FACULTY OF ENGINEERING

Scheme of Instruction &

Examination(AICTE Model Curriculum)

and Syllabi

B.E. VII and VIII Semesters
of
Four Year Degree
Program
in
B.E (Computer Science and Engineering)

(w.e.f: 2023-24)

Issued by

Dean, Faculty of Engineering

Osmania University

Hyderabad

2023

Chairperson, BoS **Dean, FoE OU**

SCHEME OF INSTRUCTION & EXAMINATION

BE (COMPUTER SCIENCE AND ENGINEERING) SEMESTER – VII

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	PC 701 CS	Distributed Systems	3		-	3	30	70	3	3
2	PE 73X CS	Professional Elective –III	3	-	-	3	30	70	3	3
3	PE 74X CS	Professional Elective –IV	3	-	-	3	30	70	3	3
4	OE-II	Open Elective -II	3	-	-	3	30	70	3	3
Practical/ Laboratory Courses										
5	PC 751 CS	Distributed Systems Lab	-	-	2	2	25	50	3	1
6	PC 752 CS	Web Technologies Lab	-	-	4	4	25	50	3	2
7	PW 751 CS	Project Work – I	-	-	6	6	50	-	-	3
8	SI 671 CS	Summer Internship	`	-	-	-	50	-	-	2
Total			**12**	**00**	**12**	**24**	**270**	**380**		**20**

Professional Elective – III		**Professional Elective – IV**	
Course Code	**Course Title**	**Course Code**	**Course Title**
PE 731 CS	Computational Complexity	PE 741 CS	Queueing Theory and Modelling
PE 732 CS	Advanced Operating Systems	PE 742 CS	Cloud Computing
PE 733 CS	Multimedia Technologies	PE 743 CS	Augmented and Virtual Reality
PE 734 CS	Software Reuse Techniques	PE 744 CS	Software Quality and Assurance
PE 735 CS	Information Retrieval System	PE 745 CS	Deep Learning
PE 736 CS	Big data analytics	PE 746 CS	Information Security

Open Elective - II	
Course Code	**Course Title**
OE 701 CS	Web Application Development
OE 702 CS	Principles of Python

Course Code	Course Title						Core / Elective
PC 701 CS	**Distributed Systems**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To acquire an understanding of the issues in distributed systems.
- To learn about Naming and synchronization with different algorithms.
- To study architectures and working of Distributed file systems, Distributed web-based system
- To expose the students to distributed transaction management, security issues and replication.
- To introduce Emerging trends in distributed computing

Course Outcomes

By the end of this course, the students will be able to

1. List the principles of distributed systems and describe the problems and challenges associated with these principles
2. Know about interposes communication and remote communication.
3. Know Distributed Computing techniques, Synchronous and Processes.
4. Know Distributed File Systems Apply Distributed web-based system. Understand the importance of security in distributed systems
5. Know distributed service oriented architecture.
6. Know about emerging trends in distributed computing.

UNIT-I

Introduction: Characteristics & Properties of Distributes Systems – Taxonomy - Types of Distributed Systems Design goals – Transparency Issues.

Architectures: Architectural Styles, System Architectures, Architectures versus Middleware, and Self-Management in Distributed Systems.

Processes: Threads, Virtualization, Software Agents, Clients, Servers, and Code Migration.

Communication: Inter process communication Mechanisms, Remote Procedure Call, Remote Method Invocation, Message-Oriented Communication, Stream- Oriented Communication, and Multicast Communication.

UNIT-II

Naming: Names, Identifiers and Addresses, Flat Naming, Structured Naming and Attribute-Based Naming.

Synchronization: Clock Synchronization, Logical Clocks, Mutual Exclusion, Global Positioningof Nodes, and Election Algorithms.

Consistency and Replication: Introduction, Data-Centric Consistency Models, Client-CentricConsistency Models, Replica Management, and Consistency Protocols.

UNIT-III

Fault Tolerance: Introduction to Fault Tolerance, Process Resilience, Reliable Client- Server Communication, Reliable Group Communication, Distributed Commit, and Recovery.

Distributed Object-Based Systems: CORBA, DCOM, GLOBE -Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security.

UNIT-IV

Distributed File Systems: File system, DFS- definition, Characteristics, Goals, SUN NFS-NFS Architecture, NFS Implementation, Protocols, The CODA file system-Design Overview, An Example, Design Rational, Implementation, The GOOGLE file system-Definition, Architectures, GFS Architecture

Distributed Web-Based Systems: Traditional Web-Based Systems, Web Services Fundamentals, The Apache Web Server, Web Server Clusters, Communication, HTTP Fundamentals, Simple Object Access

Protocol SOAP, Web Proxy Caching, Replication for Web Hosting Systems-CDN'S**,** Service-Oriented Architectures, REST and Web Services

UNIT-V

Distributed Coordination-Based Systems -- Architecture, Naming and Security

Emerging Trends in Distributed Systems - Emerging Trends Introduction, Grid Computing, Cloud Computing and its roots in distributed systems mechanisms and self-management of distributed systems, Virtualization, Service Oriented Architecture, The Future of Emerging Trends.

Map-Reduce: Example, Scaling, Programming Model, Apache Hadoop, Amazon Elastic Map Reduce, Mapreduce.net, Pig and Hive.

Suggested Readings:

1. Andrew S. Tanenbaum and Maarten Van Steen, Distributed Systems, PHI 2nd Edition, 2009.
2. Sunita Mahajan and Seema Shah, Distributed Computing, Oxford University
3. R. Hill,L. Hirsch,P.Lake,S.Moshiri, Guide to Cloud Computing,Principles andPractical, Springer, 2013.
4. R. Buyya, J. Borberg, A. Goscinski, Cloud Computing-Principles and Paradigms, Wiley,2013.
5. P. K. Sinha, Distributed Operating Systems, PHI
6. Taunenbaum, Distributed Systems: Principles and Paradigms, Pearson Prentice Hall, 2nd Edition 2007
7. Hagit Attiyaand Jennifer Welch, Distributed Computing, Fundamentals, Simulations and Advanced topics,Wiley India, 2nd Edition
8. G. Coulouris, J. Dollimore, and T. Kindberg, Distributed Systems: Concepts and Design, Published by Pearson 5th edition
9. David Reilly, Michael Reill, Java Network Programming & Distributed Computing, Addison-Wesley.

Course Code	Course Title						Core / Elective
PE 731 CS	**COMPUTATIONAL COMPLEXITY**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- Introduces basic complexity classes.
- To learn about computability techniques and computational complexity models.
- Introduces randomized algorithms and discusses how effective they are in reducing time and space complexity.

Course Outcomes

By the end of this course, the students will be able to

- Distinguish between computable and un-computable problems
- Classify decision problems into appropriate complexity classes
- Gain in-depth understanding of complexity models
- Specify what it means to reduce one problem to another, and construct reductions for simple examples.

UNIT–I

The computational model : Encodings and Languages, Some conventions, Modeling computation and efficiency, Machines as strings and the universal Turing machines, Uncomputable functions, Deterministic time and the class P.

NP and NP completeness : The class NP, Reducibility and NP-completeness, The Cook-Levin Theorem, Computation is Local, The web of reductions, Decision versus search, coNP, EXP and NEXP, More thoughts about P, NP.

UNIT–II

Diagonalization : Time Hierarchy Theorem, Space Hierarchy Theorem, Nondeterministic Time Hierarchy Theorem, Ladner's Theorem, Existence of NP-intermediate problems, Oracle machines and the limits of diagonalization.

Space complexity: Configuration graphs, Some space complexity classes, PSPACE completeness, NL completeness.

The Polynomial Hierarchy and Alternations: The classes Σ^{P}_{2} and Π^{P}_{2}, The polynomial hierarchy, Alternating Turing machines, Time versus alternations: time-space tradeoffs for SAT, Defining the hierarchy via oracle machines.

UNIT–III

Circuits: Boolean circuits, Karp-Lipton Theorem, Circuit lowerbounds, Non-uniform hierarchy theorem, Finer gradations among circuit classes, Circuits of exponential size, Circuit Satisfiability and an alternative proof of the Cook-Levin Theorem.

Randomized Computation: Probabilistic Turing machines, Some examples of PTMs, One-sided and zero-sided error: RP, coRP, ZPP, The robustness definitions, Randomness efficient error reduction, BPP $\subseteq$ P/poly, BPP is in PH, State of our knowledge about BPP, Randomized reductions, Randomized space-bounded computation.

UNIT–IV

Interactive proofs: Interactive proofs with a deterministic verifier, The class IP, Proving that graphs are not isomorphic, Public coins and AM, IP = PSPACE, The power of the prover, Program Checking, Multiprover interactive proofs (MIP).

Complexity of counting: The class #P, #P completeness, **Toda's Theorem:** PH _ P#SAT.

Cryptography: Hard-on-average problems and one-way functions, What is a random-enough string?, One-way functions and pseudorandom number generators, Applications.

UNIT–V

Decision Trees: Certificate Complexity, Randomized Decision Trees, Lowerbounds on Randomized Complexity, Some techniques for decision tree lowerbounds.

Communication Complexity: Definition, Lowerbound methods, Multiparty communication complexity, Overview of other communication models, Applications of communication complexity.

