Scheme and Syllabi

of

Four Year Degree Program of

Bachelor of Engineering (B.E)

CSE (AI&ML)

(Based on AICTE Model Curriculum)

Faculty of Engineering

Osmania University, Hyderabad – 500 007

SCHEME OF INSTRUCTION & EXAMINATION
B.E. - V SEMESTER
CSE (AI&ML)

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	P/D	Contact hrs/wk	CIE	SEE	Duration in hrs	
Theory Courses										
1	PC501CSM	Compiler Design	3	-	-	3	30	70	3	3
2	PC502CSM	Artificial Intelligence	3	1	-	4	30	70	3	4
3	PC503CSM	Operating Systems	3	-	-	3	30	70	3	3
4	PC504CSM	Web and Internet Technologies	3	-	-	3	30	70	3	3
5	PC505CSM	Speech and Natural Language Processing	3	-	-	3	30	70	3	3
6	PE-I	Professional Elective-1	3	-	-	3	30	70	3	3
Practical/ Laboratory Courses										
7	PC551CSM	Artificial Intelligence Lab	-	-	2	2	25	50	3	1
8	PC552CSM	Operating Systems Lab	-	-	2	2	25	50	3	1
9	PW533CSM	Mini Project	-	-	2	2	25	50	3	1
			18	**1**	**6**	**25**	**255**	**570**	**-**	**22**

PROFESSIONAL ELECTIVE -1	
COURSE CODE	COURSE TITLE
PE511	EMBEDDED SYSTEMS
PE512	COMPUTER GRAPHICS
PE513	OBJECT ORIENTED ANALYSIS AND DESIGN
PE514	MACHINE LEARNING FOR DATA SCIENCE
PE515	BLOCK CHAIN TECHNOLOGY

Course Code	Course Title						Core/ Elective
PC 501 CSM	**COMPILER DESIGN**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	1	-	-	30	70	3

Course Objectives

- To understand and list the different stages in the process of compilation.
- Identify different methods of lexical analysis
- Design top-down and bottom-up parsers
- Identify synthesized and inherited attributes
- Develop syntax directed translation schemes
- Develop algorithms to generate code for a target machine

Course Outcomes

- Upon completion of the course, the students will be able to:
- For a given grammar specification, develop the lexical analyzer.
- For a given parser specification, design top-down and bottom-up parsers.
- Develop syntax directed translation schemes.
- Develop algorithms to generate code for target machine.

UNIT-I

Introduction: The Structure of a Compiler, Phases of Compilation, The Translation Process, Major Data Structures in a Compiler, Bootstrapping and Porting.

Lexical Analysis (Scanner): The Role of the Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, The Lexical Analyzer Generator Lex.

UNIT-II

Syntax Analysis (Parser): The Role of the Parser, Syntax Error Handling and Recovery, Top-Down Parsing, Bottom-Up Parsing, Simple LR Parsing, More Powerful LR Parsing, Using Ambiguous Grammars, Parser Generator Yaac.

UNIT-III

Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's Applications of Syntax-Directed Translation.

Symbol Table: Structure, Operations, Implementation and Management.

UNIT-IV

Intermediate Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Backpatching, Switch-statements, Intermediate Code for Procedures.

Run-time environment: Storage Organization, Stack Allocation of Space, Access to Nonlocal Data on the Stack, Parameter passing, Heap Management and Garbage Collection.

UNIT-V

Code Generation: Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks and Flow graphs, Optimization of Basic Blocks, Peephole Optimization, Register Allocation and Assignment.

Machine-Independent Optimizations: The Principal Sources of Optimizations, Introduction to Data-Flow Analysis.

Suggested Books:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, & Jeffrey D. Ullman , ***Compilers: Principles, Techniques and Tools***, Pearson Education
2. Kenneth C. Louden***, Compiler Construction: Principles and Practice,*** *Thomson Learning Inc., 1997.*

Suggested Reference Books*:*

1. P.Trembley and P.S.Sorenson, ***The Theory and Practice of Compiler Writing***, TMH-1985.

Course Code	Course Title						Core/ Elective
PC 502 CSM	**ARTIFICIAL INTELLIGENCE**						**Core**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	1	-	-	30	70	4

Course Objectives

- Understand the importance of the field of AI by discussing its history and various applications.
- Learn about one of the basic applications of A.I, search state formulations.
- Learn methods of expressing knowledge by a machine with appropriate reasoning and different mathematics involved behind it
- Learn how to reason when an agent has only uncertain information about its task.
- Know various supervised and unsupervised learning algorithms

Course Outcomes

The students will be able to:

- Formalize a problem in the language/framework of different AI methods
- Illustrate basic principles of AI in solutions that require problem solving, search, inference
- Represent natural language/English using Predicate Logic to build knowledge through various representation mechanisms
- Demonstrate understanding of steps involved in building of intelligent agents, expert systems, Bayesian networks
- Differentiate between learning paradigms to be applied for an application

UNIT – I

Problem Solving & Search: Introduction- introduction to intelligence Foundations of artificial intelligence (AI). History of AI, Structure of Agents.

Problem Solving - Formulating problems, problem types, states and operators, state space.

Search Strategies. - Informed Search Strategies- Best first search, A* algorithm, heuristic functions, Iterative deepening A*.

Adversarial Search/ Game playing - Perfect decision game, imperfect decision game, evaluation function, alpha-beta pruning.

UNIT – II

Knowledge, Reasoning &Planning: Reasoning - Knowledge based agent, Propositional Logic, Inference, Predicate logic (first order logic),Resolution

Structured Knowledge Representation – Frames, Semantic Nets

Planning - A Simple Planning Agent, Form Problem Solving to Planning, Basic representation of plans, partial order planning, hierarchical planning.

UNIT – III

Expert Systems, Reasoning with Uncertainty: Expert System and Applications: Introduction, Phases in Building Expert Systems, Expert System Architecture, Applications. **Uncertainty** - Basic probability, Bayes rule, Belief networks, Inference in Bayesian Networks**,** Fuzzy sets, and fuzzy logic: Fuzzy logic system architecture, membership function.

Decision Making- Utility theory, utility functions.

UNIT – IV

Learning: Machine-Learning Paradigms: Introduction, Machine Learning Systems, Supervised and Unsupervised Learning, Inductive Learning, Learning Decision Trees

Artificial Neural Networks: Introduction, Artificial Neural Networks, Single-Layer Feed-Forward Networks, Multi-Layer Feed-Forward Networks

Reinforcement learning: Learning from rewards, Passive and Active reinforcement learning, Applications.

UNIT – V

Communicating & Perceiving: Introduction to NLP- Progress & applications of NLP, Components of NLP, Grammars, Parsing.

Automatic Speech Recognition (ASR) – Speech Processing, Ex: DRAGON, HARPY,

Machine Vision – Applications, Basic Principles of Vision, Machine vision techniques: Low, Middle and High-level vision.

AI Today & Tomorrow - Achievements, ubiquitous AI.

Suggested Readings:

1. Stuart Russell and Peter Norvig. *Artificial Intelligence – A Modern Approach*, 3rd Edition, Pearson Education Press, 2009.
2. Kevin Knight, Elaine Rich, B. Nair, *Artificial Intelligence*, 3rd Edition, McGraw Hill, 2008.
3. Nils J. Nilsson, *The Quest for Artificial Intelligence*, Cambridge University Press, 2009.

Course Code	Course Title						Core/ Elective
PC 503 CSM	**OPERATING SYSTEMS**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	1	-	-	30	70	3

Course Objectives

- To learn the fundamentals of Operating Systems.
- To learn the mechanisms of OS to handle processes and threads and their communication
- To learn the mechanisms involved in memory management in contemporary OS
- To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection
- To know the components and management aspects of concurrency management

Course Outcomes

- Identify System calls and evaluate process scheduling criteria of OS.
- Develop procedures for process synchronization of an OS.
- Demonstrate the concepts of memory management and of disk management
- Solve issues related to file system interface and implementation, I/O systems
- Describe System model for deadlock, Methods for handling deadlocks.

UNIT-I

Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS - Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.

UNIT-II

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads,

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling Criteria, Scheduling algorithms, multiprocessor scheduling

UNIT-III

Process Synchronization: Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Peterson's Solution, classical problems of synchronization: The Bounded buffer problem, Producer\Consumer Problem, reader's & writer problem, Dinning philosopher's problem. Semaphores, Event Counters, Monitors, Message Passing,

Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Methods for Handling: Deadlocks: Deadlock prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

UNIT-IV

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation, fragmentation and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, structure of page table, Protection and sharing, Disadvantages of paging.

Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault, Working Set, Dirty page/Dirty bit – Demand paging, Page Replacement algorithms, Trashing

UNIT-V

I/O Hardware: I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software,

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods, Free-space management, directory implementation, efficiency and performance.

Secondary-Storage Structure: Disk structure, Disk scheduling algorithms, Disk Management, RAID structure

Suggested books:

1. Operating System Concepts Essentials, 9th Edition by AviSilberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.
2. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.

