specify precedence in the execution of operations	Diction + Examples + Laboratory	Oral and practical exam
8	2	The student should be able to define constants, variables and reserved words	Define constants, variables, and reserved words	Diction + Examples + Laboratory	Oral and practical exam
9	2	Transfers	Definition of transfers	Diction + Examples + Laboratory	Oral exam and homework
10	2	The student should be able to use the input and output phrases	Input and output phrases	Diction + Examples + Laboratory	and Oral practical exam
11	2	The student should be able to use condition sentences within the program	Conditional phrases	Diction + Examples + Laboratory	Oral and practical exam
12	2	The student should be able to use condition sentences within the program	Conditional phrases	Diction + Examples + Laboratory	Oral and practical exam
13	2	The student should be able to use repetition sentences within the program	Rotary phrases (repetition sentences)	Diction + Examples + Laboratory	Oral and practical exam

14	2	The student should understand the overlap of phrases	Overlapping phrases	Diction + Examples + Laboratory	Oral and practical exam
15	2	The student should understand the overlap of phrases	Overlapping phrases	Diction + Examples + Laboratory	Oral and 1 practical exam
16	2	The student should be able to use jumping phrases	Jumping phrases	Diction + Examples + Laboratory	Oral and practical exam
17	2	The student should be able to use jumping phrases	Jumping phrases	Diction + Examples + Laboratory	Oral and 1 practical exam
18	2	The student should be able to define and use the one-dimensional matrix within the program	One-dimensional matrix	Diction + Examples + Laboratory	Oral and practical exam
19	2	The student should be able to define and use the one-dimensional matrix within the program	One-dimensional matrix	Diction + Examples + Laboratory	Oral and practical exam
20	2	The student should be able to define and use the two-dimensional matrix within the program	Two-dimensional matrix	Diction + Examples + Laboratory	Oral and practical exam
21		The student should be able to define and use the two-dimensional matrix within the program	Two-dimensional matrix	Diction + Examples + Laboratory	Oral and practical exam
22		The student should be able to call ready-made functions within the program	Subprograms (Ready-made Functions)	Diction + Examples + Laboratory	Oral and practical exam
23		The student should be able to call ready-made functions within the program	Subprograms (Ready-made Functions)	Diction + Examples + Laboratory	Oral and practical exam
24		The student should be able to call functions and methods of announcing them	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam

		within the program			
25		The student should be able to call functions and methods of announcing them within the program	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam
		The student should be able to call functions and methods of announcing them within the program	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam
27		The student should be able to call functions and methods of announcing them within the program	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam
28		The student should be able to call functions and methods of announcing them within the program	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam
29	2	The student must be able to create files, read from them and write in them	Create files, store them, and read from them	Diction + Examples + Laboratory	nd Oral a practical exam
30		The student must be able to create files, read from them and write in them	Create files, store them, and read from them	Diction + Examples + Laboratory	Oral and practical exam

47.Course Evaluation

Monthly exam (40) / Practical exam (10) / Final exam (50)

Learning and Teaching Resources.48

- Stanly B.Lippman, Josee Lajoie, and Barbara E. Moo,"C++ Prmer", forth edition, 2005.
- Juan Soulie,"C++ language Tutorial", 2008
- Any book and article which is related to the C++ language
- <http://www.cplusplus/>
- <http://www.tutorialspoint.com/>
- <http://www.w3schools.com/cpp/>

Course Description Form

49.Course Title:	
Computer Technologies and Installation Computer Techniques and Organization	
:Code Course.50	
103CISCO	
:Year /Semester.51	
2023/2024	
:Description Preparation date.52	
20/3/2024	
53.Available attendance forms:	
Physical attendance	
54.Number of Credit Hours (Total) / Number of Units (Total)	
4 hours / 6 units	
55. Course administrator name (list all, if more than one)	
Name: Dr. Mohamed Sami Mohamed (theoretical material) (Eng. Sahar Jassim Mohamed (practical material الإلكتروني البريد: dr.mohammed.sami@uodiyala.edu.iq	
56.Course Objectives	
<p>Course Objectives The student's knowledge of the components of the calculator The student's knowledge of input and output devices The student's knowledge of the types of data, methods of entering it, and how to represent it in the calculator The student's knowledge of how to process data and how to get it out Identify addressing patterns Identify I/O problems Introducing the student to the basic circuits in the installation of the electronic calculator Study the main parts of the calculator from memory, input and output units and microprocessor.</p>	<p>Course Objectives 1- Qualifying the student to teach this subject in middle and middle schools 2- Introducing students to the computer and its components, how it works and its importance in our lives through the following: 3- Mastering the student how to deal with the computer through the following: 4- The student acquires the skill of maintaining computers. 5- The student acquires the skill of applying modern practical methods in the use of computers 6- The student acquires the skill of using the best methods in programming computers 7- Develops the student's skills on computers 8- The student acquires the ability to deal on all computers. 9- The student acquires the skill in detecting all computer parts and solving all errors</p>
57.Teaching and Learning Strategies	
<p>Use of websites as well as auxiliary applications. The use of used and idle computers for purpose of scooping on parts and linking them</p>	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Lab activities and additional exercises as tasks - Scientific books. - Daily and monthly exams
58. Course Structure	

Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2 Theoretical + Practical	Introducing the operating system	Learn how to represent numbers and symbols in the calculator	Reading + Lab	Oral and written exam
2	2 Theoretical + Practical	Definition of the physical devices of the calculator	Learn about the mechanism and work of the therapist	Reading + Lab	Oral and written exam
3	2 Theoretical + Practical	Screen properties and icons on the desktop	How to fetch and implement information	Reading + Lab	Oral and written exam
4	2 Theoretical + Practical	Screen properties and icons on the desktop	Identify the types of vectors and registers and how the computation and logic unit works	Reading + Lab	Oral and written exam
5	2 Theoretical + Practical	Office Automation Software	Recognize the work of the control unit	Reading + Lab	Oral and written exam
6	2 Theoretical + Practical	Introduction to the program Microsoft office word	Identify the types of memory	Reading + Lab	Oral and written exam
7	2 Theoretical + Practical	Explanation of the program tools Microsoft office word	Identify the types of memory and the function of each memory	Reading + Lab	Oral and written exam
8	2 Theoretical + Practical	Explanation of the main menus in the program Microsoft office word	Identify the function of cache memory	Reading + Lab	Oral and written exam
9	2 Theoretical + Practical	Explanation of the preparation of pages in the program Microsoft office word	Identify the work of the input and output system and the importance of the Simos chip	Reading + Lab	Oral and written exam
10	2 Theoretical + Practical	An explanation of the print settings in the program Microsoft office word	Recognize the importance of the hard drive	Reading + Lab	Oral and written exam
11	2 Theoretical + Practical	Introduction to the program Microsoft office Excel	Recognize the difference between hard drive and laser	Reading + Lab	Oral and written exam
12	2	Explanation of	Identify the importance and	Reading	Oral and

	Theoretical + Practical	the program tools Microsoft office word	work of input and output devices	+ Lab	written exam
13	2 Theoretical + Practical	Explanation of the main menus in the program Microsoft office w Excel	Learn about the work of the keyboard,touch screen,mouse	Reading + Lab	Oral and written exam
14	2 Theoretical + Practical	Explanation of the preparation of pages in the program Microsoft office Excel	Identify the types of display screens and the work of the printer	Reading + Lab	Oral and written exam
15	2 Theoretical + Practical	An explanation of the print settings in the program Microsoft office Excel	How to transfer data between RAM and CPU	Reading + Lab	Oral and written exam
16	2 Theoretical + Practical	Explanation of functions in Microsoft office Excel	Identify the function of the translator, the assembly, the interpreter	Reading + Lab	Oral and written exam
17	2 Theoretical + Practical	Work on the C++ software environment	Learn about the architecture of each processor	Reading + Lab	Oral and written exam
18	2 Theoretical + Practical	Work on Assembly software environment	Identify the names of public and private purpose records and indexing and media records	Reading + Lab	Oral and rittenw exam
19	2 Theoretical + Practical	Identify and deal with some operating system errors	Identify the divisions of memory and physical and logical address	Reading + Lab	Oral and written exam
20	2 Theoretical + Practical	Introduction to the program Microsoft office power point	Learn about transport orders	Reading + Lab	Oral and written exam
21	2 Theoretical + Practical	Explanation of the program tools Microsoft office power point	Knowing the arithmetic commands	Reading + Lab	Oral and written exam
22	2 Theoretical + Practical	Identify problems related to output and input devices	Recognize logical commands	Reading + Lab	Oral and written exam
23	2 Theoretical + Practical	Identify problems related to output and input devices	Identify problems with input devices	Reading + Lab	Oral and written exam
24	2	Identify problems	Identify problems related to	Reading	Oral and

	Theoretical + Practical	related to output and input devices	output devices	+ Lab	written exam
--	----------------------------	--	----------------	-------	-----------------

59.Course Evaluation

Monthly exam (20) / practical exam (5) for the first semester
 Monthly exam (20) / practical exam (5) for the second semester
 Final exam (50)

Learning and Teaching Resources.60

- Computer organization: 5 (fifth) edition by Carl Hamacher, Zvonko G.Vranesic
- Computer organization and architecture: edition) by design for performance (8 William stalling
- Introduction to Computers Peter Norton available online

Course Description Form

61.Course Name:

Design of translators

:Code Course.62

304CSCO

:Year /Semester.63

2023/2024

:Description Preparation date.64

26/3/2024

65.Available attendance forms:

Physical attendance

66.Number of Credit Hours (Total) / Number of Units (Total)

2 hours / 6 units

67. Course administrator name (list all, if more than one)

Name: Teacher Rasha Sobhi Hameed

Email: dr.ahmedd.k.abbas@uodiyala.edu.iq

68.Course Objective

Course Objective

- Explain the basic functions and services provided by the translator.
- Analysis of the core components of a compiler program, including lexical analysis, syntax analysis, semantic analysis, multimedia code, optimization, and code generation.
- Study each stage of the translator's design separately in terms of how it handles inputs and how they convert them to another format to improve the efficiency of program implementation, ensure program validity, facilitate language transfer, and enable software development

	<p>productivity.</p> <ul style="list-style-type: none"> • Plan the structure and algorithms for the lexical and grammatical stages, implement coding and build the compiler based on the structure and algorithm. • Clarify the strategies used and adopted during this compilation phase. • Explore troubleshooting, reporting, and resolution methods to ensure a robust and actionable language. • Comparison and variation of different stages of the translator
69. Teaching and Learning Strategies	
strategy	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Lab activities and additional exercises as tasks - Scientific books. - Daily and monthly exams

