MANIPAL UNIVERSITY JAIPUR

School of Information Technology

Department of Information Technology
Course Hand-Out

Automata Theory & Compiler Design | IT 3202 | 4 Credits | 3 1 0 4

Session: **January 2024 – May 2024** | Faculty: **Dr. Shalini Puri** (Sec A)**, Dr. Prakash Chandra Sharma** (Sec B)**, Ms. Shikha Chaudhary** (Sec C), **Dr. Shally Vats** (Sec D), **Mr. Vijay Prakash Sharma** (Sec E)**, Mr. Rohit Kumar Gupta** (Sec F)

A. Introduction: The objective of this course is to make the students familiar with the fundamental area of computer science which will enable the students to focus on abstract models of computation. The course exposes students to the computability theory. The goal is to allow them to answer fundamental questions about problems, such as whether they can or cannot be computed. The objective is also to make the students familiar with the different types of automata with its designing approach as well as various phases of compilation process of any source code. Throughout the semester they will learn about lexical analysis, different types of parsing techniques, code generation and optimization. The goal is to allow them to answer in detail about how a compiler works and how it reports to its user's various types of errors.

B. Course Outcomes: At the end of the course, students will be able to

[3202.1]. **Identify** the fundamental concepts used in automata theory and formal languages.

[3202.2]. **Explain** the construction of finite automata and regular expressions.

[3202.3]. **Interpret** properties of context free language and **construct** push down automata.

[3202.4]. **Illustrate** the basic properties of Turing machines and **compute** the tractability and decidability with Turing machine.

[3202.5]. **Inspect** the performance of each phase of a compiler and **construct** parsers for designing compiler.

[3202.6]. **Apply** Automata theory and compiler construction concepts in different engineering applications.

C. Vision, Mission, Program Educational Objectives

Vision:

To achieve global excellence in higher education, research, and human development by adapting innovations in IT domain.

Mission:

1. To become the most inspirational department among the students and engineering aspirants who want to pursue their career in the field of Information Technology.
2. To foster academic, research, and professional excellence within the domain of Information Technology.
3. To transform young minds into competent IT professionals imbibing strong moral values.

Program Educational Objectives (PEOs)

PEO1: Confidently demonstrate the required logical and technical problem-solving skills and technical competence to work in the field of IT and IT enabled services.

PEO2: Specify, analyse, design, and implement prototypes and working solutions for a wide range of real-life problems within the specified constraints.

PEO3: Pursue higher studies, grow in the direction of research and development, pave a way forward in the field of entrepreneurship while demonstrating remarkable professionalism, confidence, and a sense of responsibility.

PEO4: Engage in life-long learning, communicate effectively, and exhibit leadership skills.

PEO5: Demonstrate sensitivity towards ethical and environmental issues, and social responsibilities.

D. PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES

[PO.1]. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

[PO.2]. **Problem analysis:** Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences

[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

[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.

[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.

[PO.6]. **The Engineers and Society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

[PO.7]. **Environment and Sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

[PO.8]. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practices.

[PO.9]. **Individual and Teamwork:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

[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:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

[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 change.

Program Specific Objectives (PSOs)

[PSO.1] To apply creativity in support of the design, simulation, implementation and inference of existing and advanced technologies.

[PSO.2] To participate & succeed in IT oriented jobs/competitive examinations that offer inspiring & gratifying careers.

[PSO.3] To recognize the importance of professional developments by pursuing postgraduate studies and positions.

E. Assessment Plan:

Criteria	Description	Maximum Marks
Internal Assessment (Summative)	MTE	30
	Quiz – 3 (Average of best 2)	20
	Two Regular Assignments or One Research Assignment	10
End Term Exam (Summative)	End Term Exam (Closed Book)	40
	Total	**100**
Attendance (Formative)	A minimum of 75% Attendance is required to be maintained by a student to be qualified for taking up the End Semester examination. The allowance of 25% includes all types of leaves including medical leaves.	
Make up Assignments (Formative)	As per department's directives	
Homework/ Home Assignment/ Activity Assignment (Formative)	As per course instructor's wish	

F. Syllabus

Introduction: Automata Theory, Mathematical Preliminaries and Notations used in Automata Theory, Basics of Set Theory, Function and Relations; **Finite Automata:** Deterministic and Non-Deterministic Finite Automata, Mealy and Moore Machines; **Regular Languages and Regular Grammars:** Regular Sets, Chomsky Hierarchy, Regular Expressions, Regular Grammar, Pumping Lemma for Regular Languages; **Context Free Languages and Grammars:** Ambiguity, Methods for Transforming Grammars; **Push Down Automata:** Context Free Languages, Non-Deterministic Push Down Automata and Deterministic Push Down Automata, Design of NPDA and DPDA; **Introduction to Turing Machine:** Basics of Turing Machine, Variations of Turing Machines; **Introduction to Compiler Design:** Structure of a Compiler, Lexical Analysis, Recognition of Tokens; Introduction to LR Parsing: Simple LR, More Powerful LR Parsers Generators; Semantic Analysis: Syntax Directed Translations; Basics of Storage Organization.