Suggested reference books:

1. Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing
2. Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley
3. Design of the Unix Operating Systems, 8th Edition by Maurice Bach, Prentice-Hall of India
4. Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati, O'Reilly and Associates

Course Code	Course Title						Core/ Elective
PC 504 CSM	**WEB & INTERNET TECHNOLOGIES**						**Core**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
C, C++, Java, DC	3	-	-	-	30	70	3

Course Objectives

- Learn various client side technologies for developing web based applications.
- Learn the concepts of JavaScript and Angular JS for adding rich GUI.
- To Know about XML applications with DTD and Schema.
- To familiarize the concepts about Servlets and JSPs in dynamic web applications.
- To learn how to establish database connectivity in web applications.

Course Outcomes

- Understand the concepts of HTML and CSS.
- Acquire the knowledge to build AJAX based applications using Javascript.
- Understand and apply the concepts of servlet framework
- Implement JSP to build interactive web applications
- Acquire the knowledge of database connectivity in web applications

UNIT-I

A Brief Introduction to Internet, The World Wide Web, Web Browsers, Web Servers, Uniform Resource Locators, MIME, HTTP

HTML5: Evolution of HTML and XHTML, Basic Syntax, Document Structure, Links, Images, Multimedia, Lists, Tables, Creating Forms. Cascading Style sheets.

UNIT-II

JavaScript: Overview, Object Orientation and JavaScript, Syntactic Characteristics, Primitives, Operators, Expressions, Input and Output, Control Statements, Objects Creation and modification, Arrays, Functions, Constructors, Pattern Matching. Manipulating DOM, HTML DOM Events, Basics of AJAX with example.

UNIT-III

XML: Introduction to XML, Syntax, XML document structure, Document Type Definition, Name spaces, XML Schemas, Display in raw XML documents, Displaying XML documents with CSS, XPath Basics, XSLT, XML Processors.

J2EE: Exploring Enterprise architecture styles, Features of EE platform, Web servers and application servers.

Database programming with JDBC: JDBC Drivers, Exploring JDBC Processes with the java.sql Package.

UNIT-IV

Servlets Technology: Exploring the Features of Java Servlet, Exploring the Servlet API, Explaining the Servlet Life Cycle, Creating a Sample Servlet, Working with ServletConfig and ServletContext Objects, Implementing Servlet Collaboration, Exploring the Session Tracking Mechanisms.

UNIT-V

JSP Technology: Advantages of JSP over Java Servlet, Architecture of a JSP Page, Life Cycle of a JSP Page, Working with JSP Basic Tags and Implicit Objects, Working with Action Tags in JSP, Exploring EL, Exploring the Elements of Tag Extensions, Tag Extension API, Working with Simple Tag Handlers Accessing Database from Servlet and JSP.

Suggested books:

1. Robert W.Sebesta: Programming the World Wide Web, 4th Edition, Pearson Education, 2009
2. Java Server Programming Java EE7 (J2EE 1.7): Black Book, (2014), Dreamtech Press
3. Porter Scobey, PawanLingras: Web Programming and Internet Technologies an E-Commerce Approach, 2nd Edition, Jones & Bartlett Learning
4. Bryan Basham, Kathy Sierra, Bert Bates: Head first Servlets & JSP, 2nd edition, OREILLY, 2008.

Course Code	Course Title						Core/ Elective
PC 505 CSM	**SPEECH AND NATURAL LANGUAGE PROCESSING**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	30	70	3

Course Objectives

- Teach students the leading trends and systems in natural language processing.
- Make them understand the concepts of morphology, syntax and semantics of the language and that they are able to give the appropriate examples that will illustrate the above mentioned concepts.
- Teach them to recognize the significance of pragmatics for natural language understanding.
- Enable students to be capable to describe the application based on natural language processing and to show the points of syntactic and semantic processing.

Course Outcomes

- To tag a given text with basic Language features
- To design an innovative application using NLP components
- To implement a rule based system to tackle morphology/syntax of a language
- To design a tag set to be used for statistical processing for real-time applications
- To compare and contrast the use of different statistical approaches for different types of NLP applications.
- Perform various language phonetic analysis

UNIT I

Introduction of NLP: Origins and challenges of NLP, Language Modeling: Grammar-based LM,Statistical LM – Regular Expressions, Automata , Morphologyand Finite State Transducers, Tokenization, stemming, Normalization,Detecting and Correcting Spelling Errors, Minimum EditDistance.

UNIT II

WORD LEVEL ANALYSIS: N-grams, Evaluating N-grams, Smoothing,Word Classes,Part-of-SpeechTagging, Rule-based, Stochastic and Transformation-basedtagging, Issuesin PoS tagging – Entropy, Hidden Markov and Maximum Entropy models, ; Named Entities

UNIT-III

SYNTACTIC ANALYSIS: Context free rules and trees – The noun Phrase – Co-ordination – Verbphrase – context free grammars – Parsing with context free grammars, Shallow parsing – Probabilistic CFG , Dependency Grammar , SemanticAnalysis: Meaning Representation-Lexical Semantics- Ambiguity-Word Sense Disambiguation. Discourse Processing: cohesion-Reference Resolution- Discourse Coherence and Structure.

UNIT_IV

Speech Fundamentals: Phonetics – speech sounds and phonetic transcription –articulatory phonetics – phonological categories and pronunciation variation– acoustic phonetics and signals –phonetic resources – articulatory and gesturalphonology.

UNIT-V

Speech synthesis – text normalization – phonetic analysis – prosodic analysis – diphonewaveformsynthesis – unit selection waveform synthesis – evaluation.

Text Books:

1. Daniel Jurafsky, James H. Martin—Speech and Language Processing: An Introduction to NaturalLanguage Processing, Computational Linguistics andSpeech, Pearson Publication, 2014.
2. Steven Bird, Ewan Klein and Edward Loper, —Natural Language Processingwith Python, FirstEdition, OReilly Media, 2009.

References:

1. Breck Baldwin, —Language Processing with Java and LingPipeCookbook,Atlantic Publisher,2015.
2. Richard M Reese, —Natural Language Processing with Java, OReilly Media, 2015.
3. Nitin Indurkhya and Fred J. Damerau, —Handbook of Natural LanguageProcessing, SecondEdition, Chapman and Hall/CRC Press,2010.
4. Tanveer Siddiqui, U.S. Tiwary, —Natural Language ProcessingandInformation Retrieval,Oxford University Press, 2008.
5. Ikrami Eldirawy , Wesam Ashour, —Visual Speech Recognition, Wiley publications , 2011
6. Himanshu Chaurasiya, —Soft Computing Implementation of AutomaticSpeech Recognition,LAP Lambert Academic Publishing,2010.
7. Kai-Fu Lee, —Automatic Speech Recognition, The Springer International Seriesin Engineeringand Computer Science, 1999.

CourseCode	Course Title						Core/ Elective
PE 511 CSM	**EMBEDDED SYSTEMS**						**ELECTIVE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	30	70	3

Course Objectives

- To provide an overview of Design Principles of Embedded System.
- Understand the fundamentals of Microcontroller based systems, basic hardware components, selection methods and attributes of an embedded system.
- To introduce and discuss Interfacing of various real world devices with 8051 microcontroller
- Comprehend the real time operating system used for the embedded system
- To expose students to the recent trends in embedded system design.

Course Outcomes

- Demonstrate the role of individual components involved in a typical embedded system.
- Describe the architectural features and instructions of Intel 8051 Microcontroller
- Apply the knowledge gained for Programming ARM for different applications.
- Expected to visualize the role of Real time Operating Systems in Embedded Systems
- Interpret the basic hardware components and their selection method based on the characteristics and attributes of an embedded system.

UNIT – I

Introduction to embedded systems: Embedded systems, Processor embedded into a system, Embedded hardware units and device in a system, Embedded software in a system, Examples of embedded systems, Design process in embedded system, Formalization of system design, Design process and design examples, Classification of embedded systems, skills required for an embedded system designer.

UNIT – II

Devices and communication buses for devices network: IO types and application with Keyboards , Serial communication devices, Parallel device ports, Sophisticated interfacing features in device ports, Wireless devices, Timer and counting devices, Watchdog timer, Real time clock, Networked embedded systems, Serial bus communication protocols, Parallel bus device protocols-parallel communication internet using ISA, PCI, PCI-X and advanced buses, Internet enabled systems network protocols, Wireless and mobile system protocols.

UNIT – III

Device drivers and interrupts and service mechanism: Programming-I/O busy-wait approach without interrupt service mechanism, ISR concept, Interrupt sources, Interrupt servicing (Handling) Mechanism, Multiple interrupts, Context and the periods for context

switching, interrupt latency and deadline, Classification of processors interrupt service mechanism from Context-saving angle, Direct memory access, Device driver programming.

UNIT – IV

Inter process communication and synchronization of processes, Threads and tasks: Multiple process in an application, Multiple threads in an application, Tasks, Task states, Task and Data, Clear-cut distinction between functions. ISRS and tasks by their characteristics, concept and semaphores, Shared data, Interprocess communication, Signal function, Semaphore functions, Message Queue functions, Mailbox functions, Pipe functions, Socket functions, RPC functions.

Unit –V

Real-time operating systems: OS Services, Process management, Timer functions, Event functions, Memory management, Device, file and IO subsystems management, Interrupt routines in RTOS environment and handling of interrupt source calls, Real-time operating systems, Basic design using an RTOS, RTOS task scheduling models, interrupt latency and response of the tasks as performance metrics, OS security issues. Introduction to embedded software development process and tools, Host and target machines, Linking and location software.

