Course Specification
(Bachelor)

Course Title:	Theory of Computation
Course Code:	CSC 1305
Program:	Bachelor in Computer Science
Department:	Computer Science
College:	Computers and Information Technology
Institution:	University of Tabuk
Version:	1.0
Last Revision Date:	27 July 2022

Table of Contents

A. General information about the course:

1. Course Identification

1. Credit hours: (3)					
2. Course type					
A.	☐ University	☐ College	☒ Department	☐ Track	☐ Others
B.	☒ Required		☐ Elective		
3. Level/year at which this course is offered: (Level 6/3rd Year)					
4. Course general Description:					
This course emphasizes computability and computational complexity theory. Topics include regular and context-free languages, deterministic and non-deterministic automata, decidable and undecidable problems, completeness, push down automata and Turing machine.					
5. Pre-requirements for this course (if any):					
CSC1204, MATH1251					
6. Co-requisites for this course (if any):					
NA					
7. Course Main Objective(s):					
Upon Completion of this course, students will be able to: • Acquire a full understanding of Automata Theory as the basis of all computer science languages design • Recognize the Automata theory concepts such as RE's, DFA's, NFA's, Stack's, Turing machines, and Grammars • Design FAs, NFAs, Grammars, languages modeling, small compilers basics • Design sample automata • Minimize FA's and Grammars of Context Free Languages					

2. Teaching mode (mark all that apply)

No	Mode of Instruction	Contact Hours	Percentage
1	Traditional classroom	45	100%
2	E-learning		
3	Hybrid • Traditional classroom • E-learning		
4	Distance learning		

3. Contact Hours (based on the academic semester)

No	Activity	Contact Hours
1.	**Lectures**	45
2.	**Laboratory/Studio**	
3.	**Field**	
4.	**Tutorial**	
5.	**Others (specify)**	
Total		45

B. Course Learning Outcomes (CLOs), Teaching Strategies and Assessment Methods

Code	Course Learning Outcomes	Code of PLOs aligned with program	Teaching Strategies	Assessment Methods
1.0	**Knowledge and understanding**			
1.1	Recognize the role of set theory and logic in the solution of computing problems	K1	Lectures	• Exams • Assignments
1.2	Illustrate the theory of finite state machine (FSM), regular expressions, context free grammar (CFG), push-down automata and Turing machine	K4		
...				
2.0	**Skills**			
2.1	Employ set theory and logic in the solution of	S1	Lectures	• Exams • Assignments

Code	Course Learning Outcomes	Code of PLOs aligned with program	Teaching Strategies	Assessment Methods
	computing problems			
2.2	Distinguish the requirements of the problems	S2		
2.3	Design sequential components based on the concept of finite state machine	S3		
2.4	Translate FSM and CFG into programs using C/C++ or C#	S3		Project
3.0	**Values, autonomy, and responsibility**			
3.1	Communicate and work (effectively, ethically, and professionally) (individually and in groups/teamwork) to accomplish all the assigned duties and projects.	V2	Lectures	Project

C. Course Content

No	List of Topics	Contact Hours
1.	Introduction: basic concepts, Languages: Alphabets, Strings and languages, finite specification of languages, Lexicographic order	3
2.	Automata Theory, Complexity theory, Concatenation, sets and relations, Kleene star, functions and relations, predicates, Regular expressions, Graphs, Boolean logic	3
3.	Finite automata, Class of Regular Languages, Language of a machine	3
4.	Deterministic finite automata, Formal Definition of DFA, Transition Function of DFA, how a DFA computes, Language Machines, Constructing a DFA complement, closed under operation.	3
5.	Non-deterministic, Non regular languages	3
6.	Non-deterministic finite automata, formal definition of NFA, Difference between DFA and NFA, Equivalences of DFA and NFA, construct NFA from a regular expression.	3
7.	Kleene's theorem and non-determinism	3
8.	Context-Free gramma CFG, Regular grammars, context free languages, deriving strings using CFG	3
9.	CFG for simple arithmetic operations, Derivation tree, Parsing, derivation, and	3

	ambiguity	
10.	Context-free language and parsing, CFG applications	3
11.	Chomsky normal form, converting CFG into Chomsky normal form, converting DFA into CFG	3
12.	Pushdown automata PDA, PDA uses stack, PDA transition, How PDA computes	3
13.	Turing Machine TM, key differences between TM and previous machines, formal definition of TM, Description of TM	3
14.	Transition function of TM, TM computation, diagrams for TM, TM configuration	3
15.	Project presentation, Review	3

Total		**45**

D. Students Assessment Activities

No	Assessment Activities *	Assessment timing (in week no)	Percentage of Total Assessment Score
1.	1st Mid Term Exam	6	15%
2.	2nd Mid Term Exam	12	15%
3.	Project	14	20%
4.	Assignments	8 and 14	10%
5.	Final Exam	16	40%

*Assessment Activities (i.e., Written test, oral test, oral presentation, group project, essay, etc.).

E. Learning Resources and Facilities

1. References and Learning Resources

Essential References	•M. Sipser. Introduction to the Theory of Computation, Course Technology Inc, 3rd Ed, 2012. •Thomas A. Sudkamp. Languages and Machines :An Introduction to the Theory of Computer Science. Third Edition, 2006, Addison Wesley; ISBN-13: 978-0321322210.
Supportive References	•Daniel I. A. Cohen. Introduction to Computer Theory. Prentice-Hall, Second Edition, 1997. •J. E. Hopcroft., R. Motwani, and J. D. Ullman Introduction to Automata Theory, Languages, and Computation, 3rd Ed, 2007, Addison Wesley.

Electronic Materials	
Other Learning Materials	

2. Required Facilities and equipment

Items	Resources
facilities (Classrooms, laboratories, exhibition rooms, simulation rooms, etc.)	Classroom
Technology equipment (projector, smart board, software)	**White board, data show projector, computer and internet connection.**
Other equipment (depending on the nature of the specialty)	

F. Assessment of Course Quality

Assessment Areas/Issues	Assessor	Assessment Methods
Effectiveness of Teaching	Faculty, Program Leaders, and Advisory Board	Both Direct and Indirect
	Students	Indirect
Effectiveness of Students Assessment	Faculty, Program Leaders, Advisory Board, and Independent Opinion	Both Direct and Indirect
Quality of Learning Resources	Faculty, Students, and Advisory Board	Indirect
The Extent to which CLOs have been Achieved	Faculty, Program Leaders, Advisory Board, and Independent Opinion	Direct (as in section B) and Indirect/Surveys
	Students	Indirect
Other	-	-

Assessors (Students, Faculty, Program Leaders, Peer Reviewer, Others (specify)

Assessment Methods (Direct, Indirect)

G. Specification Approval

COUNCIL /COMMITTEE	
REFERENCE NO.	
DATE	