# Autonomous Programme Structure (Modified) of Third Year B.Tech. Computer Engineering AY: 2019-20

T. Y. B. Tech. Computer Engineering Semester – II										
Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End Semester	Oral	Practical		
CE 3201	Theory of Computation	3	1	0	50	50	0	0	100	4
CE 3202	Artificial Intelligence and Machine Learning	3	0	0	50	50	0	0	100	3
CE 3203	Software Design and Architecture	3	1	0	50	50	0	0	100	4
PECE 3201	Programme Elective-II	3	0	0	50	50	0	0	100	3
PECE 3202	Programme Elective-III	3	0	0	50	50	0	0	100	3
CE 3204	Seminar	0	0	4	25	0	25	0	50	2
CE 3205	Artificial Intelligence and Machine Learning Laboratory	0	0	4	0	0	0	50	50	2
PECE 3203	Programme Elective-III Laboratory	0	0	2	0	0	25	0	25	1
AC 3201	Audit Course	0	0	2	0	0	0	0	0	0
	Total	15	2	12	275	250	50	50	625	22
	29			625				625	22	

### PECE 3201: Programme Elective-II

- 1. Wireless and Mobile Communication
- 2. Software Testing and Quality Assurance
- 3. Human Computer Interaction
- 4. Multimedia Systems
- 5. Swayam Online Course

# PECE 3202: Programme Elective-III

- PECE 3203: Programme Elective-III Laboratory
  - 1. Data Mining and Data Warehousing
  - 2. Embedded and Real-Time Systems
    - 3. Linux Internals
  - 4. Image Processing



DEAN ACADEMICS

MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052 **Principal** MKSSS's Cummins College of Engg. For Women, Karvenagar, Pune-52.

**APPROVED BY** Governing Body Members MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

college of Engin

Karvenagar, Pune-411052

\* uau

Cummins

# **CE 3201 THEORY OF COMPUTATION**

**Teaching Scheme** 

Lectures: 03 Hrs/Week

Tutorial : 01 Hrs/Week

### **Examination Scheme**

In Semester: 50 marks End Semester: 50 Marks Credits: 4

### **Prerequisites:**

- 1. Data Structures and Algorithms II (CE 2201)
- 2. Discrete Mathematics (CE 2103)

# **Course Objectives:**

To facilitate the learners -

- 1. Recall and understand the basics of mathematical concepts, formal languages and machines.
- 2. Understand and design different computational models like finite automata, regular expression, push down automata, context free grammar, turing machine for a given language.
- 3. Apply inter conversion between equivalent representations of a language.
- 4. Learn classification of a given problem into appropriate complexity class.

# **Course Outcomes:**

By taking this course, the learner will be able to -

- 1. Make use of fundamentals of mathematical concepts, formal languages and automata theory.
- 2. Construct different computation models like finite automata, regular expression, push down automata, context free grammar, turing machine for a given language.
- 3. Evaluate capabilities of Computational model by inter-conversion.
- 4. Classify a problem into appropriate complexity class.

# Unit 1: Introduction (06)

Finite and infinite set. Basic concepts of symbol, alphabet, string. Formal Language Definition, Problems. Finite representation of languages. Concept of Basic Machine and Finite State Machine introduction.

Regular Expression (RE): definition and operators, Regular Set, Algebraic Laws of Regular Expressions, Closure Properties of Regular Languages, Regular expression examples.

# Unit 2: Finite Automata (08)

Finite Automata (FA) - (Deterministic FA, Non-deterministic FA, C-NFA): Definition , Transition Function and language acceptance, Transition graph, Construction of FA.

Conversion of NFA with  $\in$  moves to NFA without  $\in$  moves, Conversion of NFA without  $\in$  moves to DFA, Direct Conversion of NFA with  $\in$  to DFA, Inter-conversion of RE and FA, Construction of RE equivalent to FA using Arden's Theorem. Construction of FA equivalent to RE (RE to  $\in$ -NFA,  $\in$ -NFA to DFA). Pumping Lemma for Regular languages, Limitations of FA.

# Unit 3: Context Free Grammar (07)

Grammar- Definition, representation of grammar. Context Free Grammar (CFG) - Definition, Derivation – Leftmost, Rightmost, sentential form, parse tree, ambiguous grammar and removing

ambiguity from grammar, Simplification of CFG, Normal Forms - Chomsky normal form, Greibach normal form, Closure properties of Context Free Languages (CFL), Decision properties of CFL, Chomsky hierarchy. Regular grammar- Definition, left linear, right linear grammar, Applications of grammar.

# Unit 4: Push Down Automata (07)

Definition, Notations – Transition Table form, Types of PDA (Deterministic PDA and Non Deterministic PDA), acceptance by final state, acceptance by empty stack, Construction of PDA (DPDA, NPDA), Instantaneous Description of PDA. Equivalence of PDA and CFG - Grammar to PDA conversion, Applications of PDA.

# Unit 5: Turing Machine (07)

Turing machines (TMs) - Formal Definition, TM Instantaneous Description, Transition Function, Languages of TM, Turing Machine and halting, Deterministic Turing Machines (DTM), Construction of DTM. Universal Turing Machine (UTM), Church -Turing hypothesis, Comparison between FA, PDA and TM. Turing Machine Halting Problem.