Suggested Readings:

1. Microcontroller and Embedded Systems Using Assembly and C (Second Edition),- Muhammed Ali Mazidi ,Janice Gillispie Mazidi, Rolin D. McKinlay ;2008;Pearson Publication ; ISBM : 978-81-317-1026-5 .

2. Raj Kamal, “Embedded Systems”, 2nd edition, Tata McGraw Hill, 2009.

3. Peter Barry and Patric Crowley, Intel architecture for Embedded system.

4. Wayne Wolf, “Computers as Components-principles of Embedded Computer system Design”, 1st edition, Elseveir, 2009.

5. Tammy Noergaard, ”Embedded System Architecture, A comprehensive Guide for Engineers and Programmers”, Elsevier, 2006.

Course Code	Course Title						Core/ Elective
PE 512 CSM	**COMPUTER GRAPHICS**						**ELECTIVE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
Mathematics, Engg.Drawing	3	-	-	-	30	70	3

Course Objectives

- To introduce the concept of synthetic camera model , programmable pipeline and OpenGL API
- To study different interaction modes and data structures that store 2-D and 3-D geometric objects
- To understand different transformations in 2-D and 3-D
- To study different rasterization and rendering algorithms

Course Outcomes

After completing this course, the student will be able to:-

- Describe the steps in graphics programming pipeline
- Write interactive graphics applications using OpenGL geometric primitives
- Apply affine transformations for viewing and projections
- create realistic images of 3-d objects that involve lighting shading aspects

UNIT-I

Graphics Systems and Models: Graphics system, Images, Physical and Synthetic, Imaging system, Synthetic camera model, Programming interface, Graphics architectures, Programmable pipelines.

Graphics Programming: Programming two-dimensional applications, OpenGL API, Primitives and attributes, Color, Viewing and Control functions.

UNIT-II

Input and Interaction: Input devices, Display lists & modeling, Programming event-driven input, Picking, Building interactive models, Animating interactive programs, Logic operations.

Geometric Objects: Three-dimensional primitives, Coordinate systems and frames, Frames in OpenGL, Modeling colored cube.

UNIT-III

Transformations: Affine transformations, Transformations in homogeneous coordinates, Concatenation of transformations, OpenGL transformation matrices.

Viewing: Classical and Computer views, Viewing with a computer, Positioning of camera, Simple projections, Projections in OpenGL, Hidden surface removal, Parallel-projection matrices, Perspective-projection matrices.

UNIT-IV

Lighting and Shading: Light sources, The Phong lighting model, Computational vectors, Polygonal shading, Light sources in OpenGL, Specification of matrices in OpenGL, Global illumination.

From Vertices to Frames: Basic implementation strategies, Line-segment clipping, Polygon clipping, Clipping in three dimensions, Rasterization, Anti-aliasing.

UNIT-V

Modeling & Hierarchy: Hierarchal models, Trees and traversal, Use of tree data structure, Animation, Graphical objects, Scene graphs, Simple scene graph API, Open Scene graph, Other tree structures.

Suggested Reading

1. Edward Angel, Interactive Computer Graphics: A Top-Down Approach Using OpenGL, Pearson Education, 5th edition, 2009
2. Francis S Hill Jr., Stephen M Kelley, Computer Graphics using OpenGL, Prentice-Hall Inc., 3rd Edition, 2007
3. Jim X. Chen, Foundations of 3D Graphics Programming using JOGL and Java3D, Springer Verlag, 2006
4. Hearn Donald, Pauline M Baker, Computer Graphics, 2nd edition, 1995

Course Code	Course Title						Core/ Elective
PE 515 CSM	**BLOCK CHAIN TECHNOLOGY**						**ELECTIVE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	30	70	3

Course Objectives

- Understand how block chain systems (mainly Bitcoin and Ethereum) work,
- To securely interact with them,
- Design, build, and deploy smart contracts and distributed applications,
- Integrate ideas from block chain technology into their own projects.

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

- Explain design principles of Bitcoin and Ethereum.
- Explain Nakamoto consensus.
- Explain the Simplified Payment Verification protocol.
- List and describe differences between proof-of-work and proof-of-stakeconsensus.
- Interact with a block chain system by sending and reading transactions.
- Design, build, and deploy a distributed application.
- Evaluate security, privacy, and efficiency of a given block chain system.

UNIT - I

Basics: Distributed Database, Two General Problem, Byzantine General problem and Fault Tolerance, Hadoop Distributed File System, Distributed Hash Table, ASIC resistance, Turing Complete.

Cryptography: Hash function, Digital Signature - ECDSA, Memory Hard Algorithm, Zero Knowledge Proof.

UNIT - II

Blockchain: Introduction, Advantage over conventional distributed database, Block chain Network, Mining Mechanism, Distributed Consensus, Merkle Patricia Tree, Gas Limit, Transactions and Fee, Anonymity, Reward, Chain Policy, Life of Block chain application, Soft & Hard Fork, Private and Public block chain.

UNIT - III

Distributed Consensus: Nakamoto consensus, Proof of Work, Proof of Stake, Proof of Burn, Difficulty Level, Sybil Attack, Energy utilization and alternate.

Unit - IV

Cryptocurrency: History, Distributed Ledger, Bitcoin protocols - Mining strategy and rewards, Ethereum -Construction, DAO, Smart Contract, GHOST, Vulnerability, Attacks, Sidechain, Namecoin

Unit – V

Cryptocurrency Regulation: Stakeholders, Roots of Bit coin, Legal Aspects-Cryptocurrency Exchange, Black Market and Global Economy.

Applications: Internet of Things, Medical Record Management System, Domain NameService and future of Block chain.

Case study : *Naive Blockchain construction, Memory Hard algorithm - Hashcash implementation, Direct Acyclic Graph, Play with Go-ethereum, Smart Contract Construction,Toy application using Blockchain, Mining puzzles*

Suggested Readings:

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction,Princeton University Press, 2016.
2. Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System
3. DR. Gavin Wood, "ETHEREUM: A Secure Decentralized Transaction Ledger,"Yellow paper.2014.
4. Nicola Atzei, Massimo Bartoletti, and TizianaCimoli, A survey of attacks onEthereum smart contracts

CourseCode	Course Title						Core/Elective
PE513CSM	**OBJECT ORIENTED ANALYSIS AND DESIGN**						**ELECTIVE**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	30	70	3

Course Objectives:

To introduce the basic concepts of Unified Modeling Language from defining Unified process and Core workflows

To impart knowledge on various UML diagrams for the software development

To understand the importance of each diagram in software development and understand rules to develop each diagram

Course Outcomes

Student will able to

Understand the activities in the different phases of the object-oriented development life cycle.

Model a real-world application by using a UML diagram.

Provide a snapshot of the detailed state of a system at appoint in time using object diagram.

Recognize when to use generalization, aggregation, and composition relationships.

Specify different types of business rules in a class diagram.

UNIT – I

Object-Oriented Analysis and Design: Introduction to UML

Iterative, Evolutionary, and Agile: Introduction to Unified Process, Agile Modeling, Agile Unified Process.

Inception& Use cases: Introduction, Evolutionary Requirements, Use Cases, Use case Diagrams, Activity Diagrams.

UNIT – II

Elaboration & Domain Models: Iteration Requirements and Emphasis: Core OOAD Skills, Domain Models, Class Diagrams, System Sequence Diagrams, Requirements to Design,

Package Diagram and UML Tools: Logical Architecture, Software Architecture, Package Diagrams, On to Object Design, UML CASE Tools, UML Class Diagrams.

UNIT – III

UML Class Diagrams, UML Interaction Diagrams, UML Activity Diagram and Modelling, Mapping Design to Code, UML State Machine Diagram and Modelling, Test Driven Development and Agile Concepts, Documenting Architecture, Case Studies.

UNIT – IV

UML Deployment and Component Diagram, GoF Design Patterns and Iterative Planning,

Introduction to GRASP – Methodological approach to OO Design, Architectural analysis and UML Package Design.

UNIT – V

Test Driven Development and Agile Concepts, Documenting Architecture, Case Studies.

Suggested Readings:

1. Craig Larman, "Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development", 3rd edition, Pearson 2008
2. Grady Booch, James Rumbaugh, Ivar Jacobson (2009), The Unified Modeling Language User guide, 2nd edition, Pearson Education, New Delhi, India.
3. Cay Horstmann), Object-Oriented Design and Patterns, Wiley India edition, New Delhi, India.
4. Meilir Page-Jones (2000), Fundamentals of Object Oriented Design in UML, Pearson Education and NewYork.
5.John W. Satzinger, Robert B Jackson, Stephen D Burd (2004), Object-Oriented Analysis and Design with theUnified Process, Cengage learning, India.

Course Code	Course Title						Core/ Elective
PE514CSM	**MACHINE LEARNING FOR DATA SCIENCE**						**ELECTIVE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	30	70	3

Course Objectives

The students will be able to derive practical solutions using predictive analytics. They will also understand the importance of various algorithms in Data Science.

Course Outcomes After completion of course, students would be able to:

- Apply practical solutions using predictive analytics.
- Understand the importance of various algorithms in Data Science.
- Create competitive advantage from both structured and unstructured data.
- Predict outcomes with supervised machine learning techniques.
- Unearth patterns in customer behavior with unsupervised techniques.

UNIT- I

Introduction: Algorithms and Machine Learning, Introduction to algorithms, Tools to analyze algorithms, Algorithmic techniques: Divide and Conquer, examples, Randomization, Applications.