G. Textbooks

T1. An Introduction to Formal Languages and Automata – Peter Linz, Jones and Bartlett Publishers, 6th Edition, 2016, ISBN: 9384323217.

T2. Compilers: Principles, Techniques and Tools – Alfred V Aho, Monica S Lam, Ravi Sethi, Jeffrey D Ullman, Pearson Education, 2nd Edition, 2007, ISBN: 9332518661.

H. Reference Books

R1. Introduction to Automata Theory, Languages and Computations – J. E. Hopcroft, R. Motwani, J. D. Ullman, Pearson Education India, 3rd Edition, 2008, ISBN: 9788131720479.

R2. Introduction to Languages and the Theory of Computation – John C Martin, Tata McGraw Hill, 4th Edition, 2010, ISBN: 0073191469.

R3. Theory of Computer Science: Automata, Languages and Computation – K.L.P. Mishra, N. Chandrashekharan, PHI, 3rd Edition, 2006, ISBN: 9788120329683

I. Lecture Plan:

Lec No	Topics	Session Outcome	Mode of Delivery	Corresponding CO	Mode of Assessing the Outcome
1	Course Hand-out Briefing, Introduction of Automata Theory, Mathematical Preliminaries and Notations used in Automata Theory: Symbol, Alphabet, String: Length, Concatenation, Reverse of string, Null string, Substring, Prefix and Suffix of String, Kleene Closure	Understand the basics of automata theory and basic terminologies used in formal languages and grammar	Lecture	3202.1	In Class Discussion (Formative)
2-3	Language, Grammar, and Chomsky Hierarchy of Grammar THEORY OF CHOMSKY	Understand and compare different types of languages and their grammars according to Chomsky Hierarchy	Lecture	3202.1	In Class Quiz, Home Assignment, Mid Term, End Term
4	**Finite Automata,** Formal definition & representation of Deterministic Finite Automata (DFA), Acceptance of a string by DFA	Understand the mathematical definition & representation of DFA	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term
5-9	Deterministic Finite Automata (DFA) Designing MORE QUESTIONS NEEDED Substring question	Illustrate the design principles of DFA and understand DFA designing for different design pattern.	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term
10	Minimization of DFA, Equivalence of two DFAs	Illustrate the minimization algorithms of DFA	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term
11	Formal definition & representation of Non-Deterministic Finite Automata (NDFA), Acceptance by NDFA SUDHKAR ATCHALA	Understand the mathematical definition and representation of NDFA	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term
12-13	Non-Deterministic Finite Automata (NDFA) Designing MORE QUESTIONS NEEDED	Illustrate the design principles of NDFA	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term
14	Equivalence of DFA and NFA, Conversion of NFA to DFA TUPLES OF EACH TOPICS SunSin video	Explain the basic difference between DFA and NDFA and realising the importance of NDFA	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term
15-16	Epsilon NFA, Epsilon NFA to NFA conversion Anita R Practice questions	Understand the application of NFA	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term
17-19	Design of FA with output: Construction of Mealy and Moore Machine and their equivalence GateHub Practice Questions Knowledge gate	Understand the design principles of Mealy and Moore machines Rohit tutorials	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, **Quiz-I** Home Assignment, Mid Term, End Term
20-21	**Regular Expression & Regular Languages:** Identities of Regular Expression, Regular Language to Regular Expression and Regular Expression to Regular Language Perfect computer engineering	Understand the properties of regular languages and regular expression.	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term

