JNTUA COLLEGE OF ENGINEERING (AUTONOMOUS): PULIVENDULA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
B.Tech II Year II Semester
SUBJECT: Formal Languages and Automata Theory (FLAT)

LESSON PLAN

Course Code	15ACS13			
Course Title	Formal Languages and Automata Theory			
Course Structure	Lectures	Tutorials	Practical's	Credits
	3	1	0	3
Course Coordinator /Team of Instructors	Sri G. Murali			

I. Course Overview:

The course aims to introduce the basic methods and conclusions of the Theory of Computation. At the end of the course, students learn to apply these methods to problems from different fields and be guided by the results in searching for computational solutions to the problems.

II. Prerequisite(s):

Level	Credits	Periods / Week	Prerequisites
UG	3	3	Mathematical background, Logical Thinking

III. Assessment:

FORMATIVE ASSESEMENT	
Mid Semester Test I for 20 Marks in first 2 units is conducted at the end of 9th week. Mid Semester Test II for 20 Marks in last three units is conducted at the end of the course work. 80% Marks taken from the test which secured highest marks and 20% marks from the other test is taken as final	20 Marks
Multiple Choice Mid Semester Test I for 10 Marks form first 2 units is conducted at the end of 9th week. Multiple Choice Mid Semester Test II for 10 Marks in last 3 units is conducted at the end of the course work. 80% Marks taken from the test which secured highest marks and 20% marks from the other test is taken as final	10 Marks
Total (Formative)	30 Marks

SUMMATIVE ASSESMENT	
End Semester Examination in all units is conducted for 70 Marks	70 marks
Grand Total	100 Marks

IV. Course Objectives:

1. Understand formal definitions of machine models.
2. Classify machines by their power to recognize languages.
3. Understanding of formal grammars, analysis.
4. Understanding of hierarchical organization of problems depending on their complexity.
5. Understanding of the logical limits to computational capacity.
6. Understanding of undecidable problems.

V. Course Outcomes:

1. Student can learn basic methods and conclusions of the Theory of Computation.
2. Students can learn how to apply these methods to problems from different fields and be guided by the results in searching for computational solutions to the problems.

VI. Program outcomes:

a An ability to apply knowledge of computing, mathematical foundations, algorithmic principles, and computer science and engineering theory in the modeling and design of computer-based systems to real-world problems (fundamental engineering analysis skills)

b An ability to design and conduct experiments, as well as to analyze and interpret data (information retrieval skills)

c An ability to design , implement, and evaluate a computer-based system, process, component, or program to meet desired needs, within realistic constraints such as economic, health and safety, manufacturability, and sustainability (Creative Skills)

d An ability to function effectively on multi-disciplinary teams (team work)

e An ability to analyze a problem, identify, formulate and use the appropriate computing and engineering skills for obtaining its solution (engineering problem solving skills)

f Obtaining the knowledge of algorithmic skills regarding data structures. (program oriented skills)

g An ability to communicate effectively both in writing and orally (speaking / writing skills)

h The broad education necessary to analyze the local and global impact of computing and engineering solutions on individuals, organizations, and society (engineering impact assessment skills)

i Recognition of the need for, and an ability to engage in continuing professional development and life-long learning (continuing education awareness)

j A Knowledge of structural skills which are related to theoretical skills for programming (detailed subject oriented skills).

k An ability to use current techniques, skills, and tools necessary for computing and engineering practice (practical engineering analysis skills)

l An ability to apply design and development principles in the construction of software and hardware systems of varying complexity (software hardware interface)

m An ability to recognize the importance of professional development by pursuing postgraduate studies or face competitive examinations that offer challenging and rewarding careers in computing (successful career and immediate employment).

VII. Syllabus:

FORMAL LANGUAGES AND AUTOMATA THEORY

L	T	P	C
3	1	0	3

Course Objective:

The course aims to introduce the basic methods and conclusions of the Theory of Computation. At the end of the course, students learn to apply these methods to problems from different fields and be guided by the results in searching for computational solutions to the problems.

