Cummins College of Engineering for Women

(An autonomous institute affiliated to Savitribai Phule pune university)

Karve Nagar, Pune - 411 052.

MAHARSHI KARVE STREE SHIKSHAN SANSTHA'S
CUMMINS COLLEGE OF ENGINEERING FOR WOMEN

Vision

To be globally renowned engineering institute for imparting holistic education and developing professional women leaders in engineering and technology

Syllabus Structure and Syllabus

of

T. Y. BTech
(Computer Engineering)

2023 Pattern [R0]

List of Abbreviations

Abbreviation	Title
PCC	Programme Core Course
BSC	Basic Science Course
ESC	Engineering Science Course
PE	Programme Elective Course
OE	Open Elective
VSEC	Vocational and Skill Enhancement Course
CC	Co-curricular Courses / Liberal Learning Course
IKS	Indian Knowledge System
VEC	Value Education Course
RM	Research Methodology
INTR	Internship
PROJ	Project
CEP	Community Engagement Project
RM	Research Methodology
Mm	Multidisciplinary Minor
AEC	Ability Enhancement course

Curriculum for UG Degree Course in BTech. Computer Engineering
(Academic Year: 2025-26 Onwards)

Third Year | Semester-V

Course Code	Course Title	Teaching Scheme Hours / Week			Cr	Examination Scheme			Total Marks
		L	T	P		ISE	ESE	Pr/Or	
23PCCE501	Artificial Intelligence and Machine Learning	3	0	0	**3**	50	50	00	100
23PCCE502	Software Engineering and Design Architecture	3	0	0	**3**	50	50	00	100
23PCCE503	Computer Networks	3	0	0	**3**	50	50	00	100
23PCCE504	Design and Analysis of Algorithms	3	0	0	**3**	50	50	00	100
23PECE501	Programme Elective-I	3	0	0	**3**	50	50	00	100
23MmCE501	Enterprise Resource Planning	2	1	0	**3**	25	25	00	50
23PCCE501L	Artificial Intelligence and Machine Learning Laboratory	0	0	2	**1**	25	00	25	50
23PCCE503L	Computer Networks Laboratory	0	0	2	**1**	25	00	25	50
23MmCE501L	Enterprise Resource Planning Laboratory	0	0	2	**1**	25	00	25	50
23PECE501L	Programme Elective-I Laboratory	0	0	2	**1**	25	00	25	50
	Total	**17**	**01**	**8**	**22**	**375**	**275**	**100**	**750**

L=Lecture, T=Tutorial, P= Practical, Cr= Credits, ISE =In Semester Evaluation, ESE =End Semester Examination, Pr/Or = Practical/Oral

23PECE501 Programme Elective-I
A. Digital Image Processing B. Advanced Operating System C. User Experience Design (UX/UI) D. Internet of Things

Department of Computer Engineering

23PCCE501 ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

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

Course Objectives
To facilitate the learner to

1. Introduce students to the basic concepts and techniques of Artificial Intelligence and Machine Learning
2. Utilize data pre-processing techniques for given data
3. Become familiar with supervised machine learning algorithms such as regression, classification
4. Become familiar with unsupervised machine learning algorithms such as clustering, association rule mining method
5. Become familiar with heuristics algorithms and optimization algorithms popularly used in artificial intelligence and machine learning
6. Evaluate the performance of the designed machine learning model

Course Outcomes
After completion of the course, students will be able to

1. Apply fundamental Artificial Intelligence techniques and heuristic search methods to solve the problem
2. Make use of machine learning concepts and pre-processing technique for given data
3. Examine supervised machine learning techniques for problem solving
4. Solve the problems using various unsupervised machine learning techniques
5. Apply gradient based optimization algorithms for problem solving

Unit I: Introduction to Artificial Intelligence

Artificial Intelligence, Problem solving with AI, Production Systems, Traveling Salesman Problem, Water-Jug Problem, State Space Representation, State Space Search, Tic-Tac-Toe as a State Space, Intelligent Agents, AI ethics.

Unit II: Heuristic Searches Techniques

Management of Imprecision and Uncertainty, Best First Search Algorithm, Hill Climbing, Simulated Annealing, A*Algorithm, Problem Reduction, AND–OR Graphs, The AO* Algorithm, Constraints Satisfaction, Crypt-arithmetic problem, Mini-max algorithm, Case Studies on heuristics searches.

Unit III: Introduction to Machine Learning and Data preprocessing

Types of Problems in Machine Learning, Aspects of Inputs to Training, Supervised, Unsupervised, semi supervised, reinforcement learning, overfitting, underfitting, best practices in machine learning, Data preprocessing: Data cleaning, data integration, data reduction, data transformation, dimensionality reduction and data discretization.

Unit IV: Supervised Learning

Regression, Correlation and regression, line fitting by least square, linear and multiple regression Classification, Logistic regression, Nearest Neighbour Classification: K-nn, Introduction to Decision tree and Bayesian Classification, Performance Measures: Confusion matrices, accuracy, sensitivity, specificity, precision, recall, F-measure, Methods of cross-validation, Types of Errors, RMSE, MSE etc.

Unit V: Unsupervised Learning

Introduction to Clustering methods, k-means clustering, Hierarchical clustering, DBSCAN clustering method, Outliers, Frequent pattern analysis, Apriori Algorithm, Association rule mining.

Unit VI: Optimization Algorithms

Introduction to Optimization algorithms for artificial intelligence and machine learning, Gradient based optimization, Stochastic Gradient Descent (SGD), Adaptive Moment Estimation (Adam), Root Mean Square Propagation(RMSprop), Hyperparameter tuning, Introduction to Bio inspired optimization.

Text Books

1. Vinod Chandra S. S., Anand Hareendran S., 'Artificial Intelligence and machine learning', PHI, 2014, ISBN 9789389347838
2. Sandhya Arora and Latesh Malik, "Data Science and Analytics with Python",1st Edition, University Press, 2023 ISBN: 9789393330345
3. S. Sridhar, M. Vijayalakshmi, "Machine Learning", Oxford University Press, 1st Edition, 2021, ISBN : 978-0190127275

Reference Books

1. Rodolfo Bonnin, "Machine learning for developers" , Packt publication, 2017, ISBN 978-1-78646-987-8
2. Peter Flach, "Machine Learning: The Art and Science of Algorithms that make sense of data", Cambridge University Press, 1st Edition, 2015, ISBN: 978-1-316-50611-0
3. Tom M. Michell, 'Machine Learning', McGraw Hill Education, Indian edition 2013, ISBN 978-1-25-909695-2.
4. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, "Deep Learning", MIT Press, ISBN 9780262035613
5. Jiawei Han, Micheline Kamber, Jian Pei, "Data Mining: Concepts and Techniques", 3rd Edition, 2012, Morgan Kaufmann publishing, ISBN: 978-0-12-381479-1

Online/Web/Other References

1. Nptel/coursera courses on Machine Learning and Artificial Intelligence

23PCCE502 SOFTWARE ENGINEERING AND DESIGN ARCHITECTURE

Teaching Scheme	**Examination Scheme**
Lectures: 3 Hours/Week	In Semester: 50 marks
	End Semester: 50 marks
	Credits: 3

Prerequisite: Data Structures

Course Objectives
To facilitate the learner to

1. Develop familiarity with the basic concepts of software architecture
2. Model the software requirements of a system using Unified Modeling Language 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 quality attributes
5. Get exposure to the various facets of agile software process models
6. Learn the basic concepts of refactoring in software development

Course Outcomes
After completion of the course, students will be able to

1. Understand the concepts of software architecture
2. Build structural and behavioral models using Unified Modeling Language
3. Make use of various design patterns to understand reusability in object oriented design
4. Apply various software testing techniques and quality attributes, to different problem areas
5. Utilize the agile software process model for application development
6. Inspect the refactoring methods to restructure the code

Unit I: Fundamentals of Software Architecture

Software application domains, Real world software development – features, challenges, Software development phases, Software Development Life Cycle (SDLC), SDLC Models, Software Requirements Specification (SRS), Importance of Software Architecture, Architectural Styles.

