



Arab Academy for Science and Technology & Maritime Transport
College of Computing and Information Technology

University/Academy: Arab Academy for Science and Technology & Maritime Transport

Faculty/Institute: College of Computing and Information Technology

Program: B.sc in Computer Science

Course title	Theory of Computation
Course code	CS311

Form no. (11A)
Knowledge and skills matrix for a course

Course content	Week study	Knowledge	Intellectual skills	Professional skills	General skills
Introduction	1	<ul style="list-style-type: none"> Define automata, computability theory, and complexity theory. Understand different methods of proof and essential mathematical background. 	<ul style="list-style-type: none"> 		<ul style="list-style-type: none"> Show the use of general computing facilities.
Deterministic Finite State Automata	2	<ul style="list-style-type: none"> Define and describe finite state automata. Understand how to describe languages using finite automata. 	<ul style="list-style-type: none"> Appreciate the limitations of computational models. 	<ul style="list-style-type: none"> Construct finite automata to describe languages. 	<ul style="list-style-type: none">
Non-Deterministic Finite State Automata	3	<ul style="list-style-type: none"> Define and describe non-determinism. Understand how to describe languages using non-deterministic finite automata. 	<ul style="list-style-type: none"> Differentiate between determinism and non-determinism. Appreciate the limitations of computational models. 	<ul style="list-style-type: none"> Construct nondeterministic automata to describe languages. 	<ul style="list-style-type: none">
Regular Expressions	4	<ul style="list-style-type: none"> Understand what regular expressions are. Understand how to describe languages using regular expressions. 	<ul style="list-style-type: none"> Give examples of regular languages. Appreciate the limitations of computational models. 	<ul style="list-style-type: none"> Formulate regular expression that generates a given regular language. Convert regular expressions into finite automata. Convert finite automata into regular expressions 	<ul style="list-style-type: none">
Non-Regular Languages	5	<ul style="list-style-type: none"> Understand the pigeonhole principle and the pumping lemma. 	<ul style="list-style-type: none"> Create proofs for statements regarding formal computational models. 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none">

Context-Free Grammars	6	<ul style="list-style-type: none"> Understand how to describe languages using regular context-free grammars. 	<ul style="list-style-type: none"> Appreciate the limitations of computational models. 	<ul style="list-style-type: none"> Find a context-free grammar for a context-free language. Find a parse tree, leftmost derivation and rightmost derivation for a word in a context free language. Show that a context free grammar is ambiguous. 	<ul style="list-style-type: none">
7 th week Exam	7				<ul style="list-style-type: none"> Demonstrate the ability to make use of a range of learning resources and to manage one's own learning.
Push-Down Automata	8	<ul style="list-style-type: none"> Define and describe push-down automata. Understand how to describe languages using push-down automata. Understand the relation between regular languages and context-free 	<ul style="list-style-type: none"> Appreciate the limitations of computational models. 	<ul style="list-style-type: none"> Construct push-down automata to describe languages. 	<ul style="list-style-type: none">

		languages.			
Non-Context Free Languages	9	<ul style="list-style-type: none"> Understand how to apply the pumping lemma on non-context-free languages. 	<ul style="list-style-type: none"> Create proofs for statements regarding formal computational models. 	•	•
Turing Machines	10	<ul style="list-style-type: none"> Understand how to use Turing machines to represent computable functions. 	<ul style="list-style-type: none"> Appreciate the limitations of computational models. 	<ul style="list-style-type: none"> Construct Turing Machines to describe languages. 	•
Turing Machines (continued)	11	<ul style="list-style-type: none"> How a Universal Turing machine can simulate any Turing Machine on any input. 	•	•	•
12 th week Exam	12				<ul style="list-style-type: none"> Demonstrate the ability to make use of a range of learning resources and to manage one's own learning.
Complexity Theory	13	<ul style="list-style-type: none"> Understand what time and space complexity is. 	<ul style="list-style-type: none"> Comprehend the limitations of computers in terms of the problems they can solve. 	•	•
Complexity Theory (continued)	14	<ul style="list-style-type: none"> Describe different problem classes. 	<ul style="list-style-type: none"> Comprehend the limitations of computers in terms of the problems they can 	•	•

			solve. •		
Revision.	15				

Course Instructor

Name:

Signature:

Head of Department

Name:

Signature: