FORMAL LANGUAGE & AUTOMATA THEORY

PCC – CS 403

2020-21

COURSE FILE ON FORMAL LANGUAGE & AUTOMATA THEORY PCC-CS403

Siliguri Institute of Technology

Course File

Course Title/Code: Formal Language & Automata Theory/ PCC CS 403

Department: - CSE (A & B) & IT, Semester: - 2nd Year: - 2nd Group: - All

Name of the Faculty: **Mr. Mithun Roy**
E-mail : mathmithunroy@gmail.com

Class Schedule:

Dept./Day	Monday	Tuesday	Friday
CSE(A)	10:00 AM – 10:50 AM	2:10 PM – 3:00 PM	10:50 AM – 11:40 AM
CSE(B)	10:00 AM – 10:50 AM	2:10 PM – 3:00 PM	10:50 AM – 11:40 AM
IT	2:10 PM – 3:00 PM	10:00 AM – 10:50 AM	11:40 AM – 12:30 PM

Hours of Meeting Students: - Any day (between 4:30 PM to 5:30 PM) (if required)

i) Course Objective:

- Be able to construct finite state machines and the equivalent regular expressions.
- Be able to prove the equivalence of languages described by finite state machines and regular expressions.
- Be able to construct pushdown automata and the equivalent context-free grammars. And be able to prove the equivalence of languages described by pushdown automata and context-free grammars.
- Be able to construct Turing machines and Post machines. Be able to prove the equivalence of languages described by Turing machines and Post machines.

ii) Course Outcomes:

After completion of this course the students are expected to be able to demonstrate following Knowledge, skills and attitudes

1

a) The Students will be able to:

Course Outcomes
PCC-CS403.1 Write a formal notation for strings, languages and machines.
PCC-CS403.2 Design finite automata to accept a set of strings of a language.
PCC-CS403.3 For a given language determine whether the given language is regular or not.
PCC-CS403.4 Design context-free grammars to generate strings of context-free language.
PCC-CS403.5 Determine equivalence of languages accepted by Push Down Automata and languages generated by context-free grammars.
PCC-CS403.6 Write the hierarchy of formal languages, grammars and machines.
PCC-CS403.7 Distinguish between computability and non-computability and Decidability and undecidability.

b) Once the student has successfully complete this course, he/she must be able to answer the following questions or perform/demonstrate the following:

1. What do you **understand** by Formal Language / Finite State Machine? (Level 2)
2. What are the **analytic** issues of finite language? (Level 4)
3. **Conversion** between NFA to DFA. (Level 2)
4. **Application** of Arden's Theorem. (Level 3)
5. How to **design** a DFA / NFA? (Level 6)
6. **What** is the difference between deterministic finite automata and non-deterministic finite automata? (Level 1)
7. **Show** that the following language is regular or not **L= {a^p | p is prime}.** (Level 5)
8. **Differentiate** between regular and context free language. (Level 4)

Formal Language & Automata Theory
Code: PCC-CS403
Contact: 3L

Unit	Content	Hrs/Unit	Marks/Unit
1	Introduction: Alphabet, languages and grammars, productions and derivation, Chomsky hierarchy of languages.	6	
2	Regular languages and finite automata: Regular expressions and languages, deterministic finite automata (DFA) and equivalence with regular expressions, nondeterministic finite automata (NFA) and equivalence with DFA, regular grammars and equivalence with finite automata, properties of regular languages, pumping lemma for regular languages, minimization of finite automata)	7	
3	Context-free languages and pushdown automata: Context-free grammars (CFG) and languages (CFL), Chomsky and Greibach normal forms,	6	
	nondeterministic pushdown automata (PDA) and equivalence with CFG, parse trees, ambiguity in CFG, pumping lemma for context-free languages, deterministic push down automata, closure properties of CFLs.		
4.	Context-sensitive languages: Context-sensitive grammars (CSG) and languages, linear bounded automata and equivalence with CSG.	6	
5	Turing machines: The basic model for Turing machines (TM), Turing recognizable(recursively enumerable) and Turing-decidable (recursive) languages and their closure properties, variants of Turing machines, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMsas enumerators	6	
6	Undecidability: Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice s theorem, undecidable problems about languages	6	