70. Course structure					
Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2	Introduction to translators and their objectives	Introduction to the design and history of translators	Reading Lab +	
2	2	Explain the hierarchy of programming languages and the language processing system that includes (preprocessor, translation, compiler, loader/link)	Programming languages	Reading + Lab	Oral exam
3	2	The concept of the translator, the compiler and the interpreter, the strengths, weaknesses and differences between them.	Types of translators	Reading + Lab	Oral exam
4	2	Compiler structure, explanation of compiler components.	Analysis - Synthesis model of aggregation	Reading + Lab	Oral exam
5	2	The compiler stages include lexical analysis, syntax analysis, semantic analysis, media code, optimization,	Stages of translators	Reading + Lab	Contest

		and code generation.			
6	2	The role of lexical analysis - the interaction of the lexical analyzer with the linguistic analyst	Parser	Reading + Lab	Oral exam
7	2	Scanner and proper lexical analysis	The role of the linguistic analyst	Reading + Lab	Oral exam
8	2	Base and type of lexicon, style and code of lexical analysis	Lexical lexicon, style and token	Reading + Lab	Homework
9	2	Exercises for attribute code and explain the relationship between the attribute symbol and the symbol table with examples	Token attribute	Reading + Lab	Oral exam
10	2	Lexical buffer definition, functions, and examples	Lexical buffer	Reading + Lab	Oral exam
11	2	Define the table of symbols and the services it provides with examples, its importance and its impact on the translation process	Code table	Reading + Lab	Oral exam
12	2	Provide explanations and examples of regular expressions, string operations, and language operations	Select the segment	Reading + Lab	Oral exam
13	2	Include regular definitions for building a pattern for token types (number, id, symbols, operations, and keywords) with examples.	Parser Model Design	Reading + Lab	Oral exam
14	2	Define transition schemes for tokens, define their	Transition plans	Reading + Lab	Contest

		components, and provide examples and exercises.			
15	2	Recognize reserved words and identifiers	Clip recognition	Reading + Lab	Homework
16	2	Design procedure to identify token types with examples and exercises	Learn about design code	Reading + Lab	amOral ex
17	2	Exam		Reading + Lab	Oral exam
18	2	Memory Management	Contiguous memory allocation + memory allocation	Reading + Lab	Oral exam
19	2	Memory Management	Paging and Page Table Structure	Reading + Lab	Oral exam
20	2	Memory Management	Hash and table structure	Reading + Lab	Oral exam
21		Student Application for Schools		Reading + Lab	Oral exam
22		Student Application for Schools		Reading + Lab	Oral exam
23		Student Application for Schools		Reading + Lab	Oral exam
24		Student Application for Schools		Reading + Lab	Oral exam
25		Student Application for Schools		Reading + Lab	Oral exam
26		Student Application for Schools		Reading + Lab	Oral exam
27		Memory Management	Hardware Support		Oral exam
28		Memory Management	Fragmentation		Oral exam
29	2	Disk scheduling	FCFS Scheduler + SSTF Scheduler + Scan Scheduling		Applied algorithm in LAB
30		Exam			

71.Course Evaluation

Monthly exam (30) / Practical exam (10) / Final exam (60)

72.Learning and Teaching Resources

- Abraham Silbershatz OS Concepts, Tenth Edition, 2018

Introduction to operating system design and implementation	-
www.tutorialspoint.com	-
- www.javatpoint.com/operating-system	

Course Description Form

73.Course Name:	
Logical design	
:Code Course.74	
CSLD101	
:Year /Semester.75	
2023/2024	
:Description Preparation date.76	
20/3/2024	
77.Available attendance forms:	
Physical attendance	
78.Number of Credit Hours (Total) / Number of Units (Total)	
4 hours / 6 units	
79. Course administrator name (list all, if more than one)	
Name: Prof. Dr. Intisar Kazem Abd	
Email: PureComp.intesar.abd@uodiyala.edu.iq	
80.Course Objective	
Course Objective	Enable students to know the most prominent basics of logical design Enable students to understand logical design methodology Enabling students to design logically of all kinds Enable students to compare different types of logical circuits Enable students to understand aspects of logic circuit design Enable students to connect websites to the Internet Qualifying students to teach this subject to middle and middle school students Qualifying students to work in different private and governmental sectors
81.Teaching and Learning Strategies	
strategy	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Lab activities and additional exercises as tasks - Scientific books. - Daily and monthly exams

82. Course structure					
Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2	Understand the basics of logical	Fundamentals of logical design	Reading Lab +	

		design			
2	2	Knowledge of the basics of numerical systems	Fundamentals of numerical systems	Reading + Lab	Classroom participation in preparation
3	2	Understand the types of numerical systems	Types of numerical systems	Reading + Lab	Classroom participation in preparation
4	2	Understanding logic gates	Logic gates	Reading + Lab	Classroom participation in preparation
5	2	Knowledge of calculations	Calculations	Reading + Lab	Classroom participation in preparation
6	2	Understanding complements	Complements	Reading + Lab	Classroom participation in preparation
7	2	Know the use and application of Boleyn equations	Boleyn equations	Reading + Lab	Classroom participation in preparation
8	2	معرفة مصطلح POS,SOP	مصطلح POS,SOP	Reading + Lab	Classroom participation in preparation
9	2	Use and implement K-MAP	مصطلح K-MAP	Reading + Lab	Classroom participation in preparation
10	2		examination	Reading Lab +	
11	2	Combination circle design	Half collector circle design and full collector	Reading + Lab	Classroom participation in preparation
12	2	Combination circle design	Half collector circle design and full collector	Reading + Lab	Classroom participation in preparation
13	2	Combination circle design	Half collector circle design and full collector	Reading + Lab	Classroom participation in preparation
14	2	Design of the subtraction circuit	Semi-extruder circle design and full extrusor	Reading + Lab	Classroom participation in preparation
15	2	Design of the subtraction circuit	Semi-extruder circle design and full extrusor	Reading + Lab	Classroom participation in preparation

