Course Title: Theory of Computation
Course no: CSC-251
Credit hours: 3

Full Marks: 80+20
Pass Marks: 32+8

Nature of course: Theory (3 Hrs.)

Course Synopsis: Deterministic and non-deterministic finite state machines, regular expressions, languages and their properties. Context free grammars, push down automata, Turing machines and computability, undecidable and intractable problems, and Computational complexity.

Goal: To gain understanding of the abstract models of computation and formal language approach to computation.

Course contents:

Unit 1: **14 Hrs.**

1.1 Review of Mathematical Preliminaries: Sets, Logic, Functions, Relations, Languages, Proofs.

1.2 Finite Automata: Deterministic and Non-deterministic Finite Automata, Equivalence of Deterministic and Non-deterministic Finite Automata, Finite Automata with Epsilon-Transition.

1.3 Regular Expressions and Languages, Equivalence of Regular Expressions and Finite Automata, Algebraic Laws for Regular Expressions, Properties of Regular Ranguages, Pumping Lemma for Regular Languages, Minimization of Finite State Machine.

Unit 2: **11 Hrs.**

2.1 Context-Free Grammar, Parse Trees, Derivation and Ambiguity, Normal Forms(CNF and GNF) of Context-Free Grammar, Regular Grammars, Closure Properties of Context-Free Languages, Proving a Language to be Non-Context-Free.

2.2 Push Down Automata (PDA), Language of PDA, Deterministic and Non-deterministic PDA, Equivalence of PDA's and CFG,s.

Unit 3: **10 Hrs.**

3.1 Introduction to Turing Machines, Computation by Turing Machines, Variants of Turing Machines, Non-deterministic Turing Machines, Turing Enumerable Languages.

3.2 Church's Thesis and Algorithm, Universal Turing Machines, Halting Problems, Turing Machines and Computers.

Unit 4: **10 Hrs.**

4.1 Undecidability: Recursive and Recursively Enumerable Languages, Encoding of

Turing Machine, Universal Language, Unrestricted Grammars and Chomsky Hierarchy, Unsolvable Problems by Turing Machines, Undecidable Problems, Post's Correspondence Problem.

4.2 Computational Complexity and Intractable Problems, Measuring Complexity, Class P, Class NP, NP-Completeness and Problem Reduction , NP-Complete Problems.

Text Book:

John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, **Introduction to Automata Theory, Languages, and Computation,** Second Edition, Addison-Wesley, 2001. ISBN: 81-7808-347-7

References:

1. Efim Kinber, Carl Smith, **Theory of Computing: A Gentle introduction,** Prentice- Hall, 2001. ISBN: 0-13-027961-7.

2. John Martin, **Introduction to Languages and the theory of computation,** 3rd Edition, Tata McGraw Hill, 2003, ISBN:0-07-049939-X

3. Harry R. Lewis and Christos H. Papadimitriou, **Elements of the Theory of Computation**, 2nd Edition, Prentice Hall, 1998.

Homework Assignments:

Homework assignments will be given through out the semester covering the lecture materials in each unit. The homework assignment will cover the 30% of the internal evaluation.

Pre-requisite: Discrete Mathematics, Fundamentals of Computer Programming and Data structure & algorithms.

Evaluation and Grading:

The evaluation and grading includes the 20% weitage for homework assignments and 2 mid term exam and 80 % weitage for final semester exam. The grading of the 20% internal evaluation will be as:

Homework assignment: 30% (6 marks)
First Mid-term exam: 30% (6 marks)
Second Mid-term exam: 40% (8 marks)

Homework assignment will be given in each weekend.
First Mid-Term: After completion of Unit 1.
Second Mid-Term: After Completion of Unit 3.

Course Title: System Analysis and Design
Course no: CSC-252 Full Marks: 60+20+20
Credit hours: 3 Pass Marks: 24+8+8

Nature of course: Theory (3 Hrs.) + Case Study

Course Synopsis: This course help launch the careers of successful systems analyst – or of users assuming an active role in building systems that satisfy their organization's information needs. Also provides a solid foundation of systems.

Goal: This course will provide the concept of system representation.