# Unit 6: Introduction to Undecidability (07)

A Language that is not recursively enumerable, Enumerating the binary strings, diagonalization Language, An undecidable problem that is RE, Recursive language, Complements of Recursive and RE languages, universal language, Undecidability of the universal language, classes P, NP and NP-Complete Problem

#### **Text Books:**

- 1. Hopcroft J., Motwani R., Ullman J., "Introduction to Automata Theory, Languages and Computations", Third edition, 2008, Pearson Education Asia. ISBN: 9788131720479
- 2. John C Martin. "Introduction to Language and Theory of Computation", Third edition, 2012, Tata McGraw- Hill, ISBN: 978007660489

#### **Reference Books:**

- 1. Daniel Cohen., "Introduction to Computer Theory", Second edition, 2011, Wiley Publications (India) ISBN: 9788126513345
- H.R. Lewis, C. H. Papadimitrou, "Elements of the Theory of Computation", Second edition, 2006, Prentice Hall Inc. ISBN: 8131703878
- 3. Michael Sipser, "Introduction to The Theory of Computation", Third edition, 2017 Thomson Course Technology, ISBN: 9781131525296
- Vivek Kulkarni, "Theory of Computation", Oxford university edition, 2013, ISBN 13:9780198084587

#### Web References:

1. NPTEL :: Computer Science and Engineering – Theory of Computation http://nptel.ac.in/courses/106101061

Example List of Tutorials:

- 1. Identify Complexity (n2, log n etc.) for a given code
- 2. Design of Regular Expression from Language
- 3. Design Deterministic Finite Automata
- 4. NFA design and NFA to DFA conversion
- 5. RE to NFA with null moves and NFA with null moves to NFA without null moves
- 6. Formal language to CFG and CFG to language conversion
- 7. Simplification of CFG and Chomsky Normal Form
- 8. Design of Push down Automata
- 9. Design of Turing Machine
- 10. Classification of a problem into appropriate complexity classes by reduction

# **CE 3202** Artificial Intelligence and Machine Learning

Teaching Scheme Lectures: 3 Hr/Week Examination Scheme In Semester : 50 Marks End Semester : 50 Marks Credits : 3

#### **Course Objectives:**

To facilitate the learners to-

- 1. Learn overview of classic Artificial Intelligence and basics of machine learning.
- 2. Understand various intelligent searches and knowledge representation.
- 3. Understand types of learning as well as machine learning.
- 4. Study applications in Artificial Intelligence and Machine Learning.

#### **Course Outcomes:**

By taking this course, the learner will be able to –

1. Build fundamental knowledge of AI, its applications and solve classical AI problems using different AI Techniques

2. Apply intelligent search algorithms on AI problems.

3. Make use of Knowledge Management techniques of AI for reasoning.

4. Apply the appropriate supervised / unsupervised Machine Learning (ML) method to solve the given problem.

5. Apply and examine different topics with various methods of expert system, pattern recognition, natural language processing, nature inspired computing.

#### Unit 1: Introduction to AI

Definitions of Artificial Intelligence, Artificial Intelligence Problems, Topics of Artificial Intelligence: Learning Systems, Knowledge Representation and Reasoning, Planning, Knowledge Acquisition, Intelligent Search, Logic Programming, Soft Computing, Management of Imprecision and Uncertainty, Production Systems: Traveling Salesman Problem, Water-Jug Problem, State Space Representation, State Space Search, Tic-Tac-Toe as a State Space, Branches of Artificial Intelligence.

#### **Unit 2: Heuristic Search Techniques**

Generate-and-Test, Search Techniques: Depth First Search, Breadth First Search, Best First Search Algorithm, Hill Climbing, Simulated Annealing, A\* Algorithm, Problem Reduction, AND–OR Graphs, The AO\* Algorithm, Towers of Hanoi Problem, Constraints Satisfaction: crypt-arithmetic problem, mini-max algorithm.

#### **Unit 3: Knowledge Management**

Knowledge Management, Types of Knowledge: Declarative Knowledge, Procedural Knowledge, Knowledge Representation, Approaches to Knowledge Representation, Issues in Knowledge Representation, First-order Logic: Basic Predicate Representations, Conversion of WFF to Clause Form, Resolution, Unification, Resolution Examples, Reasoning, monotonic and non-monotonic reasoning, Truth Maintenance Systems.

#### (07)

#### (07)

(07)

#### **Unit 4: Learning**

Types of Learning: Rote Learning, Learning by General Problem Solving, Concept Learning, Learning by Analogy, Learning problems and designing the learning systems, Machine Learning: Types of Problems in Machine Learning, Aspects of Inputs to Training, Learning Systems, Intelligent Agents.

# Unit 5: Machine Learning methods and models

Introduction to Supervised, Unsupervised, semi-supervised Learning, Ensemble Learning, discovery based Learning, Learning by problem solving, Reinforcement Learning, Support vector Machine, Artificial Neural Network : Perceptron, multi-layer perceptron, back propagation Neural Network, Self-organizing map.

# Unit 6: Applications in Artificial Intelligence and Machine Learning

Game Playing, Expert Systems, Natural Language Processing, Image Understanding & Computer Vision, Pattern Recognition, Virtual Reality, Nature Inspired Computing.

# **Text Books:**

1. Vinod Chandra S. S., Anand Hareendran S., 'Artificial Intelligence and machine learning', PHI, (2014), ISBN 978-81-203-4934-6.