UNIT - II

Algorithms: Graphs, maps, Map searching, Application of algorithms: stable marriages example, Dictionaries and hashing, search trees, Dynamic programming.

UNIT - III

Application to Personal Genomics: Linear Programming, NP completeness, Introduction to personal Genomics, Massive Raw data in Genomics, Data science on Personal Genomes, Interconnectedness on Personal Genomes, Case studies.

UNIT - IV

Machine Learning Introduction: Classification, Linear Classification, Ensemble Classifiers, Model Selection, Cross Validation, Holdout.

UNIT - V

Machine Learning Applications: Probabilistic modelling, Topic modelling, Probabilistic Inference, Application: prediction of preterm birth, Data description and preparation, Relationship between machine learning and statistics.

Suggested books:

1. Introduction to Machine Learning, Jeeva Jose, Khanna Book Publishing House.
2. Machine Learning, Rajiv Chopra, Khanna Book Publishing House.
3. Data Science and Machine Learning: Mathematical and Statistical Methods Machine Learning & Pattern Recognition, by Dirk P. Kroese, Zdravko Botev, Thomas Taimre, Radislav Vaisman, Chapman & Hall/Crc, 2019.
4. Hands-On Data Science and Python Machine Learning, Frank Kane, Packt Publishers, 2017.
5. https://www.edx.org/course/machine-learning-for-data-science-and-analytics

Course Code	Course Title						Core/ Elective
PW 533 CS	**MINI PROJECT**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	2	50	-	1

Course Objectives: To prepare the students

- To enhance practical and professional skills.
- To familiarize tools and techniques of systematic literature survey and documentation
- To expose the students to industry practices and team work.
- To encourage students to work with innovative and entrepreneurial ideas

Course Outcomes

- Demonstrate the ability to synthesize and apply the knowledge and skills acquired in the academic program to the real-world problems.
- Evaluate different solutions based on economic and technical feasibility
- Effectively plan a project and confidently perform all aspects of project management
- Demonstrate effective coding, written, presentation and oral communication skills

The students are required to carry out mini projects in any of the areas such as Data Structures, Microprocessors and Interfacing, Database Management Systems, Operating Systems, Design and Analysis of Algorithms, Software Engineering, Data Communications, Web Programming & Services, Computer Networks, Compiler Construction, and Object Oriented System Development.

Problems Statements are suggested to be taken can also be taken from Smart India Hackathon (SIH) Portal invited from the Ministries / PSUs / MNCs / NGOs to be worked out through.

The project could be classified as hardware, software, modeling, simulation etc. The project should involve one or many elements of techniques such as analysis, design, and synthesis.

The department will appoint a project coordinator who will coordinate the following:

1. Grouping of students (maximum of 3 students in a group)
2. Allotment of projects and project guides.
3. All projects allotment is to be completed by the 4th week of the semester so that the students get sufficient time for completion of the project.
4. Disseminate guidelines given by monitoring committee comprising of senior faculty members to the students and their guides.
5. Three periods of contact load will also be assigned to each project guide for project guidance andmonitoring at regular intervals.
6. Sessional marks are to be awarded by the monitoring committee.
7. Common norms will be established for the final presentation and documentation of the project report bythe respective departments.
8. Students are required to submit a presentation and report on the mini project at the end of the semester.

Course Code	Course Title						Core/ Elective
PC 552 CSM	**OPERATING SYSTEMS LAB**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	2	25	50	1

Course Objectives

- Learn different types of CPU scheduling algorithms
- Demonstrate the usage of semaphores for solving synchronization problem
- Understand memory management techniques and different types of fragmentation that occur in them and various page replacement policies
- Understand Banker's algorithm used for deadlock avoidance
- Learn various disk scheduling algorithms.

Course Outcomes

- Evaluate the performance of different types of CPU scheduling algorithms
- Implement producer-consumer problem, reader-writers problem, Dining philosopher's problem
- Simulate Banker's algorithm for deadlock avoidance
- Implement paging replacement and disk scheduling techniques
- Use different system calls for writing application programs.

I. CASE STUDY

Perform a case study by installing and exploring various types of operating systemson a physicalor logical (virtual) machine

II. List of Experiments (preferred programming language is C)

1. Write a C programs to implement UNIX system calls and file management
2. Write C programs to demonstrate various process related concepts.
3. Write C programs to demonstrate various thread related concepts.
4. Write C programs to simulate CPU scheduling algorithms: FCFS, SJF, Round Robin
5. Write C programs to simulate Intra & Inter-Process Communication (IPC)techniques: Pipes,Messages Queues, Shared Memory.
6. Write C programs to simulate solutions to Classical Process SynchronizationProblems: DiningPhilosophers, Producer-Consumer, Readers-Writers
7. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance.
8. Write C programs to simulate Page Replacement Algorithms: FIFO, LRU
9. Write C programs to simulate implementation of Disk Scheduling Algorithms:FCFS, SSTF

.

Course Code	Course Title						Core/Elective
PC551CSM	**ARTIFICIAL INTELLIGENCE LAB**						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
Python Programming	-	-	-	2	25	50	1

Course Objectives :

- To apply programming skills to formulate the solutions for computational problems.
- To study implementation first order predicate calculus using Prolog
- To familiarize with basic implementation of NLP with the help of Python libraries NLTK
- To understand python library scikit-learn for building machine learning models
- To enrich knowledge to select and apply relevant AI tools for the given problem

Course Outcomes

- Design and develop solutions for informed and uninformed search problems in AI.
- Demonstrate reasoning in first order logic using Prolog
- Utilize advanced package like NLTK for implementing natural language processing.
- Demonstrate and enrich knowledge to select and apply python libraries to synthesize information and develop supervised learning models
- Develop a case study in multidisciplinary areas to demonstrate use of AI.

Prerequisite: Basics of programming in Python

1. Write a program to implement Uninformed search techniques:
a. BFS
b. DFS

2. Write a program to implement Informed search techniques
a. Greedy Best first search
b. A* algorithm

3. Study of Prolog, its facts, and rules.
a. Write simple facts for the statements and querying it.
b. Write a program for Family-tree.

4. Write a program to train and validate the following classifiers for given data (scikit-learn):
a. Decision Tree
b. Multi-layer Feed Forward neural network

5. Text processing using NLTK
a. Remove stop words
b. Implement stemming
c. POS (Parts of Speech) tagging

In addition to the above programs, students should be encouraged to study implementations of one of the following

• Game bot (Tic Tac toe, 7 puzzle)
• Expert system (Simple Medical Diagnosis • Text classification
• Chat bot

SCHEME OF INSTRUCTION & EXAMINATION
B.E. - VI SEMESTER
CSE (AI&ML)

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	P/ D	Contact Hrs/Wk	CIE	SEE	Duratin in Hrs	
Theory Courses										
1	PC601CSM	Deep Learning Techniques	3	-	-	3	30	70	3	3
2	PC602CSM	Computer Networks	3	-	-	3	30	70	3	3
3	PC603CSM	Advanced Machine Learning	3	1	-	4	30	70	3	4
4	PC604CSM	Soft Computing	3	-	-	3	30	70	3	3
5	PE-II	Professional Elective-2	3	-	-	3	30	70	3	3
6	OE-I	Open Elective-1	3	-	-	3	30	70	3	3
Practical/ Laboratory Courses										
7	PC651CSM	Deep Learning Techniques Lab	-	-	2	2	25	50	3	1
8	PC652CSM	Computer Networks Lab	-	-	2	2	25	50	3	1
9	PC653CSM	Advanced Machine Learning Lab	-	-	2	2	25	50	3	1
10	PC671CSM	Summer Internship(To be evaluated during VII Semester)	-	-	-	-	-	-	-	-
			18	**1**	**6**	**25**	**255**	**570**	**-**	**21**

PROFESSIONAL ELECTIVE -2	
COURSE CODE	COURSE TITLE
PE621	QUANTUM COMPUTING
PE622	ADVANCED COMPUTER ARCHITECTURE
PE623	IMAGE PROCESSING
PE624	SOFTWARE QUALITY AND TESTING
PE625	DATA MINING
PE626	MOBILE COMPUTING

Open Elective- I

Code	Name of Subject
OE601 EE	Electrical Energy Conservation and Safety (Not for EEE & EIE Students)
OE602 EE	Reliability Engineering (Not for EEE & EIE Students)
OE611 AE	Automobile Engineering (Not for Auto. Engg. students)
OE611 ME	Entrepreneurship (Not for Mech Engg& Prod. Engg. students)
OE601 EG	Soft Skills & Interpersonal Skills
OE602 MB	Human Resource Development and Organizational Behaviour
OE601 LW	Cyber Law and Ethics
OE601 CS	Operating Systems (Not for CSE Students)
OE602 CS	OOP using Java (Not for CSE Students)
OE601 IT	Database Systems (Not for IT Students)
OE602 IT	Data Structures (Not for IT Students)

OE601 CE Disaster Mitigation (Not for Civil Engg. Students)
OE601 CSM Principles of Machine Learning(Not for AI&ML students)

Course Code	Course Title						Core/Elective
PC 601 CSM	**DEEP LEARNING TECHNIQUES**						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	**30**	**70**	**3**

Course Objectives:

- Understand the concept of neural networks, convolutional neural networks, and recurrent neural networks.
- Implement deep learning algorithms, and learn how to train deep networks.
- Gain in-depth knowledge of TensorFlow along with its functions, operations, and the execution pipeline.
- Understanding the major Architectures of Neural Networks and getting into the Convolutional neural Networks.
- Understand the applications of implementing deep learning such as image processing, natural language processing, speech recognition, deep face - facial recognition system, etc.