Unit II: Design Using Unified Modeling Language

Design concepts, SOLID principles, Importance of modeling, Introduction to Unified Modeling Language (UML), Object-oriented modeling language, Use case Diagrams, Activity Diagrams, Class Diagrams, Sequence Diagrams, Component Diagrams, Deployment Diagrams. Case studies with related diagrams.

Unit III: Design Patterns

Design Pattern concept, Classification of Design Patterns, Elements of Design Pattern, Creational Design Patterns, Structural Design Patterns, Behavioral Design Patterns , Model View Controller (MVC) pattern. Real life examples of design patterns

Unit IV: Quality Attributes and Software Testing

Understanding Quality Attributes, Requirements, Quality Attribute Scenarios and tactics for Performance, Security, Usability

Introduction to testing , Verification and Validation, V-Model, White Box testing - Structural Testing – Unit / Code functional testing, Code coverage testing, Code complexity testing, Black Box testing, Use case based testing, Testing of web Applications, Case studies and applications

Unit V: Agile Software Development Methodology

Need for Agile software development, Agile process model, Evaluating Agile , Understanding the Agile Manifesto, Agile Principles, Agile approaches, User stories, Agile plan, Product Box, Elevator Pitch, Burn down-Burn up charts, Estimation,

Agile software practices: Test-Driven Development (TDD), Continuous Integration Continuous Delivery (CICD), DevOps, Benefits of agile approach

Unit VI: Software Refactoring

Refactoring concept, Need/Importance of Refactoring, Code smells, Refactoring methods: Extract, Move Rename method, Replace method, Introduce assertion, Replace constructor, Replace error code with exception, Pull up field, Pull up method, Push down method, Push down field, Extract subclass, Extract super class, Extract interface

Text books

1. Len Bass, Paul Clements, Rick Kazman, 'Software Architecture in Practice', Pearson Education, (3rd Edition) (2013).
2. Jonathan Rasmusson, 'The Agile Samurai: How Agile Masters Deliver Great Software', Shroff Publishers and Distributors (SPD), ISBN: 978-93-5213-411-3, (2016).
3. Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides, 'Design Patterns- Elements of Reusable Object-Oriented Software', Pearson Education, (2002).
4. Martin Fowler, Kent Beck, John Brant, William Opdyke and Don Roberts, 'Refactoring: Improving The Design of Existing Code', Pearson Education, ISBN: 978-81-317-3466-7, (2017).

Reference Books

1. Grady Booch, James Rumbaugh, Ivar Jacobson, 'The Unified Modeling Language User Guide', Pearson Education, (2nd edition) (2008).

2. Srinivasan Desikan, Gopalaswamy Ramesh, 'Software Testing Principles and Practices', Pearson Education, ISBN 81-7758-121-X (2013).
3. Mark C. Layton, Steven J. Ostermiller, 'Agile Project Management for Dummies', Wiley, (2nd Edition), (2017).
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).
6. Roger S. Pressman, 'Software Engineering: A Practitioners Approach', Tata McGraw Hill, (7th Edition) (2010).

Online/Web/Other References

1. Agile Software process model: https://www.agilealliance.org/
2. Basics of Scrum: https://www.scrumalliance.org/
3. https://www.bmc.com/blogs/devops-basics-introduction/

23PCCE503 COMPUTER NETWORKS

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

Course Objectives
To facilitate the learner to

1. Distinguish the fundamental concepts of networking standards, protocols and technologies
2. Identify the role of protocols at various layers in the protocol stack
3. Select and Compare the appropriate network and protocols by understanding the given requirements for a given system
4. Get familiar with fundamental concepts of network security and recent trends in networking

Course Outcomes
After completion of the course, students will be able to

1. Apply the fundamental concepts of computer networking and recent networking trends
2. Select appropriate protocols for different applications
3. Identify the data flow between two communicating hosts using various protocols of the TCP/IP protocol stack
4. Identify the role of various address in the TCP/IP Protocol stack

Unit I: Introduction to Computer Networks

Definition, Types of Networks, Protocol Design Issues for the Network Layers, Network Topologies, Types of Transmission Medium, Wireless Transmission, Network Architectures, Client-Server, Peer-to-Peer, Hybrid, Network Devices, Bridge, Switch, Router, Gateway, Access Point, Multiplexing, Network Models, The Open Systems Interconnection (OSI) Reference Model, Transmission Control Protocol/Internet Protocol Model (TCP/IP)

Unit II: Application layer

HyperText Transport Protocol (HTTP), Simple Mail Transport Protocol (SMTP), Multipurpose Internet Mail Extensions (MIME), Post Office Protocol version 3 (POP3), File Transfer Protocol (FTP), Cookies, Dynamic Host Configuration Protocol (DHCP), Domain Name Server (DNS), Working, Proxy Server, Need and Significance, Working

Unit III: Transport Layer

Transport Layer Design Issues, Protocol Overview, Header Structure, Transmission Control Protocol (TCP) Functions, Connection Management, Error Control, Flow Control, Congestion Control, User Datagram Protocol (UDP) Overview, Typical Applications Support, TCP vs UDP, TCP and UDP Socket Primitives.

Unit IV: Network Layer

Functions of Network Layer, Design Issues, IP Protocol: Classes of IP (Network Addressing), IPv4, IPv6, Forwarding Concepts: Addressing, Network Address Translation, Subnet Masking, Next Hop, Classless Inter-Domain Routing (CIDR), Routing Algorithms: Dijkstra's, Distance Vector Routing, Link State Routing, Network Layer Protocols: Address Resolution Protocol(ARP), Reverse Address Resolution Protocol, Internet Control Messaging Protocol, Routing Protocols: Routing Information Protocol, Open Shortest Path First, Border Gateway Protocol, Unicast Routing Protocols, Multicast Routing Protocol.

Unit V: Data Link Layer

Introduction, Functions, Design Issues, Services, Framing, Error Detection and Correction, Parity Bits, Hamming Codes, Cyclic Redundancy Check (CRC), Flow Control Protocols, Unrestricted Simplex, Stop-and-Wait, Sliding Window Protocol, MAC Sub-layer: Multiple Access Protocols, Pure and Slotted ALOHA, Carrier Sense Multiple Access/Collision Avoidance (CSMA), Carrier Sense Multiple Access with Collision Detection, and CSMA/CA for Carrier Sense Multiple Access with Collision Avoidance, Introduction to Ethernet: Institute of Electrical and Electronics Engineers (IEEE) 802.3, IEEE 802.11 a/b/g/n, IEEE 802.15 Standards.

Unit VI: Network Security and Recent Trends in Communication Networks

Introduction to Network Security, Security Mechanism, Need and Services, Introduction to Classical Cryptography and Its Types, Types of Advanced Threats, Malware, Ransomware, Distributed Denial-of-Service (DDoS) Attacks, Strategies for Mitigating These Threats, Best Practices.

Text Books

1. Kurose, Ross 'Computer Networking a Top Down Approach Featuring the Internet' Pearson, (8th Edition) , (2022).
2. Forouzan B 'Data Communication and Networking with TCP IP protocol suite', Tata McGraw Hill, (6th Edition), (2022).

