Scheme of Instruction, Evaluation

and

Syllabi of

MASTER OF COMPUTER APPLICATIONS

(MCA)

With effect from Academic Year 2025-26

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
UNIVERSITY COLLEGE OF ENGINEERING
(Autonomous)
Osmania University
Hyderabad – 500 007, TG, INDIA

Estd. 1917

Estd. 1929

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, U.C.E., O.U

MASTER OF COMPUTER APPLICATIONS (MCA)

Type of course	Course Code	Course Name	Contact hours per week			Scheme of Evaluation		Credits
			L	T	P	CIE	SEE	
SEMESTER-I								
Core-I	PCC 101	Mathematical Foundations of Computer Science	3	-	-	40	60	3
Core-II	PCC 102	C Programming	3	-	-	40	60	3
Core-III	PCC 103	Object Oriented Programming Using Java	3	-	-	40	60	3
Core-IV	PCC 104	Computer Architecture	3	-	-	40	60	3
Core V	PCC 105	Probability and Statistics	3	-	-	40	60	3
Core VI	MGC 106	Managerial Economics and Accountancy	3	-	-	40	60	3
Lab-I	LCC 151	C Programming Lab	-	-	2	25	**50**	1
Lab -II	LCC 152	Java Programming Lab	-	-	2	25	**50**	1
Lab-III	HSC 151	Soft Skills Lab	-	-	2	25	**50**	1
TOTAL			**18**	**-**	**6**	**315**	**510**	**21**
SEMESTER-II								
Core – VI	PCC 201	Operating Systems	3	-	-	40	60	3
Core – VII	PCC 202	Database Management Systems	3	-	-	40	60	3
Core - VIII	PCC 203	Design and Analysis of Algorithms	3	-	-	40	60	3
Core –IX	PCC 204	Data Engineering with Python	3	-	-	40	60	3
Core – X	PCC 206	Operations Research	3	-	-	40	60	3
Core-XI	PCC 207	Data Structures	3	-	-	40	60	3
Lab- IV	LCC 251	Operating Systems Lab	-	-	2	25	**50**	1
Lab- V	LCC 252	Data Engineering with Python Lab	-	-	2	25	**50**	1
Lab- VI	LCC 253	Database Management Systems Lab	-	-	2	25	**50**	1
Lab-VII	LCC 254	Data Structure Lab	-	-	2	25	**50**	1
Mini Project	LCC 254	Mini Project	-	-	8	25	**50**	4
TOTAL			**18**	**-**	**16**	**365**	**610**	**26**

PCC101	MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE					
CORE -I						
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To learn logic theory and relations and functions
2	To study graph theory and concepts of trees
3	To gain insights into recurrence relation

Course Outcomes:	
On completion of this course, the student will be able to :	
CO-1	Understand the basic functions and relations and solve problems
CO-2	Solve the recurrence relations
CO-3	Apply various algebraic structures.
CO-4	Analyze the different applications of Computer science as graph theory problems

UNIT– I

Fundamentals of Logic: Basic Connectives and Truth Tables, Logical Equivalence, Logical Implication, Use of Quantifiers, Definitions and the Proof of Theorems.
Set Theory: Set and Subsets, Set Operations, and the Laws of Set theory, Counting and Venn Diagrams.
Properties of the Integers: The well – ordering principle, Recursive Definitions, Division Algorithms, Fundamental theorem of Arithmetic.

UNIT– II

Relations and Functions: Cartesian Product, Functions onto Functions, Special Functions, Pigeonhole Principle, Composition and Inverse Functions.
Relations: Partial Orders, Equivalence Relations and Partitions.
Principle of Inclusion and Exclusion: Principles of Inclusion and Exclusion, Generalization of Principle.

UNIT– III

Generating Functions: Introductory Examples, Definition and Examples, Partitions of Integers.
Recurrence Relations: First–order linear recurrence relation, second–order linear homogenous recurrence relation with constant coefficients..

UNIT– IV

Algebraic Structures: Algebraic System–General Properties, Semi Groups, Monoids, Homomorphism, Groups, Residue Arithmetic.

UNIT–V
Graph Theory: Definitions and examples, sub graphs, complements and graph Isomorphism, Vertex degree, Planar graphs, Hamiltonian paths and Cycles. **Trees:** Definitions, properties and Examples, Rooted Trees, Spanning Trees and Minimum Spanning Trees.