Suggested Readings:

1. Sanjeev Arora and Boaz Bara, Computational Complexity: A Modern Approach, Cambridge University Press,2009.
2. Christos H. Papadimitriou, Computational Complexity, Addition Wesley Longman Publishers.
3. T. Cormen, C. Leiserson, R. Rivest and C. Stein, Introduction to Algorithms, Third Edition, McGraw-Hill, 2009.
4. R. Motwani and P. Raghavan, Randomized Algorithms, Cambridge University Press, 1995.
5. S. Dasgupta, C. H. Papadimitriou and U. V. Vazirani, Algorithms, McGraw-Hill, 2008.

Course Code	Course Title						Core / Elective
PE 732 CS	**ADVANCED OPERATING SYSTEMS**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To get a comprehensive knowledge of the architecture of distributed systems.
- To understand the deadlock and shared memory issues and their solutions in distributed environments.
- To get a knowledge of resource Management and Distributed Scheduling
- To know the security issues and protection mechanisms for distributed environments.
- To get a knowledge of multiprocessor operating system and database operating systems.

Course Outcomes

After completing this course, the student will be able to

- Understand the design approaches of advanced operating systems
- Analyze the design issues of distributed operating systems.
- Evaluate design issues of multi-processor operating systems.
- Identify the requirements Distributed Shared Memory.
- Formulate the solutions to schedule the real time applications.

Unit - I:

Architectures of Distributed Systems - System Architecture types - issues in distributed operating systems.

Theoretical Foundations: Introduction, limitations of a distributed system, Lamport's logical clocks, vector clocks, causal ordering of messages, Global state reordering algorithm, Cuts of a distributed computation, termination detection.

Unit - II:

Distributed mutual exclusion: Lamport's algorithm, Ricart-Agrawala Algorithm, Maekawa algorithm, Suzuki-Kasami algorithm, Raymond's tree based algorithm.

Distributed Deadlock Detection -Introduction, Deadlock Handling Strategies in Distributed Systems, Issues in Deadlock Detection and Resolution, Centralized- Deadlock – Detection Algorithms, Distributed Deadlock Detection Algorithms, Hierarchical Deadlock Detection Algorithms

UNIT – III:

Distributed resource management: introduction-architecture – mechanism for building distributed file systems – design issues – log structured file systems.

Distributed Shared Memory: Architecture and Motivation, Algorithms for Implementing DSM, Memory Coherence, Coherence Protocols, Design Issues

UNIT- IV:

Multiprocessor operating systems - Basic multiprocessor system architectures – inter connection networks for multiprocessor systems – caching – hypercube architecture.

Distributed Scheduling: Issues in Load Distributing, Components of a Load Distributed Algorithm, Stability, Load Distributing Algorithms, Requirements for Load Distributing, Task Migration, Issues in task Migration

Unit –V:

Protection and security -Preliminaries, the access matrix model and its implementations.-safety in matrix model- advanced models of protection.

Data security – cryptography: Model of cryptography, conventional cryptography- modern cryptography, private key cryptography, data encryption standard- public key cryptography – multiple encryption – authentication in distributed systems.

Suggested Readings:

1. Mukesh Singhal, Niranjan G. Shivaratri, Advanced Concepts in Operating Systems, Tata McGraw-Hill Edition 2001
2. Andrew S. Tanenbaum, Maarten Van Steen, Distributed Systems Pearson Prentice Hall, Edition – 2, 2007
3. Kskhemkalyani, A and Singhal, M. Distributed computing: Principles, Algorithms, and systems. Cambridge University Press.

Course Code	Course Title						Core / Elective
PE 733 CS	**Multimedia Technologies**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To give each student a firm grounding in the fundamentals of the underpinning technologies in graphics, distributed systems and multimedia.
- To teach students about the principled design of effective media for entertainment, communication, training and education.
- To provide each student with experience in the generation of animations, virtual environments and multimedia applications, allowing the expression of creativity.
- To provide each student with a portfolio of their own completed work at the end of the programme.

Course Outcomes

By the end of this course, the students will be able to

- Demonstrate knowledge and understanding of the concepts, principles and theories of Multimedia Applications and Virtual environments.
- Analyze and solve problems related to their expertise in Multimedia Applications and Virtual Environments.
- Extend their basic knowledge to encompass new principles and practice
- Demonstrate their computing, technical and theoretical skills by developing a substantial Multimedia application.
- Plan, conduct and report on the development of a Multimedia Application

UNIT-I

Introduction to Multimedia: Concept of Non- Temporal and Temporal Media. Basic Characteristics of Non-Temporal Media; Images, Graphics, Text.

Basic Characteristics of Temporal Media: Video, Audio, and Animation. Hypertext and Hypermedia. Presentations: Synchronization, Events, Scripts and Interactivity, Introduction to Authoring Systems.

UNIT-II

Compression Techniques: Basic concepts of Compression.

Still Image Compression: JPEG Compression. Features of JPEG2000.

Video Compression: MPEG- 1&2 Compression Schemes, MPEG-4 Natural Video Compression.

Audio Compression: Introduction to speech and Audio Compression, MP3 Compression Scheme. Compression. Of synthetic. Graphical objects.

UNIT-III

Multimedia Systems Architecture: General Purpose Architecture for Multimedia Support: Introduction to Multimedia PC/Workstation Architecture, Characteristics of MMX instruction set,

I/O systems: Overview of USB port and IEEE 1394 interface, Operating System Support for Multimedia.

UNIT-IV

Multimedia Information Management: Multimedia Database Design,

Content Based Information Retrieval: Image Retrieval, Video Retrieval, Overview of MPEG-7, Design of video-on-Demand Systems.

UNIT-V

Introduction to Virtual Reality and Virtual Reality Systems: Related Technologies, Tele-operation and Augmented Reality Systems Interface to the Virtual World-Input; Head and hand trackers, data globes, hap tic input devices. Interface to the Virtual World- Output, Stereo display, head- mounted display, auto-stereoscopic displays, holographic displays, hap tic and force feedback.

Suggested Readings:

1. Andleigh and Thakarar, Multimedia System Design,PHI
2. David Hillman, Multimedia Technology & Application, Galgotia Publications.
3. Steinmetz, Multimedia Computing Communication and Application, Pearson Education.
4. John Vince, Virtual Reality Systems, Pearson Education

Course Code	Course Title						Core / Elective
PE 734 CS	**Software Reuse Techniques**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- Demonstration of patterns related to object oriented design.
- Describe the design patterns that are common in software applications.
- Analyze a software development problem and express it
- Design a module structure to solve a problem, and evaluate alternatives.
- Implement a module so that it executes efficiently and correctly.

Course Outcomes

By the end of this course, the students will be able to

- Construct a design consisting of a collection of modules.
- Exploit well-known design patterns (such as Iterator, Observer, Factory and Visitor).
- Distinguish between different categories of design patterns.
- Ability to understand and apply common design patterns to incremental/iterative development.
- Ability to identify appropriate patterns for design of given problem.
- Design the software using Pattern Oriented Architectures

UNIT I:

Software reuse success factors, Reuse driven software engineering as business, Object oriented software engineering, Applications and Component subsystems, Use case components, Object components.

Introduction: What Is a Design Pattern?, Design Patterns in Smalltalk MVC, Describing Design Patterns, The Catalog of Design Patterns, Organizing the Catalog, How Design Patterns Solve Design Problems, How to Select a Design Pattern

UNIT II:

A Case Study: Designing a Document Editor : Design Problems, Document Structure, Formatting, Embellishing the User Interface, Supporting Multiple Look-and-Feel Standards, Supporting Multiple Window Systems.

UNIT III:

Creational Patterns: Abstract Factory, Builder, Factory Method, Prototype, Singleton, Discussion of Creational Patterns.

UNIT IV:

Structural Pattern: Adapter, Bridge, Composite. Decorator, Façade, Flyweight, Proxy.

UNIT V:

Behavioral Patterns: Chain of Responsibility, Command, Interpreter, Iterator. Mediator, Memento, Observer, State, Strategy, Template Method , Visitor Discussion of Behavioral Patterns. What to Expect from Design Patterns.

Suggested Readings:

1. Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software, Pearson Education 1995
2. IvarJacabson, Martin Griss, Patrick Johnson, "Software Reuse: Architecture, Process and Organization for Business Success",ACM Press 1997.
3. Mark Grand , Pattern's in JAVA Vol-I, WileyDreamTech.
4. Mark Grand , Pattern's in JAVA Vol-II, WileyDreamTech.
5. Mark Grand, JAVA Enterprise Design Patterns Vol-III, ,WileyDreamTech
6. Eric Freeman, Head First Design Patterns, Oreilly

Course Code	Course Title						Core / Elective
PE 735 CS	**Information Retrieval Systems**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To provide the knowledge on information retrieval system capabilities.
- To delve into the concepts of Cataloging and Indexing.
- To explore into Automatic Indexing's diverse classes and Document/Term Clustering.
- To educate learners about various user search techniques.
- To discuss about information visualization and system evaluation.

Course Outcomes

By the end of this course, the students will be able to

- Understand various functionalities and capabilities of Information Retrieval System.
- Gain knowledge on cataloging and data structure methodology for IRS.
- Differentiate various clustering algorithms and indexing.
- Differentiate various user search techniques and system search techniques.
- Understand the concepts of information visualization and text search.