Reference Books

1. Andrew S Tanenbaum, David J Wetherall,'Computer Networks', Pearson, (5th Edition), (2014)
2. Stallings W 'Data and Computer Communications' Prentice Hall Pvt.Ltd. (8th Edition), (2009).
3. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff 'Unix Network Programming Volume 1', Addison-Wesley Publication, (3rd Edition), (2003).

4. Geoffrey C. Fox, Jack Dongarra, and Kai Hwang,'Distributed and Cloud Computing' Morgan Kaufmann, (1st Edition),(2011).
5. Stallings W, 'Cryptography and Network Security: Principles and Practice', Pearson, (6th Edition), (2014).

Online/Web/Other References

1. nptel.ac.in/courses/106/105/106105183
2. nptel.ac.in/courses/106/105/106105081
3. nptel.ac.in/courses/106/106/106106091
4. nptel.ac.in/courses/106/105/106105031

23PCCE504 DESIGN AND ANALYSIS OF ALGORITHMS

Teaching Scheme	**Examination Scheme**
Lecture: 3 Hours/Week	In Semester: 50 Marks
	End Semester: 50 Marks
	Credits: 3

Prerequisite

1. Data Structures
2. Discrete Mathematics and statistics

Course Objectives

To facilitate the learners

1. Understand and apply methods of analysis of algorithms
2. Learn and apply the concept of computational complexity classes for the given problem
3. Learn and apply strategies for designing the algorithms
4. Get acquainted with the concept of advanced algorithms design

Course Outcomes

After completion of the course, students will be able to

1. Apply algorithm analysis techniques to recursive and non-recursive algorithms
2. Classify problems into complexity classes by analyzing time complexity
3. Examine algorithm design techniques for solution of a problem
4. Understand the design requirements of advanced algorithms

UNIT I: Introduction

Fundamentals of Algorithmic Problem Solving, Mathematical analysis for Recursive and Non-recursive algorithms, Towers of Hanoi problem, Solving Homogeneous and non-homgeneous recurrences, Substitution method, Recursion-tree method, Master method.

UNIT II: Computational Complexity Classes

Basic Concepts of complexity classes, Non deterministic algorithms, The classes P and NP, NP Complete and NP Hard, Decision problems, Clique Decision problem, Node cover Decision problem, Directed Hamiltonian Cycle Problem, Satisfiability problem, Reducibility problem, NP Hard problems.

UNIT III: Divide-and-Conquer and Greedy Strategy

Decrease-and-Conquer, insertion sort, Topological sort, Divide and Conquer General Strategy, Control Abstraction, min/max problem, Binary Search, Quick Sort, Strassen's algorithm for matrix multiplication, Greedy Method: General strategy, control abstraction, fraction Knapsack problem, Job sequencing with Deadlines (Scheduling Algorithm), Minimal Spanning Tree algorithms(Graph Based Algorithm).

UNIT IV: Dynamic Programming

Dynamic programming general strategy, Control Abstraction, Principle of optimality, Multi stage graphs, Chained matrix multiplication, All pairs shortest path problem, Optimal Binary Search Tree problem, 0/1 Knapsack problem, Travelling Salesperson Problem, Assembly line scheduling problem, Longest Common Subsequence.

UNIT V: Backtracking and Branch-and-Bound

Backtracking General Strategy, Implicit and Explicit constraints, DFS State space tree formulation, Backtracking applications, Sum of subsets, Hamiltonian Cycle problem/Graph colouring problem, 4/8 Queens problem, Maze problem, Tower of Hanoi problem, 15-puzzle problem, solving sudoku, Branch and Bound General Strategy, BFS state space tree formulation, Traveling Salesperson Problem.

UNIT VI: Advanced Algorithms

Parallel Algorithms for Matrix Multiplication and Sorting, Genetic Algorithms, Consensus Algorithms (e.g. Blockchain), Geometric Algorithms (e.g., Graham Scan for Convex Hull), Linear Programming (e.g., Simplex algorithm), Approximation algorithms (e.g., for Knapsack, Traveling Salesperson), String matching algorithm (e.g. Rabin Karp Algorithm, Knuth-Morris-Pratt Algorithm)

Text Books

1. Ellis Horowitz, Sartaj Sahni and Sanguthevar Rajasekaran, "Fundamentals of Computer Algorithms", 2nd edition - 2018, Computer Science Press, New York,
   ISBN: 0-7167-8316-9
2. Gilles Brassard and Paul Bartley, "Fundamental of Algorithmics", Prentice Hall of India , 2010, ISBN:9788120311312
3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein,"Introduction to Algorithm", 4th edition - 2022, The MIT Press, ISBN: 9780262046305

Reference Books

1. Fayez Gebali, “Algorithms and Parallel Computing”, Willy, 2015, ISBN 9788126553891
2. Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, 2014, Pearson Education
3. A. V. Aho and J.D. Ullman, "Design and Analysis of Algorithms", Pearson Education, 2006, ISBN: 978 81 317 0205 5
4. Parag Himanshu Dave, Himanshu Bhalchandra Dave, “ Design And Analysis of Algorithms”, PEARSON Education, ISBN 81-7758-595-9

Online/Web/Other References

1. https://nptel.ac.in/courses/106/101/106101060/
2. https://www.mooc-list.com/course/algorithms-design-and-analysis-part-1-coursera
3. https://www.cse.iitb.ac.in/~akshayss/courses/cs310-2019/index

23PECE501A DIGITAL IMAGE PROCESSING

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

Course Objectives
To facilitate the learners to

1. Understand basic concepts of digital image processing
2. Learn enhancement, restoration and reconstruction techniques used for low quality images
3. Be aware of object Recognition using various segmentation methods
4. Realize the significance of image compression techniques in image processing applications
5. Understand the modern advancements in image processing

Course Outcomes
After completion of the course, students will be able to

1. Illustrate the fundamental techniques and algorithms used for image processing
2. Select the image enhancement, restoration and reconstructions techniques to improve image quality
3. Apply the different compression techniques on images
4. Choose image segmentation techniques on given images for object detection
5. Make use of the advances in image processing to real life applications

UNIT I: Introduction to Image Processing

Fundamental steps and component of the image processing system, Representation of digital images, Basic relationships between pixels, Introduction to Human Visual System, Image sensing and acquisition, Image modalities, Basic concepts in sampling and quantization, Basic operations: Convolution, Arithmetic and Logical Operations.

UNIT II: Image Enhancement Techniques

Image Pre-processing - Contrast enhancement, simple noise reduction, Some basic gray level transformations Spatial Domain Filtering, Histogram Processing, Histogram Equalization, Histogram stretching, Correlation and convolution, Smoothing filters, sharpening filters, gradient and Laplacian Frequency Domain Filtering, Introduction to Fourier Transform, Applications of a Fourier Transform, Discrete Fourier Transform (DFT).

UNIT III: Image Restoration and Reconstruction

Image deformation and geometric transformations, Basic Framework Restoration, Restoration techniques, Noise characterization, Noise restoration filters, Adaptive filters, Image Reconstruction Techniques, Analytical and Iterative Reconstruction.

UNIT IV: Image Compression Techniques

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

UNIT V: Image Segmentation, Analysis and Object Recognition

Point, Line and Edge Detection, Classification of image segmentation techniques, Edge-based Segmentation, Region based techniques, Binarization, Global Thresholding, Adaptive thresholding. Types of Edge detector, Derivative filters-Prewitt, Sobel, Canny, Introduction to Object Recognition, Object Representation (Signatures, Boundary Skeleton), Simple Boundary Descriptors, Regional descriptors-Topological feature (Texture), Morphological Operations, Basics of Set Theory, Dilation and Erosion-Dilation, Erosion, Structuring Element, Opening and Closing.