2. Kulkarni P., Joshi P., 'Artificial Intelligence: Building Intelligent Systems', PHI Learning, (2015), ISBN 978-81-203-5046-5.

# **Reference Books:**

1. Peter, Norvig, 'Artificial Intelligence: A Modern Approach', Pearson, (3 rd edition), (2014), ISBN-0-13-103805-2.

2. Elaine Rich, Kevin Knight and Nair, 'Artificial Intelligence', Tata McGraw – Hill, (3rd edition), (2012), ISBN-978-0-07-008770-5.

3. Bratko I., 'Prolog Programming for Artificial Intelligence', Pearson Education, (3rd edition), (2004). 4. Tom M. Michell, 'Machine Learning', McGraw Hill Education, Indian edition (2013), ISBN-13: 978-1-25-909695-2.

5. Ethem Alpaydin, 'Introduction to Machine Learning', PHI, (2006), ISBN-81-203-2791-8.

#### (07)

(07)

(07)

# CE 3203 Software Design And Architecture

# **Teaching Scheme**

Lectures: 3 Hrs/Week Tutorial: 1 Hr/Week **Examination Scheme** In Semester: 50 marks

End Semester: 50 marks Credits: 4

Prerequisite: Data structures and Algorithms II (CE 2201)

# **Course Objectives:**

# To facilitate the learner to -

- 1. Develop familiarity with the basic concepts of software architecture and quality attributes of a system.
- 2. Model the software requirements of a system using Unified Modeling Language (UML) to understand the architectural, structural and behavioral aspects of the system.
- 3. Understand and apply various design patterns in creating an object oriented design.
- 4. Get exposure to the various software testing techniques and methods.

# **Course Outcomes:**

# By taking this course, the learner will be able to -

- 1. Apply and analyze the concepts of software architecture and quality attributes to realize the solution of a system.
- 2. Build structural and behavioral models using Unified Modeling Language (UML).
- 3. Apply and analyze various design patterns to understand reusability in object oriented design.
- 4. Apply various software testing techniques at unit level, suitable to different problem areas.

# Unit 1: Introduction to Software Architecture

Software Development Life Cycle (SDLC), Software Requirement Specification (SRS), What is Software Architecture, Why Software Architecture is important.

# Unit 2: Design Using Unified Modeling Language (UML)

Importance of modeling, Use case Diagrams, Activity Diagrams, Class Diagrams, Sequence Diagrams.

# **Unit 3: Quality Attributes**

Understanding Quality Attributes, Quality Attribute Scenarios and Tactics - Performance, Security, Usability.

#### (08)

(06)

(08)

# **Unit 4: Creational and Structural Design Patterns**

What is Design Pattern, Classification of Design Patterns, Elements of Design Pattern, Creational Design Patterns - Singleton, Factory Method, Structural Design Patterns - Proxy, Adapter.

# Unit 5: Behavioral Design Patterns

Observer, Iterator, Model View Controller (MVC), Mediator.

# **Unit 6: Software Testing**

Introduction, Verification and Validation, White Box testing - Structural Testing – Unit / Code functional testing, Code coverage testing, Code complexity testing, Black Box testing - Equivalence Class Partitioning, Boundary Value Analysis.

# Text books:

- 1. Len Bass, Paul Clements, Rick Kazman, 'Software Architecture in Practice', *Pearson Education*, (3rd Edition)(2013).
- 2. Grady Booch, James Rumbaugh, Ivar Jacobson, '**The Unified Modeling Language User Guide**', *Pearson Education*, (2nd edition)(2008).
- 3. Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides, 'Design Patterns-Elements of Reusable Object-Oriented Software', *Pearson Education*, (2002).
- 4. Srinivasan Desikan, Gopalaswamy Ramesh, 'Software Testing Principles and Practices', *Pearson Education*, ISBN 81-7758-121-X (2013).

# **Reference books:**

- 1. Len Bass, Paul Clements, Rick Kazman, 'Software Architecture in Practice', *Pearson Education*, (2nd Edition) (2006).
- 2. Mary Shaw and David Garlan, 'Software Architecture Perspectives on an Emerging Discipline', *Prentice Hall of India*, (1996).
- 3. Richard N. Taylor, Nenad M. and Eric M. Dashofy, 'Software Architecture: Foundations, Theory and Practice', *Wiley*, (2006).
- 4. Jim Arlow and Ila Neustadt, 'UML 2 and the Unified Process –Practical Object-Oriented Analysis and Design', *Pearson Education*, (2nd edition) (2006).
- 5. Iien Burnstein, 'Practical Software Testing', Springer (India) private limited, (2005).

# **Example List of Tutorials:**

- 1. Study architectural styles and submit a report on these styles.
- 2. A case study of any website or any other large system and its architecture for quality attributes requirements such as Performance, Security, Usability and Availability.

(07)

- 3. Design a Software Requirement Specification (SRS) document for a given system.
- 4. Draw Use case diagrams for capturing and representing requirements of a given system.
- 5. Draw Activity diagrams to display the business flows for a given system.
- 6. Draw Class diagrams to identify and describe key concepts like classes, relationships and other classifiers like interfaces.
- 7. Draw Sequence diagrams to show message exchanges in a given system.
- 8. Identify suitable design patterns for a given application.
- 9. Apply various Black Box testing methods for unit testing of a sample application.
- 10. Apply various White Box testing methods for unit testing of a sample application.