UNIT-I

Introduction: Definition, Objectives, Functional Overview, Relationship to DBMS, Digital Libraries and Data Warehouses.

Information Retrieval System Capabilities: Search, Browse, Miscellaneous.

UNIT-II

Cataloging and Indexing: Objectives, Index Process, Automatic Indexing, Information Extraction.

Data Structures: Introduction, Stemming Algorithms, Inverted File Structure, N-Gram Data Structure, PAT Data Structure, Hypertext Data Structure.

UNIT-III

Automatic Indexing: Classes of Automatic Indexing, Statistical Indexing, Natural Language, Concept Indexing.

Document and Term Clustering: Introduction, Thesaurus Generation, Item Clustering, Hierarchy of clusters.

UNIT-IV

User Search Techniques: Search Statements and Binding, Selective Dissemination of Information Search, Weighted Searches of Boolean Systems, Searching the Internet and Hypertext.

UNIT-V

Information Visualization: Introduction, Cognition and Perception, Information Visualization Technologies.

Text Search Algorithms: Introduction, Software Text Search Algorithms, Hardware Text Search Systems.

Text Book:

1. Kowalski, Gerald, Mark T May bury: INFORMATION STORAGE AND RETRIEVAL SYSTEMS: Theory and Implementation, Second Edition, Kluwer Academic Press.

Reference Books:

1. Gerald Kowalski: INFORMATION RETRIEVAL Architecture and Algorithms.
2. Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval data Structures and Algorithms, Prentice Hall, 1992.
3. Modern Information Retrieval by Yates Pearson Education.
4. Information Storage & Retrieval by Robert Korfhage –John Wiley & Sons.

e-Learning Resources:

1. https://class.coursera.org/nlp/lecture/178
2. http://cosmolearning.org/courses/database-design-417/video-lectures/
3. http://nptel.ac.in/video.php?subjectId=106102064

Course Code	Course Title						Core / Elective
PE 736 CS	**Big Data Analytics**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- Understand big data for business intelligence
- Understand big data for business intelligence
- Defend big data Without SQL
- Discuss the process of data analytics using Hadoop and related tools.

Course Outcomes

By the end of this course, the students will be able to

- Demonstrate big data and use cases from selected business domains.
- Apply the knowledge of NoSQL big data management and experiment with Install, configure, and run Hadoop and HDFS.
- Analyze map-reduce analytics using Hadoop.
- Adapt Hadoop related tools such as HBase, ZooKeeper, Pig, and Hive for big data Analytics.

UNIT – I

Big Data Analytics: Define Big Data, Affects of Big Data on our daily lives, Data Sizes, Source of Big Data, Challenges of Big Data, 5 V's of Big Data, Types of Digital Data, Structured/Unstructured data-Advantages and Sources, Business Intelligence, Comparison business intelligence from traditional big data, Architecture of data warehouse, Analytical tools used for big data analytics, Hadoop environment and its components.

UNIT – II

Apache Hadoop Distributed File System (HDFS): Applications for HDFS, Architecture of HDFS, Basic File System Operations of Hadoop, Anatomy of File read/File operations write in HDFS, Parallel Copying by distcp.

The Hadoop I/O: Data Integrity in HDFS, Different file based data structures, Hadoop Compression formats, tools and algorithms in file compression, Serialization, Hadoop configuration, Apache Oozie.

UNIT – III

Developing a MapReduce Application: Execution of MapReduce job, Failures in MapReduce job, Shuffling in MapReduce, Sorting in MapReduce, MapReduce job scheduling, Hierarchy for InputFormat class, General form of map, combiner, reduce functions, Types of input formats in MapReduce, Output formats used in MapReduce.

MapReduce Features: Built-in counter groups, Map side join, Reduce side join, Use of side data distribution.

UNIT – IV

Pig Latin: Installing and Running Pig, Comparison with Databases, Different Pig Latin expression, Ways of executing Pig Programs, Built-in functions in Pig, Data Processing Operators, Pig Latin commands/ Pig in Practice.

Hive: Installing Hive, The Hive Shell, Comparison with Traditional Databases, HiveQL, Tables, User Defined Functions.

UNIT – V

Hbase: Hbase cluster members, Comman issues while running Hbase cluster (load), Characteristics of Hbase, Hbase table, Comparsion with Hbase with RDBMS.

ZooKeeper: Characteristics of ZooKeeper, Zookeeper Commands, Operations in ZooKeeper.

Suggested Readings:

1. Tom White, "Hadoop: The Definitive Guide", Third Edition, O'Reilly, 2012.
2. Eric Sammer, "Hadoop Operations", O'Reilly, 2012.
3. VigneshPrajapati, Big data analytics with R and Hadoop, 2013.
4. E. Capriolo, D. Wampler, and J. Rutherglen, "Programming Hive", O'Reilly, 2012

Course Code	Course Title						Core / Elective
PE 741 CS	**Queueing Theory and Modelling**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To provide necessary basic concepts in probability and random processes for applications such as random signals, linear systems in communication engineering.
- To understand the basic concepts of probability, one and two dimensional random variables and to introduce some standard distributions applicable to engineering which can describe real life phenomenon.
- To understand the concept of queueing models and apply in engineering.
- To understand the significance of advanced queueing models.
- To provide the required mathematical support in real life problems and develop probabilistic models which can be used in several areas of science and engineering.

Course Outcomes

By the end of this course, the students will be able to

1. Understand the fundamental knowledge of the concepts of probability and have knowledge of standard distributions which can describe real life phenomenon.
2. Understand the basic concepts of one and two dimensional random variables and apply in engineering applications.
3. Apply the concept of random processes in engineering disciplines.
4. Acquire skills in analyzing queueing models.
5. Understand and characterize phenomenon which evolve with respect to time in a probabilistic manner

UNIT I

Probability and Random Variables:

Probability – Axioms of probability – Conditional probability – Baye‘s theorem - Discrete and continuous random variables – Moments – Moment generating functions – Binomial, Poisson, Geometric, Uniform, Exponential and Normal distributions.

UNIT II

Two - Dimensional Random Variables:

Joint distributions – Marginal and conditional distributions – Covariance – Correlation and linear regression – Transformation of random variables – Central limit theorem (for independent and identically distributed random variables).

UNIT III

Random Processes

Classification – Stationary process – Markov process - Poisson process – Discrete parameter Markov chain – Chapman Kolmogorov equations – Limiting distributions.

UNIT IV

Queueing Models

Markovian queues – Birth and death processes – Single and multiple server queueing models – Little‘s formula - Queues with finite waiting rooms – Queues with impatient customers : Balking and reneging.

UNIT V

Advanced Queueing Models

Finite source models - M/G/1 queue – Pollaczek Khinchin formula - M/D/1 and M/EK/1 as special cases – Series queues – Open Jackson networks.

TEXT BOOKS:

1. Gross, D., Shortle, J.F, Thompson, J.M and Harris. C.M., ―Fundamentals of Queueing Theory", Wiley Student 4th Edition, 2014.
2. Ibe, O.C., ―Fundamentals of Applied Probability and Random Processes", Elsevier, 1st Indian Reprint, 2007.

REFERENCES:

1. Hwei Hsu, "Schaum‘s Outline of Theory and Problems of Probability, Random Variables and Random Processes", Tata McGraw Hill Edition, New Delhi, 2004.
2. Taha, H.A., "Operations Research", 9th Edition, Pearson India Education Services, Delhi, 2016.
3. Trivedi, K.S., "Probability and Statistics with Reliability, Queueing and Computer Science Applications", 2nd Edition, John Wiley and Sons, 2002.
4. Yates, R.D. and Goodman. D. J., "Probability and Stochastic Processes", 2nd Edition, Wiley India Pvt. Ltd., Bangalore, 2012.

Course Code	Course Title						Core / Elective
PE742 CS	**Cloud Computing**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To understand the concept of cloud computing
- To understand the various issues in cloud computing.
- To familiarize themselves with the lead players in cloud.
- To appreciate the emergence of the cloud as the next generation computing paradigm.

Course Outcomes

By the end of this course, the students will be able to

- Define Cloud Computing and related concepts and describe the characteristics, advantages, risks and challenges associated with cloud computing.
- Explore virtualization technique
- Explore characterize various cloud service models, cloud deployment models
- Illustrate the use of various cloud services available online

Unit - I

Introduction - Historical Development -System Models for Distributed and Cloud Computing; Cloud Computing Architecture – The Cloud Reference Model – Cloud Characteristics

Cloud Deployment Models: Public, Private, Community, Hybrid Clouds, Challenges and Risks, Cloud Delivery Models: IaaS, PaaS, SaaS.

Unit - II:

Virtual Machines & Cloud Computing Mechanism: Levels of Virtualization, Virtualization Structures//Tools, and Mechanisms,

Types of hardware virtualization: Full virtualization - partial virtualization - para virtualization Desktop virtualization Virtualization of CPU, Memory and I/O Devices, Virtual Clusters and Resource Management, **Cloud Infrastructure Mechanism:** Cloud Storage, Cloud Usage Monitor .

