# Fall Semester L-2, T-II

| COURSE INFORMATION | | | |
|---|---|---|---|
| Course Code<br>Course Title | : CSE 217<br>: Theory of Computation | Lecture Contact Hours<br>Credit Hours | : 3.00<br>: 3.00 |

**PREREQUISITE**

N/A

**CURRICULUM STRUCTURE**

Outcome Based Education (OBE)

SYNOPSIS/RATIONALE

To learn how problems can be efficiently solved on a model of computation using algorithms and the elementary ways in which a computer works.

**OBJECTIVE**

1. Understand the mathematical foundations of computation including automata theory.
2. Have a solid foundation of the theory of formal languages and grammars.
3. Analyse and design finite automata, pushdown automata, Turing machines, formal languages and languages, and grammars.

**LEARNING OUTCOMES & GENERIC SKILLS**

| No. | Course Learning Outcome | Bloom's Taxonomy | CP | CA | KP | Assessment Methods |
|---|---|---|---|---|---|---|
| CO1 | Applying and evaluating the mathematical foundations of computation including mathematical proofs for computation. | C3, C5 | 1 | | 1, 3 | T, F |
| CO2 | Analysing and creating finite automata and regular expressions for regular languages. | C4, C6 | 1, 2 | | 1, 3, 5 | T, F |
| CO3 | Analysing and creating context free grammar and pushdown automata for context free languages. | C4, C6 | 1, 2 | | 1, 5 | T, F |
| CO4 | Understanding and analysing Turing machines and investigating the limits of algorithmic solvability. | C2, C4 | 1 | | 1, 3 | T, F |

(CP- Complex Problems, CA-Complex Activities, KP-Knowledge Profile,T – Test ; F – Final Exam)

**COURSE CONTENT**

Finite automata: deterministic finite automata, nondeterministic finite automata, equivalence and conversion of deterministic and nondeterministic finite automata, Regular languages: regular expressions, nonregular languages, the pumping lemma; Context free languages; Context free grammars, Chomsky normal form, Greibach Normal Form; Pushdown automata; Turing Machines: basic machines, configuration, computing with Turing machines, combining Turing machines; Decidability.

**SKILL MAPPING**

| No. | Course Learning Outcome | PROGRAM OUTCOMES (PO) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| CO1 | Identify the mathematical foundations of computation including mathematical proofs for computation. | | H | | | | | | | | | | |
| CO2 | Design and/or analyse finite automata and regular expressions for regular languages. | | | H | | | | | | | | | |
| CO3 | Design and/or analyse context free grammar and pushdown automata for context free languages. | | | | H | | | | | | | | |
| CO4 | Illustrate Turing machines and investigate the limits of algorithmic solvability. | | | | H | | | | | | | | |

(H – High, M- Medium, L-low)

**TEACHING LEARNING STRATEGY**

| Teaching and Learning Activities | Engagement (hours) |
|---|---|
| Face-to-Face Learning | |
| Lecture | 42 |
| Practical / Tutorial / Studio | - |
| Student-Centred Learning | - |
| Self-Directed Learning | |
| Non-face-to-face learning | 84 |
| Revision | 21 |
| Assessment Preparations | - |
| Formal Assessment | |
| Continuous Assessment | 2 |
| Mid Term Exam | 1 |
| Final Examination | 3 |
| Total | 153 |

**TEACHING METHODOLOGY**

Lectures, class performances, class tests, mid-term exam, final exam.

**COURSE SCHEDULE**

| Week | Lecture | Topics | LECTURER |
|---|---|---|---|
| **1** | Lec 1<br>Lec 2<br>Lec 3 | Automata, Computability, and Complexity, Mathematical Notation and Terminology, Sets, Sequences and Tuples, Functions and Relations, Strings and Languages, Definitions, Theorems and Proofs. | |
| **2** | Lec 4<br>Lec 5<br>Lec 6 | Finite Automata<br>Formal Definition of a Finite Automaton<br>Examples of Finite Automata | **Class Test 1** |
| **3** | Lec 7<br>Lec 8-9 | Formal Definition of Computation<br>Designing Deterministic Finite Automata | |
| **4** | Lec 10<br>Lec 11<br>Lec 12 | The Regular Operations<br>Union operation, Concatenation operation, Star operation, Closure under the Regular Operations | |
| **5** | Lec 13<br>Lec 14<br>Lec 15 | Nondeterminism<br>Equivalence of NFAs and DFAs<br>Closure under the Regular Operations | **Class Test 2** |
| **6** | Lec 16<br>Lec 17<br>Lec 18 | Regular expressions<br>Formal definition of a regular expression | |
| **7** | Lec 19 | Nonregular Languages, | |

| | | | |
|---|---|---|---|
| | Lec 20<br>Lec 21 | The Pumping Lemma for Regular Languages. | |
| **8** | Lec 22<br>Lec 23<br>Lec 24 | Context-Free Languages<br>Context-Free Grammars<br>Formal Definition of CFG | **Mid Term** |
| **9** | Lec 25<br>Lec 26<br>Lec 27 | Examples of CFG, Designing CFG<br>Ambiguity | |
| **10** | Lec 31<br>Lec 32-33 | Chomsky Normal Form I<br>Chomsky Normal Form II | |
| **11** | Lec 28<br>Lec 29<br>Lec 30 | Pushdown Automata<br>Formal Definition of a Pushdown Automaton<br>Examples of Pushdown Automata. | **Class Test 3** |
| **12** | Lec 34<br>Lec 35<br>Lec 36 | Non-context-free languages<br>The pumping lemma for context-free languages and proofs | |
| **13** | Lec 37<br>Lec 38<br>Lec 39 | Turning Machines, Formal Definition of a<br>Turing Machine, Examples of Turing Machines. | |
| **14** | Lec 40-42 | Decidability, decidable languages,<br>Decidable problems concerning Regular languages | |

## ASSESSMENT STRATEGY

| Components | | Grading | CO | Blooms Taxonomy |
|---|---|---|---|---|
| Continuous Assessment (40%) | Test 1-3 | 20% | CO1 | C3, C5 |
| | | | CO2 | C4, C6 |
| | | | CO3 | C4, C6 |
| | Class Performance | 5% | CO3 | C4, C6 |
| | Mid term | 15% | CO2 | C4, C6 |
| | | | CO3 | C4, C6 |
| Final Exam | | 60% | CO1 | C3, C5 |
| | | | CO2 | C4, C6 |
| | | | CO3 | C4, C6 |
| | | | CO4 | C2, C4 |
| Total Marks | | 100% | | |

**(CO = Course Outcome, C = Cognitive Domain, P = Psychomotor Domain, A = Affective Domain)**

## REFERENCE BOOKS


1. Introduction to the Theory of Computation, 3rd edition, 2012- Michael Sipser.
2. Introduction to Automata Theory, Languages, and Computation. Addison-Wesley Longman Publishing Co., Inc., 3rd ed., 2006 - J. E. Hopcroft, R. Motwani, and J. D. Ullman.
3. Elements of the Theory of Computation. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2nd edition, 1997- H. R. Lewis and C. H. Papadimitriou.