# **CE 3205** Artificial Intelligence and Machine Learning Laboratory

# **Teaching Scheme**

Practical: 4 Hr/Week

Examination Scheme Practical: 50 Marks Credits : 2

# **Course Objectives:**

To facilitate the learners to-

- 1. Experiment Artificial Intelligence and machine learning concepts from syllabus.
- 2. Experiment AI searches like A\*, Min-max algorithm.
- 3. Understand monotonic and non-monotonic knowledge representation.
- 4. Experiment classification and clustering algorithms.

# **Course Outcomes:**

By taking this course, the learner will be able to-

- 1. Implement and analyse various intelligent searching techniques.
- 2. Apply Knowledge Management techniques to implement truth maintenance system / Expert system.
- 3. Choose the appropriate supervised Machine Learning (ML) method and solve the given problem.
- 4. Choose the appropriate Unsupervised ML method and solve the given problem.

# Example list of Assignments:

# Assignments Group A (Mandatory)

- 1. Study: Learning simple statements in Prolog
- 2. Implement DFS/BFS for simple water jug problem
- 3. Implement A\* algorithm for 8 puzzle problem
- 4. Implement Unification algorithm
- 5. Represent knowledge using Prolog by implementing small expert system
- 6. Implement Best first search algorithm

# Assignments Group B (Any 3)

- 1. Write a program to implement Min-max algorithm for game playing
- 2. Write a program to implement Perceptron in artificial neural network
- 3. Write a program to implement SOM
- 4. Write a program to implement SVM/backpropagation learning algorithm

# Assignment Group C

Develop any one machine learning tool for application: character/sign classification

# PECE 3201 Software Testing And Quality Assurance

# Teaching Scheme

Lectures: 3 Hrs/Week

Examination Scheme In Semester: 50 marks End Semester: 50 marks Credits: 3

#### **Prerequisites:** -

#### **Course Objectives:**

# To facilitate the learner to -

- 1. Develop familiarity with the fundamental concepts and the process of software testing.
- 2. Gain comprehensive knowledge about various software testing techniques and methods.
- 3. Study various software testing strategies.
- 4. Get exposure to the quality assurance process and its role in software development.
- 5. Learn the essential features of various automated testing tools used for testing different types of applications.

# **Course Outcomes:**

# By taking this course, the learner will be able to -

- 1. Apply various concepts and process of software testing , testing metrics and quality assurance to different scenarios.
- 2. Apply and analyze various software testing techniques and strategies to different problem areas.
- 3. Build the essential test cases at various phases of software testing life cycle.
- 4. Apply and compare techniques of automated testing and modern testing tools for testing various types of applications.

#### **Unit 1: Introduction**

Need of testing, Basics of Software Testing, Testing Principles, Goals, Software Testing Life Cycle, Defects, Defect management, Verification and validation, Test Plan.

#### **Unit 2: Black Box Testing**

Introduction, Need of black box testing, Testing Methods - Requirements based testing, Positive and negative testing, Equivalence Class Partitioning, Boundary value analysis, Decision table / Cause effect graphing, State based testing, Domain testing, Examples of Black-Box testing.

#### **Unit 3: Testing Strategies and System Testing**

Unit, Integration, System, Acceptance testing, Usability testing, Regression testing, Scenario testing, Adhoc testing, Functional, Performance testing, Stress testing, Security testing, Alpha-Beta testing.

#### (08)

(07)

### **Unit 4: Testing Metrics and Quality Assurance**

Testing Metrics and measurements, Types of metrics – Project, Progress, Productivity, Software quality, Quality control and assurance, Quality factors, Software Quality Assurance (SQA) Model - Six Sigma, Ishikawa's Seven Basic Tools.

# Unit 5: White Box Testing

Introduction, Need of white box testing, Testing types, Static testing, Structural Testing – Unit / Code functional testing, Code coverage testing, Code complexity testing, Challenges in White box testing, Examples of White-Box testing.

# **Unit 6: Recent Trends and Automated Testing**

Agile Testing, Model based testing, Need for Automation, Keyword driven automation, Data driven automation, Manual testing versus Automated testing, Automated Testing Tools, Selection of tool, Study of Testing tools and frameworks (such as Selenium, JUnit, Bugzilla).

# Text books:

- 1. Iien Burnstein, 'Practical Software Testing', Springer (India) private limited (2005).
- 2. Srinivasan Desikan, Gopalaswamy Ramesh, **Software Testing Principle and Practices**', *Pearson Education*, ISBN 81-7758-121-X (2013).
- 3. Nageshwar Rao Pusuluri, '**Software Testing Concepts and Tools'**, *Dreamtech press*, ISBN 81-7722-712-2 (2008).

# **Reference books:**

- 1. Ron Patton, '**Software Testing**', *Pearson Education*, ISBN-13: 978-0-672-32798-8 (Second Edition) (2013).
- 2. Stephen H Kan, 'Metric and Model in Software Quality Engineering', *Pearson Education* ISBN 81-297-0175-8 (Second Edition) (2006).
- 3. William Perry, 'Effective Methods for Software Testing', *Wiley Publication*, ISBN 81-265-0893-0 (Third Edition) (2006).
- 4. Dr. K.V.K.K. Prasad, 'Software Testing Tools', *Dreamtech Press* ISBN: 10:81-7722-532-4 (2008).
- 5. Naresh Chauhan, '**Software Testing Principles and Practices**', Oxford *University Press*, ISBN 0-19-806184-6 (2011).