Unit – III:

State Management Database – Cloud Management Mechanism: Remote Administration System, Resource Management System, SLA Management System, State Management Database – Cloud Management Mechanism: Remote Administration System, Resource Management System, SLA Management System,

Unit- IV:

Cloud Security and Trust Management, Data Security in the Cloud: An Introduction to the Idea of Data Security, The Current State of Data Security in the Cloud, CryptDb: Onion Encryption layers DET, RND, OPE, JOIN, SEARCH, HOM, and Homomorphic Encryption.

Unit –V:

Case Studies: Google App Engine (GAE) – GAE Architecture – Functional Modules of GAE – Amazon Web Services (AWS) – GAE Applications – Cloud Software Environments – Eucalyptus – Open Nebula – Open Stack, Migrating into a Cloud Introduction, Challenges while migrating to Cloud, Broad approaches to migrating into the cloud why migrate -deciding on cloud migration, the Seven-step model of migration into a cloud, Migration Risks and Mitigation

Suggested Reading:

1. Anthony T.Velte , Toby J. Velte Robert Elsenpeter, Cloud computing a practical approach - TATA McGraw- Hill , New Delhi – 2010
2. Thomas Erl, ZaighamMahood, Ricardo Puttini, ―Cloud Computing, Concept, Technology and Architecture‖, Prentice Hall, 2013.
3. K. Chandrasekhran, Essentials of cloud Computing: CRC press, 2014
4. John W. Rittinghouse, James F. Ransome, "Cloud Computing: Implementation, Management, and Security ". CRC Press 2009
5. Anthony T.Velte , Toby J. Velte Robert Elsenpeter, Cloud computing a practical approach - TATA McGraw- Hill , New Delhi – 2010
6. Raluca Ada Popa, Catherine M.S. Redfield, NickolaiZeldovich, and Hari Balakrishnan, "CryptDB: Protecting Confidentiality with encrypted Query Processing", 23rd ACM Symposium on Operating Systems Principles (SOSP 2011), Cascais, Portugal October 2011.
7. Craig Gentry, A Fully Homomorhic Encryption Scheme, September 2009.
8. David Marshall, Wade A. Reynolds, "Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center", Auerbach Publications, 2006.
9. Rajkumar Buyya, James Broberg, Andrzej Goscinski,Cloud Computing (Principles and Paradigms), Edited by John Wiley & Sons, Inc. 2011

Course Code	Course Title						Core / Elective
PE 743 CS	**Augmented And Virtual Reality**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To establish and cultivate a broad and comprehensive understanding of AR and VR applications evolving and commercially viable field of Computer Science.

Course Outcomes

After completing this course, the student will be able to

- Understand fundamental computer vision, computer graphics and human-computer interaction techniques related to AR/VR.
- Understand geometric modeling and Virtual environment.
- Relate and differentiate AR/VR technology.
- Use various types of Hardware and software in virtual Reality systems
- Implement Virtual/Augmented Reality applications.

UNIT-I

Introduction to Virtual Reality: Virtual Reality and Virtual Environment: Introduction, Computer graphics, Real time computer graphics, Flight Simulation, Virtual environment requirement, benefits of virtual reality, Historical development of VR, Scientific Landmark.

UNIT-II

Computer Graphics And Geometric Modelling: Introduction, The Virtual world space, positioning the virtual observer, the perspective projection, human vision, stereo perspective projection, Colour theory, Conversion From 2D to 3D, 3D space curves, 3 D boundary representation, Simple 3D modelling, 3D clipping,Illumination models, Reflection models, Shading algorithms.

Geometrical Transformations: Introduction, Frames of reference, Modelling transformations, Instances, Picking, Flying, Scaling the VE, Collision detection.

UNIT-III

Virtual Environment: Input: Tracker, Sensor, Digital Gloves, Movement Capture, Video-based Input, 3D Menus & 3D Scanner etc.Output: Visual /Auditory / Haptic Devices.

Generic VR system: Introduction, Virtual environment, Computer environment, VR technology, Model of interaction, VR Systems.

Animating the Virtual Environment: Introduction, The dynamics of numbers, Linear and Nonlinear interpolation, the animation of objects, linear and non-linear translation, shape & object in between, free from deformation, particle system.

Physical Simulation: Introduction, Objects falling in a gravitational field, Rotating wheels, Elastic collisions, projectiles, simple pendulum, springs, Flight dynamics of an aircraft.

UNIT-IV

Augmented Reality: Taxonomy, technology and features of augmented reality, difference between AR and VR, Challenges with AR, AR systems and functionality, Augmented reality methods, visualization techniques for augmented reality, enhancing interactivity in AR environments, evaluating AR systems.

UNIT-V

Development Tools and Frameworks: Human factors: Introduction, the eye, the ear, the somatic senses.
Hardware: Introduction, sensor hardware, Head-coupled displays, Acoustic hardware, Integrated VR systems.
Software: Introduction, Modelling virtual world, Physical simulation, VR toolkits, Introduction to VRML.

Suggested Readings:

1. Grigore C. Burdea, Philippe Coiffet , Virtual Reality Technology, Wiley 2016.
2. Alan B. Craig, Understanding Augmented Reality, Concepts and Applications, Morgan Kaufmann, 2013.
3. Alan Craig, William Sherman and Jeffrey Will, Developing Virtual Reality Applications, Foundations of Effective Design, Morgan Kaufmann, 2009.
4. John Vince, "Virtual Reality Systems ", Pearson Education Asia, 2007.
5. Anand R., "Augmented and Virtual Reality", Khanna Publishing House, Delhi.

Course Code	Course Title						Core / Elective
PE 744 CS	**Software Quality & Assurance**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- Introduces components of software quality assurance systems before, during, and after software development.
- Presents a framework for software quality assurance and explains the individual components of the framework.

Course Outcomes

After completing this course, the student will be able to

- Judge the software quality factors for a defined project
- Integrate the quality activities in the project life during implementation
- Propose some corrective and preventive actions for maintaining software quality
- Manage components of software quality in the project life cycle
- Process SQA management standards in a project

UNIT-I

Introduction to Software Quality: Definition, Software quality factors, Overview of the components of the Software Quality Assurance(SQA) System

UNIT-II

Pre-project software quality components: Contract review, Development and quality plans

SQA components in the project life cycle: Integrating quality activities in the project life cycle, Reviews, Assuring the quality of software maintenance components, CASE tools.

UNIT-III

Software quality infrastructure components: Procedures and work instructions, Supporting quality devices, Staff training and certification, Corrective and preventive actions, Configuration management, Documentation control.

UNIT-IV

Management components of software quality: Project progress control, Software quality metrics, Costs of software quality,

UNIT-V

Standards, certification and assessment: Quality management standards (ISO 9001 and ISO 9000-3, CMM, CMMI, SPICE, ISO/IEC 15504), SQA project process standards – IEEE software engineering standards

Suggested Readings:

1. Daniel Galin, Software Quality Assurance : From Theory to Implementation, Addison Wesley, 2003.
2. Stephen Kan, Metrics and Models in Software Quality Engineering (2nd Edition), Addison Wesley, 2002.
3. Schulmeyer, G. Gordon and McManus, James, (eds), Handbook of Software Quality Assurance, 3rd Ed. Prentice Hall, 1999.

Course Code	Course Title						Core / Elective
PE 745 CS	**Deep Learning**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- Understand the fundamentals of neural networks.
- Know issues in optimization of neural networks algorithms and understand regularization.
- Learn about network architectures such as convolutional neural networks, recurrent neural networks and long short term memory cells.
- Understand the application of deep networks to Computer Vision, NLP
- Learn about adversarial learning models

Course Outcomes

After completing this course, the student will be able to

- Demonstrate the fundamentals of neural networks and their training.
- Illustrate the optimization methods for deep neural networks.
- Differentiate between various architectures of CNNs, RNN
- Apply the relevant architecture to applications of Computer Vision and NLP
- Illustrate architecture of GANs and their applications

Unit I

Introduction: History of Deep Learning, McCulloch Pitts Neuron, Multilayer Perceptrons(MLPs), Sigmoid Neurons, Feed Forward Neural Networks, Back propagation

Unit II

Activation functions: Sigmoid, ReLU, Hyperbolic Functions, Softmax

Optimization: Types of errors, bias-variance trade-off, overfitting-underfitting, Cross Validation, Feature Selection, Gradient Descent (GD), Momentum Based GD, Stochastic GD, Regularization (dropout, drop connect, batch normalization), Hyper parameters

Unit III

Introduction to CNNs, Architecture, Convolution/pooling layers, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet.

Vision Application: Object Detection – As classification, region proposals, RCNN, YOLOarchitectures

Unit IV

Introduction to RNNs, basic building blocks of RNNs and other architectural details, GRU, LSTMs

Encoder Decoder Models, Seq2Seq models

NLP application: Language Translation (Machine Translation) - Attention mechanism.

