**COURSE DESCRIPTION FORM**

**INSTITUTION** FAST - National University of Computers and Emerging Sciences

BS - Computer Science

**PROGRAM (S) TO BE**

**EVALUATED**

1. **Course Description**

(Fill out the following table for each course in your computer science curriculum. A filled out form should not be more than 2-3 pages.)

| **Course Title** | Theory of Computation / Automata | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Course Code** | CS3005 | | **Credit Hours** | | | 3 + 0 |
| **Prerequisites by Course(s)** | Discrete Structures | | **Semester** | | | Spring 2022 |
| **Assessment Instruments**  (with tentative weights) | **Semester Work 20%** (at least 3 assignments and 3 quizzes)  **Midterm 30%** (2 Mid semester exam – Week 6 and Week 11)  **Final 50%** (Comprehensive end of semester exam) | | | | | |
| **Course Coordinator** | Muhammad Shahzad | | | | | |
| **Office Hours** | Details displayed outside my Basement (Old library- CS Block). | | | | | |
| **Current Catalog 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 | | | | | |
| **Textbook** (or **Laboratory Manual** for Laboratory Courses) | 1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation 2. P. Linz. Introduction to Formal Languages and Automata, 6th edition, 2017 (or 5th or 4th edition), Jones and Barlett 3. Daniel I. A. Cohen, Introduction to Computer Theory | | | | | |
| **Reference Material** | 1. John Martin, Introduction to Languages and the Theory of Computation, Third Edition 2. Michael Sipser, Introduction to Theory of Computation 3. Instructor Notes | | | | | |
| **Course Goals** | | **A.** **Course Learning Outcomes (CLOs)** | | --- | |  | | | **CLO No.** | **Course Learning Outcomes** | **Bloom Taxonomy** | **Tools** | | --- | --- | --- | --- | | **CLO-1** | Explain and manipulate the different concepts in automata theory and formal languages such as formal proofs, automata, regular expressions, Turing machines etc | C2 (Understand) | A1, Q1 | | **CLO-2** | Prove properties of languages, grammars and automata with rigorously formal mathematical methods | C2 (Understand) | A2, Q2 | | **CLO-3** | Design of automata, RE and CFG | C3 (Apply) | Q3,  M1, F1 | | **CLO-4** | Transform between equivalent NFAs, DFAs and Res | C3 (Apply) | M1, F1 | | **CLO-5** | Define Turing machines, PDA machines performing simple tasks | C2 (Understand) | A3,M2,F1  Q3 | | *Tool: A = Assignment, Q = Quiz, M = Midterm, F=Final, CEP = Complex Engineering Problem.* | | | |  | **B. Program Learning Outcomes** | | | | | --- | --- | --- | --- | | **PLO 1** | Computing Knowledge | Apply knowledge of mathematics, natural sciences,  computing fundamentals, and a computing  specialization to the solution of complex computing  problems. | | | **PLO 2** | Problem Analysis | Identify, formulate, research literature, and analyse  complex computing problems, reaching substantiated  conclusions using first principles of mathematics,  natural sciences, and computing sciences. | | | **PLO 3** | Design/Develop Solutions | Design solutions for complex computing problems and design systems, components, and processes that meet specified needs with appropriate consideration for public health and safety, cultural,  societal, and environmental considerations. | | | **PLO 4** | Investigation & Experimentation | Conduct investigation of complex computing problems using research based knowledge and research based methods | | | **PLO 5** | Modern Tool Usage | Create, select, and apply appropriate techniques,  resources and modern computing