22-23	Equivalence of Finite Automata and Regular Expressions (FA to RE (Arden's Theorem), and RE to FA), Equivalence of Finite Automata and Regular Grammar (FA to RG and RG to FA) neso Sundeep Saradhi Kanthety	Understand the equivalence between FA and RE, FA and regular grammar. Easy computer engineering	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term
24-25	Identifying non-regular languages using Pumping Lemma Easy Theory 2 hour vid (for practice) Sundeep saradhi	Apply pumping lemma to identify non-regular languages.	Lecture, Practice questions	3202.1 3202.2	In Class Quiz, Home Assignment, Mid Term, End Term
26-28	**Context Free Grammar (CFG) and Context Free Language:** Construction of CFG, Derivation trees, Ambiguity in CFGs, Left Recursion, Left Factoring Anita R 21 min video for construction	Understand the properties of CFLs.	Lecture, Practice questions	3202.1 3202.3	In Class Quiz, Home Assignment, Mid Term, End Term
29-31	Simplification of CFGs, Normal Forms for CFGs: CNF, Pumping Lemma for CFGs Knowledge gate	Understand the normal forms of CFGs. Apply pumping lemma to identify non-CFLs.	Lecture, Practice questions	3202.3	In Class Quiz, Home Assignment, Mid Term, End Term
32-33	**Pushdown Automata (PDA):** Definition, Instantaneous Description, Language of PDA, Acceptance by final state and empty state, PDA Designing, Types of PDA: NPDA and DPDA,	Design pushdown automata and understanding the properties of its variants. Design PDA questions to be done	Lecture, Practice questions	3202.1 3202.3	In Class Quiz, **Quiz-2** Home Assignment, **Mid Term,** End Term
34-35	Equivalence of PDA and CFGs: PDA to CFGs, CFGs to PDA, Two Stack PDA	Understand the equivalence between PDA and CFGs.	Lecture, Practice questions	3202.3	In Class Quiz, Home Assignment, End Term
36-38	**Introduction to Turing machines**: basic model, instantaneous description, design of Turing machines, Language acceptance by TM, properties of recursive and recursively enumerable languages.	Design of Turing machines, application of Turing machines and understanding the properties of recursive and recursively enumerable languages. Manish Tiwari DR Pushpa choudhary	Lecture, Practice questions	3202.4	In Class Quiz, Home Assignment, End Term
39-40	**Introduction to compilers**: structure and phases of a compiler, lexical analysis phase of a compiler	Summarize the structure and roles of each phase of a compiler. Illustrate the role of a lexical analyser.	Flipped Classroom	3202.5	In Class Quiz, Home Assignment, End Term
41-43	Introduction to syntax analysis and parsing, computing FIRST and FOLLOW, LL(1) Handle Pruning	Understand the working principles of Top-down parsers. Question practice of LL1	Lecture, Practice questions	3202.5 3202.6	In Class Quiz, **Quiz-3** Home Assignment, End Term
44-49	LR(0), SLR(1), CLR(1) and LALR(1) parsing	Develop and compare the working principles of different types of bottom-up parsers.	Lecture, Practice questions	3202.5 3202.6	In Class Quiz, Home Assignment, End Term
50-52	Introduction to Syntax directed translations, S-attributes, I-attributes and L-attributes, Building annotated trees	Understand and develop syntax directed translation scheme and annotated trees	Lecture, Practice questions	3202.5 3202.6	In Class Quiz, Home Assignment, End Term
53-54	Type Checking: Rules for Type Checking, Storage Organization.	Understanding concepts of type checking and its various rules. Summarize the storage organization.	Flipped Classroom	3202.5 3202.6	In Class Quiz, Home Assignment, End Term

J. Course Articulation Matrix: (Mapping of COs with POs)

CO	STATEMENT	CORRELATION WITH PROGRAM OUTCOMES												CORRELATION WITH PROGRAM SPECIFIC OUTCOMES		
		PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
[IT3202.1]	**Identify** the fundamental concepts used in automata theory and formal languages.	3	3	1	1			1	1				2	1	1	
[IT3202.2]	**Explain** the construction of finite automata and regular expressions.	3	3	3	1	1			1	1	1		1	2	1	1
[IT3202.3]	**Interpret** properties of context free language and **construct** push down automata.	2	3	3	1				1	1			1	2		
[IT3202.4]	**Illustrate** the basic properties of Turing machines and computing the tractability and decidability with Turing machine.	2	3	3	1		1		1	1	1		1	2	1	1
[IT3202.5]	**Inspect** the performance of each phase of a compiler and **construct** parsers for designing compiler.	2	3	3	1				1	1	1		1	1	1	
[IT3202.6]	**Apply** Automata theory concepts in engineering applications like designing of compilers, etc.	1	2	3	1				1		1	1	1	1	2	1

1-Low Correlation; 2- Moderate Correlation; 3- Substantial Correlation