16	2	Circuit Design	Comparative circle design		Classroom participation in preparation
17	2	Circuit Design	Comparative circle design	Reading + Lab	Classroom participation in preparation
18	2		examination	Reading Lab +	
19	2	Circuit Design	Multiplication circuit design	Reading + Lab	Classroom participation in preparation
20	2	Circuit Design	Multiplication circuit design	Reading + Lab	Classroom participation in preparation
21	2	Circuit Design	Design of hopper circuits	Reading + Lab	Classroom participation in preparation
22	2	Circuit Design	Design of hopper circuits	Reading + Lab	Classroom participation in preparation
23	2	Circuit Design	Design of hopper circuits	Reading + Lab	Classroom participation in preparation
24			exam		
25					

83.Course Evaluation

Monthly exam (40) / Practical exam (10) / Final exam (50)

Learning and Teaching Resources.84

1- Digital system : principle & applications

Digital by: M.Morais Mand -

2-Digital system : principle & applications

1. Digital by: M.Morais Mand

3-vu.puressi.odiyala.7.iq

Course Description Form

85.Course Name:

Networks & Communication

86.Course Code:

402CSCN

87.Semester / Year:

2023/2024

88.Description Preparation Date:

25/3/2024

89.Available Attendance Forms:

Physical attendance

90.Number of Credit Hours (Total) / Number of Units (Total)

4 Hours / 6 Unites

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

Name: Assist. Prof. Dr. Hassan Hadi Saleh

Email: hassan.hadi@uodiyala.edu.iq

92.Course Objectives

Course Objectives

The Networks and Communications course is designed to provide students with a deep understanding of the concepts, principles and mechanisms behind communications and networking systems. The main objectives of the Networks and Communications course include:

- The concept of communications and networks: Introducing students to the basic concepts of networks and their devices, including their types, intermediate connection methods, protocols, and standards.
- Networking devices: Explaining the devices and their types for networks and communications, such as routers, network cards, hubs, and network splitters.
- Connection topology: Teaching students how to connect networks of many types, such as star, tree, and hybrid ring connections, and the differences between them.
- Types of networks: Introducing students to the existing types of networks, such as local networks, the Internet, or wireless networks.
- Transmission media: Exploring the media through which data is transferred, such as wired connections and their types, or transmission via wireless media, such as wave frequencies.
- Digital and analog signals: Introducing students to distinguishing between them and differentiating between them.
- Network protocols: Explaining the layers of the traditional protocol and knowing the work of each layer and how they interrelate, in addition to presenting other protocols.
- Addressing: Introducing students to the mechanisms and methods of addressing in sending and receiving, and ways to know each address, such as the real and the virtual, and the benefits of each.

93.Teaching and Learning Strategies

Strategy

- Traditional lectures and discussion method.
- Laboratory activities and additional exercises as assignments.
- Scientific books.
- Daily and monthly exams

94. Course Structure

Week	Hours	Required Learning	Unit or subject name	Learning	Evaluation
------	-------	-------------------	----------------------	----------	------------

		Outcomes		method	method
1	2	Introduction to Computer Networks	Introduction to Computer Networks	Lecture +LAB	
2	2	Fundamentals of Data communication, Characteristics of data communication	Fundamentals of Data communication, Characteristics of data communication	Lecture +LAB	Oral Exam
3	2	Network Model	Network Model	Lecture +LAB	Oral Exam
4	2	Network components	Network components	Lecture +LAB	Oral Exam
5	2	Data and Signal fundamentals	Data and Signal fundamentals	Lecture +LAB	Monthly Exam
6	2	Transmission technology	Transmission technology	Lecture +LAB	Oral Exam
7	2	Analog Signals, Digital Signals	Analog Signals, Digital Signals	Lecture +LAB	Oral Exam
8	2	Network media of transmission and connectivity	Network media of transmission and connectivity	Lecture +LAB	homework
9	2	Network Types (LAN,MAN,WAN)	Network Types (LAN,MAN,WAN)	Lecture +LAB	Monthly Exam
10	2	Network devices	Network devices	Lecture +LAB	
11	2	Networking Topologies (Bus, Star, Ring, Mish,...)	Networking Topologies (Bus, Star, Ring, Mish,...)	Lecture +LAB	homework
12	2	Data Communication and Protocol Architecture	Data Communication and Protocol Architecture	Lecture +LAB	Oral Exam
13	2	Network reference models	Network reference models	Lecture +LAB	Oral Exam
14	2	Networking layers (OSI Model, 7 layers)	Networking layers (OSI Model, 7 layers)	Lecture +LAB	Quiz
15	2	OSI Model	OSI Model	Lecture +LAB	Monthly Exam
16	2	Link two pc with server	Physical LAYER		
17	2	Two Modes Operating	Data link LAYER	Lecture +LAB	Oral Exam
18	2	IoT	Network LAYER	Lecture +LAB	Oral Exam
19	2	FTP	Logical address (Address space, IPv4, IPv6 addressing) Dynamic Addressing, routable and non-routable	Lecture +LAB	Oral Exam
20	2	IPv4	Classless & Class full	Lecture +LAB	Monthly Exam
21		Students applied to the schools			
22		Students applied to the schools			

23		Students applied to the schools			
24		Students applied to the schools			
25		Students applied to the schools			
26		Students applied to the schools			
27		IPv6	Modulation technologies	Lecture +LAB	Oral Exam
28		VoiP	IoT	Lecture +LAB	Oral Exam
29	2	VPN	Datagram networks routing table, efficiency, delay	Lecture +LAB	Monthly Exam
30		Wireless	IEEE and networking standards	Lecture +LAB	

95.Course Evaluation

Monthly exam (30) / Practical exam(10) / final exam (60)

96.Learning and Teaching Resources

- Computer Networking, seventh edition, Behrouz A. Forouzan,
- Fourth Edition ,Computer Networks ,ANDREW S. TANENBAUM