UNIT VI: Advances in Image processing Applications

Applications of Image Processing in Healthcare, Remote Sensing, Satellite Imaging, Entertainment Artificial Intelligence Techniques for Image Enhancement, Image Analysis, Pattern Generation and Object Detection, Introduction to Video processing.

Text Books

1. R.C.Gonzalez, R.R. Woods, 'Digital Image Processing',ISBN-13 : 978-9353062989, Pearson Education; Fourth edition (July 2018); Pearson India
2. Sridhar S. 'Digital Image Processing', Oxford University Press, (Second Edition), (2016)
3. S. Jayaraman, S. Esakkirajan, T. Veerakumar , 'Digital Image processing', ISBN 978-0-07-014479-8, Mcgraw Hills Publication (Tenth reprint),(2013)

References Books

1. Milan Sonka, Hlavac, Boyle, 'Digital Image Processing and Computer Vision', ISBN 978-81-315-0555-7,Cengage Learning (Sixth Indian Reprint) ,(2011)
2. Sandipan Dey, 'Hands-On Image Processing with Python: Expert techniques for advanced image analysis and effective interpretation of image data', ISBN-13: 9781789341850, Packt, (Nov. 2018)
3. B. Chanda, D.Datta Mujumdar 'Digital Image Processing And Analysis', PHI, ISBN 978-81-203- 4325-2, (Second Edition),(2013)
4. Anil Jain, 'Fundamentals of Digital Image Processing', PHI, ISBN-81-203-0929-4 (Indian Reprint), (1995)
5. Basudeb Bhatta ' Remote Sensing and GIS', Oxford University Press, I978-0-19-807239-3 (Second Edition)(2014)

Online/Web/Other References

1. NPTEL series – https://archive.nptel.ac.in/courses/106/105/106105032/
2. NPTEL series – https://onlinecourses.nptel.ac.in/noc22_ee116/preview
3. NPTEL series – ://httpsarchive.nptel.ac.in/courses/108/103/108103174/

23PECE501B ADVANCED OPERATING SYSTEM

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

Prerequisites

1. Data Structures
2. Operating Systems
3. Digital Systems and Computer Organization

Course Objectives
To facilitate the learner to

1. Understand the Operating system booting process and advance file system concepts
2. Learn init() process and multithreading concepts
3. Learn Security and Protection issues with respect to an operating system
4. Explore the advanced concepts of operating systems

Course Outcomes
By taking this course, the learners will be able to

1. Apply design aspects of kernel for operating system services
2. Make use of multithreading concepts to solve synchronization problems
3. Utilize the concepts of Security and Protection for virtualization in operating system
4. Interpret the architecture and design issues of mobile operating system
5. Understand the advancements in operating system

Unit I : Overview of Operating systems and Kernel

Multiprocessor Operating Systems, System Architectures, Structures of OS, OS design issues Introduction to kernel, Types of kernel (monolithic,micro,exo), Operating system booting process, Buffer management, Buffer Cache, Internal Representation of Files, Systems calls for the files systems, File management, Access methods, Free Space Management and Swap space, Disk management, Process Management, Memory Management , I/O Management.

Unit II : Inter-Process Communication and Synchronization

init() process, Signals, Pipes and Named Pipes (FIFOs), Threads and pthread library, Multithreading models (M to 1, 1 to 1, M to M model), Threading issues, Fork() and exec() system calls, Thread Cancellation, Thread specific data, Producer-Consumer Problem and Solutions using Pipes , Init Process in Android Mobiles, Device Drivers, Concepts and Implementation Details.

Unit III : Security and Virtualization in OS

OS Security Models and Access Control, Authentication, Authorization, and Encryption, Threats, Vulnerabilities, Security Policies and Protection Mechanisms, Virtual Machines and Hypervisors, Virtualization and Types of Virtualization, Containerization and Docker, Resource isolation and allocation in virtual environments, Virtualization Security.

Unit IV : Mobile Operating System

Introduction to mobile operating system, mobile operating system architecture, power management and approaches to power management, kernel structure, Run time issues, Mobile OS security architecture, Permissions and sandboxing, case study of Android OS.

Unit V: Case study and Advancements in Operating Systems

Distributed OS, Network OS, Real Time Operating Systems (RTOS), Machine Learning and AI in OS, Operating Systems for Embedded Systems.

Text Books

1. Silberschatz, Galvin, Gagnes, "Operating System Concepts", John Wiley & Sons, (8/e), ISBN: 9971- 51-388-9.
2. William Stallings, "Operating System-Internals and Design Principles ", Prentice Hall India, (5/e) ISBN: 81-297-0 1 094-3.
3. Maurice J. Bach, "The Design of the Unix Operating System", Pearson Education, ISBN: 81-7758- 770-6.

Reference Books

1. Pramod Chandra P. Bhatt "An Introduction to Operating Systems: Concepts and Practices", 4th edition, PHI, ISBN 978-81-203-4836-3
2. Milan Milenkovic, "Operating Systems", TMH, (2/e), ISBN: 0-07-044700-4.
3. Andrew S. Tanenbaum, "Modern Operating Systems", Prentice Hall India, (4/e), ISBN: 81-203- 2063-8.

23PECE50C USER EXPERIENCE DESIGN (UX/UI)

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

Course Objectives
To facilitate the learner to

1. Understand the basic concepts of user interface/user experience Design in order to design with intention
2. Achieve a deep understanding of the entire lifecycle of the design process
3. Provide a visual understanding of the product to make user interaction as easy and efficient as possible
4. Understand various design technologies for mobile and web to help avoid common mistakes and meet user requirements

Course Outcomes
After completion of the course, students will be able to

1. Analyse the relationship between user interface and user experience to create user-centered designs
2. Apply various user experience research methods to improve user experience
3. Make use of various design principles to create intuitive wireframes and prototypes
4. Understand mobile and web-based user experience designs by integrating emerging interaction technologies

Unit I: Introduction to User Experience

User Experience, Relationship between user interface (UI) and user experience (UX), Understanding User Experience, importance of design, Overview of the design process, Usability, Design Psychology and User Perception, Mental Model, Cognitive Model in UX, The role of emotions in user interactions, Universal Design, User centered design, Addressing inclusivity, accessibility (WCAG guidelines), and ethical considerations like dark patterns and data privacy.

Unit II: UX Research and Analysis

User requirements, types of user research, Qualitative research, Quantitative research. Tools to collect user requirements – personal observation, interviews, questionnaire, User/Expert reviews User requirement analysis, Understanding target audience and client requirements, Competitive analysis, Affinity mapping, Creating user Personas and User Journey Maps, Information Architecture, Heuristic evaluation and usability testing.

Unit III: User Experience Design Process

Design principles and rules, Shneiderman's golden rules, Normans seven principles, Nielsens ten heuristics with example of its use, Gestalt principles of design - Aesthetics in UI design, Visual Communication Design - effective visual communication for graphical user interface. UX Design Tools, atomic design, component-based UI, Design Systems, Designing for diverse audiences, localization and globalization in UI/UX.

Unit IV: Wireframing and Prototyping

Wireframing, Purpose, Low-fidelity and high-fidelity, Basic considerations in wireframing-device, size, behaviour, interaction, Prototyping basics Interactive prototypes and user flows Creating wireframes using tools (Figma, Adobe XD, Sketch, Balsamiq), Usability tests, heatmaps, and analytics.

