

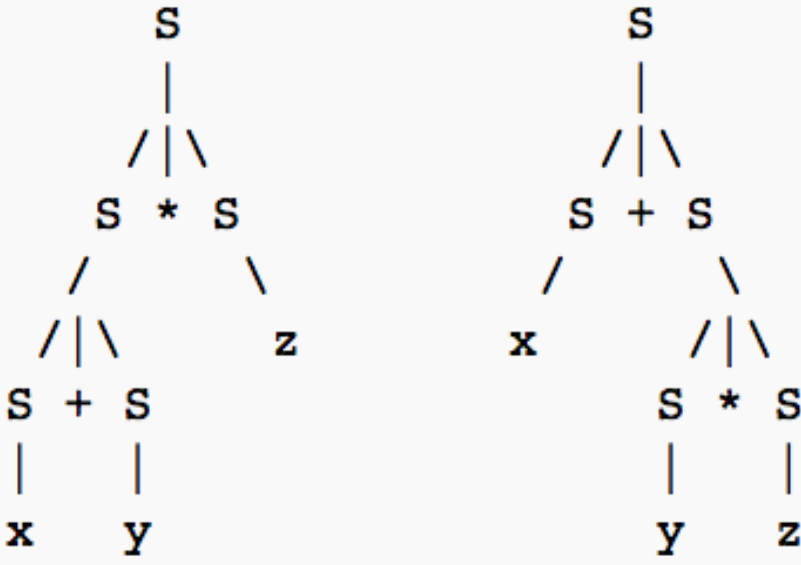
## Module (Course Syllabus) Catalogue 2022-2023

College/ Institute	Erbil Technical Engineering	
Department	Information Systems Engineering	
Module Name	Computational Theory	
Module Code	Computational Theory	
Degree	Technical Diploma <input type="checkbox"/> Bachler <input checked="" type="checkbox"/> High Diploma <input type="checkbox"/> Master <input type="checkbox"/> PhD <input type="checkbox"/>	
Semester	Full year course	
Qualification	PhD	
Scientific Title	Lecturer	
ECTS (Credits)	6	
Module type	Prerequisite <input type="checkbox"/> Core <input checked="" type="checkbox"/> Assist. <input type="checkbox"/>	
Weekly hours	4	
Weekly hours (Theory)	( 2 )hr Class	(93.5)Total hrs Workload
Weekly hours (Practical)	( 2 )hr Class	(68.5)Total hrs Workload
Number of Weeks	15	
Lecturer (Theory)	Dr. Sara Raouf Muhamad Amin	
E-Mail & Mobile NO.	<a href="mailto:Sara.muhamad@epu.edu.iq">Sara.muhamad@epu.edu.iq</a> 07504881488	
Lecturer (Practical)	Mr. Muhamad Adham	
E-Mail & Mobile NO.	+964 750 4995745	
Websites		

# Course Book

<p><b>Course Description</b></p>	<p>This subject is a part of theory of computation. It comprises the fundamental mathematical properties of computer hardware, software, and certain applications thereof. In studying this subject we seek to determine what can and cannot be computed, how quickly, with how much memory, and on which type of computational model. The subject has obvious connections with engineering practice, and as in many sciences, it also has purely philosophical aspects.</p>				
<p><b>Course objectives</b></p>	<ol style="list-style-type: none"> <li>1. Introduce students to the mathematical foundations of computation including automata theory; the theory of formal languages and grammars; the notions of algorithm, decidability, complexity, and computability.</li> <li>2. Enhance/develop students' ability to understand and conduct mathematical proofs for computation and algorithms.</li> </ol>				
<p><b>Student's obligation</b></p>	<p>The attendance of students in both lectures and labs will have extra credit. He / she is required to continuously follow the lectures, submits homework and assignments. Expect quizzes any time. This part of the assessment defined in section <b>Assessment scheme</b>.</p>				
<p><b>Required Learning Materials</b></p>	<p>Java or C++ or any Programming Language and a computer device</p>				
<p><b>Evaluation</b></p>	<p><b>Task</b></p>		<p><b>Weight (Marks)</b></p>	<p><b>Due Week</b></p>	<p><b>Relevant Learning Outcome</b></p>
	<p>Paper Review</p>				
	<p>Assignments</p>	<p>Homework</p>	<p>5</p>	<p>15</p>	
		<p>Class attendance</p>	<p>2</p>	<p>15</p>	
		<p>Report</p>	<p>10</p>	<p>2</p>	<p>Academic writing</p>
<p>Seminar</p>		<p>10</p>	<p>1</p>	<p>presentation</p>	