c) Topic/Unit/Chapter Layout

Topic/Unit/Chapter	Lecture Hours
Unit - I	4
Unit -II	4
Unit -III	5
Unit - IV	7
Unit - V	11
Unit - VI	2
Unit - VII	5
Total	38 Hrs.

d) Textbooks:

1) **Introduction to Automata Theory Language and Computation, Hopcroft H.E. and Ullman J. D., Pearson education.**
2) Theory of Computer Science, Automata Languages and computation", Mishra and Chandrashekaran, 2nd edition, PHI.
3) Formal Languages and Automata Theory, C.K.Nagpal, Oxford

e) Reference Books:

1) Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley
2) Introduction to languages and the Theory of Computation, John C Martin, TMH
3) Elements of Theory of Computation, Lewis H.P. & Papadimitrou C.H. Pearson, PHI.

f) Evaluation Scheme:

Evaluation Criteria		Marks
First Internal Exam	Average of First & Second Internal Exam	15
Second Internal Exam		15
Quiz/ Assignments (2.5 * 4)		10
Attendance		5
University Exam		70
Total		100

g) Lesson Plan

Week	SN	Lectures	Assignment
1	1	**Introduction**	
	2	Symbol, Alphabet, String.	
	3	Set, Different Operations with examples.	
	4	Grammars with examples.	
2	5	**Deterministic finite automaton (DFA) with examples.**	
	6	Different examples on DFA.	
	7	Non-deterministic finite (NFA) automaton. Difference between DFA and NFA. Examples.	
	8	Transition diagrams and Language recognizers. Examples on some DFA.	
3	9	**Minimization of DFA with some examples.**	
	10	Finite Automata: NFA with € transitions - Significance, acceptance of languages.	
	11	Equivalence between NFA with and without € transitions with some examples.	
	12	NFA to DFA conversion (Thomson Construction) with some worked out examples.	**Assignment - I**
	13	**Revision on Finite Automata**	
4	14	**Regular Languages: Regular sets**.	
	15	Regular expressions, identity rules. Arden's theorem state and prove.	
	16	Different Examples.	
	17	Constructing finite Automata for a given regular expressions, Regular string accepted by NFA/DFA.	
	18	Different Examples.	
	19	Pumping lemma of regular sets. With examples.	
	20	Closure properties of regular sets. (Prove)	
5	21	**Grammar Formalism: Regular grammars-right linear and left linear grammars.**	
	22	Equivalence between regular linear grammar and FA.	
	23	Inter conversion, Context free grammar.	
	24	Derivation trees, sentential forms. Right most and leftmost derivation of strings. (Concept only).	**Assignment - II**
	25	Context Free Grammars, Ambiguity in context free grammars.	
	26	Minimization of Context Free Grammars.	
	27	Chomsky Normal Form [CNF] with examples.	
	28	Greibach Normal Form [GNF] with examples.	
	29	Pumping Lemma for Context Free Languages.	

	30	Push Down Automata: Push down automata, definition.	
	31	Equivalence of CFL and PDA, inter conversion. (Proofs not required).	
6	32	**Context Sensitive Grammar with examples.**	**Assignment - III**
	33	Linear Bounded Automata with examples.	
7	34	**Turing Machine: Turing Machine, definition, model.**	
	35	Design of TM, Computable functions.	
	36	Church's hypothesis, counter machine.	
	37	Types of Turing machines (proofs not required).	
	38	Universal Turing Machine, Halting problem.	
	39	**Revision on Unit - IV**	**Assignment - IV**
	40	Discuss on University QP	