Course Description Form

97.Course Name:

Computer Architecture

:Code Course.98

305CSCA

:Year /Semester.99

2023/2024

:Description Preparation date .100

25/3/2024

101. Available attendance forms:

Physical attendance

102.	Number of Credit Hours (Total) / Number of Units (Total)	
	2 hours / 4 units	
103.	Course administrator name (list all, if more than one)	
	Name: Assoc. Prof. Saad Abdulaziz Shaaban Email: saad.shaban@uodiyala.edu.iq	
104.	Course Objective	
Course Objective	<ul style="list-style-type: none"> • Explain the basic functions and services performed by the computer system. • Analysis of the basic components of a computer system, including processing, input and output processes. • Clarify the types of memory systems in the computer and ways to manage them. • And identify ways to connect memory to the CPU and input and output devices. • Learn how to make a memory address map. • Clarify the types of volatile memory and ways of writing in it. • Clarify the differences between the types of physical and imaginary addresses and the relationship between them. • Explain the methods of memory management using software hardware or physical hardware. • Clarify the most important ways to protect memory. • Identify ways to connect memory with peripherals using vectors and interfaces. • Learn about the basics of the CPU and the ways it is connected with peripherals. • Identify the structure and organization of the core parts of the CPU and the types of recorders. • Explain the ways of data transfer within the CPU through examples. • Learn about the CPU command cycle and ways to fetch and implement it. • Boycott procedures during the implementation of the directives and methods of their application. • Discuss the basic principles of the processor controller and methods of conducting it • Identify the types of precise instructions and the difference between them and follow the steps of their implementation through practical examples. • Illustrate the idea of parallel treatment, execution lines and the difference between their different types with mathematical examples. • Identify the concept of data reliability in the computer system and deal with branch instructions. 	
105.	Teaching and Learning Strategies	
strategy	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Additional activities and exercises as tasks. - Scientific books. - Daily and monthly exams. 	

106. Course structure					
Week	Hours	Required Learning	Unit or Subject Name	Learning	Evaluation

		Outcomes		method	method
1	2	Introduction to Memory	Introduction to memory and memory system hierarchy	Reading	
2	2	Main memory types	Main memory and its types RAM,ROM	Reading	Oral exam
3	2	Memory address map	Memory address map, how to link it to the processor + daily test	Reading	Oral exam
4	2	Volatile memory	Volatile memory and its data transfer methods from the main memory	Reading	Oral exam
5	2	Volatile memory2	Volatile memory and its data transfer methods from the main memory	Reading	Oral exam
6	2	Volatile and phantom memory	Write and initialize volatile memory, phantom memory, address space and memory space +daily test	Reading	Oral exam
7	2	Memory pages	Memory address mapping using pages	Reading	Oral exam
8	2	Associative memory and its tables	Associative memory page table and ways to replace it	Reading	Oral exam
9	2	Memory Management Systems	Memory, Software and Physical Management Systems + Daily Test	Reading	Oral exam
10	2	Fragmented memory pages	Fragmented memory pages and mathematical examples	Reading	Oral exam
11	2	Memory protection	Protect memory and data access rights	Reading	Oral exam
12	2	Input and output interface	I/O interface and peripherals +Daily Test	Reading	Oral exam
13	2	Input and output vectors	Models of input and output vectors and interfaces	Reading	Oral exam
14	2	Transfer data between computer parts	Data transfer between parts of the central computer and input and output devices	Reading	Oral exam
15	2	Direct memory access	Direct Memory Access (DMA)	Reading	Oral exam
16	2	Exam			
17	2	CPU connections and I/O modules	CPU and I/O Communications	Reading	Oral exam
18	2	CPU	CPU Basics and Parts	Reading	Oral exam
19	2	CPU2	CPU parts and data paths +Daily test	Reading	Oral exam
20	2	Guidelines Course	CPU Routing Cycle, Fetching and Processing Instruction	Reading	Oral exam
21	2	Handling Directives and Interruptions	Examples of processing simple calculations, procedures for interrupting directives	Reading	Oral exam

22	2	Control Unit	Processor controller and its function +Daily test	Reading	Oral exam
23	2	Process Control Methods	Process control methods, physical control and precision program control	Reading	Oral exam
24	2	Exact Directives	Accurate instructions and their types, vertical and horizontal	Reading	Oral exam
25	2	Accurate Instructions2	Examples of processing precise instructions + daily test	Reading	Oral exam
26	2	Parallel processing and execution lines	Parallel processing and pipelining execution lines	Reading	Oral exam
27	2	Execution Lines Wizard	Execution Lines Wizard, Execution Line Types	Reading	Oral exam
28	2	Sports Execution Lines and the lines of implementation of the instructions	Arithmetic Pipeline and Instruction pipeline + daily test	Reading	Oral exam
29	2	Data reliability and branching instruction	Data dependency in the computer system and dealing with branch instructions	Reading	Oral exam
30		Exam			

107. Course Evaluation

Monthly exam (40) / Final exam (60)

108. Learning and Teaching Resources

.third edition ‘Computer Architecture Maurice Manu -

- www.tutorialspoint.com

Course Description Form

109.	Course Name:
	Advanced databases
	:Code Course .110
	403CSOS
	:Year /Semester .111
	2023/2024
	:Description Preparation date .112
	20/3/2024
113.	Available attendance forms:

Physical attendance	
114.	Number of Credit Hours (Total) / Number of Units (Total)
4 hours / 6 units	
115.	Course administrator name (list all, if more than one)
Name: Assoc. Prof. Qasim Radam Mahmoud Email: pure.comp.qasim.radam@uodiyala.edu.iq	
116.	Course Objective
Course Objective	<ul style="list-style-type: none"> • Explain the basic functions and services provided by the database management system. • Analysis of the basic components of a database management system • Types of database management systems. • Introduction to structured query language. • Explain how to create tables and add data to them by writing their own queries. • Explain the basic operations that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc). • Create relationships between tables by querying. • Explain the queries for comparing literal strings. • Explain aggregate operations on digital fields. • Explanation of scaling functions in the query language and how to write them • Explain the orders to arrange data in ascending or descending order. • Main query and subquery. • Use commands (IN, Exists) in subqueries. • Use commands (Any, All) with subqueries. • Explain operations on groups (Union, Intersect, difference). • Explain joint queries or joins queries. • Explanation of System Data queries.
117.	Teaching and Learning Strategies
strategy	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Lab activities and additional exercises as tasks - Scientific books. - Daily and monthly exams

118. Course structure					
Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2	Understand what a database management system is	Introduction to Database Management System	Diction + examples	Oral exam
2	2	Understand what a database management system is	Introduction to Database Management System	Diction + Examples	Oral exam
3	2	Know the basic components of a database system	Analysis of the basic components of a database management system (entity,	Diction + Examples	Oral exam

			properties, relationships)		
4	2	The student should know the types of database management systems	Types of database management systems.	Diction + Examples	Oral exam
5	2	The student should know the types of database management systems	Types of database management systems.	Diction + Examples	Oral exam
6	2	The student should know the types of database management systems	Types of database management systems.	Diction + Examples	Oral exam
7	2		examination		
8	2	Learn about structured query language	Introduction to structured query language.	Diction + Examples + Laboratory	Oral exam
9	2	The ability to create tables within a specific database by querying.	Explain how to create tables and add data to them by writing their own queries.	Diction + Examples + Laboratory	Oral and practical exam
10	2	The ability to write queries for operations that can be performed on data within tables.	Explain the basic operations that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc).	Diction + Examples + Laboratory	Oral and practical exam
12	2	The ability to write queries for operations that can be performed on data within tables.	Explain the basic operations that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc).	Diction + Examples + Laboratory	Oral exam and homework
13	2	The ability to write queries for operations that can be performed on data within tables.	Explain the basic operations that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc).	Diction + Examples + Laboratory	Oral and practical exam
14	2	The ability to write queries for operations that can be performed on data within tables.	Explain the basic operations that can be performed on the data within the tables (adding, deleting, modifying, displaying, etc).	Diction + Examples + Laboratory	Oral and practical exam
15	2		examination		
13	2	The student should be able to establish relationships	Create relationships between tables by querying.	Diction + Examples +	Oral and practical exam

		between tables		Laboratory	
14	2	The student should be able to write queries for comparison between literal strings.	Explain the queries for comparing literal strings.	Diction + Examples + Laboratory	Oral and practical exam
15	2	The student should be able to write queries for comparison between literal strings.	Explain the queries for comparing literal strings.	Diction + Examples + Laboratory	Oral and practical exam
16	2	The student should be able to write queries to perform aggregate operations on table data	Explain aggregate operations on digital fields.	Diction + Examples + Laboratory	Oral and practical exam
17	2	The student should be able to write queries to perform aggregate operations on table data	Explain aggregate operations on digital fields.	Diction + Examples + Laboratory	Oral and practical exam
18	2		examination		
19	2	The student should be able to write sizing queries	Explanation of scaling functions in the query language and how to write them	Diction + Examples + Laboratory	Oral and practical exam
20	2	The student should be able to write sizing queries	Explanation of scaling functions in the query language and how to write them	Diction + Examples + Laboratory	Oral and practical exam
21	2	The student should be able to write queries in ascending and descending order.	Explain the orders to arrange data in ascending or descending order.	Diction + Examples + Laboratory	Oral and practical exam
22	2	The student should be able to write queries that link the external and internal query	Main query and subquery.	Diction + Examples + Laboratory	Oral and practical exam
23	2	The student should be able to write queries that link the external and internal query	Main query and subquery.	Diction + Examples + Laboratory	Oral and practical exam
24	2		examination		

25		The student should be able to write queries that include commands (In, Exists)	Use commands (IN, Exists) in subqueries.	Diction + Examples + Laboratory	Oral and practical exam
26	2	The student should be able to write queries that include commands (Any, All)	Use commands (Any, All) in subqueries.	Diction + Examples + Laboratory	Oral and practical exam
27	2	The student should be able to write queries for operations on groups.	Explain operations on groups (Union, Intersect, difference).	Diction + Examples + Laboratory	Oral and practical exam
28	2	The student should be able to write queries for linking data from more than one table and show them.	Explain joint queries or joins queries.	Diction + Examples + Laboratory	Oral and practical exam
29	2	The student should be able to write queries for linking data from more than one table and show them.	Explain joint queries or joins queries.	Diction + Examples + Laboratory	Oral and practical exam
30		The student should be able to write queries for system data	Explanation of System Data queries.	Diction + Examples + Laboratory	Oral and practical exam

119. Course Evaluation

Monthly exam (30) / Practical exam (10) / Final exam (60)

Learning and Teaching Resources .120

- Hector Georrica, Jeffrey D. Ullman and Jennifer Widom, "Database System The Compleat Book", Pearson Prentice Hall, 2002-2009.
- C.Avgerou And T. Cornford, Developing Information System: concepts, Issues, and Practices, 2nd Ed., Macmillian Press, 1998.