	Essay			
	Project			
	Quiz	8	2	
	Lab.	15	1	
	Midterm Exam	10	1	Student evaluation
	Final Exam	40	2	Student evaluation
Total	100	25		
<b>Specific learning outcome:</b>	<p>1- Students will learn several formal mathematical models of computation along with their relationships with formal languages.</p> <p>2- In particular, they will learn regular languages and context free languages which are crucial to understand how compilers and programming languages are built</p> <p>3- Students will understand that there are limitations on what computers can do, and learn examples of unsolvable problems.</p> <p>4- Students will learn that certain problems do not admit efficient algorithms, and identify such problems.</p>			
<b>Course References:</b>	<ol style="list-style-type: none"> <li>1. Introduction to the Theory of Computation by Michel Sipser, 2nd Ed., Cengage Learning, 2005.</li> <li>2. Introduction to Automata Theory, Languages, and computations by John E. Hopcroft, Rajeev Motwani, Jeffery D. Ullman, 2nd Ed., 2001</li> </ol>			
<b>Course topics (Theory)</b>			<b>Week</b>	<b>Learning Outcome</b>
Introduction to Computational Theory			1	General information about FF
Deterministic finite automata (DFA) & Nondeterministic finite automata (NFA).			2	Design DFA and NFA
Equivalence of DFA and NFA,			3	Convert NFA to DFA

Myhill Nerode Theorem – Table Filling Method	4	Minimizing DFA
Finding a String in a text	5	How compiler work
Regular expressions $(a + b(b + ab)^*aa)^*$	6	What is regular language
Midterm exam	7	Evaluation the student
Mealy Machine and Moore Machine	8	Design Moore Machine
Context Free Grammar (CFG) arise in linguistics where they are used to describe the structure of sentences and words in a natural language	9	Design CFG
		
Simplification of CFG	10	How to simplify CF
Push Down Automata is a finite automata with extra memory called stack which helps <b>Pushdown automata</b> to recognize Context Free Languages.	11	Design an automata

Turing Machines and reduction	12	Turing Machines
Turing Machine Programming Techniques	13	Programm Turing Machine
Review and final exam.	14	Students final evaluation

### Questions Example Design

Q1/ design a DFA that accept any strings that ends with 00 or 11

Solution:

Q: {q0, q1, q2, q3, q4}

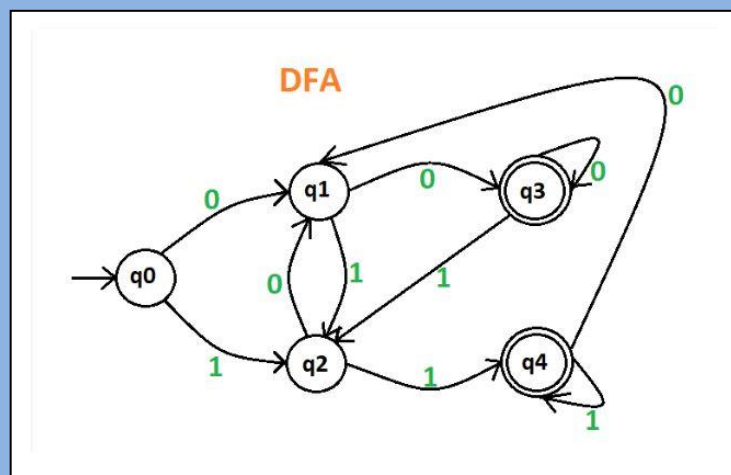
$\Sigma$ : {0,1}

q<sub>0</sub> = q0

F = {q3, q4}

	0	1
q0	q1	q2
q1	q3	q2
q2	q1	q4
q3	q3	q3
q4	q1	q4

$\delta =$



Q2/ Find the equivalent DFA for the NFA given by  $M = \{A, B, C\}, (a, b), \delta, A, \{C\}$  where  $\delta$  is given by