Unit V

Adversarial Learning Models: Generative and discriminative models, Architectural and training details of Generative Adversarial Networks (GANs), Loss functions, Conditional GAN, RC GAN

Vision Application: Image to Image Translation – pix2pix GAN

Suggested Readings:

1. Ian Goodfellow, YoshuaBengio, Aaron Courville. Deep Learning, the MIT press, 2016
2. Bengio, Yoshua. " Learning deep architectures for AI." Foundations and trends in Machine Learning 2.1, Now Publishers, 2009
3. Rajiv Chopra, Deep Learning, Khanna Book Publishing, Delhi 2020.
4. https://nptel.ac.in/courses/106/106/106106184/
5. https://www.coursera.org/specializations/deep-learning

Course Code	Course Title						Core / Elective
PE 746 CS	**Information Security**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To learn legal and technical issues in building secure information systems
- To provide an understanding of network security
- To expose the students to security standards and practices

Course Outcomes

After completing this course, the student will be able to

- Describe the steps in Security Systems development life cycle (SecSDLC)
- Understand the common threats and attack to information systems
- Understand the legal and ethical issues of information technology
- Identify security needs using risk management and choose the appropriate risk control strategy based on business needs
- Use the basic knowledge of security frameworks in preparing security blue print for the organization
- Usage of reactive solutions, network perimeter solution tools such as firewalls, host solutions such as antivirus software and Intrusion Detection techniques and knowledge of ethical hacking tools
- Use ethical hacking tools to study attack patterns and cryptography and secure communication protocols
- Understand the technical and non-technical aspects of security project implementation and Accreditation

UNIT-I

Introduction: History, Critical Characteristics of Information, NSTISSC Security Model, Components of an Information System, Securing the Components, Balancing Security and Access, The SDLC, The Security SDLC.

Need for Security: Business Needs, Threats, Attacks, and Secure Software Development

UNIT-II

Legal, Ethical and Professional Issues: Legal Bodies, Ethics and Information Security.

Risk Management: Overview, Risk Identification, Risk Assessment, Risk Control Strategies, selecting a Risk Control Strategy, Quantitative versus Qualitative Risk Control Practices, Risk Management Discussion Points.

UNIT-III

Planning for Security: Security policy, Standards and Practices, Security Blue Print, Security Education, Continuity strategies.

Security Technology: Firewalls and VPNs: Physical Design, Firewalls, Protecting Remote connections.

UNIT-IV

Security Technology: Intrusion Detection, Access Control, and other Security Tools: Intrusion Detection and Prevention Systems-Scanning, and Analysis Tools- Access Control Devices.

Cryptography: Foundations of Cryptology, Cipher methods, Cryptographic Tools, Protocols for Secure Communications.

UNIT-V

Implementing Information Security: Information security project management, Technical topics of implementation, Non-Technical Aspects of implementation, Security Certification and Accreditation.

Information Security Maintenance: Security management models, Maintenance model.

Suggested Readings

1. Michael E Whitman and Herbert J Mattord, *Principles of Information Security*, Cengage Learning, 6 th Edition 2018
2. Atul khate, Cruptographu and Network Security" 4 th edition , Tata Mc Graw Hill , 2019
3. Nina Godbole, "Information Systems Security: Security Management, Metrics, Frameworks and Best Practices" Second Edition, WILEY 2017
4. Gupta Sarika, "Information and Cyber Security", Khanna Publishing House, Delhi
5. V.K. Pachghare, "Cryptography and Information Security", PHI Learning

Course Code	Course Title						Core/Elective
OE 701 CS	**Web Application Development**						**Elective**
	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	-	-	-	30	70	3

Course Objectives:

- To develop dynamic web applications using the concepts of HTML 5.0 and CSS
- To understand the document structure and schemas and represent data in that format

Course Outcomes:

Students will able to

- Design and develop dynamic web sites using Html 5.0, CSS
- Develop web content publishing applications that accesses data in XML

UNIT I

Introduction to Web Application Development:

World Wide Web, Web Browsers, Web Servers, Uniform Resource Locators, HTTP.

UNIT II

HTML5: Introduction, Links, Images, Multimedia, Lists, Tables, Creating Forms, CSS

UNIT III

XML : XML document structure, Document Type Definition, Namespaces, XML Schema, Displaying raw XML documents, Displaying XML documents with CSS, XPath Basics, XSLT, XML Processors.

Unit IV

Java Script : JavaScript and Forms Variables, Functions, Operators, Conditional Statements and Loops, Arrays DOM, Strings, Event and Event Handling, Java Script Closures.

Unit V

PHP Programming: Overview of PHP, General Syntactic Characteristics, Primitives, Operations, Expressions, Control Statements, Arrays, Functions, Pattern matching, Form handling, Files, Cookies, Session Tracking

Suggested Readings:

1. Robert W. Sebesta, ―Programming with World Wide Web‖, 8 th Edition, Pearson Education, 2008.
2. John Pollak, ―JQuery - A Beginners Guide‖, McGraw Hill Education, 2014.
3. AgusKurniawan,‖AngularJS Programming by Example‖,PE Press, First Edition
4. Colin J Ihrig, : Full Stack JavaScript Development with MEAN, SitePoint, 2015 Edition
5. Raul Estrada,:Fast Data Processing Systems with SMACK Stack,Packt, December 2016
6. W. Jason Gilmore "Beginning PHP and MySQL" From Novice to Professional, 4th Edition, Apress
7. Dr R.V.V.S.V.Prasad, T.V.Satya Sheela, M.Praveen Kumar, Web Programming, Shree Publishing house

Course Code	Course Title						Core/Elective
OE 702 CS	**Principles of Python Programming**						**Elective**
	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	-	-	-	30	70	3

Course Objectives:

The objectives of this course is to impart knowledge of

- To learn how to use variables, lists, tuples and dictionaries in python programs.
- To learn how to write control statements and defining modules in python
- To learn how to read and write files and handle

Course Outcomes:

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

- Understand and use the basics of python programming and core data structures like lists, set, tuple and dictionaries.
- Create, run and manipulate python programs by using python modules and writing functions.
- Understand and apply different file handling

Unit-1

Introduction to Python Programming Language, Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, The type() Function and Is Operator, Dynamic and Strongly Typed Language

Unit-2

Control Flow Statements, The if Decision Control Flow Statement, The if…else Decision Control Flow Statement, The if…elif…else Decision Control Statement, Nested if Statement, The while Loop, The for Loop, The continue and break Statements, Catching Exceptions Using try and except Statement

Unit-3

Functions, Built-In Functions, Commonly Used Modules, Function Definition and Calling the Function, The return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, Command Line Arguments.

Unit-4

Strings, Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Files, Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, The Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules

Unit-5

Lists, Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, The del Statement.

Dictionaries, Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, The del Statement,

Tuples and Sets, Creating Tuples, Basic Tuple Operations, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Tuple Methods, Using zip() Function, Sets, Set Methods, Traversing of Sets, Frozenset.

Suggested Readings

1. Gowrishankar S, Veena A, **"Introduction to Python Programming",** 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372
2. Jake VanderPlas, **"Python Data Science Handbook: Essential Tools for Working with Data",** 1st Edition, O'Reilly Media, 2016. ISBN-13: 978-1491912058
3. Wesley J Chun, **"Core Python Applications Programming",** 3rd Edition, Pearson Education India, 2015. ISBN-13: 978-9332555365

Course Code	Course Title						Core/Elective
PC 751 CS	**Distributed Systems Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	-	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

- To implement client and server programs using sockets
- To learn about working of NFS
- Understanding Remote Communication and Interprocess Communication
- To use Map- reduce model for distributed processing
- To develop mobile applications

Course Outcomes

After completing this course the student will be able to

- Write programs that communicate data between two hosts
- Configure NFS
- To implement inter process communication and remote communication
- Use distributed data processing frameworks and mobile application toolkits

List of Experiments to be performed:

1. Implementation of FTP Client
2. Implementation of Name Server
3. Implementation of Chat Server
4. Understanding of working of NFS(Includes exercises on Configuration of NFS)
5. Write a program to implement hello world service using RPC or Write a program to implement date service using RPC.
6. Implement a word count application which counts the number of occurrences of each word a large collection of documents Using Map Reduce model.
7. Develop an application using 3-tier architectures

Course Code	Course Title						Core/Elective
PC 752 CS	**Web Technologies Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	4	25	50	2

Course Objectives:

The objectives of this course is to impart knowledge of

- To develop skills in designing & implement static and dynamic websites.
- To develop server-side programming for handling user requests
- To develop database processing and communication.

Course Outcomes:

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

- Analyze a web page and identify its elements and attributes.
- Apply Cascading Style Sheets web pages for a good aesthetic sense of design.
- Build dynamic web pages using JavaScript.
- Develop server-side scripting using Middleware Technologies for various application scenarios
- Facilitate back-end Database communication for users via Middleware Technologies

LIST of EXPERIMENTS

1. Static Website Development
   A. Create a static website using HTML tables.
   B. Create a registration form using HTML forms.
   C. Apply various CSS attributes and styles.