# Web References

- 1. http://www.seleniumeasy.com/selenium-tutorials
- 2. https://www.tutorialspoint.com/junit
- 3. https://www.bugzilla.org

#### (07)

# (07)

(07)

# **PECE 3201 Human Computer Interaction**

**Teaching Scheme** 

Lectures: 3 Hrs /week

**Examination Scheme** 

In Semester: 50 marks End Semester: 50 marks Credits: 3

# **Course Objectives:**

To facilitate the learner to-

1. Identify the main modes of human computer interaction.

2. Identify the common pitfalls in data analysis, interpretation and presentation.

3. Understanding the use of prototyping and evaluation in design.

4. Understand the advanced techniques of Human Computer Interaction.

# **Course Outcomes:**

By taking this course, the learner will be able to:

1. Apply the concepts of HCI to enhance the user experience.

2. Select the appropriate data gathering techniques and establish the requirements for the good design.

3. Apply the fundamental aspects of designing and evaluating the interfaces.

4. Compare the advanced techniques of Human Computer Interaction.

# **Unit 1: Introcuction to Interactive Design**

What is HCI – design, models, evaluation, Need to understand people, computers and methods. Humans – Memory, Attention Span, Visual Perception, psychology, ergonomics. Computers – speed, interfaces, widgets, and effects on interaction. Understanding Users, Universal Design, User-centered design.

# **Unit 2: Design Process and Interaction Styles**

HCI in the Software Process, HCI design principles and rules, Shneiderman's golden rules, Normans seven principles, Nielsens ten heuristics with example of its use. Interaction Styles, Direct Manipulation - Menu selection, Form Fill-in and Dialog Boxes

# **Unit 3: Establishing Requirements**

Understanding importance of identifying the requirements, Different kinds of requirements, Data gathering for requirements, Data analysis, Data interpretation and presentation, Task description and analysis.

# **Unit 4: Design, Prototyping, and Construction**

Prototyping and construction, Conceptual design, Physical design, User Persona, Using scenarios in design, Using prototypes in design and support for design, Handling errors and designing help.

# **Unit 5: Evaluation Approaches**

Importance of evaluation, Evaluation approaches and methods, Evaluation case studies, Determine, Explore, Choose, Identify, Decide, Evaluate (DECIDE): A Framework to guide evaluation.

# **Unit 6: New Interaction Technologies**

Explicit and Implicit Human Computer Interaction, User Interfaces and Interaction for Four Widely Used Devices, Hidden User Interface via Basic smart Devices, Hidden User Interface via Wearable and Implanted Devices.

# (08)

(06)

(07)

(06)

#### (07)

(08)

# **Text books:**

 Rogers, Sharp, Preece, 'Interaction Design', Wiley Publications (India), (Third edition), (2014).
Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, 'Designing the User Interface: Strategies for Effective Human-Computer Interaction', Pearson Education Limited (India),(2010).

3. Stefan Poslad, 'Ubiquitous Computing', Wiley Publications (India), (2014).

# **Reference Books:**

1.Alan Dix, 'Human Computer Interaction', Pearson Education Limited (Third edition), (2004). 2.Wilbert O. Galitz, 'The Essential Guide to User Interface Design', Wiley Publications (Second edition), (2003).

3.John M. Carroll, 'Human-Computer Interaction', Pearson Education Limited, (2002). 4.Don Norman, 'The Design of Everyday Things', Basic Books, A member of the Perseus Books Group, (2013).

# PECE 3201 – Multimedia Systems

**Teaching Scheme:** 

Lectures: 3Hrs/Week

Examination Scheme: In-Semester: 50 Marks End-Semester: 50 Marks Credits: 3

#### **Course Objectives:**

#### To facilitate the learners to -

- Understand basics of Multimedia Systems.
- Understand various file formats.
- Learn Multimedia editing tools.
- Analyze various compression techniques.
- Learn advances in multimedia.

#### **Course Outcomes:**

#### By taking this course, the learner will be able to

- Build the knowledge of multimedia systems and its characteristics.
- Utilize text and audio file formats and compression techniques in multimedia applications.
- Apply digital image and video processing techniques useful in multimedia applications.
- Build the knowledge of animation and Virtual reality concepts.
- List and analyse advances in multimedia.

#### Unit – I: Introduction to Multimedia

What is Multimedia? (Text, Graphics, Audio, Video, Animation), Multimedia presentation and production, Multimedia Authoring Tools (Various tools for creation and editing of Multimedia Projects), Hardware and Software requirement for Multimedia, Multimedia Applications

#### Unit – II: Text and Audio

Text - Introduction, About Fonts and Faces, Using Text in Multimedia, Font Editing and Design Tools, Text Compression (HUFFMAN, LZ, LZW), File Formats (TXT, DOC, RTF, PDF, PS), Hypertext and Hypermedia.

Audio – Introduction, Characteristics of Sound, Elements of Sound System, Digital Audio, Synthesizer, MIDI, Audio File Formats (WAV, VOC, MP3), Audio Processing Softwares.

#### Unit – III: Images

Digital Image, Basic steps for image processing, Image file formats (BMP, TIFF), Image Compression (RLE, JPEG), Image Manipulation, Image processing softwares.