Course Description Form

121.	Course Name:
	Design of translators
	:Code Course .122
	304CSCO
	:Year /Semester .123
	2023/2024
	:Description Preparation date .124

26/3/2024

125. Available attendance forms:

Physical attendance

126. Number of Credit Hours (Total) / Number of Units (Total)

2 hours / 6 units

127. Course administrator name (list all, if more than one)

Name: Teacher Rasha Sobhi Hameed

Email: dr.ahmedd.k.abbas@uodiyala.edu.iq

128. Course Objective

Course Objective

- Explain the basic functions and services provided by the translator.
- Analysis of the core components of a compiler program, including lexical analysis, syntax analysis, semantic analysis, multimedia code, optimization, and code generation.
- Study each stage of the translator's design separately in terms of how it handles inputs and how they convert them to another format to improve the efficiency of program implementation, ensure program validity, facilitate language transfer, and enable software development productivity.
- Plan the structure and algorithms for the lexical and grammatical stages, implement coding and build the compiler based on the structure and algorithm.
- Clarify the strategies used and adopted during this compilation phase.
- Explore troubleshooting, reporting, and resolution methods to ensure a robust and actionable language.
- Comparison and variation of different stages of the translator

129. Teaching and Learning Strategies

strategy

- Traditional lectures and discussion method.
- Lab activities and additional exercises as tasks
- Scientific books.
- Daily and monthly exams

130. Course structure

Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2	Introduction to the design and history of translators	Introduction to translators and their objectives	Reading Lab +	
2	2	Programming languages	Explain the hierarchy of programming languages and the language processing system that includes (preprocessor, translation, compiler, loader/link)	Reading + Lab	Oral exam
3	2	Types of translators	The concept of the translator, the compiler and the interpreter, the strengths, weaknesses and differences between them.	Reading + Lab	Oral exam

4	2	Analysis - Synthesis model of aggregation	Compiler structure, explanation of compiler components.	Reading + Lab	Oral exam
5	2	Stages of translators	The compiler stages include lexical analysis, syntax analysis, semantic analysis, media code, optimization, and code generation.	Reading + Lab	Contest
6	2	Parser	The role of lexical analysis - the interaction of the lexical analyzer with the linguistic analyst	Reading + Lab	Oral exam
7	2	The role of the linguistic analyst	Scanner and proper lexical analysis	Reading + Lab	Oral exam
8	2	Lexical lexicon, style and token	Base and type of lexicon, style and code of lexical analysis	Reading + Lab	Homework
9	2	Token attribute	Exercises for attribute code and explain the relationship between the attribute symbol and the symbol table with examples	Reading + Lab	Oral exam
10	2	Lexical buffer	Lexical buffer definition, functions, and examples	Reading + Lab	Oral exam
11	2	Code table	Define the table of symbols and the services it provides with examples, its importance and its impact on the translation process	Reading + Lab	Oral exam
12	2	Select the segment	Provide explanations and examples of regular expressions, string operations, and language operations	Reading + Lab	Oral exam
13	2	Parser Model Design	Include regular definitions for building a pattern for token types (number, id, symbols, operations, and keywords) with examples.	Reading + Lab	Oral exam
14	2	Transition plans	Define transition schemes for tokens, define their components, and provide examples and exercises.	Reading + Lab	Contest
15	2	Clip recognition	Recognize reserved words and identifiers	Reading + Lab	Homework
16	2	Learn about design code	Design procedure to identify token types with examples and exercises	Reading + Lab	Oral exam
17	2	Exam			
18	2	Automata Ltd (FA)	Finite Automata (FA) types (NFA and DFA) with components with examples	Reading + Lab	Oral exam

19	2	Convert NFA to DFA	Explain and apply the algorithm (subset construction) to build a DFA from an NFA with examples	Reading + Lab	exam Oral
20	2	Convert RE to FA	Applying the Thomson Building Algorithm with Examples	Reading + Lab	Oral exam
21		Lexical error	Enumerate types with examples of lexical error and how to program the scanner	Reading + Lab	Oral exam
22		Syntax Analysis (Analyzer)	Syntax analysis, roles, and location of parser functions in the compiler form with examples	Reading + Lab	Oral exam
23		Context-free grammar (CFG)	For context-free rules and components (CFGs)	Reading + Lab	Oral exam
24		Derivatives and Analysis Trees	Strategies for Derivations and Analysis Trees with Examples	Reading + Lab	Oral exam
25		Rules issues	Detect grammar problems and apply algorithms to exclude them (ambiguity, left iteration, left-factor analysis)	Reading + Lab	Oral exam
26		Types of Arabic trees	Top-down analyzer and bottom-up	Reading + Lab	Oral exam
27		Types of analyzer from top to bottom	Undo and not retreat with examples		Oral exam
28		Recursive Ratio Analysis	Recursive Ratio Analysis Design		Oral exam
29	2	PPM Predictive Analysis Method	First, follow the stack table entry string and parser using PPM		Applied algorithm in LAB
30		Exam			

131. Course Evaluation

Monthly exam (40) / Practical exam (10) / Final exam (50)

132. Learning and Teaching Resources

- Principles, Techniques and Tools of Translators, by ALFRED, V. Aho; Monica, S. Lam; Jeffrey, Ullman. Pearson Education, 2004
- Introduction to translator design. by Mugensen, Turbine Egidios. Springer Nature, 2024.
- A practical approach to building a translator. By Watson, Des Springer, 2017.

