INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal, Hyderabad -500 043

COMPUTER SCIENCE AND ENGINEERING

COURSE DESCRIPTOR

Course Title	**THEORY OF COMPUTATION**				
Course Code	AIT002				
Programme	B.Tech				
Semester	IV	CSE | IT			
Course Type	Foundation				
Regulation	IARE - R16				
Course Structure	**Theory**			**Practical**	
	Lectures	**Tutorials**	**Credits**	**Laboratory**	**Credits**
	3	-	3	-	-
Chief Coordinator	Mr. P Anjaiah, Assistant Professor				
Course Faculty	Ms. A Jayanthi, Assistant Professor Ms. Uma Shankari, Assistant Professor Ms. Ramya Sree, Assistant Professor				

I. COURSE OVERVIEW:

Introduction to the theory of computation, including models of computation such as Turing machines; theory of programming languages, including grammars, parsing, syntax and semantics. This course in reached to student by power point presentations, lecture notes, and assignment questions ,previous model question papers, multiple choice questions and question bank of long and short answers.

II. COURSE PRE-REQUISITES:

Level	Course Code	Semester	Prerequisites	Credits
UG	ACS002	II	Data Structures	4
UG	AHS013	III	Discrete Mathematical Structures	4

III. MARKS DISTRIBUTION:

Subject	SEE Examination	CIA Examination	Total Marks
THEORY OF COMPUTATION	70 Marks	30 Marks	100

IV. DELIVERY / INSTRUCTIONAL METHODOLOGIES:

✔	Chalk & Talk	✔	Quiz	✔	Assignments	✔	MOOCs
✔	LCD / PPT	✔	Seminars	✘	Mini Project	✔	Videos
✘	Open Ended Experiments						

V. EVALUATION METHODOLOGY:

The course will be evaluated for a total of 100 marks, with 30 marks for Continuous Internal Assessment (CIA) and 70 marks for Semester End Examination (SEE). Out of 30 marks allotted for CIA during the semester, marks are awarded by taking average of two CIA examinations or the marks scored in the make-up examination.

Semester End Examination (SEE): The SEE is conducted for 70 marks of 3 hours duration. The syllabus for the theory courses is divided into five units and each unit carries equal weightage in terms of marks distribution. The question paper pattern is as follows. Two full questions with "either" or "choice" will be drawn from each unit. Each question carries 14 marks. There could be a maximum of two sub divisions in a question.

The emphasis on the questions is broadly based on the following criteria:

50 %	To test the objectiveness of the concept.
50 %	To test the analytical skill of the concept OR to test the application skill of the concept.

Continuous Internal Assessment (CIA):
CIA is conducted for a total of 30 marks (Table 1), with 25 marks for Continuous Internal Examination (CIE), 05 marks for Quiz/ Alternative Assessment Tool (AAT).

Table 1: Assessment pattern for CIA

Component	**Theory**		**Total Marks**
Type of Assessment	CIE Exam	Quiz / AAT	
CIA Marks	25	05	30

Continuous Internal Examination (CIE):
Two CIE exams shall be conducted at the end of the 8^{th} and 16^{th} week of the semester respectively. The CIE exam is conducted for 25 marks of 2 hours duration consisting of two parts. Part–A shall have five compulsory questions of one mark each. In part–B, four out of five questions have to be answered where, each question carries 5 marks. Marks are awarded by taking average of marks scored in two CIE exams.

Quiz / Alternative Assessment Tool (AAT):
Two Quiz exams shall be online examination consisting of 25 multiple choice questions and are be answered by choosing the correct answer from a given set of choices (commonly four). Marks shall be awarded considering the average of two quizzes for every course. The AAT may include seminars, assignments, term paper, open ended experiments, five minutes video and MOOCs.