#### Unit – IV: Video

Types of Video Signals, Analog Video, Digital Video, Video File Formats and CODEC (AVI, MPEG), Video Editing Softwares.

(08)

(07)

(07)

#### Unit – V: Animation and Virtual Reality

Animation- Introduction, Uses, Types, Principles, Animation on Web, 3D animation, Rendering, Animation Softwares Virtual reality - Introduction, Forms, Applications, Software Requirements, Devices, VRML

#### Unit VI: Introduction to Advances in Multimedia

Introduction, Challenges of Multimedia Information processing, Watermarking, Organization, Storage and retrieval Issues, Neural Networks for multimedia processing, Multimedia Processors

#### **Text Books:**

- 1. Ranjan Parekh, **'Principles of Multimedia'**, *McGraw Hills education*, (2<sup>nd</sup> edition), (2013)
- 2. Ralf Steinmetz, Klara Nahrstedt, **'Multimedia: Computing, Communications and Applications'**, *Pearson*, (8<sup>th</sup> Impression 2011)
- 3. Nigel Chapman & Jenny Chapman, **'Digital Multimedia'**, *Wiley Publications*, (2<sup>nd</sup> edition) (2004)

#### **Reference Books:**

- 1. Ze-Nian Li, Marks S. Drew, **'Fundamentals of Multimedia'**, *Pearson Education*, (2005)
- 2. Tay Vaughan , **'Multimedia: Making it work'**, *Tata McGraw-Hill*, (8<sup>th</sup> edition), (2011)
- 3. Judith Jeffcoate, 'Multimedia in Practice', Prentice Hall of India, (2003)

(07)

# PECE 3202 Data Mining and Data Warehousing

Teaching Scheme

Lectures: 3 Hrs /week

**Examination Scheme** In Semester: 50 Marks

End Semester: 50 marks Credits: 3

Prerequisite: Database Management Systems (CE 3102)

# **Course Objectives:**

# To facilitate the learners to -

- 1. Understand the concepts and techniques of data mining and data warehousing.
- 2. Apply various data pre-processing and visualization techniques.
- 3. Design and model a data warehouse and its components.
- 4. Compare and analyze various Data Mining algorithms based on performance parameters.
- 5. Understand advances in the field of Data Mining.

# **Course Outcomes:**

# By taking this course, the learner will be able to -

- 1. Demonstrate the need, importance and procedure of building a Data Warehouse (DW) to solve any Business Intelligence (BI) problem
- 2. Choose and apply appropriate pre-processing techniques to make data ready for further analysis
- 3. Design a Data warehouse model for the given application
- 4. Compare and analyze the strengths and weaknesses of various data mining algorithms
- 5. Understand the advances in the field of Data Mining.

#### Unit 1: Introduction to Data Warehousing and Data Mining

Introduction to data warehousing and data mining, Evolution of decision support systems, operational data Vs. historical data (Data Warehouse data), importance of data preparation for data mining, types of data mining techniques, various data mining functionalities, data mining task primitives, integration of operational system and Data Warehousing system.

# **Unit 2: Data Preprocessing**

Introduction / overview of data pre-processing; Descriptive data summarization – Measuring central tendency, dispersion, range, quartiles, variance and standard deviation of data, Graphical displays of descriptive data summaries; Data cleaning, Data Integration, Data Transformation, Data Reduction.

# Unit 3: Data Warehouse and Online Analytical Processing (OLAP) Technology

3-tier Data Warehouse architecture, data warehouse design process; Modelling subject(s), dimensions and measures, Multidimensional data modelling, Introduction to OLAP, OLAP operations, Data cube generation, Concept hierarchy generation, Case study on designing a Data warehouse for a given application.

#### (08)

# (07)

# Unit 4: Data mining Functionalities – I

# Unit 5: Data mining Functionalities – II

Classification and Regression, Decision Tree Induction, Bayesian Classification, Nearest Neighbor approach; Mining frequent patterns and Association Rules – Apriori Algorithm, Outlier analysis.

Data mining process, Types of Data Mining Systems; Cluster Analysis - Types of Data In Cluster Analysis, Categorization of Major Clustering Methods, k-means clustering, Density based Clustering.

# Unit 6: Advances in Data Mining

Information Retrieval and Text Mining, Multimedia Data Mining, Graph Mining, Mining World Wide Web, Stream, Time series and Sequence data mining, Applications and trends in Data Mining.

# **Text Books:**

- 1. Han, J., and Kamber, M., 'Data Mining: Concepts and Techniques', Morgan Kaufmann, (3 rd edition), (2011)
- 2. Tan P.N., Steinbach M., Kumar V., 'Introduction to Data Mining', Addison Wesley, (2 nd edition), (2006)

# **Reference Books:**

- 1. W. H. Inmon, 'Building the Data Warehouse', Wiley, (4 th edition).
- 2. Alex Berson, Stephen J, 'Data Warehousing, Data Mining, & OLAP', Tata McGraw-Hill, (2004).
- 3. Dunham M.H., 'Data Mining: Introductory and Advanced Topics', Prentice Hall, (2003).
- 4. Miller T. W., 'Data and Text Mining A Business Applications Approach', Pearson education, (2008).
- 5. Maimon O., Rokach L., 'Data Mining and Knowledge Discovery Handbook', Springer, (2009).
- 6. Pujari A K, 'Data Mining Techniques', Universities Press, (2010).