1. Understand formal definitions of machine models.
2. Classify machines by their power to recognize languages.
3. Understanding of formal grammars, analysis
4. Understanding of hierarchical organization of problems depending on their complexity
5. Understanding of the logical limits to computational capacity
6. Understanding of undecidable problems

UNITI:

Introduction:

Basics of set theory, Relations on sets, Types of Formal Proof, Deductive proofs, Functions, Types of functions, Proofing Techniques, proof by Contra positive, Proof by contradiction, Counter examples, Fundamentals of Automata Theory:- Alphabets, Strings, Languages, Problems, Grammar formalism, Chomsky Hierarchy

Finite Automata: An Informal picture of Finite Automata, Deterministic Finite Automata (DFA), Non Deterministic Finite Automata (NFA), Finite Automata with Epsilon transitions (ϵNFA or NFA- ϵ), Finite Automata with output, Conversion of one machine to another,Minimization of Finite Automata, Myhill-Nerode Theorem.

UNIT – II:

Regular Languages: Regular Expressions (RE), Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic laws for Regular Expressions, The Arden’s Theorem, Using Arden’s theorem to construct RE from FA, Pumping Lemma for RLs, Applications of Pumping Lemma, Equivalence of Two FAs, Equivalence of Two REs, Construction of Regular Grammar from RE, Constructing FA from Regular Grammar, Closure properties of RLs, Decision problem’s of RLS, Applications of REs and FAs

UNIT – III:

Context Free Grammars and Languages: Definition of Context Free Grammars (CFG), Derivations and Parse trees, Ambiguity in CFGs, Removing ambiguity, Left recursion and Left factoring, Simplification of CFGs, Normal Forms, Linear grammars, Closure properties for CFLs, Pumping Lemma for CFLs, Decision problems for CFLs, CFG and Regular Language

UNIT – IV:

Push Down Automata (PDA): Informal introduction, The Formal Definition, Types of PDA, Graphical notation, Instantaneous description, The Languages of a PDA, Equivalence of PDAs and CFGs, Deterministic Push Down Automata, Two Stack PDA.

UNIT – V:

Turing Machines and Undecidability: Basics of Turing Machine (TM), Transitional Representation of TMs, Instantaneous description, Non Deterministic TM, Conversion of

Regular Expression to TM, Two stack PDA and TM, Variations of the TM, TM as an integer function, Universal TM, Linear Bounded Automata, TM Languages, Unrestricted grammar, Properties of Recursive and Recursively enumerable languages, Undecidability, Reducibility, Undeciadable problems about TMs, Post's Correspondence Problem(PCP), Modified PCP.

Text Books:
1."Introduction to Automata Theory, Languages, and Computation", Third Edition, John E.Hopcroft, Rajeev Motwani, Jeffery D. Ullman, PEARSON.

Reference Books:
1. "Introduction To Languages And The Theory of Computation", John C Martin, The McGraw-Hill Companies, Third Edition. (TATA McGRAW HILL)
2. "Theory of Computation", Vivek Kulkarni, OXFORD.
3. Introduction to the Theory of Computation, Michel Sipser, 2nd Edition, Cengage Learning
4. "Theory of computer Science Automata, Languages and Computation", K.L.P. Mishra, N.Chandrasekaran, PHI, Third Edition.
5. "Fundamentals of the Theory of Computation, Principles and Practice", Raymond Greenlaw, H. James Hoover, MK (MORGAN KAUFMANN)
6. Finite Automata and Formal Language A Simple Approach, A.M. Padma Reddy, Pearson

VIII. Course Plan:

The course plan is meant as a guideline. There may probably be changes.

S.No	**Date**	**Topic**	Objectives	**Outcomes**	**Reference**
			UNIT- I		
1	**28.11.19(2)**	**Introduction:** Basics of set theory, Relations on sets, Types of Formal Proof, Deductive proofs, Functions, Types of functions	The main objective is that basics of set theory and functions	Students will able to know importance of set theory	T:1:1.1:1.2 R1:1.1-1.3 R3:0.1-0.2

2	**02.12.19(1)**
	04.12.19(1)
		Proofing Techniques, proof by Contra positive, Proof by contradiction, Counter examples,	Its objective is that introduction of proofing techniques	Students will able to know the proofing techniques	T:1:1.3:1.4
					R3:0.3-0.4