Course Contents

Unit 1. Overview of Systems Analysis and Design **4 Hrs.**

Introduction to system analysis and design, Types of Information Systems and Systems Development, Developing Information Systems and the Systems Development Life cycle, Systems analysis and design tools

Unit 2. Modeling Tools for Systems Analyst **5 Hrs.**

Modeling with Data Flow Diagrams, Drawing DFDs with CASE, Modeling with Entity – Relationship Diagrams

Unit 3. Structured Methodologies **6 Hrs.**

The need for a Structured Methodology, CASE as an Enabling Technology, Advantages and Disadvantages of Modeling and Data Dictionaries, Other Specification Tools

Unit 4. Systems Analysis **8 Hrs.**

Systems planning and initial Investigation, Information Gathering, The tools of Structured Analysis, Feasibility Study, Cost/ Benefit Analysis

Unit 5. Systems Design **8 Hrs.**

The process and Stages of systems Design, Input/ Output Forms Design, File Organization and Data Base Design

Unit 6. System Implementation **8 Hrs.**

System Testing and quality Assurance, Implementation and Software Maintenance, Hardware / Software Selection and the Computer Contract, Project Scheduling and Software

Unit 7. Object-Oriented Analysis and Design **6 Hrs.**

Object-Oriented Development Life Cycle, the Unified Modeling Language, Use-Case Modeling, Object Modeling: Class Diagrams, Dynamic Modeling: State Diagrams Dynamic Modeling: Sequence Diagramming, Analysis Verses Design

Case studies: Student must have to do one case study covers all chapters.

Text books: Jeffrey A. Hoffer, Joey F. George, Joseph S. Valacich, **Modern Systems Analysis and Design,** Pearson Education, Second Edition

References: Englewood Cliffs, New Jersey, **Systems Analysis and Design.** Jeffrey L. Whitten, Loonnie D. Bentley, 5rd Edition, **Systems Analysis and Design Methods.**
Grady Booch, Pearson Education, **Object Oriented analysis and design with applications**.

Course Title: Database Management System
Course no: CSC-253 Full Marks: 60+20+20
Credit hours: 3 Pass Marks: 24+8+8

Nature of course: Theory (3 Hrs.) + Lab (3 Hrs.)

Course Synopsis: This is a first database course for B.Sc. Computer Science and Information Technology students. It introduces the fundamentals of database technology. Topics covered include: database concepts, Database System Architecture, E-R model, relational model, database design theory, database languages, transaction management, concurrency control and database recovery.

Goal: There are two principle objectives for this course.

- To introduce the fundamental concepts and methods necessary for the design and use of a database systems.
- To provide practical experience in applying these concepts and methods using commercial database management systems.

Course Contents:

Unit 1: **17 Hrs.**

Introduction: Characteristics of database approach, Advantages of using DBMS, Database concept and architecture, Data Abstraction, Data Models, Instances and schema, Data independence, schema architecture, Database Languages, Database Manager, Database Administrator, Database Users

Data models: Entity-relationship Model: Entities and entity sets, Relationship and Relationship sets, Attributes, Mapping constraints, Keys, Weak and Strong entity types, E-R Diagrams, Reducing E-R Diagrams to Tables, Specialization and Generalization, Aggregation, Design E-R Database Schema
Relational Model: Structure of Relational Database, The Relational Algebra, The Tuple Relational Calculus, The Domain Relational Calculus, Modifying the Database, Views.
Historical Models: basic concepts of Hierarchical and Network Models.
Relational Commercial Languages: SQL and Query By Example (QBE)

Unit 2: **14 Hrs.**

Integrity & Security: Domain Constraints, Referential Integrity, Assertion and Triggers, Authorization & Authentication, Data encryption.
Theory of database design: Functional dependencies, trivial and non trivial dependencies, closure of a set of functional dependencies, irreducible sets of dependencies.
Normalization: non-loss decomposition and functional dependencies, first, second, and third normal forms, Dependency preservation, Boyce-Codd normal form.

Unit 3: **14 Hrs.**

Transaction Processing: Desirable properties of transactions, Implementation of atomicity and durability, Concurrent executions, Schedules and recoverability, testing for Serializability.
Concurrency Control: Overview of Concurrency Control, Locking techniques, Lock-based protocols, Timestamp-based protocols, Commit protocols, Optimistic technique, Granularity of data items, Time stamp ordering multi version concurrency control, Deadlock handling - detection and resolution.

Database Recovery: Failure Classification, The Storage Hierarchy, Transaction Model, Log-Based recovery, Buffer Management, Checkpoints, Shadow Paging, Failure with Loss of Non-volatile Storage.

Laboratory works: The course involves a mini project using any one of the popular Commercial database packages like Oracle, MySql, MS SQLServer, MS Access etc.

Prerequisite: Be familiar with at least one high-level programming language such as C, C++ or Java. Introduction to Operating Systems, Data Structures and Algorithms.