Unit V: UX Design for Mobile and Web

Important Differences from the Desktop. Smartphone and Tablet, UI mobile components and patterns, Elements of Mobile Design, Web user Interface, Responsive and adaptive design, Mobile-first approach, Designing for iOS and Android (Material Design & Human Interface Guidelines), Best practices for web and mobile usability, Using game mechanics like rewards, progress indicators, and social proof to improve user engagement, The role of language in UI, writing clear error messages, tooltips, and onboarding instructions to enhance user experience.

Unit VI: Interaction Technologies and Trends

Zero UI, Explicit and Implicit Human Computer Interaction – Gesture interfaces, Speech Recognition, Tangible interfaces, Auditory Interfaces, Natural Language Interfaces, User Interfaces and Interaction for Four Widely Used Devices. Hidden User Interface via Basic smart Devices, Hidden User Interface via Wearable and Implanted Devices, Virtual and Augmented Reality. Case studies of successful UI and UX designs in AI applications. AI-driven design solutions like personalization, chatbots, and predictive UI elements.

Text Books

1. Interaction Design: Beyond Human-Computer Interaction: Book by Helen Sharp Jenny Preece, and Yvonne Rogers, ISSN1119547253, 5th Edition, Willey Publication (2019)
2. Wilbert O. Galitz'Wiley The Essential Guide to User Interface Design' 3rd Edition Apr 2007

Reference Books

1. Don Norman, 'The Design of Everyday Things', Basic Books, A member of the Perseus Books Group, (2013)

2. Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, 'Designing the User Interface: Strategies for Effective Human-Computer Interaction', Pearson Education Limited (India),(2010)

3 "Designing Interfaces: Patterns for Effective Interaction Design" by Jenifer Tidwell

4 "Design for Cognitive Bias" by David Dylan Thomas

Online/Web/Other References

1. https://www.interaction-design.org/courses/user-experience-the-beginner-s-guide
2. https://www.coursera.org/learn/user-experience-design#syllabus

23PECE501D INTERNET OF THINGS

Teaching Scheme	**Examination Scheme**
Lectures: 3 Hours/week	In semester: 50 marks
	End semester: 50 marks
	Credit: 3

Prerequisite: Microprocessor and Microcontroller

Course Objectives
To facilitate the learners to

1. To understand the Internet of Things with different components and its design process
2. To design a small real time system for Internet of Things using different devices
3. To understand security mechanisms to provide security for integration of various components
4. To understand the integration of Internet of Things with cloud services and new emerging technologies

Course Outcomes
By taking this course, the learner will be able to

1. Understand the components and design process of Internet of Things
2. Examine Internet of Things devices for real time applications
3. Select security mechanism for Internet of Things ecosystem
4. Make use of cloud services and emerging technologies for Internet of Things

Unit I: Introduction to Internet of Things

Definition and characteristics of Internet of Things (IoT), Vision, Ecosystem, Reference Model, Physical Design Model, Logical Design, Functional Block, Communication models, Communication API's, IoT enabling Technologies, IoT Levels and Deployment Templates, Applications of IoT, IoT & Machine to Machine (M2M), IoT platform Design methodology, Steps involved in IoT system Design methodology.

Unit II: Embedded Devices and Programming for Internet of Things

Sensors, Actuators and Devices, Introduction to Arduino, Beagle Bone Black, Raspberry Pi, Python Programming for IoT devices, IoT Simulation Platforms like Tinker cad, IoTIFY, MATLAB, NetSim, BevyWise IoT Simulator, NS-3, Cooja.

Unit III: Internet of Things Security

Security risks in IoT systems like data breaches, unauthorized access and malware, different layer IoT security protocols- Message Queue Telemetry Transport (MQTT), Extensible Messaging and Presence Protocol (XMPP), Data Distribution Services (DDS), Advanced Message Queuing Protocol (AMQP), Constrained Application Protocol (COAP), Representational State Transfer (REST), Encryption methods used for IoT systems, Authentication and access control for devices, Privacy concerns and legal implications, Case studies- Home automation, Smart cities, Agriculture.

Unit IV: Internet of Things - Cloud Synergy

Introduction to cloud computing in IoT, Cloud services for IoT, Data storage and management, Cloud-to-Device (C2D) and Device-to-Cloud (D2C) communication, IoT Data management on Cloud, Case studies- IoT in smart homes, connected vehicles, Industrial Automation.

Unit V: Recent Trends in Internet of Things

IoT integration with emerging technologies like Edge computing, Digital Twins, 5G, Blockchain, Industry 4.0, Artificial Intelligence, Future of IoT, Smart Agriculture, Smart Irrigation, Autonomous Systems, Wearables Devices.

Textbooks

1. Arshdeep Bagha, Vijay Madisetti, 'Internet of Things – A Hands-on-approach', Universities Press (2014).
2. Srinivasa K.G., Siddesh G.M., Hanumantha Raju R., 'Internet of Things',Cengage Publication.
3. Rajkamal, 'Internet Of Things: Architecture and Design Principles' McGraw Hill Education (India) Private Limited.

Reference Books

1. Ovidiu Vermesan, Peter Friess, 'Internet of Things – Converging Technologies for Smart Environments and Integrated Ecosystems', River Publishers.
2. Honbo Zhou, 'The Internet of Things in the Cloud', CRC Press(2013).
3. Peter Waher, 'Learning Internet of Things', Packt Publishing (2015).

23MmCE501 ENTERPRISE RESOURCE PLANNING

Teaching Scheme	**Examination Scheme**
Lectures: 2 Hours/Week	In Semester : 25 Marks
Tutorial: 1 Hour/Week	End Semester: 25 Marks
	Credit: 3

Prerequisites: Enterprise Information System

Course Objectives
To facilitate the learner to

1. Understand the fundamentals of Enterprise Resource Planning (ERP)
2. Explore ERP-related technologies, tools and their applications
3. Learn the benefits and implementation lifecycle of ERP systems
4. Understand real-world ERP applications and trends

Course Outcomes
After completion of the course, students will be able to

1. Understand the fundamental concepts of Enterprise Resource Planning (ERP)
2. Identify relevant ERP modules related to specific business applications
3. Apply the phases of the ERP implementation lifecycle to specific business applications
4. Utilize knowledge of various ERP tools, technologies and trends to support data-driven decisions

Unit I: Introduction

Enterprise, Enterprise Resource Planning (ERP), Evolution of ERP, ERP Architecture and Components, Various Modules of ERP, Integrated Enterprise applications, Advantages and Challenges, Reasons of ERP failure, Benefits of ERP

Unit II: ERP and Related Technologies

Business Process Reengineering (BPR), Product Lifecycle Management (PLM), Supply Chain Management (SCM), Customer Relationship Management (CRM), Financial Management, Plant Maintenance, Quality Management, Materials Management, Enterprise Decision Support System Case Studies : ERP in Banking, SAPR / 3 at Tata Steel, Pantaloon, Automotive company

Unit III: ERP Implementation Lifecycle

Different phases of ERP implementation lifecycle-Pre-evaluation Screening, Package Evaluation, Project Planning Phase, Gap Analysis, Reengineering, Configuration, Implementation Team Training, Testing, Going Live, End-user Training, Post-implementation (Maintenance mode).