2. Dynamic Website Development
   A. Develop dynamic Web content using JavaScript.
   B. Develop a student registration form with Validation support using JavaScript.
   C. Develop a dynamic website using JavaScript Event Handling
   D. Develop DOM manipulation using JavaScript

3. Server-side programming using Node.js
   A. Deployment of Node.js and built-in Node.js modules
   B. Implement file system in Node.js
   C. Introduction to Express.js, Router
   D. Request Parameter handling, Request Response Objects
   E. Develop MVC Architecture
   F. Create REST services in Node.js

4. Server-side programming using React.js
   A. Introduction to React Components, rendering HTML, JSX
   B. Execute React props, events, lists, forms, and router
   C. Implement a REST service from Node.js, and display a SPA

5. MongoDB and React JS
   A. Essentials of No-SQL database
   B. Creating collections, retrieving data, inserting, updating, and querying the databases
   C. Connection with Node.js

6. CASE STUDY: A full-fledged MERN stack application

Course Code	Course Title						Core/Elective
PW 751 CS	**Project Work - I**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	6	50	-	3

Course Objectives:

The objectives of this course is to impart knowledge of

- 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:

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

- 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 written and oral communication skills

The department can initiate the project allotment procedure at the end of VI semester and finalize it in the first two weeks of VII semester.

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

Collection of project topics/descriptions from faculty members(Problems can also be invited from the industries)

- Grouping of students (max 3 in a group)
- Allotment of project guides
-

The aim of project work is to develop solutions to realistic problems applying the knowledge and skills Obtained in different courses, new technologies and current industry practices .This requires students to understand current problems in their domain and methodologies to solve these problems. To get awareness on current problems and solution techniques, the first 4 weeks of VII semester will be spent on special lectures by faculty members, research scholars, post graduate students of the department and invited lectures by engineers from industries and R&D institutions. After completion of these seminars each group has to formalize the project proposal based on their own ideas or as suggested by the project guide. Seminar schedule will be prepared by the coordinator for all the students from the 5th week to the last week of the semester which should be strictly adhered to.

Each group will be required to:

1. Submit a one-page synopsis before the seminar for display on notice board.
2. Give a 30 minutes' presentation followed by 10 minutes' discussion.
3. Submit a technical write-up on the talk.

Atleast two teachers will be associated with the Project Seminar to evaluate students for the award of sessional marks which will be on the basis of performance in all the 3 items stated above.
The seminar presentation should include the following components of the project:

- Problem definition and specification
- Literature survey
- Broad knowledge of available techniques to solve a particular problem.
- Planning of the work, preparation of bar (activity)charts
- Presentation- oral and written.

Course Code	Course Title						Core/Elective
SI 752 CS	**Summer Internship**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	-	-	-	-	50	-	2

Course Objectives:

The objectives of this course is to impart knowledge of

- To give an experience to the students in solving real life practical problems with all its constraints.
- To give an opportunity to integrate different aspects of learning with reference to real life problems.
- To enhance the confidence of the students while communicating with industry engineers and give an opportunity for useful interaction with them and familiarize with work culture and ethics of the industry.

Course Outcomes:

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

- Able to design/develop a small and simple product in hardware or software.
- Able to complete the task or realize a pre-specified target, with limited scope, rather than taking up a complex task and leave it.
- Able to learn to find alternate viable solutions for a given problem and evaluate these alternatives with reference to pre-specified criteria.
- Able to implement the selected solution and document the same

Summer Internship is introduced as part of the curriculum for encouraging students to work on problems of interest to industries. A batch of two or three students will be attached to a person from an Industry / R & D Organization / National Laboratory for a period of 4 weeks. This will be during the summer vacation following the completion of the VI semester course. One faculty member will act as an internal guide for each batch to monitor the progress and interacts with the Industry guide.

After the completion of the project, students will submit a brief technical report on the project executed and present the work through a seminar talk to be organized by the department. Award of sessional marks are based on the performance of the student at the work place and awarded by industry guide and internal guide (25Marks) followed by presentation before the committee constituted by the department (25Marks).One faculty member will coordinate the overall activity of Summer Internship.

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

SCHEME OF INSTRUCTION
BE (COMPUTER SCIENCE AND ENGINEERING) CSE-Semester -VIII

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	PE 85X CS	Professional Elective -V	3	-	-	3	30	70	3	3
2	OE-III	Open Elective – III	3	-	-	3	30	70	3	3
Practical/ Laboratory Courses										
3	PW861 CS	Project Work – II	-	-	16	16	50	100	3	8
Total			**06**	**-**	**16**	**16**	**110**	**240**	**-**	**14**

Professional Elective –V		**Open Elective - III**	
Course Code	**Course Title**	**Course Code**	**Course Title**
PE 851 CS	Information Theory and Coding	OE 801 CS	Software Engineering
PE 852 CS	Internet of Things	OE 802 CS	Neural Networks
PE 853 CS	Robotics		
PE 854 CS	Secure Software Engineering		
PE 855 CS	Natural Language Processing		
PE 856 CS	Intellectual Property Rights		

Course Code	Course Title						Core / Elective
PE 851 CS	**Information Theory And Coding**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To acquire the knowledge in measurement of information and errors.
- Understand the importance of various codes for communication systems.
- To design encoder and decoder of various codes.
- To know the applicability of source and channel codes.

Course Outcomes

By the end of this course, the students will be able to

- Explain concept of Dependent & Independent Source, measure of information, Entropy, Rate of Information and Order of a source
- Model the continuous and discrete communication channels using input, output and joint probabilities
- Determine a codeword comprising of the check bits computed using Linear Block codes, cyclic codes & convolutional codes
- Design the encoding and decoding circuits for Linear Block codes, cyclic codes, convolutional codes, BCH and Golay codes.

UNIT-I

Information Theory: Introduction, Measure of information, Information content of message, Average Information content of symbols in Long Independent sequences, Average Information content of symbols in Long dependent sequences, Markov Statistical Model for Information Sources, Entropy and Information rate of Mark off Sources.

UNIT-II

Source Coding: Encoding of the Source Output, Shannon's Encoding Algorithm, Source coding theorem, Prefix Codes, Kraft McMillan Inequality property- KMI, Huffinan codes.

UNIT-III

Information Channels: Communication Channels, Discrete Communication channels Channel Matrix, Joint probability Matrix, Binary Symmetric Channel, System Entropies. Mutual Information, Channel Capacity, Channel Capacity of Binary Symmetric Channel.

UNIT-IV

Error Control Coding: Introduction, Examples of Error control coding, methods of Controlling Errors, Types of Errors, types of Codes, Linear Block Codes: matrix description of Linear Block Codes, Error detection & Correction capabilities ofLinear Block Codes, Single error correction Hamming code, Table lookup Decoding using Standard Array.

UNIT-V

Binary Cyclic Codes: Algebraic Structure of Cyclic Codes, Encoding using an (n-k) Bit Shift register, Syndrome Calculation, Error Detection and Correction

Convolution Codes: Convolution Encoder, Time domain approach, Transform domain approach, Code Tree, Trellis and State Diagram, The Viterbi Algorithm).

Suggested Readings:

10. K. Sam Shanmugam, Digital and Analog Communication Systems, John Wtley India Pvt Ltd, 1996
11. Simon Haykin, Digital Communication, John Wtley India Pvt Ltd, 2008.
12. Ranjan Bose, ITC and Cryptography, TMH, II edition, 2007
13. J. Das, S.K.Mullick, P. K. Chatterjee, Principles of Digital Communication, Wiley, 1986-Technology &Engineering
14. Bernard Sklar, Digital Communications- Fundamentals and Applications, SecondEdition, Pearson Education, 2016, ISBN: 9780134724058.
15. HariBhat, Ganesh Rao, Information Theory and Coding , Cengage, 2017.
16. Todd K Moon ,Error Correction Coding,Wiley Std. Edition, 2006

Course Code	Course Title						Core / Elective
PE 852 CS	**Internet of Things**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
	3	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To introduce the terminology, technology and its applications
- To introduce the Python Scripting Language which is used in many IoT devices
- To introduce the Raspberry PI platform, that is widely used in IoT applications
- To introduce the implementation of web-based services on IoT devices

Course Outcomes

After completing this course, the student will be able to

- Understanding of IoT value chain structure (device, data cloud), application areas and technologies involved.
- Understand IoT sensors and technological challenges faced by IoT devices, with a focus on wireless, energy, power, and sensing modules
- Market forecast for IoT devices with a focus on sensors
- Explore and learn about Internet of Things with the help of preparing projects designed for Raspberry Pi

Unit - I:

Introduction to Internet of Things- Definition and Characteristics of IoT, Sensors, Actuators, Physical Design of IoT – IoT Protocols, IoT communication models, IoT Communication APIs, IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Embedded Systems, IoT Levels and Templates, Domain Specific IoTs – Home, City, Environment, Energy, Agriculture and Industry.

Unit - II:

IoT Software defined networks, network function virtualization, difference between SDN and NFV for IoT, Basics of IoT System Management with NETCOZF, YANG- NETCONF, YANG, SNMP NETOPEER

UNIT – III:

IoT Physical Devices and Endpoints- Introduction to Arduino and Raspberry Pi- Installation, Interfaces (serial, SPI, I2C), Programming – Python program with Raspberry PI with focus on interfacing external gadgets, controlling output, reading input from pins.