#### Web References:

- 1. www.autonlab.org/tutorials : Statistical Data mining Tutorials
- 2. www-db.standford.edu/ullman/mining/mining.html : Data mining lecture notes
- 3. ocw.mit.edu/ocwweb/slon-School-of-management/15-062Data-MiningSpring2003/course home/index.htm : MIT Data mining open course ware
- 4. www.kdnuggets.com : Data mining resources

# (07)

# (08)

# PECE 3203 Data Mining and Data Warehousing Laboratory

# Teaching Scheme

Practical: 2 Hrs / week

**Examination Scheme** Oral: 25 marks Credit: 1

# **Course Objectives:**

# To facilitate the learners to -

- 1. Model and build a data mart / data warehouse.
- 2. Study and analyze various open source data sets to pre-process them using open source data mining tools.
- 3. Implement data mining algorithms to discover interesting patterns.
- 4. Analyze results of data mining algorithms

# **Course Outcome:**

By taking this course, the learner will be able to –

- 1. Study and process raw data to model and build a data warehouse, using appropriate schema
- 2. Experiment with large open source datasets by applying pre-processing tools and techniques
- 3. Build and analyze various data mining algorithms on real time data
- 4. Implement advanced Data Mining functionalities such as Text Mining and Mining unstructured data .

# **Example List of Assignments**

# Assignments Group A (Mandatory)

- 1. Explore WEKA Data Mining / Machine Learning Toolkit and perform the following operations: Understand the features of WEKA toolkit, Study the arff file format, explore the available data sets in WEKA.
- 2. Load any one dataset in Weka and observe the following : List the attribute names and their types, Number of records in each dataset, class attribute (if any), Plot Histogram, Determine the number of records for each class, Visualize the data in various dimensions; Apply various pre-processing tasks; Apply classification OR clustering algorithms on the chosen dataset and observe the results
- 3. Implement K-means clustering algorithm using a programming language that you are familiar with such as Java / Python. Compare the performance of your algorithm on the dataset, used in Weka, on different parameters such as accuracy, scalability, efficiency etc. by changing input parameter value such as K.

# Assignments Group B (Any 2)

- 1. Implement DBSCAN clustering algorithm. Compare the performance of your algorithm on the dataset, used in Weka, on different parameters such as accuracy, scalability, efficiency etc.
- 2. Implement a decision tree classification algorithm. Compare the performance of your algorithm on the dataset, used in Weka, on different parameters such as accuracy, scalability, efficiency etc.
- 3. Implement Apriori, a Frequent Pattern Analysis algorithm. Assume suitable data. Compare the performance of your algorithm on the dataset, used in Weka, on different parameters such as accuracy, scalability, efficiency etc.
- 4. Implement Information Retrieval using TF / IDF algorithm. Assume suitable data.

# Assignments Group C (Any 1)

- 1. Build a Data Warehouse / Data Mart (using open source tools like Pentaho or other data warehouse tools like Microsoft-SSIS etc.) Identify source tables and populate sample data Analyze which multidimensional model (Star, snowflake and Fact constellation) will be best suited for the given application and design the schema (Example Applications can be Banking, Insurance, Finance, Healthcare, Manufacturing, Automobile, etc.)
- 2. Study any of the existing data warehouse / data repository / ... and prepare your report based on data / model / tools and techniques / software used etc.
- 3. Download, install and study the features of any open source data mining
- 4. compare its features with Weka.

Teaching Scheme

Lecture : 3 Hrs/week

#### **Examination Scheme**

In semester : 50 marks End semester : 50 marks Credits : 3

# **Course Objectives:**

# To facilitate the learner to-

- 1. Understand basic concepts of digital image processing.
- 2. Learn and apply image enhancement and Image Segmentation techniques.
- 3. Understand object Recognition, Image Restoration and reconstructions.
- 4. Learn and apply image compression techniques and Understand image processing applications.

# **Course Outcome:**

# By taking this course, the learner will be able to -

- 1. Identify basic steps of digital image processing.
- 2. Select, Examine, Justify the image enhancement techniques and Image Segmentation techniques on images.
- 3. Identify Image Restoration, reconstructions techniques.
- 4. Examine image compression techniques. Identify fundamental steps required for different image processing applications.

# **Unit 1: Introduction to Image Processing**

Introduction to digital image processing: Origin, usage and application of image processing, Fundamental steps and component of image processing system, introduction to Human Visual System, Image sensing and acquisition, Basic concepts in sampling and quantization, representation of digital images. Elements of matrix theory

# **Unit 2: Image Enhancement Techniques**

Basic image preprocessing (contrast enhancement, simple noise reduction, color balancing), some basic gray level transformations, Histogram Processing, Arithmetic Operations, Spatial filtering, Smoothing and Sharpening Spatial filters, Image Enhancement in the Frequency Domain, Gaussian filters, Homomorphic filtering.

# Unit 3: Image Compression

Introduction to Image Compression and its need, Coding Redundancy, Classification of Compression Techniques (Lossy and Losless - JPEG, RLE, Huffman, Shannon fano), Scalar & Vector Quantization.

# Unit 4: Image Restoration & Reconstruction.

Model of Image degradation, Noise Models, Classification of image restoration techniques, Blinddeconvolution techniques, Lucy Richardson Filtering, Wiener Filtering.

