CSC 2204 Finite Automata Theory and Formal Languages

COURSE INFORMATION SHEET

PROGRAMME: **Computer Science**	DEGREE: **BS**
COURSE: **Finite Automata Theory and Formal Languages**	SEMESTER**: 4** CREDITS:**3**
COURSECODE: **CSC2204**	COURSE TYPE: **CORE**
COURSE AREA/DOMAIN: **Computer Science**	CONTACTHOURS: **3 hours/Week.**
CORRESPONDINGLAB COURSECODE(IFANY): **Nil**	LAB COURSENAME: **Nil**

COURSE OBJECTIVES:

The major objective of this course is to introduce the student to the concepts of theory of computation in computer science. The student should acquire insights into the relationship among formal languages, formal grammars, and automata.

COURSE DESCRIPTION:

Finite State Models: Language definitions preliminaries, Regular expressions/Regular languages, Finite automata (FAs), Transition graphs (TGs), NFAs, Kleene’s theorem, Transducers (automata with output), Pumping lemma and non-regular language Grammars and PDA: CFGs, Derivations, derivation trees and ambiguity, Simplifying CFLs, Normal form grammars and parsing, Decidability, Context sensitive languages, grammars and linear bounded automata (LBA), Chomsky’s hierarchy of grammars Turing Machines Theory: Turing machines, Post machine, Variations on TM, TM encoding, Universal Turing Machine, Defining Computers by TMs.

TEXT/REFERENCE BOOKS:

Introduction to computer theory, 2nd edition, Daniel I. A. Cohen, John Wiley & Sons Inc.
Automata, Computability and Complexity: Theory and Applications, by Elaine Rich, Pearson Prentice Hall, 2011.
An Introduction to Formal Languages and Automata, by Peter Linz, 4th edition, Jones & Bartlett Publishers, 2006.
Theory of Automata, Formal Languages and Computation, by S. P. Eugene, Kavier, 2005, New Age Publishers.

SYLLABUS:

Week	Topics
1.	Introduction to the course, Languages, Definitions, Operations.
2.	Basics of Finite Automata, Deterministic Finite Automata (DFA).
3.	Non-deterministic Finite Automata (NDFA), Finite Automata with Epsilon Transitions.
4.	Equivalence of Deterministic and Nondeterministic Finite Automata, Basics of Regular Expressions, Regular Expressions and Languages.
5.	Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions.
6.	Kleene theorem.
7.	FAs with Output.
8.	Mid Term Paper.
9.	Transducers s Models of Sequential Circuits.
10.	Decision Properties of Regular Languages, Basics of Context-Free Grammars and Context Free Languages, Regular vs. Context-Free grammars.
11.	Parse Trees, Derivation Order and Derivation Trees, Applications of CFGs, Ambiguity in Grammars and Languages.
12.	Basics of Pushdown Automata (PDA), The Language of a PDA, Equivalence of PDAs and CFGs.
13.	Simplification of Context-Free Grammars and Normal Forms, Designing CFG for a Specific language, Properties and Applications of Context-Free Languages.
14.	Turing Machines, Post Machines.
15.	Variations on TMs, Encoding of TMs.

COURSE LEARNING OUTCOMES (CLOs):

CLO	DESCRIPTION	Domain	BT Level*
1	Describe the fundamental concepts of formal language theory and their properties.	C	1
2	Design languages and machines using using different representations such as REs, DFAs, NFAs, CFL, CFGs, PDAs, Turing Machines and TM Variants.	C	2
3	Transform between equivalent notations and machines.	C	3
4	Implement machines.	C	3
*BT=Bloom’s Taxonomy, C=Cognitive domain, P=Psychomotor domain, A=Affective domain.			

MAPPING COURSE LEARNING OUTCOMES (CLOs) ANDPROGRAM LEARNINGOUTCOMES (PLOs):

	PROGRAM LEARNING OUTCOMES (PLOs)										
	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**
CLO.1				X							
CLO.2			X	X							
CLO.3			X	X							
CLO.4			X								

PROGRAM LEATNING OUTCOMES (PLOs):

PLO	DESCRIPTION
c	The ability to think critically, perform scientific analysis and develop solutions for typical Computer Science problems.
d	In depth knowledge in advanced and evolving areas in Computer Science.