tools, including  prediction and modeling for complex computing  problems. | | | **PLO 6** | Society Responsibility | Apply reasoning informed by contextual knowledge to  assess societal, health, safety, legal, and cultural  issues relevant to context of complex computing  problems. | | | **PLO 7** | Environment and Sustainability | Understand and evaluate sustainability and impact of  professional computing work in the solution of  complex computing problems | | | **PLO 8** | Ethics | Apply ethical principles and commit to professional  ethics and responsibilities and norms of computing  practice. | | | **PLO 9** | Individual and Team Work | Function effectively as an individual, and as a member  or leader in diverse teams and in multi-disciplinary  settings. | | | **PLO 10** | Communication | Communicate effectively on complex computing  activities with the computing community and with s  ociety at large. | | | **PLO 11** | Project Mgmnt and Finance | Demonstrate knowledge and understanding of management principles and economic decision making and apply these to one's  own work as a member or a team. | | | **PLO 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 changes. | |  | **C. Relation between CLOs and PLOs**  (CLO: Course Learning Outcome, PLOs: Program Learning Outcomes) | | | | | | | | | | | | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | | **PLOs** | | | | | | | | | | | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | | **CLOs** | 1 |  |  |  |  |  |  |  |  |  |  | | 2 |  |  |  |  |  |  |  |  |  |  | | 3 |  |  |  |  |  |  |  |  |  |  | | 4 |  |  |  |  |  |  |  |  |  |  | | 5 |  |  |  |  |  |  |  |  |  |  | | | | | | | |
| **Topics Covered in the Course** (Tentative plan)  Please note:   1. Students are expected to go through the suggested reading topics from at least one reference book and internet, before & after each class. 2. Representative topic of suggested chapters are given week-wise (on right) as **Chapter [Ullman] / [Cohen]**. Please read the full chapter(s). 3. Apart from graded assignments, students are expected to discuss and solve exercises at the end of each chapter. | | **Week** | **Lecture** | **Topics** | **CLO** | **Chapters** | **Assessments** | | --- | --- | --- | --- | --- | --- | | **1.** | **1.** | Discussion on Course Outline, Introduction to Finite Automata | CLO-1 | [HMU]:1.1  [Linz]: 1 |  | | **2.** | What does automata means? Introduction to Languages,  Alphabets, Strings | CLO-1 | [HMU]: 1.5  [DC]: 1 |  | | **3.** | Kleene Star Closure, Regular Expression (RE) | CLO-3 | [DC]: 4  [HMU]: 3  [JM]: 3  [Linz]: 3 |  | |  |  |  |  |  |  | | **2.** | **1.** | Equivalent RE, Finite Automaton (FAs), Equivalent FAs | CLO-3 | [HMU]: 2  [JM]: 2  [MS]: 1 |  | | **2.** | FA corresponding to finite languages, Transition Graph | CLO-3 | [Linz]: 2.1  [DC]: 6 |  | | **3.** | Continued |  |  |  | |  |  |  |  |  |  | | **3.** | **1.** | Examples of TGs: accepting all strings, accepting none, starting with b, not ending in b, containing aa, containing aa or bb. | CLO-4 | [Linz]: 2  [DC]: 6 | ***Assignment 1***  ***Friday Release Week 3*** | | **2.** | Generalized Transition Graph | CLO-4 | [DC]: 6 | | **3.** | Language accepted by NFA, Recursive definition of NFA | CLO-4 | [Linz]: 2.2  [HMU]:2.3 | |  |  |  |  |  |  | | **4.** | **1.** | Basis Clause and Inductive Clause of NFA | CLO-4, 2 | [Instructor  Notes] | ***Quiz no 1*** | | **2.** | NFA with Λ Transitions, Language accepted by NFA- Λ, Definition of Λ-Closure, Basis Clause and Inductive Clause of NFA- Λ | CLO-4 | [Instructor  Notes] |  | | **3.