# (08)

(07)

#### (07)

# Unit 5:

# Image Segmentation, Analysis and Object Recognition.

Introduction to feature extraction: Edges, Lines & corners detection, Texture & shape measures. Segmentation & thresholding, region extraction, edge (Canny) and region based approach, use of motion in segmentation.

Introduction to Object Recognition, Object Representation(Signatures, Boundary Skeleton), Simple Boundary Descriptors, Regional descriptors (Texture).

# Unit 6: Advances in Image processing Applications

(06)

Medical Image Processing, Face detection, Iris Recognition, Remote Sensing, Synthetic-aperture radar (SAR) Image Processing

# **Text Books:**

- 1. R.C. Gonzalez, R.R. Woods, '**Digital Image Processing**', ISBN 978-81-317-2695-2,*Person* (Third Edition) ,(2011)(62 copies)
- 2. S.Jayaraman, S. Esakkirajan, T. Veerakumar , **'Digital Image processing'**, ISBN 978-0-07-014479-8, *Mcgraw Hills Publication (Tenth reprint)*, (2013)

# **References:**

- Sonka, Hlavac, Boyle, 'Digital Image Processing and Computer Vision', ISBN 978-81-315-0555-7,*Cenage Learning* (Sixth Indian Reprint), (2011)
- 2. B. Chanda, D.Datta Mujumdar '**Digital Image Processing And Analysis'**, *PHI*, ISBN 978-81-203-4325-2, (Second Edition),(2013)
- 3. Sridhar S. 'Digital Image Processing', Oxford University Press, (Second Edition),(2016)
- 4. Anil Jain, '**Fundamentals of Digital Image Processing'**, *PHI*, ISBN-81-203-0929-4 (Indian Reprint) ,(1995)

5Basudeb Bhatta ' **Remote Sensing and GIS'** Oxford University Press, ISBN 978-0-19-807239-3 (Second Edition)(2014)

(08)

# Teaching Scheme

Practical: 2Hrs/week Credits: 1 **Examination Scheme** Oral : 25 Marks

# **Course Objectives:**

# To facilitate the learners to -

- 1. Learn Basics Image Processing operations like image Read, Write, add, subtract.
- 2. Understand and apply algorithms used for image enhancement, edge detection.
- 3. Design Image Processing application using various techniques.
- 4. Learn and use different Image Processing Tools.

# **Course Outcome:**

# By taking this course, the learner will be able to -

- 1. Make use of basic operations of image processing on the given image.
- 2. Apply and analyze image enhancement techniques and image segmentation techniques
- 3. Develop small image processing application
- 4. Make use of Image Processing tool.

# **Example list of Assignments:**

# Group A: (Mandatory)

- 1. Write a program to create a simple image file in .tiff format, and display it .
- 2. Write a program to perform Intensity Transformation techniques on given image.
- 3. Write a program for image enhancement techniques.

# Group B: (Any Three)

- 1. Write a program using derivative filtering techniques for edge detection .
- 2. Write a program to illustrate Morphological transformation using Dilation.
- 3. Write a program to illustrate Morphological transformation using Erosion.
- 4. Write a program to illustrate Image Restoration techniques.

# Group C: (Any One)

Develop any one of the Image processing application using MATLAB/OpenCV (in Limited Scope).

- 1. Medical Image Processing
- 2. Face detection
- 3. Iris Recognition
- 4. Finger Print detection

# **CE 3204 SEMINAR**

**Teaching Scheme** Practical : 4 Hrs./week **Examination Scheme** 

In semester : 25 marks Oral : 25 marks Credits : 2

# Course Objectives:

# To facilitate the learners :-

- 1. To identify the topic based on current engineering trends/ social problems/ new technologies.
- 2. To explore the basic principles of communication (verbal and non verbal) and active, empathetic listening, speaking and writing techniques.
- 3. To produce relevant technical documents by following best practices of technical writing.
- 4. To understand the basic principles of presentation, technical writing techniques for seminar.

# **Course Outcome:**

# By taking this course, the learner will be able to :

- 1. Select appropriate/reserach topic and write a technical report and present it to audience.
- 2. Be familiar and use the basic technical writing concepts and terms such as audience analysis,

jargon, format, visuals and presentation.

- 3. Improve skills to read, understand and interpret material on technology.
- 4. To enhance technical communication and presentation skills.

# **General Guidelines for Seminar:**

- Seminar is an individual student activity.
- The area/domain must be selected under the guidance of institute guide.
- Each student will select a topic in the current/new trends of Computer Engineering and Technology beyond the scope of syllabus avoiding the repetition in consecutive years.
- Student should do literature survey based on IEEE/ACM/Springer/Digital Library papers or technical Magazines/books, specify knowledge area, brief technical knowledge about the topic.
- Each student will make a seminar presentation based on the domain topic using audio/video aids for a duration of 20-25 minutes.
- Student have to submit the technical seminar report in the department.

# **Guidelines for assessment:**

- Internal guide will evaluate students on understanding of topic, punctuality and Timely Completion of Report, Paper presentation/Publication and Attendance.
- An external examiner(s) panel will be assessing the seminar work based on these parameters Understanding of Topic, flow of Contents, Presentation, report, Paper presentation/Publication, Question and Answers, Active Participation.

# **References:**

- 1. Research papers from reputed journals/transactions- references necessary for the Project.
- 2. Reference books/Magazines for conceptual technical support.