Table (2-1)

	a	b
--	---	---

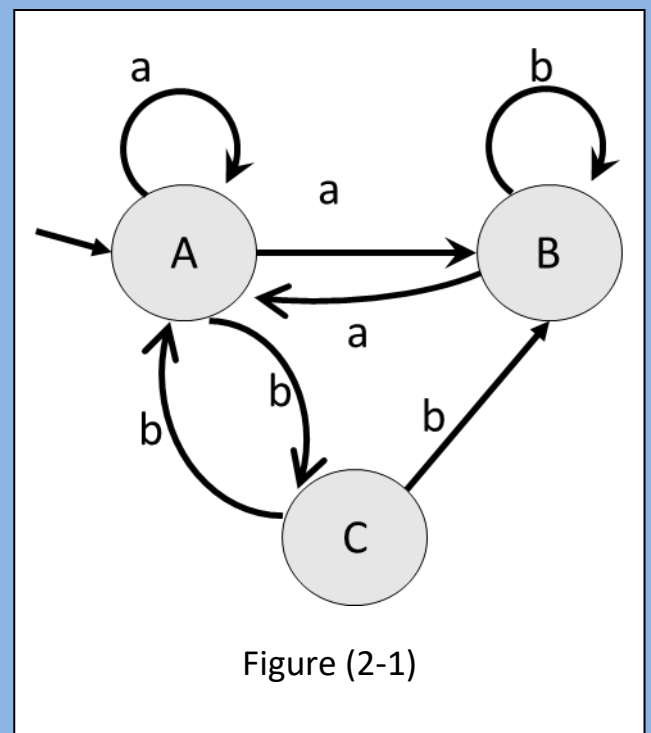
A	A,B	C
B	A	B
C	-	A,B

**Solution:**

From the given table (2-1) we designed NFA as in figure (2-1). Then we will create table (2-2) as shown below.

**Table (2-2)**

	a	b
A	AB	C
AB	AB	BC
BC	A	AB
C	D	AB
D	D	D



From the table (2-2) we create DFA as shown in figure (2-2)

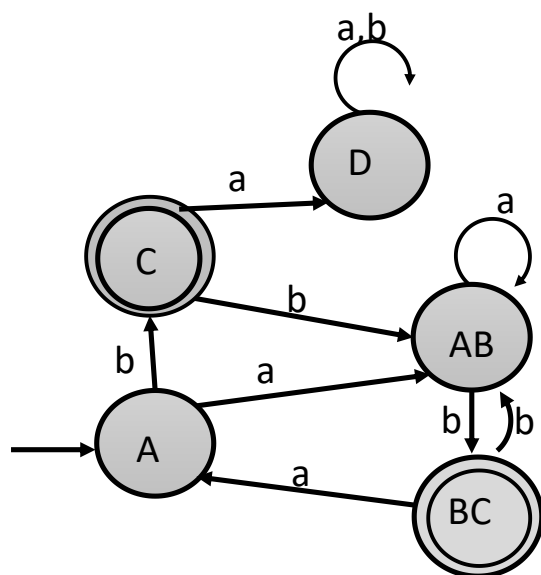


Figure (2-2) DFA

**Extra notes:**

### External Evaluator

I found that this course book is easy to learn and beneficial in building reflective learners who took the time to think through their work and monitor their own progress. Very clear sub-titles, good detail in explaining the subjects and good application examples have been presented.

**Assist. Lect. Najat Yohana Danha**