UNIT- IV:

Controlling Hardware- Connecting LED, Buzzer, Switching High Power devices with transistors, Controlling AC Power devices with Relays, Controlling servo motor, speed control of DC Motor, unipolar and bipolar Stepper motors Sensors- Light sensor, temperature sensor with thermistor, voltage sensor, ADC and DAC, Temperature and Humidity Sensor DHT11, Motion Detection Sensors, Wireless Bluetooth Sensors, Level Sensors, USB Sensors, Embedded Sensors, Distance Measurement with ultrasound sensor

Unit –V:

IoT Physical Servers and Cloud Offerings– Introduction to Cloud Storage models and communication APIs Webserver – Web server for IoT, Cloud for IoT, Python web application framework designing a RESTful web API. Industrial IoT : Agriculture, Healthcare, Activity Monitoring

Suggested Readings:

1. Arshdeep Bahga and Vijay Madisetti, Internet of Things - A Hands-on Approach, Universities Press, 2015, ISBN: 9788173719547
2. Matt Richardson & Shawn Wallace, Getting Started with Raspberry Pi, O'Reilly (SPD), 2014, ISBN: 9789350239759 3. Raspberry Pi Cookbook, Software and Hardware Problems and solutions, Simon Monk, O'Reilly (SPD), 2016, ISBN 7989352133895

3. S. Misra, A. Mukherjee, and A. Roy, 2020. Introduction to IoT. Cambridge University Press.
4. S. Misra, C. Roy, and A. Mukherjee, 2020. Introduction to Industrial Internet of Things and Industry 4.0. CRC Press.
5. Peter Waher, 'Learning Internet of Things', Packt Publishing, 2015 3. Editors Ovidiu Vermesan
6. Peter Friess,'Internet of Things – From Research and Innovation to Market Deployment', River Publishers, 2014
7. N. Ida, Sensors, Actuators and Their Interfaces, SciTech Publishers, 2014.

Course Code	Course Title						Core / Elective
PE 853 CS	**Robotics**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To understand the functions of the basic components of a Robot. To study the use of various types of End of Effectors and Sensors
- To impart knowledge in Robot Kinematics and Programming
- To learn Robot safety issues and economics.

Course Outcomes

By the end of this course, the students will be able to

- to apply the basic engineering knowledge for the design of robotics

Unit-I

Fundamentals of robot

Robot - Definition - Robot Anatomy - Co-ordinate Systems, Work Envelope Types and Classification-Specifications-Pitch, Yaw, Roll, Joint Notations, Speed of Motion, Pay Load- Robot Parts and their Functions-Need for Robots-Different Applications.

Unit II

Robot drives systems and end effectors

Pneumatic Drives-Hydraulic Drives-Mechanical Drives-Electrical Drives-D.C. Servo Motors, Stepper Motors, A.C. Servo Motors-Salient Features, Applications and Comparison of all these Drives, End Effectors-Grippers-Mechanical Grippers, Pneumatic and Hydraulic- Grippers, Magnetic Grippers, Vacuum Grippers; Two Fingered and Three Fingered Grippers; Internal Grippers and External Grippers; Selection and Design Considerations.

Unit III

Sensors and machine vision

Requirements of a sensor, Principles and Applications of the following types of sensors- Position sensors - Piezo Electric Sensor, LVDT, Resolvers, Optical Encoders, pneumatic Position Sensors, Range Sensors Triangulations Principles, Structured, Lighting Approach, Time of Flight, Range Finders, Laser Range Meters, Touch Sensors ,binary Sensors., Analog Sensors, Wrist Sensors, Compliance Sensors, Slip Sensors, Camera, Frame Grabber, Sensing and Digitizing Image Data- Signal Conversion, Image Storage, Lighting Techniques, Image Processing and Analysis-Data Reduction, Segmentation, Feature Extraction, Object Recognition, Other Algorithms, Applications- Inspection, Identification, Visual Serving and Navigation.

Unit IV

ROBOT KINEMATICS AND ROBOT PROGRAMMING

Forward Kinematics, Inverse Kinematics and Difference; Forward Kinematics and Reverse Kinematics of manipulators with Two, Three Degrees of Freedom (in 2 Dimension), Four Degrees of freedom (in 3 Dimension) Jacobians, Velocity and Forces-Manipulator Dynamics, Trajectory Generator, Manipulator Mechanism Design-Derivations and problems. Lead through Programming, Robot programming Languages-VAL Programming-Motion Commands, Sensor Commands, End Effector commands and simple Programs.

Unit V

Implementation and robot economics

RGV, AGV; Implementation of Robots in Industries-Various Steps; Safety Considerations for Robot Operations - Economic Analysis of Robots.

Suggested Readings:

1. Klafter R.D., Chmielewski T.A and Negin M., “Robotic Engineering - An Integrated Approach”, Prentice Hall, 2003.
2. Groover M.P., “Industrial Robotics -Technology Programming and Applications”, McGraw Hill, 2001.
3. Craig J.J., “Introduction to Robotics Mechanics and Control”, Pearson Education, 2008
4. Deb S.R., “Robotics Technology and Flexible Automation” Tata McGraw Hill Book Co., 1994.
5. Koren Y., “Robotics for Engineers", Mc Graw Hill Book Co., 1992.
6. Fu.K.S.,Gonzalz R.C. and Lee C.S.G., “Robotics Control, Sensing, Vision and Intelligence”, McGraw Hill Book Co., 1987.
7. Janakiraman P.A., “Robotics and Image Processing”, Tata McGraw Hill, 1995.
8. Rajput R.K., “Robotics and Industrial Automation”, S.Chand and Company, 2008.
9. Surender Kumar, “Industrial Robots and Computer Integrated Manufacturing”, Oxford and IBH Publishing Co. Pvt. Ltd., 1991.

Course Code	Course Title						Core / Elective
PE 854 CS	**Secure Software Engineering**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To understand that how the security aspects of software development are embedded into the system to be developed.
- Able to learn secure architecture design, secure coding, secure deployment and secure software development methodologies.

Course Outcomes

Upon successful completion of this course, the student will be able to:

- Explain why security is a software issue.
- Detail the principles and practices of secure software design.
- Describe the principles and practices of secure software coding and testing.
- Integrate biblical principles within the field of secure software engineering.

UNIT – I

Security a software Issue: Introduction, The Problem, Software Assurance and Software Security, Threats to software security, Sources of software insecurity, Benefits of detecting software security defects early, managing secure software development

What Makes Software Secure: Defining Properties of secure software, Influencing the security properties of software, Asserting and specifying desired security properties?

UNIT – II

Requirements Engineering for secure software: Introduction, The SQUARE process Model, Requirements elicitation and prioritization.

UNIT – III

Secure Software Architecture and Design: Introduction, Software Security Practices for Architecture and Design: Architectural risk analysis, Software Security Knowledge for Architecture and Design: Security Principles, Security Guidelines, and Attack Patterns.

Secure Coding and Testing: Introduction, Code analysis, Coding Practices, Software Security Testing, Security Testing considerations throughout the SDLC.

UNIT – IV

Security and Complexity:

System Assembly Challenges: Introduction, Security Failures, Functional and Attacker Perspectives for Security Analysis, System Complexity Drivers and Security, Deep Technical Problem Complexity.

UNIT – V

Governance and Managing for More Secure Software:

Governance and security, Adopting an Enterprise Software Security Framework, How much security is enough?, Security and project management, Maturity of Practice.

Suggested Readings:

1. Julia H Allen, Sean J Barnum, Robert J Ellison, Gary McGraw, Nancy R Mead, "Software Security Engineering: A Guide for Project Managers", Addison Wesley, 2008
2. Ross J Anderson, "Security Engineering: A Guide to Building Dependable Distributed Systems", 2nd Edition, Wiley, 2008.
3. Howard, M. and LeBlanc, D., "Writing Secure Code", 2nd Edition, Microsoft Press, 2003
4. Jason Grembi, "Developing Secure Software", First Edition, Cengage Learning, 2008.
5. Gary R. McGraw, "Software Security: Building Security", AddisonWesley Software Security Edition, 2006.
6. Richard Sinn, "Software Security: Theory, Programming and Practice", First Edition, Cengage Learning, 2009.

Course Code	Course Title						Core / Elective
PE 855 CS	**Natural Language Processing**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- Introduce to some of the problems and solutions of NLP and their relation to linguistics and statistics.
- This course introduces the fundamental concepts and techniques of natural language Processing (NLP).
- Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information

Course Outcomes

By the end of this course, the students will be able to

- Show sensitivity to linguistic phenomena and an ability to model them with formal grammars.
- Understand and carry out proper experimental methodology for training and evaluating empirical NLP systems
- Manipulate probabilities, construct statistical models over strings and trees, and Estimate parameters using supervised and unsupervised training methods.
- Design, implement, and analyze NLP algorithms
- Design different language modelling Techniques.

UNIT - I

Finding the Structure of Words: Words and Their Components, Issues and Challenges, Morphological Models

Finding the Structure of Documents: Introduction, Methods, Complexity of the Approaches, Performances of the Approaches

UNIT - II

Syntax Analysis: Parsing Natural Language, Treebanks: A Data-Driven Approach to Syntax, Representation of Syntactic Structure, Parsing Algorithms, Models for Ambiguity Resolution in Parsing, Multilingual Issues

UNIT - III

Semantic Parsing: Introduction, Semantic Interpretation, System Paradigms, Word Sense Systems, Software.

UNIT - IV

Predicate-Argument Structure, Meaning Representation Systems, Software.