** | Conversion of NFA- Λ to equivalent NFA | CLO-4 | [Instructor  Notes] |  | |  |  |  |  |  |  | | **5.** | **1.** | Conversion of NFA to equivalent DFA | CLO-4 | [Instructor  Notes] | ***Assignment 1 submission Monday Week 5*** | | **2.** | Equivalence of DFAs, NFAs and NFA- Λ | CLO-4 | [Instructor  Notes] | | **3.** | Kleene’s Theorem Part-1 & Part-2 | CLO-4 | [JM]: 3.4,  3.5  [DC]: 7 | |  |  |  |  |  |  | | **6.** | **1.** |  |  |  |  | | **2.** | **Mid-I Examination** |  |  |  | | **3.** |  |  |  |  | |  |  |  |  |  |  | | **7.** | **1.** | Complement of Regular Language and Complement of DFA,  Intersection of Regular Languages | CLO-3 | [Instructor  Notes] |  | | **2.** | Properties of RLs | CLO-2,3 | [Linz]: 4  [HMU]: 4 |  | | **3.** | Pumping Lemma | CLO-3 | [HMU]: 4.1  [JM]: 2.4 |  | |  |  |  |  |  |  | | **8.** | **1.** | Minimization of DFA | CLO-4 | [Instructor  Notes]  [HMU]: 4.4 | ***Quiz no 2***  ***Assignment 2***  ***Friday Release Week 8*** | | **2.** | Mealy & Moore Machines | CLO-4 | [DC]: 9 |  | | **3.** | Conversion between Mealy & Moore Machines | CLO-4 | [Instructor  Notes] |  | |  |  |  |  |  |  | | **9.** | **1.** | Regular Grammars, Linear Grammar, Context-free Languages (CFL), Context-free grammars (CFG). | CLO-2,3 | [DC]: 13  [MS]: 2 |  | | **2.** | Parse Trees, Derivations and ambiguity and Chomsky-normal-form grammars (CNF), Null Production | CLO-3 | [DC]: 20  [Linz]: 6.2 |  | | **3.** | Trees, Polish Notations, Total Language Tree | CLO-3 | [JM]: 4.4 |  | |  |  |  |  |  |  | | **10.** | **1.** | Push down automata (PDA) | CLO-5 | [JM]: 5  [DC]: 17  [MS]: 2.2 | ***Assignment 2 2 submission Monday Week***  ***10*** | | **2.** | Deterministic PDA, Pumping Lemma for CFG | CLO-5 |  |  | | **3.** | NPDA and CFG Equivalence | CLO-5 |  |  | |  |  |  |  |  |  | | **11.** | **1.** |  |  |  |  | | **2.** | **Mid-II Examination** |  |  |  | | **3.** |  |  |  |  | |  |  |  |  |  |  | | **12.** | **1.** | Turing Machines (TM) Intro & Formalities | CLO-5 | [MS]: 3.1  [Linz]: 9 |  | | **2.** | Designing TM as Acceptors/Transducers | CLO-5 | [Linz]: 9 |  | | **3.** | Turing’s Thesis, Turing Machine Variations | CLO-5 | [MS]: 3.2  [DC]: 27 |  | |  |  |  |  |  |  | | **13.** | **1.** | Universal Turing Machine Decidability | CLO-5 | [JM]: 7.8  [DC]: 27 | ***Assignment 3***  ***Friday Release Week 13*** | | **2.** | Recursive vs. recursively enumerable | CLO-5 | [JM]: 8  [DC]: 28 |  | | **3.** | Continued |  |  | |  |  |  |  |  |  | | **14.** | **1.** | Decidable Problem and Undecidable Problem, | CLO-4 | [JM]: 9  [HMU]: 9  [MS]: 4 | ***Quiz no 3*** | | **2.** | Continued |  |  |  | | **3.** | Continued |  |  |  | |  |  |  |  |  |  | | **15.** | **1.** | Reducibility, Reduction problems | CLO-5 | [MS]: 5.1 | ***Assignment 3 submission Monday Week***  ***15*** | | **2.** | The Chomsky Hierarchy | CLO-1,2,3 | [JM]: 8.3 | | **3.** | Continued |  |  | |  |  |  |  |  |  | | **16.** | **1.** | **Revision** |  |  |  | | | | | | |
| **Class Time Spent on**  (in credit hours) | **Theory** | **Problem Analysis** | | **Solution Design** | **Social and Ethical Issues** | |
| 5 | 15 | | 28 | 0 | |
| **Oral and Written Communications** | Every student is required to submit at least 3 assignments and 3 quizzes with no oral presentations. | | | | | |

**Instructor Name \_\_\_\_Muhammad Shahzad\_\_\_\_\_\_\_\_\_**

**Instructor Signature \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Date \_\_\_\_16-01-2023\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**