Course Outcomes:

After completing this course, students will be able to:

- To understand the fundamentals of deep learning.
- To be able to understand deep learning algorithms and design neural network.
- To be able to train and implement a neural network.
- To be able to have knowledge about convolutional neural networks.
- To be able to apply neural networks in various fields.

UNIT – I

What is deep learning? Artificial intelligence, Machine learning, and Deep learning - Artificial intelligence -Machine learning - Learningrepresentations from data - The "deep" in deep learning -Understanding how deep learning works, in three figures -What deep learning has achieved so far- The promise of AI

UNIT – II

Getting started with neural networks - Anatomy of a neural network - Layers: the building blocks of deep learning - Models:networks of layers - Loss functions and optimizers: keysto configuring the learning process

The Neural Network-Building Intelligent Machines, The Limits of Traditional Computer Programs, The Mechanics of Machine Learning, The Neuron, Expressing Linear Perceptrons as Neurons, Feed-Forward Neural Networks, Linear Neurons and Their Limitations, Sigmoid, Tanh

UNIT – III

Training Feed-Forward Neural Networks - The Fast-Food Problem - Gradient Descent - The Delta Rule and Learning Rates - Gradient Descent with Sigmoidal Neurons - The Backpropagation Algorithm - Stochastic and Minibatch Gradient Descent - Test Sets, Validation Sets, and Overfitting - Preventing Overfitting in Deep Neural Networks

Implementing Neural Networks in TensorFlow - What is TensorFlow? - How Does TensorFlow Compare to Alternatives?- Installing TensorFlow - Creating and Manipulating TensorFlow Variables - TensorFlow Operations - Placeholder Tensors - Sessions in TensorFlow - Navigating Variable Scopes and Sharing Variables - Managing Models over the CPU and GPU - Specifying the Logistic Regression Model in TensorFlow - Logging and

Training the Logistic Regression Model - Leveraging TensorBoard to Visualize Computation Graphs and Learning -Building a Multilayer Model for MNIST in TensorFlow

UNIT – IV

Introduction to Major Architectures of Deep Networks–Unsupervised Pretrained Networks (UPNs), Convolutional Neural Networks (CNNs), Recurrent Neural Networks, Recursive Neural Networks

Convolutional Neural Networks -Neurons in Human Vision - The Shortcomings of Feature Selection - Vanilla Deep Neural Networks Don't Scale - Filters and Feature Maps - Full Description of the Convolutional Layer - Max Pooling - Full Architectural Description of Convolution Networks - Closing the Loop on MNIST with Convolutional Networks - Accelerating Training with Batch Normalization.

UNIT –V

Deep Learning Applications - Large Scale Deep Learning - Computer Vision - Speech Recognition - Natural Language Processing - Other Applications

Suggested Reading:

1. Nikhil Buduma and Nicholas Locascio - Fundamentals of Deep Learning : Designing Next-Generation Machine Intelligence Algorithms – First Edition - O'Reilly , 2017
2. Francois Chollet-Deep Learning with Python-Second Edition,Manning Publications, 2017.
3. Josh Patterson and Adam Gibson- Deep Learning: A Practitioner's Approach - First Edition - O'Reilly , 2017
4. Ian Goodfellow, Yoshua Bengio, Aaron Courville - Deep Learning – Second Edition-MIT Press , 2016

Course Code	Course Title						Core/ Elective
PC 602 CSM	**COMPUTER NETWORKS**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	30	70	3

Course Objectives

- To develop an understanding of communication in modern network architectures from a design and performance perspective.
- To understand Data Transmission standards and MAC protocols.
- To introduce the protocols functionalities in Network Layer and Transport Layer.
- To understand DNS and supportive application protocols.
- To provide basic concepts of Cryptography.

Course Outcomes

After completing this course, the student will be able to:

- Explain the functions of the different layer of the OSI and TCP/IP Protocol.
- Understand wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) describe the function of each block.
- Illustrate network layer and transport layer protocols. For a given problem related TCP/IP protocol developed the network programming.
- Configure DNS , EMAIL, SNMP, Bluetooth, Firewalls using open source available software and tools.
- Identify the types of encryption techniques.

UNIT - I

Data communication Components: Representation of data and its flow Networks,Layered architecture, OSI and TCP/IP model, Transmission Media.

Techniques for Bandwidth utilization: Line configuration, Multiplexing - Frequency division, Timedivision and Wave division, Asynchronous and Synchronous transmission, XDSL, wireless LAN

UNIT - II

Data Link Layer and Medium Access Sub Layer: Error Detection and Error Correction -Fundamentals, Block coding, Hamming Distance, CRC;

Flow Control and Error control protocols: Stop and Wait, Go back – N ARQ, Selective Repeat ARQ,Sliding Window, and Piggybacking.

Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA/CD, CDMA/CA

UNIT - III

Network Layer: Switching techniques (circuit and packet) , Logical addressing – IPV4, IPV6, sub nettingconcepts.

Inter-networking: Tunnelling, Fragmentation, congestion control, Internet control

protocols:ARP,RARP, BOOTP and DHCP.
Delivery, Forwarding and Unicast Routing protocols, Gateway protocols.

UNIT - IV

Transport Layer: Process to Process Communication, Elements of transportprotocol, IntroductionSocket Programing.
Internet Transport Protocols: UDP, TCP, SCTP; Quality of Service, QoS improvingtechniques: LeakyBucket and Token Bucket algorithm.

UNIT - V

Application Layer: Domain Name Space (DNS), EMAIL, SNMP, Bluetooth, VOIP.

Basic concepts of Cryptography: Network Security Attacks, firewalls, symmetric encryption, Data encryption Standards, public key Encryption (RSA), Hash function, Message authentication, Digital Signature.

Suggested books

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGrawHill.
2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice HallIndia.

Suggested reference books

2. Computer Networks, 8th Edition, Andrew S. Tanenbaum, Pearson New InternationalEdition.
3. Internetworking with TCP/IP, Volume 1, 6th Edition Douglas Comer, Prentice Hall ofIndia.
4. TCP/IP Illustrated, Volume 1, W. Richard Stevens, Addison-Wesley, United States ofAmerica.

Course Code	Course Title						Core / Elective
PC 603 CSM	**Advanced Machine Learning**						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**1**	**-**	**-**	**30**	**70**	**4**

Course Objectives

- To introduce advanced concepts and methods of Machine learning.
- To develop an understanding of the role of machine learning in massive scale automation.
- To design and implement various machine learning algorithms in the range of real world applications.

Course Outcomes

After completing this course, the student will be able to

1. Understand advanced concepts of machine learning.
2. Design various machine learning algorithms.
3. Implement machine learning algorithms in the range of real world applications.

Unit I

Artificial neural network: Introduction to ANN, Perceptron,Cost function,Gradient checking, Multi layer perceptron and back propagation algorithm.

Unit II

Bayesian learning: Probability theory and Bayes rule.Naive Bayes learning algorithm,
Bayes Nets.

Unit III

Decision trees: Representing concepts as Decision trees, Recursive induction of Decision trees, Best splitting attribute: Entropy and Information gain, Searching for simple trees and Computational complexity, Overfitting, noisy data and pruning.

Unit IV

Reinforcement Learning : Reinforcement Learning through feedback network, function approximation.

Unit V

Ensemble methods : Bagging,Boosting and learning with ensembles. Random forests.

Suggested Readings:

1. Tom Mitchel, Machine learning Mc Graw Hill,1997.
2. Jeeva Jose,Introduction to Machine learning,Khanna book publishing,2020
3. Rajiv chopra,Machine Learning,Khanna book publishing,2021.
4. Uma N Dulhare,Khaleel Ahmad,Khairol AmaliBin Ahmad, Machine Learning and Big Data Concepts,Algorithms,Tools and Applications,Scrivener Publishing,Wiley,2020.
5. Ethem Apaydin, Introduction to Machine learning, 2e, the MIT press, 2010.

Course Code	Course Title						Core/ Elective
PC 604 CSM	**SOFT COMPUTING**						**Core**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	30	70	3

Course Objectives

Objectives of the course

- Classify the various soft computing frame works
- Be familiar with the design of neural networks, fuzzy logic and fuzzy systems
- Learn mathematical background for optimized genetic programming

Course Outcomes

Upon completion of the course, the students will be able to:

- Learn about soft computing techniques and their applications.
- Learn about fuzzy logic, various fuzzy systems and their functions.
- Use fuzzy rules and reasoning to develop decision making and expert system
- Choose and design suitable neural network for real time problems
- Understand the genetic algorithm concepts and their applications

UNIT-I

Introduction to Soft Computing: Soft computing constituents, characteristics of neuro-computing and soft computing, difference between hard computing and soft computing, some applications of soft computing techniques, concepts of learning and adaptation.

UNIT-II

Fuzzy logic: Introduction to classical/crisp sets and fuzzy sets, classical/crisp relations and fuzzy relations, tolerance and equivalence relations, non-iterative fuzzy sets.