Unit IV: Emerging Tools and Trends in ERP

ERP Tools like ERPNext, Odoo, Apache OFBiz, JD-Edwards-EnterpriseOne, Microsoft Dynamic CRm-Module, Salesforce

Emerging Trends- AIML in ERP, BlockChain and IOT in ERP, Cloud-based ERP solutions, Big Data and Analytics in ERP

Text Books

1. Alexis Leon, Enterprise Resource Planning, Tata McGraw Hill, 4th Edition, 2019
2. Rajesh Ray, Enterprise Resource Planning, McGraw Hill Education, 1st Edition, 2017

Reference Books

1. Veena Bansal, Enterprise Resource Planning- A managerial perspective, Pearson, 2013
2. V. K. Garg and N. K. Venkitakrishnan, Enterprise Resource Planning: Concepts andPractice, Second Edition, 2011
3. Ashim Raj Singla, Enterprise Resource Planning, Cengage Learning India Pvt. Ltd., New Delhi, 2008

Web References:

1. https://www.scribd.com/document/261279208/Emerging-Trends-in-ERP-Technology
2. ERPNext Documentation: https://docs.erpnext.com
3. Odoo Official Docs: https://www.odoo.com/documentation
4. Apache OFBiz: https://ofbiz.apache.org
5. Salesforce Trailhead (Free learning): https://trailhead.salesforce.com
6. Microsoft Dynamics CRM Docs: https://learn.microsoft.com/en-us/dynamics365

Suggestive list of Tutorials

1. Explore various ERP systems - Enterprises, modules, tools used : Observations in terms of Benefits, limitations, design constructs (GUI etc.)
2. Identify one real-world ERP implementation failure. Identify the probable reasons for failure. Highlight "gaps" and suggest solutions.
3. Use Excel or GanttProject to create a Gantt chart covering ERP implementation phases (12 months).
4. Model a product's supply chain management module from raw material to end product creating customer records and simulating lead generation.
5. Design the ERP system which focuses on managing the customer relationship lifecycle, starting from lead generation to sales closure. The process involves identifying and engaging potential clients who may benefit from ERP solutions and systematically guiding them through the sales funnel. Effective lead generation strategies include

content marketing, social media campaigns, email marketing, and offering valuable webinars or demos.

6. Design a basic balance sheet and monthly budget (using Excel) which include income, expenses, assets, and liabilities. Design a ERP system for a mid-sized enterprise to understand how core financial operations are managed through the ERP's Financial module. As part of the preparatory analysis, manually simulate key financial processes such as journal entry creation, ledger posting, and trial balance preparation for common transactions like capital infusion, asset purchases, credit sales, payments, and receipts.
7. Design a decision support dashboard assuming emerging trends and technologies of ERP.
8. Presentations of ERP case studies.

23PCCE501L ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING LABORATORY

Teaching Scheme	**Examination Scheme**
Practical: 2 Hours/Week	In Semester: 25 Marks
	End Semester: 25 Marks
	Credits: 1

Prerequisite

1. Programming Skills Development Laboratory

Course Objectives

To facilitate the learner to

1. Implement some AI based heuristics techniques, pre-processing operations on given data
2. Implement supervised machine learning algorithms such as regression, classification
3. Implement unsupervised machine learning algorithms such as clustering, association rule mining method
4. Implement artificial neural networks and its learning algorithms
5. Implement a small machine learning application and evaluate the performance of the designed machine learning model

Course Outcomes

After completion of the course, students will be able to

1. Apply heuristics search techniques to solve the given problem
2. Make use of pre-processing operations on given data
3. Utilize classification and regression machine learning techniques to solve the problem
4. Experiment with clustering and association rule mining techniques of machine learning to solve the problem
5. Develop a machine learning application using different Artificial Intelligence or Machine learning techniques

Preamble

A large part of the lab would be for understanding the basic concepts of artificial intelligence, machine learning and implementation of some real-world simple applications. Assignment statements are in brief and should be implemented in JAVA/Python programming language. Motivation here is that students should be able to code the basic algorithm and also should be able to make use of built-in functions available in different libraries of Java/Python. Faculty members are encouraged to expand problem statements with variations. Assignments can be framed and expanded in such a way that it explores concepts, logic of solution and simple application. Students will be encouraged to solve open problems in different domains. Faculty will appropriately adopt assignments on similar lines as the examples shown here. Group A assignments are on artificial intelligence, pre-processing data, supervised learning methods such as classification and regression, and simple logic gates implementation using artificial neural networks. Group B assignments are on unsupervised learning and Group C assignment is on case study implementation for different applications.

Suggestive List of Assignments

Group A: (Mandatory)

Explore language used for Machine Learning Python/Java and perform the following operations: Understand the basic functionality, visualization of data. Study the different file format, explore the available data sets and its usage using programming language.

1. Robots that need to manage liquids (like in a lab) to perform tasks efficiently. Use the water Jug algorithm to solve this and test it for all test cases.
2. Solve a puzzle like the 8-puzzle by representing the puzzle state as a graph and using A* / best first to find the sequence of moves leading to the solved state.
3. Suppose that the data for analysis includes the attribute age (a) Plot an equal-width histogram of width 10. (b) Sketch examples of each of the following sampling techniques: SRSWOR, SRSWR, cluster sampling, and stratified sampling. Use samples of size 5 and the strata "youth," "middle-aged," and "senior." (c) scale and also normalize the data. Pre-process the given data as given here.
4. BMI and body fat of persons are given. Use this BMI to predict the body fat of a person. Implement predictive modelling using regression analysis using a programming language that you are familiar with such as Java/Python. Fit the model and predict the value for a given problem.
5. Data given for the SPEED and AGILITY rating of 20 college athletes and whether they were drafted by a professional team. Implement k-nn classification technique of Machine learning using a programming language that you are familiar with such as Java/Python. Compare the performance of classification by changing the value of k for the given data.

Group B: (Any Two)

1. You have a list of shopping items purchased by many people. Find out what are the frequently purchased combination of 2 items. Implement Apriori, a Frequent Pattern Analysis algorithm using Java/Python.
2. A Hospital Care chain wants to open a series of Emergency-Care wards within a region. We assume that the hospital knows the location of all the maximum accident-prone areas in the region. They have to decide the number of the Emergency Units to be opened and the location of these Emergency Units, so that all the accident-prone areas are covered in the vicinity of these Emergency Units. The challenge here is to decide the location of these Emergency Units so that the whole region is covered. Implement a 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 by changing input parameter value such as K.
3. Implement AO* algorithm/ minimax algorithm of game playing.

Group C: (Any One)

1. Develop a machine learning case study for readily available data sets using the techniques studied, and evaluate the designed and implemented model.
2. Develop a game using Artificial Intelligence.

23PCCE503L COMPUTER NETWORKS LABORATORY

Teaching Scheme	**Examination Scheme**
Practical: 2 Hours/Week	In Semester: 25 Marks
	End Semester: 25 Marks
	Credits: 1

Course Objectives

To facilitate the learner to

1. Learn computer network topologies and types of network
2. Use modern tools for network traffic analysis and various networking configurations
3. To learn network programming
4. To develop an understanding of various protocols, modern technologies and applications

Course Outcomes

After completion of the course, students will be able to

1. Demonstrate error and flow control techniques
2. Build networks by configuring switches and routers
3. Experiment with LAN and WAN protocol behavior using modern tools
4. Develop Client-Server architectures and prototypes

Preamble

This lab is designed to provide students with an overview of the concepts and fundamentals of computer networks. The lab includes: communication concepts and techniques in a layered network architecture, routing, network congestion, network topologies, network configuration and management, network model components, layered network models (such as the OSI reference model and TCP/IP networking architecture) and their protocols, as well as various types of networks (LAN, MAN, WAN, and wireless networks) and their protocols. Students are expected to complete any 4–5 assignments from Groups A and B, and any 1–2 assignments from Group C.