3	**05.12.19(2)**	Fundamentals of Automata Theory:- Alphabets, Strings, Languages, Problems, Grammar formalism, Chomsky Hierarchy	Basic fundamentals of Automata theory	Students comes to know the fundamentals of Automata theory	T:1:1.5 R6:1.1-1.2 R6:4.2

4	**09.12.19(1)**
	11.12.19(1)
		Finite Automata: An Informal picture of Finite Automata, Deterministic Finite Automata (DFA), Non Deterministic Finite Automata (NFA)	Introduction of DFA and NFA	Students will able to know about NFA and DFA	T:2:2.1:2.3
					R1:2.1-2.6
					R3:1.1-1.2
					R4:3.1-3.7
					R6:1.3-1.5

5	**12.12.19(2)** **.**	Finite Automata with Epsilon transitions (ϵNFA	Objective is introduction of	Students comes to know	T:2:2.5 R4:3.8

	16.12.19(1)	or NFA- ϵ), Finite Automata with output	automata with epsilon transitions	the automata with epsilon transitions	R6:1.7-1.10 R6:2.1

6	**18.12.19(1)**
	19.12.19(2)
		Conversion of one machine to another, Minimization of Finite Automata, Myhill-Nerode Theorem.	Conversion of DFA to NFA and NFA to DFA	Students comes to know the conversion procedure of NFA to DFA	T:2:2.5
					R4:3.9
					R6:2.3-2.5

UNIT - II					

7	**23.12.19(1)**
	26.12.19(2)
		Regular Languages: Regular Expressions (RE), Finite Automata and Regular Expressions, Applications of Regular Expressions	Introduction to regular languages and their applications	Students will able to know about regular languages	T:3:3.1:3.3
					R1:3.1-3.3
					R4:5.1-5.2
					R6:2.6-2.7

8	**30.12.19(1)**
	01.01.20(1)
	02.01.20(2)
	06.01.20(1)
		Algebraic laws for Regular Expressions, The Arden's
		Theorem, Using Arden's theorem to construct RE from FA, Pumping Lemma for RLs, Applications of Pumping Lemma, Equivalence of Two FAs, Equivalence of Two REs
			Objective is that to learn Arden's, applications of pumping lemma, equivalence of two finite automata's	Students comes to know the applications of pumping lemma	T:3:3.4
					T:4:4.1:4.2
					R4:5.3-5.6
					R6:3.1-3.6

9	**08.01.20(1)**
	10.01.20(2)
	13.01.20(1)
	15.01.20(1)
		Construction of Regular Grammar from RE, Constructing FA from Regular Grammar, Closure properties of RLs, Decision problem's of RLS, Applications of REs and FAs	Construction of regular grammar form FA and closure properties of regular languages	Students will able to learn closure properties of regular languages	T:4:4.3:4.4
					R6:4.3-4.4

UNIT - III					

10	**16.01.20(2)**
	22.01.20(1)
		Context Free Grammars and Languages: Definition of Context Free Grammars (CFG), Derivations and Parse trees	Objective is learn about CFG's and Parse trees	Students will able to learn about CFG's and Parse trees	T:5:5.1:5.3
					R1:4.1-4.5
					R3:2.1
					R4:6.1-6.2

11	**23.01.20(2)**
	27.01.20(1)
	29.01.20(1)
		Ambiguity in CFGs, Removing ambiguity, Left recursion and Left factoring	Procedure to remove ambiguity from CFG's and to remove left recursion and left factoring	Students comes to know the procedure to remove ambiguity from CFG's and to remove left recursion and left factoring	T:5:5.4
					R3:2.1
					R4:6.3
					R6:4.5-4.11

12	**30.01.20(2)**
	03.02.20(1)
	05.02.20(1)
		Simplification of CFGs, Normal Forms, Linear grammars, Closure properties for CFLs	It's objective is to learn normal forms and closure properties of CFL's	Students will able to learn normal forms and closure properties of CFL's	T:7:7.1:7.3
					R4:6.3-6.4

13	**06.02.20(2)**
	10.02.20(1)
	12.02.20(1)
		Pumping Lemma for CFLs, Decision problems for CFLs, CFG and Regular Language	Objective is learn about pumping lemma for CFL's	Students will able to learn about pumping lemma for CFL's	T:7:7.4
					R4:6.5-6.6