Membership functions: fuzzification, methods of membership value assignments, defuzzification, lambda cuts for fuzzy sets and fuzzy relations, defuzzification methods.

UNIT-III

Fuzzy arithmetic and fuzzy measures: Fuzzy rule base and approximate reasoning, truth values and tables in fuzzy logic, fuzzy propositions, formation of rules, decomposition and aggregation of rules, fuzzy reasoning, fuzzy inference systems, fuzzy decision making, fuzzy logic control systems, fuzzy expert systems.

UNIT-IV

Introduction Neural Network: Fundamental concept, evolution of neural networks, models of artificial neural networks, important technologies, applications, McCulloch, Pitts Neuron, linear separability, Hebb network.

Supervised learning network: Perception networks, adaptive linear neuron, multiple adaptive linear neurons, back propagation network, radial basis function network.

Unsupervised learning networks: Kohonenself-organizing feature maps, learning vector quantization, counter propagation networks, adaptive resonance theory network.

UNIT-V

Genetic Algorithm: Difference between traditional algorithms and GA, genetic algorithm and search space, general genetic algorithm, operators, generational cycle, in genetic algorithm, stopping condition for genetic algorithm flow, constraints in genetic algorithm, schema theorem, classification of genetic algorithm, genetic programming, multilevel optimization.

Suggested Readings:

1. J.S.R.Jang, C.T.Sun and E.Mizutani, "Neuro-Fuzzy and Soft Computing", Pearson Education 2004.
2. S.N.Sivanandam, S.N.Deepa "Principles of Soft Computing" Second Edition, Wiley Publication.
3. Timothy J.Ross, "Fuzzy Logic with Engineering Applications", McGraw-Hill,1997.
4. Davis E.Goldberg, "Genetic Algorithms: Search, Optimization and Machine Learning", Addison Wesley, N.Y.
5. S.Rajasekaran and G.A.VijayalakshmiPai "Neural Networks, Fuzzy Logic and Genetic Algorithms" PHI Learning.

Course Code	Course Title						Core/Elective
SI 671CSM	**SUMMER INTERNSHIP**						Core
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	-			-

Course Objectives

- To train and provide hands-on experience in analysis, design, and programming of information systems by means of case studies and projects.
- To expose the students to industry practices and team work.
- To provide training in soft skills and also train them in presenting seminars and technical report¬ writing

Course Outcomes

After completing this course, the student will be able to

- Get Practical experience of software design and development, and coding practices within Industrial/R&D Environments.
- Gain working practices within Industrial/R&D Environments.
- Prepare reports and other relevant documentation.

Summer Internship is introduced as part of the curricula of encouraging students to work on problems of interest to industries. A batch of three students will be attached to a person from the Government or Private Organisations/Computer Industry/Software Companies/R&D Organization for a period of 4-6 weeks.

This will be during the summer vacation following the completion of the III-year Course. One faculty coordinator will also be attached to the group of 3 students to monitor the progress and to interact with the industry coordinate (person from industry).

The course schedule will depend on the specific internship/training experience. The typical time per topic will vary depending on the internship

- Overview of company/project
- Safety training
- Discussions with project teams
- Background research, review of documents, white papers, and scientific papers
- Planning, designing, and reviewing the planned work
- Executing the plans
- Documenting progress, experiments, and other technical documentation
- Further team discussions to discuss results
- Final report writing and presentation

After the completion of the project, each student will be required to:

1. Submit a brief technical report on the project executed and
2. Present the work through a seminar talk (to be organized by the Department)

Note: Students have to undergo summer internship of 4-6 weeks at the end of semester VI and credits will be awarded after evaluation in VII semester.

Course Code	Course Title						Core/ Elective
PC 651 CSM	**DEEP LEARNING TECHNIQUES LAB**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	2	25	50	1

Course Objectives

- Understand the concepts of Artificial Neural Networks and Deep Learning concepts.
- Implement ANN and DL algorithms with Tensorflow and Keras.
- Gain knowledge on Sequence learning with RNN.
- Gain knowledge on Image processing and analysis with CNN
- Get information on advanced concepts of computer vision.

Course Outcomes After learning the concepts of this course, the student is able to

- Develop ANN without using Machine Learning/Deep learning librarie
- Understand the Training ANN model with back propagation
- Develop model for sequence learning using RNN
- Develop image classification model using ANN and CNN.
- Generate a new image with auto-encoder and GAN.

List of Programs

1. Create Tensors and perform basic operations with tensors
2. Create Tensors and apply split & merge operations and statistics operations.
3. Design single unit perceptron for classification of iris dataset without using predefined models
4. Design, train and test the MLP for tabular data and verify various activation functions and optimizers tensor flow.
5. Design and implement to classify 32x32 images using MLP using tensorflow/keras and check the accuracy.
6. Design and implement a simple RNN model with tensorflow / keras and check accuracy.
7. Design and implement LSTM model with tensorflow / keras and check accuracy.
8. Design and implement GRU model with tensorflow / keras and check accuracy.
9. Design and implement a CNN model to classify multi category JPG images with tensorflow / keras and check accuracy. Predict labels for new images.
10. Design and implement a CNN model to classify multi category tiff images with tensorflow / keras and check the accuracy. Check whether your model is overfit / underfit / perfect fit and apply the techniques to avoid overfit and underfit like regulizers, dropouts etc.
11. Implement a CNN architectures (LeNet, Alexnet, VGG, etc) model to classify multi category Satellite images with tensorflow / keras and check the accuracy. Check whether your model is overfit / underfit / perfect fit and apply the techniques to avoid overfit and underfit.
12. Implement an Auto encoder to de-noise image.
13. Implement a GAN application to convert images.

Text Books:

1. Data Science for Beginners- Comprehensive Guide to Most Important Basics in Data Science, Alex Campbell.
2. Artificial Intelligence Technologies, Applications, and Challenges- Lavanya Sharma, Amity University , Pradeep Kumar Garg, IIT Roorkee, India.
3. Artificial Intelligence Fundamentals and Applications- Cherry Bhargava and Pardeep Kumar Sharma, CRC Press.

Course Code	Course Title						Core/ Elective
PC 652 CSM	**COMPUTER NETWORKS LAB**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
DC	-	-	-	2	30	70	1

Course Objectives

- Learn to communicate between two desktop computers.
- Learn to implement the different protocols
- Be familiar with socket programming.
- Be familiar with the various routing algorithms
- Be familiar with simulation tools.
- To use simulation tools to analyze the performance of various network protocols

Course Outcomes

After completing this course, the student will be able to:

- Implement various protocols using TCP and UDP.
- Program using sockets.
- Use simulation tools to analyze the performance of various network protocols.
- Implement and Analyze various routing algorithms.

1. Running and using services/commands like tcpdump, netstat, ifconfig, nslookup, FTP, TELNET andtraceroute. Capture ping and trace route PDUs using a network protocol analyzer and examine.
2. Configuration of router, switch . (using real devices or simulators)
3. Socket programming using UDP and TCP (e.g., simple DNS, data & time client/server, echoclient/server, iterative & concurrent servers)
4. Network packet analysis using tools like Wireshark, tcpdump, etc.
5. Network simulation using tools like Cisco Packet Tracer, NetSim, OMNeT++, NS2, NS3, etc.
6. Study of Network simulator (NS) and Simulation of Congestion Control Algorithms using NS.Performance evaluation of Routing protocols using Simulation tools.
7. Programming using raw sockets
8. Programming using RPC

Note: The Instructor may add/delete/modify/tune experiments, wherever he/she feels in a justifiedmanner.

LABORATORY REQUIREMENTS

HARDWARE:

1. Standalone desktops

SOFTWARE:

1. C / C++ / Java / Python / Equivalent Compiler
2. Network simulator like NS2/NS3/OPNET/ CISCO Packet Tracer / Equivalent

Course Code	Course Title						Core/ Elective
PC 653 CSM	**ADVANCED MACHINE LEARNING LAB**						**CORE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	2	30	70	1

Course Objectives

Implement the machine learning concepts and algorithms in any suitable languageof choice

- To implement classification algorithms.
- To implement regression algorithms.
- To implement clustering techniques.
- To implement neural networks.
- Learn to implement the different protocols
- Implement Decision trees.

Course Outcomes

After completing this course, the student will be able to:

- Implement various protocols using classification and regression techniques.
- Implement clustering mechanisms
- Implement Decision trees.
- Implement and Analyze various random forest techniques.

Implementation of following machine learning algorithms in various projects using python:

1. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
2. Build an Artificial Neural Network by implementing the Backpropagationalgorithm *and test the same using appropriate data sets.*
3. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
4. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
5. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
6. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.
7. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data setfor clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML libraryclasses/API in the program.
8. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.

CourseCode	Course Title						Core/Elective
PE 621CSM	**QUANTUM COMPUTING**						**ELECTIVE**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	**30**	**70**	**3**

Course Objectives:
The objective of this course is to impart necessary knowledge to the learner so that he/she can develop and implement algorithm and write programs using these algorithms.

