

1. Course Name:	
COMPUTATION THEORY	
2. Course Code:	
3. Semester / Year: 2025-2026	
Semester	
4. Description Preparation Date:	
17-11-2025	
5. Available Attendance Forms:	
Weekly (Theory: 5 hours,)	
6. Number of Credit Hours (Total) / Number of Units (Total)	
Theory: 5 Hours Total Units: 125	
7. Course administrator's name (mention all, if more than one name)	
Name: sarah haytham Email: sarah.haytham@muc.edu.iq	
8. Course Objectives	
Course Objectives	This course aims to teach students how to solve problems efficiently in a computational model, using algorithms, which are the programming foundation for all programming languages. The field is divided into three main branches: automata theory and language computational theory, and computational complexity theory, which together represent fundamental capabilities and limitations of computers.
9. Teaching and Learning Strategies	
Strategy	1. -knowledge and understanding A1-enable the student to know and understand theoretical principles of languages and logical expressions A2-the student should describe how the machine works for mathematical operations and languages and stages it goes through to solve problems A3-enable the student to learn how to design machines for languages, arithmetic and logical operations. B-subject-specific skills B1-logical thinking B2-give the student tasks to design and solve problems and build a tree showing the work of the machine and how to move to motivate the student to acquire skills to solve software problems. C-Thinking Skills C1-ability to work together and lead group C2-ability to solve problems and think collectively
2. Course Structure	
Week	Required Learning Outcomes
1	Introduction, languages, examples and applications
2	Regular expression +examples
3	Introduction to Finite automata +examples
4	Deterministic finite automata + rules and examples
5	Non- Deterministic finite automata + rules and examples
6	How to Convert NFA to DFA + rules and examples

7	How to Convert NFA with e-move to NFA Without e-move + rules and examples
8	Equivalence of mealy and moor machine
9	Introduction to grammar, PSG, CSG, CFG
10	Regular grammar, Left linear grammar, Right linear grammar
11	Trees, left and right most derivation, top-down-bottom-up
12	Chomsky Normal Form(NFA) + rules and examples
13	Pushdown automata + rules and examples
14	Trees, left and right most derivation, top-down-bottom-up
15	Mid Term Exam

3. Learning and Teaching Resources

The course strategy aim to explain how the terminology, notations and techniques of computer the are necessary for the undergraduate students of the computer science to develop their programm skills. also include:- -Workshops and projects. -Assessment Methods - Exams.

Course Description Form