VI. HOW PROGRAM OUTCOMES ARE ASSESSED:

Program Outcomes (POs)		Strength	Proficiency assessed by
PO 1	**Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.	3	Assignments
PO 2	**Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.	3	Lectures, Assignments
PO 3	**Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.	2	Assignments
PO 4	**Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.	2	Guest Lettuces
PO 5	**Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.	2	Seminars

3 = High; 2 = Medium; 1 = Low

VII. HOW PROGRAM SPECIFIC OUTCOMES ARE ASSESSED:

Program Specific Outcomes (PSOs)		Strength	Proficiency assessed by
PSO 1	**Professional Skills:** The ability to research, understand and implement computer programs in the areas related to algorithms, system software, multimedia, web design, big data analytics, and networking for efficient analysis and design of computer-based systems of varying complexity.	2	Lectures, Assignments
PSO 2	**Problem-Solving Skills:** The ability to apply standard practices and strategies in software project development using open-ended programming environments to deliver a quality product for business success.	-	-
PSO 3	**Successful Career and Entrepreneurship:** The ability to employ modern computer languages, environments, and platforms in creating innovative career paths, to be an entrepreneur, and a zest for higher studies.	1	Guest Lectures

3 = High; 2 = Medium; 1 = Low

VIII. COURSE OBJECTIVES (COs):

The course should enable the students to:	
I	Comprehend abstract, mathematical models of computation and use them to solve computational problems.
II	Interpret the relationship between formal languages in Chomsky's hierarchy and different Machines.
III	Analyze and explain the behavior of push-down automata.
IV	Understand the limits and capacities of Turing‘s machines to recognize languages.

IX. COURSE LEARNING OUTCOMES (CLOs):

CLO Code	CLO's	At the end of the course, the student will have the ability to:	PO's Mapped	Strength of Mapping
AIT002.01	CLO 1	Use the definitions and notations for sets, relations and functions in defining and study Finite Automata	PO1; PO2	3
AIT002.02	CLO 2	Knowledge on formal languages and Kleene's Theorem to intend programming languages	PO1; PO2; PO3	2
AIT002.03	CLO 3	Construct deterministic and nondeterministic finite state automata (DFA and NFA) for solving simple decision problems.	PO1; PO2; PO4; PO5	2
AIT002.04	CLO 4	Perform conversions between nondeterministic finite automata and deterministic finite automata and regular expressions and finite state automata to gain knowledge about formal proofs in computer science	PO1; PO2; PO3; PO4	2
AIT002.05	CLO 5	Knowledge on recursive definitions of regular languages, regular expressions and the use of regular expressions to represent regular languages	PO1; PO2; PO3; PO4	2
AIT002.06	CLO 6	Detailed knowledge on the relationship between regular expressions and finite automata	PO2; PO3	2
AIT002.07	CLO 7	Identify that few languages are not regular by using Pumping lemma	PO4	2
AIT002.08	CLO 8	Knowledge on Left Linear grammar, Right Linear grammars and converting grammars into Finite Automata.	PO1; PO2; PO5	2
AIT002.09	CLO 9	Understand the fundamental role played by Context-Free Grammars (CFG) in designing formal computer languages with simple examples	PO2; PO3	2
AIT002.10	CLO 10	Knowledge on Context-Free Grammars so that able to prove properties of Context-Free Grammars.	PO2	3
AIT002.11	CLO 11	Identify relationship between regular languages and context-free grammars	PO1; PO2; PO3	2
AIT002.12	CLO 12	Use the pumping lemma for Context Free Languages to show that a language is not context-free	PO2; PO4	2
AIT002.13	CLO 13	Understand the equivalence between Context-Free Grammars and Non-deterministic Pushdown Automata	PO1; PO2; PO3	2
AIT002.14	CLO 14	Understand deterministic Pushdown Automata to parse formal language strings by using (i) top down or (ii) bottom up techniques	PO2; PO4; PO5	2
AIT002.15	CLO 15	Knowledge on converting Context-Free Grammars into pushdown automata to identify the acceptance of a string by the Context Free Language	PO1; PO2	3
AIT002.16	CLO 16	Understand the path processing computation using Turing Machines (Deterministic and Non-Deterministic) and Church-Turing Thesis in computers.	PO1; PO2; PO4; PO5	1