Course Outcomes:
After completing this course, students will be able to:

- Explain the working of a Quantum Computing program, its architecture and program model
- Develop quantum logic gate circuits
- Develop quantum algorithm
- Program quantum algorithm on major toolkits

UNIT-I

Introduction to Quantum Computing: Motivation for studying Quantum Computing, Major players in the industry (IBM, Microsoft, Rigetti, D-Wave etc.) Origin of Quantum Computing, Overview of major concepts in Quantum Computing Qubits and multi-qubits states, Bra-ket notation: Bloch Sphere representation, Quantum Superposition, Quantum Entanglement

UNIT-II

Math Foundation for Quantum Computing: Matrix Algebra: basis vectors and orthogonality, inner product and Hilbert spaces, matrices and tensors, unitary operators and projectors, Dirac notation, Eigen values and Eigen vectors.

UNIT-III

Building Blocks for Quantum Program: Architecture of a Quantum Computing platform, Details of q-bit system of information representation: Block Sphere, Multi-qubits States, Quantum superposition of qubits (valid and invalid superposition), Quantum Entanglement, Useful states from quantum algorithmic perceptive e.g. Bell State, Operation on qubits: Measuring and transforming using gates. Quantum Logic gates and Circuit: Pauli, Hadamard, phase shift, controlled gates.

UNIT-IV

Programming model for a Quantum Computing Program: Steps performed on classical computer, Steps performed on Quantum Computer, Moving data between bits and qubits.

UNIT-V

Quantum Algorithms: Amplitude amplification, Quantum Fourier Transform, Phase Kick-back, Quantum Phase estimation, Quantum Walks

Suggested Readings:

1. Michael A. Nielsen, “Quantum Computation and Quantum Information”, Cambridge University Press. October 2000.
2. David McMahon, “Quantum Computing Explained”, Wiley 2007
3. IBM Experience: https://quantumexperience,ng,bluemix.net
4. Microsoft Quantum Development Kit https://www.microsoft.com/en-us/quantum/development-kit
   Forest SDK PyQuil: https://pyquil.readthedocs.io/en/stable/

Course Code	Course Title						Core/ Elective
PE 622 CSM	**DATA MINING**						**ELECTIVE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
-	3	-	-	-	30	70	3

Course Objectives

- To introduce the basic concepts of data Mining and its applications
- To understand different data mining like classification, clustering and Frequent Pattern mining
- To introduce current trends in data mining
- To understand, pre-process and analyze the basic concepts of Data Attributes
- To explore the various data mining techniques (Association Analysis, Classification, Clustering) adapted on data as per the requirement

Course Outcomes

- Organize and Prepare the data needed for data mining using preprocessing techniques
- Implement the appropriate data mining methods like classification, clustering or Frequent Pattern mining on a given data set
- Define and apply metrics to measure the performance of various data mining algorithms
- Understanding the importance of data mining application and using the most appropriate approach or trend for the realistic strategy

UNIT-I

INTRODUCTION: What is Data Mining? The process of knowledge discovery in databases, predictive and descriptive data mining techniques, supervised and unsupervised learning techniques. Major issues in Data Mining. Getting to know your data: Data objects and attributed types. Basic statistical descriptions of data. Data visualization, Measuring data similarity and dissimilarity.

UNIT-II

MINING FREQUENT PATTERNS, ASSOCIATIONS AND CORRELATIONS: Basic Concepts and methods, Frequent Item set Mining Methods, Sequential Pattern Mining conceptsand Pattern evaluation methods.

UNIT-III

CLASSIFICATION: Basic concepts, Decision tree, Decision rules, Bayes classification methods, Advance methods, Bayesian Belief Network, K-Nearest Neighbor (KNN) classifier, Classification byback propagation, Support vector machine.

UNIT-IV

CLUSTER ANALYSIS: Concepts and Methods: Type of data in Cluster Analysis, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Evaluation of clustering.

UNIT-V

DATA MINING TRENDS AND RESEARCH FRONTIERS: Mining Complex Data Types, Other Methodologies of Data Mining, Data Mining Applications, Data Mining and Society, Data Mining trends.

Suggested Readings:

1. Jiawei Han, Micheline Kamber, Jin Pei, Data Mining: Concepts & Techniques, 3rd Edition.,Morgon Koffman ,2011
2. Vikram Pudi, P. Radha Krishna, Data Mining, Oxford University Press, 1st Edition, 2009.
3. Pang-Ning Tan, Michael Steinbach, A Karpatne, and Vipin Kumar, Introduction to DataMining, 2nd Ed., Pearson Education, 2018.
4. J Zaki Mohammed and Wagner Meira, Data Mining and Analysis: Fundamental Conceptsand Algorithms, Cambridge University Press, 2014

References:

1. Vipin Kumar, Pang-Ning Tan, Michael Steinbach, *Introduction to Data Mining*, AddisonWesley, 2006.
2. G Dong, J Pei, Sequence Data Mining, Springer, 2007.

Course Code	Course Title						Core/ Elective
PE 623 CSM	**ADVANCED COMPUTER ARCHITECTURE**						**ELECTIVE**
Prerequisite	Contact Hours Per Week				CIE	SEE	Credits
	L	T	D	P			
--	3	-	-	-	30	70	3

Course Objectives

- An overview of computer architecture, which stresses the underlying design principles and the impact of these principles on computer performance. General topics include design methodology, processor design, control design, memory organization, system organization, and parallel processing.

Course Outcomes

After completing this course, the student will be able to:

- Know the classes of computers, and new trends and developments in computer architecture
- Understand pipelining, instruction set architectures, memory addressing.
- Understand the performance metrics of microprocessors, memory, networks, and disks
- Understand the performance and efficiency in advanced multiple-issue processors.
- Understand symmetric shared-memory architectures and their performance.

UNIT-I

Introduction - What is computer architecture? Software-hardware interface. Performanceand Power. Performance metrics. Performance measurement. Benchmark programs.

UNIT-II

Instructions- Instruction Set. Operations. Operands and addressing modes. Role of compilers and system software. Understanding implementation of function calls and returns, array references, pointers.

UNIT-III

Computer Arithmetic- Signed integers. Floating point. Rounding and accuracy. Addition and Subtraction. Multiplication. Division

Processor - Data path elements. Data path control.

UNIT-IV

Pipelining - Speedup. Pipeline hazards. Stalling. Forwarding. Branch prediction. Exceptions. Speculation. Multiple issue.

Dynamic scheduling; Cache memory- Locality of reference. Cache organization and access. Multilevel caches. Performance. Cache coherence.

UNIT-V

Virtual Memory- Hardware support for address translation, page fault handling. Translation look aside buffer, Hardware-software interface.

Input/Output- Hard disk. Flash memory. I/O interfacing. Memory mapped I/O. Interrupt driven I/O. Direct memory access. Redundant arrays of inexpensive disks; Introduction to Multi-core architecture, Multi-processors. Clusters.

Suggested Readings:

1. David A. Patterson and John L. Hennessy, *Computer Organization and Design: The Hardware and Software Interface*, Morgan Kaufmann Publishers, 4th Edition.(2009)
2. John L. Hennessy and David A. Patterson, Computer Architecture: *A Quantitative Approach*, Morgan Kaufmann Publishers (2007)

Course Code	Course Title						Core / Elective
PE 624 CSM	**Mobile Computing**						**ELECTIVE**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To introduce basics of wireless voice and data communication technologies
- To build working knowledge on various telephone and satellite networks
- To study the working principles of wireless LANs and standards
- To study principles of adhoc networks and routing
- To gain knowledge on integration of mobile networks into Internet
- To build skills in working with wireless application protocols to develop mobile applications.

Course Outcomes

After completing this course, the student will be able to

1. Understand and apply various techniques involved in planning and construction stages.
2. Implement Adhoc Network Routing protocols.
3. Mini based project based on tracking, localization and routing in wireless networks.
4. Implement file transfer, access and authentication based applications for mobile computing.

UNIT-I

Introduction – Wireless transmission – Frequencies for radio transmission – Signals – Antennas – Signal Propagation – Multiplexing – Modulations – Spread spectrum – MAC – SDMA – FDMA – TDMA – CDMA – Cellular Wireless Networks.

UNIT-II

Telecommunication systems – GSM – GPRS – DECT – UMTS – IMT-2000 – Satellite Networks - Basics – Parameters and Configurations – Capacity Allocation – FAMA and DAMA – Broadcast Systems – DAB - DVB.

UNIT-III

Wireless LAN – IEEE 802.11 - Architecture – services – MAC – Physical layer – IEEE 802.11a - 802.11b standards – HIPERLAN – Blue Tooth.

UNIT-IV

Mobile IP, Dynamic Host Configuration Protocol, Routing in MANETs: DSDV, DSR, AODV and ZRP. MANETS vs VANETs

UNIT-V

Traditional TCP – classical TCP improvements – WAP, and WAP 2.0.
Mobile Transaction models, File Systems and Mobility Management

Suggested Readings:

1. Jochen H. Schiller, *Mobile Communications*, Addison Wesley, Second Edition, 2003.
2. William Stallings, *Wireless Communications and Networks*, PHI/Pearson Education, 2002.
3. Kaveh Pahlavan, Prasanth Krishnamurthy, *Principles of Wireless Networks*, Prentice Hall, 2003.
4. Uwe Hansmann, LotharMerk, Martin S. Nicklons and Thomas Stober, *Principles of MobileComputing*, Springer, 2003.
5. Krzysztof Wesolowski, *Mobile Communication Systems*, John Wiley and Sons Ltd, 2002.