Textbooks: A. Silberschatz, H.F. Korth, and S. Sudarshan, Database System Concepts, 4th Edition, McGraw Hill (ISBN: 0-07-120413)

References:

1. C. J. Date, An Introduction to Database Systems, 8th Edition, Addison Wesley

2. Raghu Ramakrishnan, and Johannes Gehrke, Database Management Systems, McGraw-Hill, 2003. (ISBN: 0-07-246563-8)

3. Ramez Elmasri and Shamkant B. Navathe, Fundamentals of Database Systems, 4th Edition, Pearson Addison Wesley; 2003, (ISBN: 0321122267)

Homework Assignments: Homework assignments can be given according to the course covered throughout the semester.

Computer Usage: Windows or Linux based PC or workstation, Commercial database package installed in the Database server.

Category Content: Science Aspects: 50%
Design Aspects: 50%

Course Title: Computer Graphics
Course no: CSC-254 **Full Marks**: 60+20+20
Credit hours: 3 **Pass Marks**: 24+8+8

Nature of course: Theory (3 Hrs.) + Case Study

Course Synopsis: This Graphics hardware, software, and applications, data structures for graphics, graphics languages, models for 2D and 3D objects, clipping, hidden surface elimination, depth buffer, raster graphics, shading rendering, splines tools.

Goal: The objective of this course is to understand the theoretical foundation of 2D and 3D graphics.

Course Contents:

Unit 1. **5 Hrs.**

Introduction, Advantage of Computer Graphics, Areas of Applications, Hardware and Software for Computer Graphics. (Hard Copy, Display Technologies), Random Scan Display System, Video Controller, Random Scan Display Processor. Raster Graphics, Scan Conversion Algorithms (Line, Circle, Ellipse), Area Filling (Rectangle, Ellipse), Clipping (Lines, Circle, Ellipse), Clipping Polygons.

Unit 2. **10 Hrs.**

Geometrical Transformations, Homogenous coordinates, 2D and 3D Transformations, Matrix Representations, Window to View Port Transformation. 3D Viewing, Projections, Mathematics of Projections.

Unit 3. **15 Hrs.**

3D Object Representation, Representing Curves and Surfaces, (Polygon Meshes, Parametric Cubic Curves, Quadratic Surface), Solid Modeling (Sweep Representation, Boundary Representation, Spatial Partitioning Representation)

Unit 4. **12 Hrs.**

Visible Surface Determination, Various Techniques, Algorithms for Visible Surface Detection, (Z- Buffer, List priority, Scan Line Algorithms), Shading and Illumination models.

Unit 5. **3 Hrs.**

Introduction to virtual Reality and Animation.

Laboratory works: All algorithms covered in the text to be implemented in PHIGS/OpenGL in C/C++.

Text / References books:

1. Foley, J. D., A. V. Dam, S. K. Feiner, J. F. Hughes, *Computer Graphics Principle and Practices*, Addison Wesley Longman, Singapore Pvt. Ltd., 1999.
2. Hearn Donald, M. P. Baker, *Computer Graphics*, 2E, Prentice Hall of India Private Limited, New Delhi, 2000.

Course Title: Introduction to Cognitive Science
Course no: CSC-255 **Full Marks**: 60+20+20
Credit hours: 3 **Pass Marks**: 24+8+8

Nature of course: Theory (3 Hrs.) + Lab (3 Hrs.)

Course Synopsis: An introduction to cognitive science and its relation with other sciences. It covers briefly the area of Artificial Intelligence, Computational models and connectionist approach.

Goal:

- The student will gain an introductory understanding of what it means to say that intelligence is computational The student will:
    - Acquire a good understanding of what an algorithm is and learn how to implement algorithms in the programming language LISP
    - Develop an introductory understanding of formal models for computation, the limits of computation, the Chomsky hierarchy, and the Turing-Church hypothesis
- The student will study some of the modern attempts to demonstrate a computational model for intelligence through an introduction to the discipline of artificial intelligence, including introductions to knowledge representation, search, and artificial neural networks.
- Finally, the student will explore some of the positions taken in the ongoing discussion of this issue. In Philosophy and Linguistics, we will begin with Descartes, and look (and discuss) Turing, Gelernter, Newell and Simon, Penrose, Searle, and others, finishing with a partial response to Descartes given to us by Chomsky and others.

Course Contents:

Unit 1. Introduction to the Problem **6 Hrs.**

Cognitive Science and other Science, Descartes, Marr, Algorithms and Computation, Turing's response to Descartes, Application related system in the Cognitive Science.