Suggestive List of Assignments

Group A: Mandatory

1. Demonstrate the different types of topologies and types of transmission media by using a packet tracer tool.
2. Design an IP scheme for a WAN network (minimum 3 networks) using Cisco Packet Tracer tool (Static Routing).
3. Set Up a WAN which contains wired as well as wireless LAN by using a packet tracer tool. Demonstrate transfer of a packet from LAN 1 (wired LAN) to LAN2 (Wireless LAN).

4. Write a program for error detection and correction for 7/8 bits ASCII codes using Hamming Codes or CRC.
5. Simulate Go back N and Selective Repeat Modes of Sliding Window Protocol in Peer-to-Peer mode.

Group B: (Any Four)

1. Use packet Tracer tool for configuration of 3 router network using one of the following protocol RIP/OSPF
2. Write a program using TCP/UDP socket for wired network for following a. Say Hello to Each other b. File transfer c. Calculator
3. Write a program for DNS lookup. Given an IP address as input, it should return URL and vice versa.
4. Write a program to demonstrate Sub-netting and find subnet masks.
5. Configuring Ftp server for file upload /download using Cisco Packet Tracer.
6. Demonstrate the TCP protocol (see packet drops, re-transmissions, Window size reductions & growths etc) Using tools like tc, iperf3 and similar tools

Group C: (Any One)

1. Setup a wired LAN. It should include preparation of cables, testing of cable using line tester, configuration machine using IP addresses, testing of the network and demonstrating the packets captured traces using suitable open source tools like the Wire shark Packet Analyzer Tool.
2. Creation and configure a network of Virtual Machines. eg: create 2 local virtual machines on host and ping the Virtual Machine.
3. Simulate a VANET scenario using suitable modern tools like NS3 or SUMO.

23MmCE501L ENTERPRISE RESOURCE PLANNING LABORATORY

Teaching Scheme	**Examination Scheme**
Practical: 2 Hours/Week	In Semester: 25 Marks
	End Semester: 25 Marks
	Credits: 1

Course Objectives
To facilitate the learner to

1. Understand the fundamental concepts and related technologies of ERP
2. Learn the benefits of ERP and understand the working of various modules of ERP
3. Understand the implementation life cycle of ERP
4. Learn and explore various ERP tools

Course Outcomes
After completion of the course, students will be able to

1. Understand open-source ERP tools, their architecture and functionalities
2. Design an ERP system defining business requirements, specific business processes, dataflows, user friendly interfaces and dashboards
3. Build relevant ERP modules such as Customer Relationship Management (CRM), Supply Chain Management (SCM) modules in an open-source ERP environment to simulate real-world enterprise
4. Develop ERP dashboards using ERP tools to visualize purchases, sales, inventory, finance, customer relationships, human resources and all relevant reports

Preamble
Enterprise Resource Planning (ERP) laboratory will be conducted using project-based learning. A group of 4 to 5 students will choose a specific business application for which they will design and implement all relevant business processes, dataflows, user friendly interfaces and dashboards using Open-Source ERP tools like ERPNext, Odoo, ERP5 etc.

Suggestive List of Assignments

Project Based Learning ERP System

A group of 4 to 5 students will choose a specific business application for which they will design and implement all relevant business processes, dataflows, user friendly interfaces and dashboards using Open Source ERP tools like ERPNext, Odoo, ERP5 etc.

1. Create Developer login : Get to know the UI, CRM flow configuration, User mode Vs Admin /Configuration mode, Lightning UI Vs Classic UI, Customize lead status to DemoPlanned, DemoDone.
2. Database modeling (for example, project is ERP Vendor selection for Cummins College): Rename Account Standard object label to ERP Vendor, Customize list view columns, Create a custom object erpProduct, with Tab, Add a currency field MaxPrice, Create relationship field of lookup/masterDetail type on erpProduct, Create an Application named cumminsERP and add tab, Many to Many Relationship, SOQL and executing queries using developer console.
3. Apex Programming Language Part I and Security Considerations:

   a. Standard Profiles and custom profile (create from standard platform profile) : Configure this custom profile so that it can not see one custom field (MaxPrice) on the custom object created (erpProduct), Achieve similar effect for another field using Layout-Profile combination, Create one user with custom Standard Profile and login with this user to see how these security restrictions work.

   b. Simple Apex class using Developer Console : Illustrate a simple Crud operation, AddProduct, EditProduct, DeleteProduct, Create a TestClass for the Apex Class
4. Apex Programming Language Part II – Triggers : Develop a Licensing pricing scheme using trigger (Add a custom object Student) as

   up to 200 total students 100 Rs / Student / Year,

   200 – 500 total students 80 Rs / Student / Year,

   500 and more students 70 Rs / Student / Year
5. Flows part I : Create an Approval flow so that, If a particular vendor invoice amount &lt; 50,000 only HOD approval, If a particular vendor invoice amount &gt; = 50,000 INR both HOD and Principal approval
6. Flows part II: Implement Licensing scheme using flow
7. SDLC and DevOps: Managed Package, UnManaged Package, DEMO only as a licensed org is needed for the purpose. Illustrate creation of Sandboxes, Managed vs unmanaged package creation and deployment, Case where data can be migrated.
8. Service Cloud : Customer creates a complain by 2 ways:

   a. Calls support person and he or she creates a record on Case object (make sure Support cloud app is selected using App Launcher)

   b. WebToCase feature where customers fill a form on a website and a case is created.
9. Demonstration of Marketing cloud
10. Latest trends Salesforce – AgentForce

23PECE501AL DIGITAL IMAGE PROCESSING LABORATORY

Teaching Scheme	**Examination Scheme**
Practical: 2 Hours/Week	In Semester: 25 Marks
	End Semester: 25 Marks
	Credits: 1

Course Objectives

To facilitate the learners to

1. Learn and use different Image Processing tools
2. Learn Basics Image Processing operations on different image file formats
3. Understand and apply methods used for image enhancement, restoration and reconstruction
4. Extend the image processing knowledge to a real world problem

Course Outcomes

After completion of the course, students will be able to

1. Experiment with modern image processing tools
2. Apply fundamental operations on a given image
3. Make use of enhancement, restoration and filtering techniques on images to improve quality
4. Develop advanced image processing techniques on image dataset

Preamble

A Digital Image Processing Laboratory will enhance the basic understanding of digital images and image computational techniques. It will also equip the students for analysis, enhancement, and manipulation of digital images using computational techniques. By utilizing software tools like OpenCV, PIL and other libraries from Python, the laboratory enables hands-on experience in real-world applications such as medical imaging, satellite imagery, remote sensing, autonomous vehicles, security and surveillance, etc. Project Based Learning will engage students in solving complex, practical problems in the area of their choice. By applying AI/ML techniques to images, students will be able to experience enhancement in automation, accuracy, and efficiency in analyzing and interpretation of images.

Suggestive List of Assignments

Mandatory Assignments

1. Study different file formats and write a program to create a simple image file and display it.
2. Write a program to perform Arithmetic, Logical and intensity transformation operations.
3. Write a program to perform geometric transformations on images.

Download or acquire noisy, unclean image dataset from area of interest. Analyze the issues in the selected dataset. Perform following operations on the selected images. Analyze the results for chosen image processing problems.

1. Perform image enhancement using Histogram and filtering operations.
2. Utilize Morphological transformations for image enhancement.
3. Apply suitable Image Restoration techniques.
4. Make use of image segmentation methods for object detection.
5. Experiment with suitable AI/ML based techniques on the chosen image dataset.