Course Code	Course Title						Core / Elective
PE 625 CSM	**Image Processing**						**ELECTIVE**
Prerequisites	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To introduce basics of visual perception, sampling, quantization and representation of digital images
- To introduce spatial domain and frequency domain filtering techniques necessary for image processing operations.
- To learn advanced image analysis techniques such as image compression, image segmentation, and object recognition
- To learn techniques of colour image processing, multi resolution methods, wavelets and morphological processing

Course Outcomes

After completing this course, the student will be able to

1. Analyse images in the frequency domain using various transforms
2. Design and implement algorithms that perform image processing operations such as histogram equalization, enhancement, restoration, filtering and denoising
3. Explain colour spaces, restoration and enhancement of colour images
4. Develop simple object recognition systems

UNIT-I

Image Processing: Introduction, Examples, Fundamental steps, Components, Elements of visual perception, Light and Electromagnetic Spectrum, Image sensing and Acquisition, Image Sampling and Quantization, Basic relationships between pixels.

Intensity Transformations and Spatial Filtering: Background, some basic intensity transformation functions, Histogram processing, Fundamentals of Spatial filtering, smoothing spatial filters, sharpening spatial filters, Combining Spatial Enhancement Methods.

UNIT-II

Filtering in the Frequency Domain: Background, Preliminary concepts, Sampling and Fourier Transformof Sampled Functions, Discrete Fourier Transform (DFT) of one variable, Extension to functions of two variables, Some Properties of the 2-D Discrete Fourier Transform, Basics of Filtering in the Frequency Domain, Image Smoothing, Image Sharpening, Homomorphic Filtering.

Image Restoration: Noise Models, Restoration in the presence of noise Only-Spatial Filtering, Periodic Noise Reduction by Frequency Domain Filtering.

Linear Degradation, Position-invariant Degradation, Estimating the Degradation Function, Inverse Filtering, Minimum Mean Square Error Filtering, Constrained Least Squares Filtering, Geometric Mean Filter.

UNIT-III

Colour Image Processing: Colour fundamentals, Colour models, Pseudocolour Image Processing, Basics of Full-colour Image Processing, Colour Transformations, Smoothing and Sharpening, Colour-based Image Segmentation, Noise in Colour Images, Colour Image Compression.

Wavelets and Multi resolution Processing: Background, Multiresolution Expansions, Wavelet Transforms in One Dimension, The Fast Wavelet Transform, Wavelet Transforms in Two Dimensions, Wavelet Packets.

UNIT-IV

Image Compression: Fundamentals, Image Compression Models, Elements of Information Theory, Error-free Compression, Lossy Compression, Image Compression Standards, Some Basic Compression Methods.
Morphological Image Processing: Preliminaries, Erosion and Dilation, Opening and Closing, The Hit-or-Miss Transformation, Some Basic Morphological Algorithms, Some Basic Gray-Scale Morphological Algorithms.

UNIT-V

Image Segmentation: Fundamentals, Point, Line and Edge Detection, Thresholding, Region-based Segmentation, Segmentation using Morphological Watersheds, The use of Motion in Segmentation.
Object Recognition: Patterns and Pattern Classes, Recognition based on Decision-theoretic Methods, Structural Methods.

Suggested Readings:

1. Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, PHI Learning Pvt. Limited,3rd Edition, 2008.
2. William K. Pratt, *Digital Image Processing*, John Wiley & Sons, Inc., 3rd Edition, 2001.

Course Code	Course Title						Core / Elective
PE 626 CSM	**Software Quality and Testing**						**ELECTIVE**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To understand the challenges of Software Quality and the need for integration of quality activities in project life cycle
- To introduce supporting software quality devices
- To introduce software quality metrics and Quality Assurance models
- To understand the steps in software testing process and taxonomy of testing tools

Course Outcomes

After completing this course, the student will be able to

4. Describe the role of quality assurance activities in the software process
5. Compare several process improvement models such as CMM, CMMI, PCMM, and ISO9000
6. Describe several process metrics for assessing and controlling a project
7. Describe how available static and dynamic test tools can be integrated into the software development environment

UNIT - I

The Software Quality Challenge, Introduction Software Quality Factors, The Components of the Software Quality Assurance System – Overview, Development and Quality Plans.

UNIT - II

Integrating Quality Activities in the Project Life Cycle, Assuring the Quality of Software Maintenance Components, CASE Tools and their effect on Software Quality, Procedure and Work Instructions, Supporting Quality Devices, Configuration Management, Documentation Control, Project Progress Control.

UNIT - III

Software Quality Metrics, Costs of Software Quality, Quality Management Standards - ISO 9000 and Companion ISO Standards, CMM, CMMI, PCMM, Malcom Balridge, 3 Sigma, 6 Sigma, SQA Project Process Standards – IEEE Software Engineering Standards.

UNIT - IV

Building a Software Testing Strategy, establishing a Software Testing Methodology, Determining Your Software Testing Techniques, eleven – Step Software Testing Process Overview, Assess Project Management Development Estimate and Status, Develop Test Plan, Requirements Phase Testing, Design Phase Testing, Program Phase Testing, Execute Test and Record Results, Acceptance Test, Report Test Results, Test Software Changes, Evaluate Test Effectiveness.

UNIT - V

Testing Client / Server Systems, Testing the Adequacy of System Documentation, Testing Web-based Systems, Testing Off – the – Shelf Software, testing in a Multiplatform Environment, Testing Security, testing a Data Warehouse, Creating Test Documentation, Software Testing Tools, Taxonomy of Testing Tools, Methodology to Evaluate Automated Testing Tools, Load Runner, Win Runner and Rational Testing Tools, Java Testing Tools, JMetra, JUNIT and Cactus.

Suggested Readings:

1. Daniel Galin, *Software Quality Assurance–From Theory to Implementation*, Pearson Education.2004
2. Mordechai Ben Menachem / Garry S. Marliss, *Software Quality–Producing Practical, ConsistentSoftware*,BS Publications, 2014
3. William E. Perry, *Effective Methods for Software Testing*, 2nd Edition, Wiley.
4. Srinivasan Desikan, Gopalaswamy Ramesh, *Software Testing, Principles and Practices*, 2006.PearsonEducation.
5. K.V.K.K. Prasad, *Software Testing Tool*, Wiley Publishers

Web Resources:

1. http://www.sei.cmu.edu/cmmi/
2. www.ibm.com/software/awdtools/tester/functional/index.html
3. www.ibm.com/software/awdtools/test/manager/
4. *java-source.net/open-source/testing-tools*
5. www.junit.org
6. *java-source.net/open-source/web-testing-tools*

Course Code	Course Title						Core/ Elective
OE 601 CSM	**PRINCIPLES OF MACHINE LEARNING**						**OPEN ELECTIVE -1**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives:

- To introduce students to the basic concepts of Data Science and techniques of Machine Learning.
- To develop skills of using recent machine learning software for solving practical problems.
- To gain experience of doing independent study and research.

Course Outcomes: After learning the contents of this course the student is able to

1. Design and implement machine learning solutions of classification, regression problems.
2. Evaluate and interpret the results of the machine learning algorithms.
3. Evaluate exploratory data analysis and Data preparation and preprocessing on different datasets.
4. Calculate Statistical measurements of the given data.
5. Analyze and identify the best algorithm matches for a given dataset.

UNIT – I

Introduction: What is Machine Learning, Use Machine Learning, and Types of Machine Learning Systems: supervised, unsupervised, semi-supervised, Reinforcement Learning, Batch and Online Learning, Main Challenges of Machine Learning.

UNIT – II

Descriptive Statistics: Data representation, types of data- nominal, ordinal, interval and continuous, central tendency- calculating mean mode median, mean vs median, variability, variance, standard deviation, Mean Absolute Deviation using sample dataset, finding the percentile, interquartile range, Box Plot, Outlier, whisker, calculating correlation, covariance, causation.

Exploratory data analysis, Data preparation and preprocessing, Data visualization.

UNIT – III

Regression: Introduction to Regression analysis, measure of linear relationship, Regression with stats models, Determining coefficient, meaning and significance of coefficients, coefficient calculation with least square method, Types of regression, Simple Linear Regression, Using Multiple features, Polynomial Regression, Metrics for Regression: MSE, RMSE, MAE.

UNIT – IV

Classification: Classification problem, Probability based approach, Logistic Regression- log-odd, sigmoid transformation, Metrics: Confusion Matrix, Accuracy, Error Rate, Precision, Recall, ROC curve, F1 score, and introduction to gradient descent.

UNIT – V

Non Parametric & SVM classification: About Non parametric classification, Decision Trees: Entropy, Gain ratio, Information Gain, Splitting criteria,

Ensemble Method: Introduction to Random Forest, Accuracy measure & performance

Instance based learning- Introduction, KNN algorithm, Distance measures, model building, locally weighted regression, radial basis functions, SVM classifier, hyper-plane, slack variables, geometric transformation kernel trick, kernel transformation.

TEXT BOOKS / REFERENCES:

1. Booz, Allen, Hamilton, The Field Guide to Data Science
2. AurélienGéron, Hands-On Machine Learning with Scikit-Learn and TensorFlow,O'Reilly Media, 2017-03-10
3. Peter Harrington, Machine Learning in Action, Manning Publications
4. Python For Data Analysis by wes McKinny 2nd edition,O'REILLY publications.
5. Jason Brownlee data analysis for machine learning.