CLO Code	CLO's	At the end of the course, the student will have the ability to:	PO's Mapped	Strength of Mapping
AIT002.17	CLO 17	Knowledge on non-halting Turing Machine accepted by Recursively Enumerable Languages	PO1; PO4	1
AIT002.18	CLO 18	Understand the power of the Turing Machine, as an abstract automaton, that describes computation, effectively and efficiently	PO1; PO4	1
AIT002.19	CLO 19	Theory of Computation is important in programming language design, parsers, web-scrappers, Natural Language Processing (NLP), and is at the heart of modern compiler architectures.	PO1; PO2; PO5	3
AIT002.20	CLO 20	Process the knowledge and skills for employability and to succeed in national and international level competitive exams.	PO5	2

3 = High; 2 = Medium; 1 = Low

X. MAPPING COURSE LEARNING OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

(CLOs)	Program Outcomes (POs)												Program Specific Outcomes (PSOs)		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CLO 1	3	2												2	
CLO 2	3	3	2											2	
CLO 3	3	3		2	2								2		
CLO 4	3	2	3	2										3	
CLO 5	2	3	2	2										3	
CLO 6		3	2										2		
CLO 7				3									2		
CLO 8	2	3			2									2	
CLO 9		2	3											2	
CLO 10		3												2	
CLO 11	2	3	2										2	2	1
CLO 12		3		3									2	2	
CLO 13	3	2	2										2	2	
CLO 14		3		2	2									2	
CLO 15	2	3											2	2	
CLO 16	3	3		2	2									3	
CLO 17	3			2										1	

(CLOs)	Program Outcomes (POs)												Program Specific Outcomes (PSOs)		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CLO 18	3			2									2	2	
CLO 19	3	2			2									2	
CLO 20					1								2		1

3 = High; 2 = Medium; 1 = Low

XI. ASSESSMENT METHODOLOGIES – DIRECT

CIE Exams	PO1, PO2, PO3, PO4, PO5	SEE Exams	PO1, PO2, PO3, PO4, PO5	Assignments	PO1, PO2 PO3, PO4, PO5	Seminars	-
Laboratory Practices	-	Student Viva	-	Mini Project	-	Certification	-
Term Paper	-						

XII. ASSESSMENT METHODOLOGIES - INDIRECT

✔	Early Semester Feedback	✔	End Semester OBE Feedback
✘	Assessment of Mini Projects by Experts		

XIII. SYLLABUS

UNIT-I	FINITE AUTOMATA
Fundamentals: Alphabet, strings, language, operations; Introduction to finite automata: The central concepts of automata theory, deterministic finite automata, nondeterministic finite automata, an application of finite automata, finite automata with epsilon transitions.	
UNIT-II	**REGULAR LANGUAGES**
Regular sets, regular expressions, identity rules, constructing finite automata for a given regular expressions, conversion of finite automata to regular expressions, pumping lemma of regular sets, closure properties of regular sets (proofs not required), regular grammars-right linear and left linear grammars, equivalence between regular linear grammar and finite automata, inter conversion.	
UNIT-III	**CONTEXT FREE GRAMMARS**
Context free grammars and languages: Context free grammar, derivation trees, sentential forms, right most and leftmost derivation of strings, applications. Ambiguity in context free grammars, minimization of context free grammars, Chomsky normal form, Greibach normal form, pumping lemma for context free languages, enumeration of properties of context free language (proofs omitted).	
UNIT-IV	**PUSHDOWN AUTOMATA**
Pushdown automata, definition, model, acceptance of context free language, acceptance by final state and acceptance by empty stack and its equivalence, equivalence of context free language and pushdown automata, inter conversion;(Proofs not required);Introduction to deterministic context free languages and deterministic pushdown automata.	
UNIT-V	**TURING MACHINE**
Turing machine: Turing machine, definition, model, design of Turing machine, computable functions, recursively enumerable languages, Church's hypothesis, counter machine, types of Turing machines (proofs not required), linear bounded automata and context sensitive language, Chomsky hierarchy of	

languages.
Text Books:
1. John E. Hopcroft, Rajeev Motwani, Jeffrey D.Ullman, ―Introduction to Automata, Theory, Languages and Computation‖, Pearson Education, 3rd Edition, 2007.
Reference Books:
1. John C Martin, ―Introduction to Languages and Automata Theory, Tata McGraw-Hill, 3rd Edition, 2007.
2. Daniel I.A. Cohen, ―Introduction to Computer Theory, John Wiley & Sons, 2nd Edition, 2004.