UNIT - V

Discourse Processing: Cohension, Reference Resolution, Discourse Cohension and Structure

Language Modeling: Introduction, N-Gram Models, Language Model Evaluation, Parameter Estimation, Language Model Adaptation, Types of Language Models, Language-Specific Modeling Problems, Multilingual and Crosslingual Language Modeling

Suggested Readings:

1. Daniel M. Bikel and ImedZitouni,Multilingual natural Language Processing Applications: From Theory to Practice –Pearson Publication
2. Tanvier Siddiqui, U.S. Tiwary - Natural Language Processing and Information Retrieval
3. Daniel Jurafsky& James H Martin, Speech and Natural Language Processing - Pearson Publications

Course Code	Course Title						Core / Elective
PE 856 CS	**Intellectual Property Rights**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- The knowledge on world trade organization, trade agreements and investments.
- The importance of intellectual property rights to develop trade mark law, copy right law and patent law.
- The new developments in the law of intellectual property rights in order to bring progressive changes towards a free market society and international trade practices under the trade related Intellectual Property Rights Agreement (TRIPS).

Course Outcomes

By the end of this course, the students will be able to

- Classify the intellectual property rights to provide the legal rights, patents, trademarks, copyrights and trade secrets.
- Relate the World Intellectual Property organization to protect intellectual property rules and policies.
- Identify the world trade organization agreements for trade related intellectual properties rights and investments.
- Outline the importance of intellectual property in organizations of different industrial sectors for the purpose of product and technology development.
- Infer the geographical Indications of international development of law for policy and legal issues.
- Interpret the purpose in category of marks for the international trademark registration.
- Extend the fundamentals of copyright law and originality of material for the rights of reproduction.
- Demonstrate the international copyright law with respect to ownership for the registration of copyright.
- Summarize the trade secrets determination, misappropriation and protection for submission and litigation.
- Utilize the new international developments for trademarks law, copyright law and patent law.

UNIT – I

Introduction to Intellectual property: Introduction, types of intellectual property, international Organizations, agencies and treaties, importance of intellectual property rights.

UNIT – II

Trade Marks: Purpose and function of trademarks, acquisition of trade mark rights, protectable matter, Selecting, and evaluating trade mark, trade mark registration processes.

UNIT – III

Law of copy rights: Fundamental of copy right law, originality of material, rights of reproduction, rights to perform the work publicly, copy right ownership issues, copy right registration, notice of copy right, international copy right law.

Law of patents: Foundation of patent law, patent searching process, ownership rights and transfer

UNIT – IV

Trade Secrets: Trade secrete law, determination of trade secrete status, liability for misappropriations of trade secrets, protection for submission, trade secrete litigation.

Unfair competition: Misappropriation right of publicity, false advertising.

UNIT – V

New development of intellectual property: new developments in trade mark law; copy right law, patent law, intellectual property audits. International overview on intellectual property, international – trade mark law, copy right law, international patent law, and international development in trade secrets law.

Suggested Readings:

1. Deborah. E. Bouchoux, Intellectual property right, Cengage learning.
2. Prabuddha ganguli, Intellectual property right – Unleashing the knowledge economy Tata McGraw Hill Publishing company ltd

Course Code	Course Title						Core / Elective
OE 801 CS	**Software Engineering**						**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 software development- processes from defining a product to shipping and maintaining that product
- To impart knowledge on various phases, methodologies and practices of software development
- To understand the importance of testing in software development and study various testing
- strategies and software quality metrics.

Course Outcomes

By the end of this course, the students will be able to

- Acquired working knowledge of alternative approaches and techniques for each phase of software development.
- Judge an appropriate process model(s) assessing software project attributes and analyze necessary requirements for project development eventually composing SRS.
- Creation of visual models to describe (non-) algorithmic solutions for projects using various design principles.
- Acquire skills necessary as an independent or as part of a team for architecting a complete software project by identifying solutions for recurring problems exerting knowledge on patterns.

UNIT-I

Introduction to Software Engineering:

A generic view of Process: Software Engineering, Process Framework, CMM Process Patterns,Process Assessment.

Process Models: Prescriptive Models, Waterfall Model, Incremental Process Models, Evolutionary Process Models, Specialized Process Models, The Unified Models, Personal and Team Process Models, Process Technology, Product and Process.

An Agile view of Process: Introduction to Agility and Agile Process, Agile Process Models

UNIT-II

System Engineering:Computer-based Systems, The System Engineering Hierarchy, Business Process Engineering, Product Engineering, System Modeling.

Requirements Engineering: A Bridge to Design and Construction, Requirements Engineering Tasks, Initiating Requirements Engineering Process, Eliciting Requirements, Developing Use- Cases, Building the Analysis Model, Negotiating Requirements, Validating Requirements.

UNIT-III

Building the Analysis Model: Requirements Analysis Modeling Approaches, Data Modeling Concepts, Object-Oriented Analysis, Scenario-based Modeling, Flow-oriented Modeling, Class- based Modeling, Creating a Behavioral Model.

Design Engineering: Design within the context of SE, Design Process and Design Quality, Design Concepts, The Design Model, Pattern-based Software Design.

UNIT-IV

Creating an Architectural Design: Software Architecture, Data Design, Architectural Styles and Patterns, Architectural Design.

Modeling Component-Level Design: Definition of Component, Designing Class-based Components, Conducting Component-level Design, Object Constraint Language, Designing Conventional Components.

Performing User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, Design Evaluation.

UNIT-V

Testing: Strategies:A Strategic Approach to Conventional Software Testing, Test Strategies for O-O Software.

Tactics: Software Testing Fundamentals, Black-box and White-box Testing, Basis Path Testing, Control Structure Testing, O-O Testing Methods.

Debugging: Debugging Techniques, The Art of Debugging.

Suggested Readings:

1. Roger S. Pressman, Software Engineering: A Practitioner's Approach, 7th Edition, McGraw Hill, 2009
2. Ali Behforooz and Frederick J. Hudson, Software Engineering Fundamentals, Oxford University Press, 1996
3. Pankaj Jalote, An Integrated Approach to Software Engineering, 3rd Edition, Narosa Publishing House, 2008

Course Code	Course Title						Core / Elective
OE 802 CS	Neural Networks						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- Understand the basics of ANN and comparison with Human brain.
- Acquire knowledge on Generalization and function approximation of various ANN architectures.
- Understand reinforcement learning using neural networks
- Acquire knowledge of unsupervised learning using neural networks.

Course Outcomes

By the end of this course, the students will be able to

- Obtain the fundamentals and types of neural networks. The student will have a broad knowledge in developing the different algorithms for neural networks.
- Analyze neural controllers
- Have a broad knowledge in Fuzzy logic principles and will be able to determine different methods of Deffuzification.

UNIT-I

Introduction to Artificial Intelligence System: Neural Network, Fuzzy logic, Genetic Algorithm
Fundamentals of Neural Networks: What is Neural Network, Model of Artificial Neuron, Learning rules and various activation functions.

UNIT-II

Neural Network Architecture: Single layer Feed-forward networks. Multilayer Feed-forward networks. Recurrent Networks.

UNIT-III

Back propagation Networks: Back Propagation networks, Architecture of Back-propagation(BP) Networks, Back-propagation Learning, Variation of Standard Back propagation algorithms.

UNIT-IV

Associative Memory: Autocorrelators, Heterocorrelators, Wang et al's Multiple Training Encoding Strategy, Exponential BAM, Associative Memory for Real coded pattern pairs, Applications.

UNIT-V

Introduction about Fuzzy set theory: Fuzzy versus Crisp, Crisp and fuzzy sets, Crisp and Fuzzy relations.

Suggested Readings:

1. Bishop, C. M. Neural Networks for Pattern Recognition. Oxford University Press. 1995.
2. Neural Networks,Fuzzy Logic and Genetic Algorithms, by S.Rajasekaran and G.A. Vijayalakshmi Pai.
3. Neuro-Fuzzy Systems, Chin Teng Lin, C. S. George Lee, PHI.
4. Build_Neural_Network_With_MS_Excel_sample by Joe choong.

Course Code	Course Title						Core / Elective
PW 861 CS	**Project Work - II**						**Elective**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	**16**	**50**	**100**	**8**

Course Objectives

- 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

By the end of this course, the students will be able to

- Demonstrate the ability to synthesize and apply the knowledge and skills acquired in the academic program to 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 written and oral communication skills

The aim of Project work –II is to implement and evaluate the proposal made as part of Project Work - I. Students can also be encouraged to do full time internship as part of project work-II based on the common guidelines for all the departments. The students placed in internships need to write the new proposal in consultation with industry coordinator and project guide within two weeks from the commencement of instruction.

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

1. Re-grouping of students - deletion of internship candidates from groups made as part of project Work-I
2. Re-Allotment of internship students to project guides
3. Project monitoring at regular intervals

All re-grouping/re-allotment has to be completed by the 1st week of VIII semester so that students get sufficient time for completion of the project.

All projects (internship and departmental) will be monitored at least twice in a semester through student presentation for the award of sessional marks. Sessional marks are awarded by a monitoring committee comprising of faculty members as well as by the supervisor. The first review of projects for 25 marks can be conducted after completion of five weeks. The second review for another 25 marks can be conducted after 12 weeks of instruction.

Common norms will be established for the final documentation of the project report by the respective departments. The students are required to submit draft copies of their project report within one week after completion of instruction.

Note: Three periods of contact load will be assigned to each project guide.