UNIT - IV					

14	**13.02.20(2)**
	17.02.20(1)
		Push Down Automata (PDA): Informal introduction, The Formal Definition	The main objective of this is introduction of push down automata	Students comes to know about PDA	T:6:6.1
					R1:5.1-5.2
					R3:2.2
					R4:7.1-7.2
					R6:5.1-5.2

15	**19.02.20(1)**
	20.02.20(2)
	24.02.20(1)
		Types of PDA, Graphical notation, Instantaneous description, The Languages of a PDA	Learn about different types of PDA's	Students will able to learn types of Push Down Automata's	T:6:6.2
					R4:7.3
					R6:5.3-5.5

16	**26.02.20(1)**
	27.02.20(2)
	02.03.20(1)
		Equivalence of PDAs and CFGs, Deterministic Push Down Automata	Its objective is to know about Deterministic Push Down Automata	Students will able to learn equivalence of PDAs and CFGs, Two stack PDA	T:6:6.3:6.4
					R1:5.3-5.4
					R4:7.4
					R6:5.6-5.9

17	**04.03.20(1)**	Two Stack PDA	Its objective is to know about Deterministic Push Down Automata	Students will able to learn equivalence of PDAs and CFGs, Two stack PDA	T:6:6.3:6.4 R1:5.3-5.4 R4:7.4 R6:5.6-5.9
UNIT – V					

18	**05.03.20(2)**
	09.03.20(1)
		Turing Machines and Undecidability: Basics of Turing Machine (TM), Transitional Representation of TMs, Instantaneous description, Non Deterministic TM	Introduction to Turing Machines and non deterministic TM	Students comes to know about Turing Machine and Transitional representation of TM	T:8:8.1:8.2
					R1:7.1-7.2
					R3:3.1
					R4:9.1-9.2
					R6:7.1-7.6

19	**11.03.20(1)**
	12.03.20(2)
		Conversion of Regular Expression to TM, Two stack PDA and TM, Variations of the TM	The main objective is conversion of Regular expression to TM, Variation of the TM	Students will able to learn how to convert Regular expression to TM	T:8:8.3
					R1:7.3-7.7
					R3:3.2
					R4:9.6-9.7
					R6:7.7-7.10

20	**16.03.20(1)**
	18.03.20(1)
		TM as an integer function, Universal TM, Linear Bounded Automata, TM	The objective is know about Universal TM, Linear bounded automata and	Students will able to learn about Universal TM,	T:8:8.4
					R4:9.8-9.10
					R6:7.11-7.12

		Languages, Unrestricted grammar	unrestricted grammar	Linear bounded automata and unrestricted grammar	
21	**20.03.20(2)**	Properties of Recursive and Recursively enumerable languages, Undecidability, Reducibility	Know about the properties of Recursive and recursively enumerable languages	Students comes to learn about properties of Recursive and recursively enumerable languages, Undecidability, and reducibility	T:9:9.1:9.2 R1:8.1-8.4 R3:4.1-4.2 R3:5.1-5.3 R4:10.1-10.3

22	**23.03.20(1)**
	25.03.20(1)
		Undeciadable problems about TMs, Post's Correspondence Problem(PCP), Modified PCP	The main objective is problems about TMs and Post's correspondence problem	Students will able to learn problems about TMs	T:9:9.3:9.4
					R1:9.1-9.5
					R4:10.4-10.6
					R6:8.2-8.6

IX. Mapping course outcomes leading to the achievement of the program outcomes:

Course Outcomes	**Program Outcomes**												
	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**
1	H	S	S		H								
2	H		H		H								

S = Supportive **H = Highly Related**

Justification of Course syllabus covering Course Outcomes:

By covering the syllabus a student can understand how to apply computation methods to problems from different fields and be guided by the results in searching for computational solutions to the problems.

Justification of CO's –PO's Mapping Table:

1. By mapping CO-1 to the PO's A, B, C, and E, which are related to the course CO1: Student can learn basic methods and conclusions of the Theory of Computation.
2. By mapping CO-2 to the PO's A, C, and E, which are related to the course CO2: students learn how to apply these computation methods to problems from different fields and be guided by the results in searching for computational solutions to the problems.