XIV. COURSE PLAN:

The course plan is meant as a guideline. Probably there may be changes.

Lecture No	Topics to be covered	Course Learning Outcomes (CLOs)	Reference
1	Alphabet, strings, language, operations	CLO 1	T1: 1.5-1.6
2	Introduction to finite automata: The central concepts of automata theory	CLO 1	T1: 2.1-2.2
3	Deterministic finite automata	CLO 3	T1: 2.2-2.3
4-5	Nondeterministic finite automata	CLO 3	T1: 2.3-2.4
6	An application of finite automata	CLO 4	T1: 2.4-2.5
7	Finite automata with epsilon transitions	CLO 2	T1: 2.5-2.6
8-9	Finite Automata with output: Moore and Melay Machines	CLO 3	T1: 2.7-2.9
10	Regular sets, regular expressions, identity rules	CLO 5	T1: 3.1-3.2
11	Constructing finite automata for a given regular expressions	CLO 5	T1: 3.1-3.2
12-13	Conversion of finite automata to regular expressions	CLO 5	T1: 3.1-3.2
14	Pumping lemma of regular sets	CLO 5	T1: 4.1-4.2
15	Closure properties of regular sets (proofs not required)	CLO 6	T1: 4.1-4.2
16-17	Regular grammars-right linear and left linear grammars	CLO 7	T1: 4.4-4.5
18	Equivalence between regular linear grammar and finite automata, inter conversion.	CLO 7	T1: 4.4-4.5
19	Context free grammar	CLO 8	T1: 5.1-5.2
20-22	derivation trees, sentential forms, right most and leftmost derivation of strings	CLO 9	T1: 5.1-5.2
23	Ambiguity in context free grammars	CLO 10	T1: 5.4-5.5
24-25	Minimization of context free grammars	CLO 11	T1: 7.4-7.5
26-27	Chomsky normal form, Greibach normal form	CLO 12	T1: 7.4-7.5
28-29	Pumping lemma for context free languages, properties	CLO 13	T1: 7.2-7.3
30	Pushdown automata, definition, model	CLO 14	T1: 6.1-6.2
31-33	Acceptance by final state and acceptance by empty stack and its equivalence	CLO 14	T1: 6.2
34-35	Equivalence of context free language and pushdown automata, inter conversion.	CLO 15	T1: 6.3
36	Deterministic context free languages and deterministic push down automata	CLO 16	T1: 6.4
37-38	Turing machine: Turing machine, definition, model	CLO 17	T1: 8.1-8.2

Lecture No	Topics to be covered	Course Learning Outcomes (CLOs)	Reference
39-40	Design of Turing machine, computable functions,	CLO 18	T1: 8.1-8.2
41-43	Recursively enumerable languages, Types of Turing machines and Church's hypothesis.	CLO 19	T1: 8.2-8.6
44-45	Linear bounded automata and context sensitive language, Chomsky hierarchy of languages.	CLO 20	T1: 9.1- 9.8

XV. GAPS IN THE SYLLABUS - TO MEET INDUSTRY / PROFESSION REQUIREMENTS:

S NO	Description	Proposed actions	Relevance with POs	Relevance with PSOs
1	Finite automata with output	Seminars / Guest Lectures / NPTEL	PO 1, PO 2, PO 3	PSO 1, PSO 2
2	Deterministic Pushdown Automata	Seminars / Guest Lectures / NPTEL	PO 2, PO 3	PSO 1
3	JFLAP Automation Tool	Assignments / Laboratory Practices	PO 1, PO 3, PO 4	PSO 2

Prepared by:
Mr. P Anjaiah, Assistant Professor

HOD, CSE