Course Description Form

133. Course Name:

Structured Programming

:Code Course .134

403CSOS

		:Year /Semester	.135
		2023/2024	
		:Description Preparation date	.136
		20/3/2024	
137.	Available attendance forms:		
	Physical attendance		
138.	Number of Credit Hours (Total) / Number of Units (Total)		
	5 hours / 6 units		
139.	Course administrator name (list all, if more than one)		
	Name: Assoc. Prof. Qasim Radam Mahmoud		
	Email pure.comp.qasim.mahmood@uodiyala.edu.iq		
140.	Course Objective		
Course Objective	<ul style="list-style-type: none"> • Explain how to write algorithms and flowcharts to find solutions to the different problems to be programmed. • Teach the student the general structure of the C++ program and its basic components. • Explain reading and writing commands in C++ . • Explain the conditional sentences of various types and structures in the C++ language. • Explain the repetition sentences of various types and structures in the C++ language. • Explanation of matrices of various types and adapted to dealing with them within the C++ program. • Teaching the student the mechanism of writing and calling sub-programs within the general program of the C++ language. • Enable the student to deal with text files (opening, writing in them, deleting them, closing them) through the program . C++ • Enable the student to be able to analyze various problems and write programs to solve various problems using the C++ language. 		
141.	Teaching and Learning Strategies		
strategy	<ul style="list-style-type: none"> - Traditional lectures and discussion method. - Lab activities and additional exercises as tasks - Scientific books. - Daily and monthly exams 		

142. Course structure					
Week	Hours	Required Learning Outcomes	Unit or Subject Name	Learning method	Evaluation method
1	2	Introduction to C++ programming	Introduction to C++ programming	Diction + examples	Oral exam
2	2	Write algorithms and flowcharts to solve various problems	How to write algorithms and flowcharts to solve various problems	Diction + Examples	Oral exam
2	2	Write algorithms and flowcharts to solve various problems	How to write algorithms and flowcharts to solve various problems	Diction + Examples	Oral exam

		problems			
3	2	General structure of the C++ language program	Components of the general structure of the C++ language program	Diction + Examples + Laboratory	and Oral practical exam
4	2	Ability to define data	Define data types	Diction + Examples + Laboratory	and Oral practical exam
5	2	Use arithmetic and logical operations	Definition of arithmetic and logical operations	Diction + Examples + Laboratory	Oral and practical exam
6	2	Learn how to increase , decrease and equality	Processes of increasing, decreasing, and equalizing	Diction + Examples + Laboratory	Oral and practical exam
7	2	Ability to write comments and prioritize operations	Comment phrases and specify precedence in the execution of operations	Diction + Examples + Laboratory	Oral and practical exam
8	2	The student should be able to define constants, variables and reserved words	Define constants, variables, and reserved words	Diction + Examples + Laboratory	Oral and practical exam
9	2	Transfers	Definition of transfers	Diction + Examples + Laboratory	Oral exam and homework
10	2	The student should be able to use the input and output phrases	Input and output phrases	Diction + Examples + Laboratory	Oral and practical exam
11	2	The student should be able to use condition sentences within the program	Conditional phrases	Diction + Examples + Laboratory	Oral and practical exam
12	2	The student should be able to use condition sentences within the program	Conditional phrases	Diction + Examples + Laboratory	Oral and practical exam
13	2	The student should be able to use repetition sentences within the program	Rotary phrases (repetition sentences)	Diction + Examples + Laboratory	Oral and practical exam
14	2	The student should understand the overlap of phrases	Overlapping phrases	Diction + Examples + Laboratory	Oral and practical exam
15	2	The student should understand the	Overlapping phrases	Diction + Examples	Oral and practical

		overlap of phrases		+ Laboratory	exam
16	2	The student should be able to use jumping phrases	Jumping phrases	Diction + Examples + Laboratory	Oral and practical exam
17	2	The student should be able to use jumping phrases	Jumping phrases	Diction + Examples + Laboratory	Oral and practical exam
18	2	The student should be able to define and use the one-dimensional matrix within the program	One-dimensional matrix	Diction + Examples + Laboratory	Oral and practical exam
19	2	The student should be able to define and use the one-dimensional matrix within the program	One-dimensional matrix	Diction + Examples + Laboratory	Oral and practical exam
20	2	The student should be able to define and use the two-dimensional matrix within the program	Two-dimensional matrix	Diction + Examples + Laboratory	Oral and practical exam
21		The student should be able to define and use the two-dimensional matrix within the program	Two-dimensional matrix	Diction + Examples + Laboratory	Oral and practical exam
22		The student should be able to call ready-made functions within the program	Subprograms (Ready-made Functions)	Diction + Examples + Laboratory	Oral and practical exam
23		The student should be able to call ready-made functions within the program	Subprograms (Ready-made Functions)	Diction + Examples + Laboratory	Oral and practical exam
24		The student should be able to call functions and methods of announcing them within the program	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam
25		The student should be able to call functions and methods of announcing them	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam

		within the program			
		The student should be able to call functions and methods of announcing them within the program	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam
27		The student should be able to call functions and methods of announcing them within the program	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam
28		The student should be able to call functions and methods of announcing them within the program	Creating functions and methods of declaring and calling them	Diction + Examples + Laboratory	Oral and practical exam
29	2	The student must be able to create files, read from them and write in them	Create files, store them, and read from them	Diction + Examples + Laboratory	Oral and practical exam
30		The student must be able to create files, read from them and write in them	Create files, store them, and read from them	Diction + Examples + Laboratory	Oral and practical exam

143. Course Evaluation

Monthly exam (40) / Practical exam (10) / Final exam (50)

Learning and Teaching Resources

- Stanly B.Lippman, Josee Lajoie, and Barbara E. Moo,"C++ Primer", forth edition, 2005.
- Juan Soulie,"C++ language Tutorial", 2008
- Any book and article which is related to the C++ language
- <http://www.cplusplus/>
- <http://www.tutorialspoint.com/>
- <http://www.w3schools.com/cpp/>