23PECE501BL ADVANCED OPERATING SYSTEM LABORATORY

Teaching Scheme	**Examination Scheme**
Practical: 2 Hours /Week	In Semester: 25 Marks
	End Semester: 25 Marks
	Credits: 1

Prerequisite

1. Data Structures Laboratory
2. Operating Systems Laboratory

Course Objectives

To facilitate the learner to

1. Understand basic commands of LINUX Operating System
2. Understand advanced operating system concepts for File Management, socket, and device drivers
3. Learn concepts of virtualization
4. Understand the problems related to operating system design

Course Outcomes

After completion of the course, students will be able to

1. Experiment with fundamental Linux utility commands
2. Develop a program for Character devices, socket, Device drivers
3. Make use of file encryption and access control mechanism in operating system
4. Solve real-world problems related to operating system design

Preamble

The Advanced Operating System Laboratory is designed to provide students with in-depth practical exposure to the internal mechanisms and components of modern operating systems. Operating systems serve as the foundation of all computing environments. This laboratory emphasizes hands-on experience with advanced OS concepts such as file operations, file encryption, socket programming, use of device drivers, IPC using pipes. By implementing these assignments, students will develop a deeper understanding of how operating systems are designed, implemented, and optimized for performance and security. Group A covers core Linux commands, IPC using pipes and implementing basic file security techniques. Group B focuses on system-level programming, including socket programming and device driver development. Group C is an open ended assignment.

Suggestive List of Assignments

Group A : (Mandatory)

1. Exploration of Linux commands - File handling utilities, Process utilities, Disk utilities, Networking commands, Text Processing utilities and Back-up utilities.
2. Write a program to display the i-node information for a given text file, image file.
3. Write a simple file system that supports basic file operations: open(), read(), write(), display ( ), update ( ) , delete ( ) and close().
4. Write a program to implement Producer-Consumer Problem using Pipes.
5. Write a program to implement basic file encryption or access control mechanisms.

Group B: (Any Three)

6. Write a program for implementation of socket in python to share files.
7. Write a device driver for a parallel port.
8. Write a simple kernel module to interact with a character device (e.g., /dev/mydevice).
9. Write a program to implement a basic file system driver as a kernel module.
10. Write a Python code for creating a virtual file system on Linux environment.

Group C: (Any One)

1. Implement Linux Kernel module for Process Scheduling, Memory mgmt, I/O Management.
2. Design and Implementation of a Modular Android Operating System Framework.
3. Design pseudo operating systems.

23PECE501CL USER EXPERIENCE DESIGN LABORATORY

Teaching Scheme
Practical: 2 Hours/Week

Examination Scheme
In Semester: 25 Marks
End Semester: 25 Marks
Credit: 1

Course Objectives
To facilitate the learner to

1. Understand users' needs, experiences, behaviours and goals
2. Learn how visual perception affects the viewing experience
3. Explain Why you made design decisions, through presentations of assignments

Course Outcomes
After completion of the course, students will be able to

1. Explore techniques to understand user, their needs, values, abilities, and limitations
2. Design innovative, user-friendly interfaces for both mobile and/or web applications
3. Evaluate existing interface designs and redesign services and products to elevate the user experience
4. Utilize the industry-standard tools and deliverables essential to professional user interface design practices

Preamble
The Laboratory will be conducted using Project Based Learning Pedagogy. Students can form a group of maximum 4 students and register their project statement. Students can perform following assignments with respect to their problem statement. Students can use any free-ware tools like Figma, Canva or Adobe XD.

Suggestive List of Assignments

1. Tool exploration Adobe XD, Figma
2. Design User persona for the users of selected product / system.
3. Conduct a contextual inquiry for the selected product / system.
4. Design a user journey map highlighting key touchpoints, emotions, and pain points based on the personas created for the selected product / system.
5. Design appropriate icons or Logo and create a user interface for the system using various interaction styles pertaining to the selected product / system.
6. Demonstrate your screen design for diverse users (Rural, Specially abled etc.)
7. Design appropriate wireframes focusing on layout and navigation for the selected product / system.
8. Conduct heuristic evaluation of the prototype developed by your classmates. Give suggestions for redesigning the features of the product. Justify your selection and the changes/design you made.
9. Design a Mobile App/Website or the selected product / system. (Mini Project).

23PECE501DL INTERNET OF THINGS LABORATORY

Teaching Scheme	**Examination Scheme**
Practical: 2 Hours/Week	In Semester: 25 Marks
	End Semester: 25 Marks
	Credit: 1

Course Objectives
To facilitate the learners to

1. Understand various development boards used for Internet of Things (IoT)
2. Learn and understand the fundamentals of sensor-based applications
3. Implement and solve the problems using high level language
4. Develop mini applications on IoT boards with proper design

Course Outcomes
By taking this course, the learner will be able to

1. Model real world problems using Internet of Things (IoT)
2. Design a system using sensor-based components
3. Experiment with Internet of Things on various development boards
4. Develop IoT applications for distributed environment

Preamble
A large part of the lab would be for understanding the basic concepts of Internet of Things, machine learning and implementation of some real-world simple applications. Assignment statements are brief and should be implemented in the Python programming language. Motivation here is that students should be able to code the basic scenario and should be able to make use of built-in functions available in different libraries of Python and design a solution for real time problems. Faculty members are encouraged to expand problem statements with variations. Assignments can be framed and expanded in such a way that they explore concepts, logic of solutions and simple application. Students will be encouraged to solve open problems in different domains. Faculty will appropriately adopt assignments on similar lines as the examples shown here.

Suggestive List of Assignments

Group A: (Mandatory)

1. Study of Raspberry-Pi, Beagle board, Arduino and other microcontrollers.
2. Study of different operating systems for Raspberry-Pi /Beagle board. Understanding the process of OS installation on Raspberry-Pi /Beagle board.
3. Develop an application to read about environmental temperature. If temperature crosses a threshold value, the application indicated user using LEDSs.
4. Understanding the connectivity of Raspberry-Pi /Beagle board circuit with IR sensor. Write code snippets to detect obstacles and notify users using LEDs.
5. Understanding and connectivity of Raspberry-Pi /Beagle board with camera. Write an application to capture and store the image.

Group B: (Any 2)

1. Understanding and connectivity of Raspberry-Pi /Beagle board with a Zigbee module. Write a network application for communication between two devices using Zigbee.
2. Assignments on BeagleBone Black:
   A. Write an application using Beagle board to control the operation of a stepper motor.
   B. Write an application using Beagle board to control the operation of a hardware simulated traffic signal.
   C. Write an application using a Beagle board to control the operation of a hardware simulated lift elevator.
3. Assignments on Cloud of Things:
   A. Write a server application to be deployed on Raspberry-Pi /Beagle board. Write client applications to get services from the server application.
   B. Create a small dashboard application to be deployed on cloud. Different publisher devices can publish their information and interested applications can subscribe.
   C. Create a simple web interface for Raspberry-Pi/Beagle board to control the connected LEDs remotely through the interface.

Group C: (Any 1)
Sample Mini Project Statements:

1. Develop a Real time application like smart home with following requirements: When user enters the house the required appliances like fan, light should be switched ON. Appliances should also be controlled remotely by a suitable web interface. The objective of this application is that students should construct a complete Smart application in groups.
2. Develop a Real time application like a smart home with following requirements: If anyone comes to the door the camera module automatically captures his image, send it to the email account of the user or send notification to the user. The door will open only after the user's approval.