Title (Units): **COMP4067 Theory of Computation (3,2,1)**

Course Aims: To introduce formal language theory, formal machine models, tractability and computability.

Prerequisite: MATH1205 Discrete Mathematics

Course Intended Learning Outcomes (CILOs):
Upon successful completion of this course, students should be able to:

No.	Course Intended Learning Outcomes (CILOs)
	Knowledge
1	Explain the concepts of non-deterministic and deterministic finite automata, regular languages, pushdown automata, context-free languages, Turing machines, Church's hypothesis, computability and complexity theory
2	Describe the formal relationships among machines, languages and grammars
	Professional Skill
3	Articulate the power and limitations of different formalisms
4	Solve problems using formal methods
	Attitude
5	Articulate the importance of rigorous solution to computational problems

Calendar Description: This course aims to introduce some fundamental concepts in theoretical computer science. The topics include non-deterministic and deterministic finite automata, regular languages, context-free languages, pushdown automata, Church's hypothesis, Turing machines, computability, and complexity theory.

Teaching and Learning Activities (TLAs):

CILOs	Type of TLA
1-5	Students will learn concepts on theory of computations in lectures and tutorials
1-4	Students will practise the formal methods used in long take-home assignments and examinations

Assessment:

No.	Assessment Methods	Weighting	CILOs to be addressed	Description of Assessment Tasks
1	Continuous Assessment	50%	1-5	Continuous assessments are designed to measure how well students have learned the basic concepts of formal methods. A set of assignments is designed to measure how well students have learned the concepts.
2	Examination	50%	1-4	Final examination questions are designed to see how far students have achieved in understanding of formal methods.

Assessment Rubrics:

	Excellent (A)	Good (B)	Satisfactory (C)	Marginal Pass (D)	Fail (F)
Formal methods	• Demonstrate a thorough understanding on (i) regular languages and finite state automata, (ii)	• Demonstrate a good understanding on (i) regular languages and finite state automata, (ii)	• Demonstrate a considerable understanding on (i) regular languages and finite state automata, (ii)	• Demonstrate a minimal understanding on (i) regular languages and finite state automata, (ii)	• Unable to demonstrate an understanding on (i) regular languages and finite state

	Excellent (A)	Good (B)	Satisfactory (C)	Marginal Pass (D)	Fail (F)
	context free grammars and (iii) Turing machine	context free grammars and (iii) Turing machine	context free grammars and (iii) Turing machine	context free grammars and (iii) Turing machine	automata, (ii) context free grammars and (iii) Turing machine
Computability and complexity theory	• Can describe and explain the concepts of computability • Can analyze the complexity of a given problem	• Can describe and explain mostly the concepts of computability • Can analyze the complexity of a given problem with a high degree of effectiveness	• Can describe and explain considerable concepts of computability • Can analyze the complexity of a given problem with some degree of effectiveness	• Can describe some concepts of computability • Can describe the complexity of a given problem	• Cannot describe the concepts of computability • Cannot describe the complexity of a given problem
Problem solving skills	• Can effectively and correctly apply formal methods to solve a given problem	• Can correctly apply formal methods to solve a given problem	• Can apply formal methods to solve a given problem with some degree of effectiveness	• Can apply some formal methods to solve a substantial part of a given problem	• Cannot apply formal methods to solve a given problem

Course Content and CILOs Mapping:

Content		CILO No.
I	Mathematical Notations, Definitions and Terminology	1
II	Regular Expressions and Finite State Automata	1-2, 4, 5
III	Context-Free Languages and Pushdown Automata	1-2, 4, 5
IV	Computability Theory	1, 3, 5
V	Complexity Theory	1, 3, 5

References:

- M. Sipser. Introduction to the Theory of Computation, 3rd Edition, Course Technology, 2014.
- J. E. Hopcroft, R. Motwani and J. D. Ullman. Introduction to Automata Theory, Languages, and Computation, 3rd Edition, Addison Wesley, 2006.

Course Content:

<u>Topic</u>

I. Mathematical Notations, Definitions and Terminology
   A. Sets, sequences, tuples, graphs, trees
   B. Functions and relations
   C. Inductive proofs

II. Regular Expressions and Finite State Automata
   A. Non-deterministic finite automata and deterministic finite automata
   B. Regular expressions
   C. Variants of finite automata
   D. Properties of regular languages

III. Context-Free Languages and Pushdown Automata

A. Context-free grammars
B. Pushdown automata
C. Derivation trees
D. Chomsky normal form and ambiguity of context-free grammars
E. The pumping lemma

IV. Computability Theory
A. Church’s hypothesis
B. Turing machines and their variants
C. Recursive and recursively enumerable languages

V. Complexity Theory
A. Polynomial time and space
B. Intractable problems
C. Some classical intractable problems