Unit 2. Brief Introduction to Artificial Intelligence **13 Hrs.**

History and background of Artificial Intelligence, Knowledge representation, Human information processing and problem solving, Search, Expert system, Introduction of Neural Networks.

Unit 3. Computation **11 Hrs.**

Introduction, Basic Model for Computation, The Turing Machine, Computational and Language: the Chomsky hierarchy, The Physical Symbols Systems Hypothesis, Illustration of practical examples.

Unit 4. Approaches **15 Hrs.**

The connectionist approach, Different models and tool: Gelernter, Penrose, Pinker, Searle; Response to Descartes: Natural Language Processing, Parameters in the Natural Language Processing.

Text / Reference books:

1. Thinking about consciousness / David Papineau, Oxford: Clarendon Press New York: Oxford University Press, 2002.
2. Copeland, Jack: *Artificial Intelligence: A Philosophical Introduction*. Blackwell Publishers.
3. Cognition in a digital world / edited by Herre van Oostendorp, Mahwah, N.J.: L. Erlbaum Associates, 2003
4. The evolution and function of cognition / Felix Goodson, Mahwah, N.J.: Lawrence Erlbaum Associates, Publishers, 2003.

Course Title: Technical Writing
Course no: ENG-256 **Full Marks**: 80+20
Credit hours: 3 **Pass Marks**: 32+8

Nature of course: Theory (3 Hrs.)

Course Synopsis: This course offers a number of tools for writing in technical fields, by presenting clear explanations of key concepts and skills in written communication. The writing process is placed in a systems approach that integrates readings, planning, writing, and revising. Other features include suggestions about how to select, organize and present information in reports, papers and other documents.

Goal: This course presets the types of writing skills students need to have for a career in technology.

Unit 1. **15 Hrs.**

Turk & Kirkman, Writing is communicating, Thinking about aim and audience, Organization and layout of information, the use of headings and numberings, Rutherfoord, Foundations (audience, language an style, organization), Grammar Units (subjects and verbs, agreement: pronouns; pronoun references; avoiding shifts; modifiers; clause and simple sentence; compound sentences; complex an d compound – complex sentences; fragments, run-ones, and comma splices; transition words; parallelism).

Unit 2. **15 Hrs.**

Turk & Kirkman, Algorithms for complex possibilities and procedures, style for Readability, Writing with a computer, Informative summaries, Choosing and using tables, illustrations and graphic presentation techniques; Rutherfoord, , Writing Elements, (Technical definitions, technical descriptions, summaries, graphics, instructions, comparisons and contrast), Mechanics Units, (Capital letters; abbreviations and acronyms; end punctuation; commas; parentheses; dashes, brackets, ellipses, slashes, and hyphens; apostrophes; quotations).

Unit 3. **15 Hrs.**

Turk & Kirkman, Writing instructions, Writing descriptions and explanations, Writing letters and memoranda, Writing minutes and reports of proceedings, Writing in Examinations: Rutherfoord, Formes of Technical Communications (technical reports; forms, memos, and e-mail; business; letters; presentations; the job search: resume and letters), Appendices (common symbols and abbreviations; tips for word processing; sample reports; irregular verbs; job applications.

Text books:

Truck, Christopher & John Krikman. **Effective Writing: Improving scientific, technical and business communications**. Second edition. London and New York: E & FN Spoon, 1989. First Indian Reprint, 2003. ISBN 0-19-14660-1.
Futherford, Andreas J. **Basic Communications Skills for Technology**. Second Edition. Pearson Education, 2001. First Indian Reprint, 2001. ISBN 81-7808-281-0

Reference Books:

Lannon, John M. **Technical Writing**, Sixth Edition. New York: HarperCollins 1994. ISBN 0-673-52294-6.
Raman, Meenakshi, and Sangeeta Sharma. **Technical Communications: Principles and Practice**. New Delhi: Oxford University Press, 2004. ISBN 0-19-566804-9.
Gerson, Sharon J., and Stenven M. Gerson. **Technical Writing: Process and Product**. Third Edition. Pearson Education Asia, 2000. First Indian Reprint, 2001. ISBN 81-7808-381-7.
Mohan, Krishna, and Meera Banergi. **Developing Communication Skills**. New Delhi: Macmillan, 1990. ISBN 0-333-92919-5
Wehmeier, Sally, Chief Ed. **Oxford Advanced Learner's Dictionary of Current English**. Oxford University Press, 2005. ISBN 0-19-431665-3.
Lafferty, Petter, and Julian Rowe, eds. **The Hutchinson Dictionary of Science**. Oxford: Helicon, 1993,. ISBN 009-177151-X