Uka Tarsadia University

C. G. Patel Institute of Technology

B. Tech.

Semester – 6

030090609/030080609

THEORY OF COMPUTATION

EFFECTIVE FROM July-2017

Syllabus version: 1.02

SEMESTER-6
Theory of Computation
(030090609/030080609)

Credits: 3 (Theory) **Contact hours per week: 3 (Theory)**

Objective:

- To introduce fundamental mathematical and computational principles that are foundation of computer science.
- To illustrate how the computational problems can be solved efficiently.
- To acquaint students about finite representations of languages and machines.
- To introduce conceptual tools that can be used in compiler and programming languages.
- To provide a formal connection between algorithmic problem solving and theory of languages.

Outcome:

Upon completion of the course, the student shall be able to

CO1	To able to think analytically and intuitively for problem solving situations in related areas of theory and computation.
CO2	To understand concepts and application of mathematical induction and notation.
CO3	To construct finite states machines and equivalent regular expressions.
CO4	Master context-free languages, push-down automata and Turing recognizable languages.
CO5	Understand key notions such as algorithms, computability, decidability and complexity through problem solving
CO6	To able to solve real time problems of networking, compiler designing, natural language processing etc.

B. Tech.	Subject	Hours
Sem 6	030090609/030080609 Theory of Computation	3 hrs/week
	(Theory)	3 Credits

Sr. No.	Topic	Hours
	Unit – I	
1	**Basic Mathematical Object:** Basic Mathematical Notations and Set Theory, Logic Functions and Relations, Language Definitions	2
2	**Mathematical Inductions and Recursive Definitions:** Proofs, The Principle of Mathematical Induction, Recursive Definitions	2
	Unit – II	
3	**Regular Expressions and Finite Automata:** Regular Expressions ,Regular and Non Regular Languages Finite Automata, Automata with output-Moore machine, Mealy machine, Union, intersection and complement of regular languages	5
4	**Nondeterminism and kleene's theorem:** Non deterministic Finite Automata, Conversion from NFA to FA, NFA- ^ to NFA, Kleene's theorem	3
5	**Regular and non regular languages:** Minimization of Finite automata, Regular And Non Regular Languages – pumping lemma	2
	Unit – III	
6	**Context Free Grammar:** Introduction to CFG, Union, Concatenations and *S Notations of CFL, Regular grammar, Killing Null Production, Killing Unit Production, Derivations trees and Ambiguity, Unambiguous CFG for Algebraic Expressions, Chomsky Normal Form	9
	Unit – IV	
7	**Pushdown Automata:** Introduction to PDA, deterministic PDA, Equivalence of CFG and PDA, Parsing	6
8	**Context-Free and non-Context-Free Languages:** Pumping lemma for CFL, Intersections and Complements of CFL, Non-CFL	4
	Unit – V	

9	**Turing Machine (TM):** Introduction to TM, Computing partial functions with TM, Combining TM, Variations Of TM, Non Deterministic TM, Universal TM, Model Of Computation And Church Turning Thesis	8
Unit – VI		
10	**Recursively Enumerable Languages:** Recursively and Enumerable Languages, Context sensitive, languages and Chomsky hierarchy	2
11	**Computable Functions:** Introduction to computable function, Primitive Recursive Functions, Primitive Recursive Predicates	2

Text Books:

1. John C. Martin – "Introduction to Languages and Theory of Computation", TMH.
2. Daniel Cohen – "Introduction to computer theory", Wiley Publication

Reference Books:

1. Hopcroft, Motwani, Ullman – "Automata Theory, Languages and Computation", Pearson Education.
2. A.V.Aho, Ravi Sethi, J.D.Ullman – "Compiler Tools Techniques", Addison Wesley.

Course objectives and Course outcomes mapping:

- To introduce fundamental mathematical and computational principles that are foundation of computer science: CO2, CO5
- To illustrate how the computational problems can be solved efficiently: CO1, CO6
- To acquaint students about finite representations of languages and machines: CO3, CO4
- To introduce conceptual tools that can be used in compiler and programming languages: CO3, CO4, CO6
- To provide a formal connection between algorithmic problem solving and theory of languages: CO1, CO5

Course units and Course outcome mapping:

	CO1	CO2	CO3	CO4	CO5	CO6
Mathematical Notation & Techniques		√			√	
Regular Languages and Finite Automata			√			√
Context Free Grammar				√		√
Pushdown Automata				√		√
Turing Machine (TM)				√		√
Recursive Enumerable Languages and Computable Functions	√					

Programme Outcomes

- **PO 1: Engineering knowledge:** An ability to apply knowledge of mathematics, science, and engineering
- **PO 2: Problem analysis:** An ability to identify, formulate, review research literature, and analyse engineering problems reaching substantiated conclusions using principles of mathematics, and engineering sciences.
- **PO 3: Design/development of solutions:** An ability to design solutions for engineering problems and to design a component, system, or process that meet the specified needs with appropriate consideration for the public health and safety, along with the cultural, societal, and environmental considerations.
- **PO 4: Conduct investigations of complex problems:** An ability to use research-based knowledge and techniques including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- **PO 5: Modern tool usage:** An ability to create, select, and apply appropriate techniques, resources, and modern engineering tools for prediction and modelling to complex engineering activities with an understanding of the limitations.
- **PO 6: The engineer and society:** Achieve professional success with an understanding and appreciation of ethical behaviour, social responsibility, and diversity, both as individuals and in team environments.
- **PO 7: Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge for sustainable development to articulate a comprehensive world view that integrates diverse approaches to sustainability.
- **PO 8: Ethics:** Apply ethical principles and commit to professional ethics, responsibilities and norms of the engineering practice.
- **PO 9: Individual and team work:** An ability to analyse the local and global impact of computing on individuals, organizations, and society.
- **PO 10: Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- **PO 11: Project management and finance:** Implement project management knowledge, processes, lifecycle and the embodied concepts, tools and techniques in order to achieve project success.
- **PO 12: Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological changes.

Programme Outcomes and Course Outcomes mapping:

Programme Out come	Course outcomes					
	CO1	CO2	CO3	CO4	CO5	CO6
PO1	√	√	√	√	√	√
PO2	√	√	√	√	√	√

PO3	√	√	√	√	√	√
PO4			√	√	√	√
PO5			√	√		
PO6	√	√	√	√	√	√
PO7	√	√	√	√	√	√
PO8	-					
PO9				√	√	√
PO10	-					
PO11	-					
PO12	√	√	√	√	√	√