Suggested Reading:

1	Mott Joe L Mott, Abraham Kandel, and Theodore P Baker, **Discrete Mathematics for Computer Scientists & Mathematicians**, Prentice Hall NJ, 2nd Edition, 2015
2	P. Tremblay and R Manohar **Discrete Mathematical Structures with Applications to Computer Science**, Mc Graw Hill, 1987
3	R.K. Bisht and H.S. Dhami, **Discrete Mathematics** Oxford Higher Education, 2015
4	Bhavanari Satyanarayana, Tumuru kotaVenkata Pradeep Kumar and Shaik Mohiddin Shaw, **Mathematical Foundation of Computer Science,** BSP, 2016

PCC102	C PROGRAMMING					
CORE -II						
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To understand the basic concepts of programming and the C language.
2	To develop problem-solving skills using C programming.
3	To learn how to write, compile, and debug C programs.
4	To gain proficiency in using fundamental data types, control structures, and functions.
5	To explore advanced topics like pointers, arrays, strings, structures, and file handling.

Course Outcomes:	
On completion of this course, the student will be able to :	
CO1	Explain the basic structure of C programs, data types, operators, and expressions, and apply them in problem solving.
CO2	Write and debug C programs using control structures (decision making and looping) for solving computational problems.
CO3	Apply modular programming concepts using functions, recursion, and parameter passing mechanisms and also demonstrate effective use of arrays, strings, structures, unions, and enumerations for data representation and manipulation.
CO4	Implement memory management using pointers, pointer arithmetic, and dynamic memory allocation functions.
CO5	Develop real-world applications in C by integrating concepts of modularity, efficiency, and error handling.

UNIT– I

Introduction to C and Basic Concepts : Introduction to Programming, History and Features of C, Structure of a C Program, Compilation and Execution, Data Types, Variables, and Constants, Operators and Expressions, Input / Output Functions: printf(), scanf(), getchar(), putchar().

UNIT– II

Control Flow Statements: Conditional Statements, Looping Statements, Jump Statements, break: Exiting a loop or a switch statement, continue: Skipping the rest of the current iteration, go to: Unconditional jump (its use and pitfalls).

Practical examples.

UNIT– III

Functions and Arrays

- Functions: Introduction to Functions: Modular programming, advantages of using functions, Function Declaration, Definition, and Call, Function Prototypes, Passing Arguments to Functions, : Call by value and call by reference, Return Statement, Recursion
- Arrays: Introduction to Arrays, Accessing Array Elements, One-Dimensional Arrays, Multi-Dimensional Arrays:
- Strings: Introduction to Strings, String Declaration and Initialization, String I/O: gets(), puts(), scanf(), printf(), Standard String Library Functions: strlen(), strcpy(), strcat(), strcmp().Practical examples.

UNIT– IV

Pointers and Dynamic Memory Allocation

- Pointers: Introduction to Pointers, The & (Address-of) and * (Dereference) Operators, Pointer Arithmetic, Pointers and Arrays, Pointers and Strings, Pointers to Pointers, Pointers and Functions.
- Dynamic Memory Allocation: Introduction, Memory Allocation Functions: malloc(), calloc(), realloc(), free()
- Practical examples: Creating a dynamic array.

UNIT–V

Structures, Unions, and File Handling

- Structures: Introduction to Structures, Structure Declaration and Initialization, Accessing Structure Members, Arrays of Structures, Pointers to Structures, Nested Structures.
- Unions: Introduction to Unions, Declaration and Accessing Members:
- File Handling: Introduction, Types of Files: Text files and binary files, File Pointers, Opening and Closing Files, File I/O Functions, Character I/O, String I/O, Formatted I/O, Block I/O.
- Error Handling in File Operations: ferror(), feof().

Suggested Reading:

1	"The C Programming Language" by Brian W. Kernighan and Dennis M. Ritchie (K&R):
2	"C Programming: A Modern Approach" by K.N. King"
3	"Let Us C" by Yashavant Kanetkar
4	Programming in C, by Pradip Dey & Manas Ghosh, Oxford University Press 2011

PCC103	OBJECT ORIENTED PROGRAMMING USING JAVA					
CORE -III						
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40 Marks	

Course Objectives:	
1	To Learn the basics of object oriented programming
2	To study Java I/O mechanisms and develop graphics based JAVA programs
3	To learn the basic of Swing framework.

Course Outcomes:	
On completion of this course, the student will be able to :	
CO-1	Explain OOPs features and concepts
CO-2	Use various built-in Java classes and methods
CO-3	Create window based Java programs

UNIT– I

Object Oriented System Development: Understanding Object Oriented Development, Understanding Object Concepts, Benefits of Object Oriented Development.
Java Programming Fundamentals: Introduction, Overview of Java, Data Type, Variables and Arrays, Operators, Control statements, Classes, Methods, Inheritance, Packages and Interfaces, Inner Classes.

UNIT– II

I/O basics, Stream and Byte classes, Character Streams, Reading Console input and output, Print Writer Class, String Handling, Exceptions Handling, Multithreaded Programming.

UNIT– III

Exploring Java Language, Collections Overview, Collections Interfaces, Collections Classes, Iterators, Random Access Interface, Maps, Comparators, Arrays, Legacy classes and interfaces, Sting Tokenizer, Bit Set, Date, Calendar, Timer.

UNIT– IV

Introducing AWT working with Graphics: AWT Classes, Working with Graphics.
Event Handling: Two Event Handling Mechanisms, The Delegation Event Model, Event Classes, Source of Events, Event Listener Interfaces.

AWT Controls: Control Fundamentals, Labels, Using Buttons, Applying Check Boxes, Checkbox Group, Choice Controls, Using Lists, Managing Scroll Bars, Using Text Field, Using Text Area, Understanding Layout Managers, Menu bars and Menus, Dialog Boxes, File Dialog, Handling events by Extending AWT Components, Exploring the controls, Menus and Layout Managers.

UNIT–V
Introduction to Swing Package, Java I/O classes and interfaces, Reading and Writing Files, Serialization, Introduction to Java Network Programming, Object Class, Exploring Image package.

Suggested Reading:

1	Herbert Schildt, **The Complete Reference Java**, 9th Edition, Tata Mc Gra Hill, 2005.
2	Bruce Eckel, **Thinking in Java**, 4th Edition, Pearson Education, 2009
3	Dietel and Dietel, **Java: How to Program,** 5th Edition, Prentice Hall, 2007
4	James M Slack, **Programming and Problem solving with JAVA**, Thomson Learning, 2002
5	C Thomas Wu, **An Introduction to Object Oriented programming with Java**, Tata Mc Graw Hill, 2005

PCC104	COMPUTER ARCHITECTURE					
CORE -IV						
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To Learn the basics of data representations and register micro operations
2	To study CPU architecture and Computer Arithmetic algorithms
3	To learn the basics of I/O organization.

Course Outcomes*:*	
On completion of this course, the student will be able to :	
CO-1	Apply data representation methods
CO-2	Understand the CPU architecture and write Computer Arithmetic algorithms
CO-3	Analyze the I/O operations basics

UNIT– I

Data Representation: Data types, Complements, Fixed and Floating Point representations, and Binary codes.

Over view of Computer Function and Inter connections: Computer components, Interconnection structures, Bus interconnection, Bus structure, and Data transfer.

UNIT– II

Register Transfer Micro operations: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic, Logic and Shift micro operations, Arithmetic Logic Shift Unit. **Basic Computer Organization and Design:** Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory reference instruction, Input-Output and Interrupt.

UNIT– III

Micro programmed Control: Control memory, Address Sequencing, Micro program example, Design of Control Unit.

Central Processing Unit: General Register Organization, Stack Organization, Instruction formats, Addressing modes, Data Transfer and Manipulation, and Program control. **Computer Arithmetic:** Addition and Subtraction, Multiplication, Division,
And Floating Point Arithmetic Operations.

UNIT– IV

Memory Organization: Memory Hierarchy, Main Memory, RAM and ROM, Auxiliary memory, Associative memory, Cache memory, Virtual memory, Memory Management hardware.

UNIT–V
Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt, Direct Memory Access (DMA), I/O Processor, Serial Communication. **Pipeline Processing:** Arithmetic, Instruction and RISC Pipe lines. **Assessing and Understanding Performance:** CPU performance and its factors, Evaluating performance.

Suggested Reading:

1	Morris Mano M, **Computer System Architecture**, Pearson Education India, 3rd Edition, 2007.
2	William Stallings, **Computer Organization and Architecture**, PHI, 7th Edition, 2008.
3	David A Patterson, John L Hennessy, **Computer Organization and Design**, Morgan Kaufmann, 5t h Edition, 2013.
4	Carl Hamacher Zvonko Vranesic, Safwat Zaky, **Computer Organization,** Tata Mc Graw-Hill Education, 5th Edition, 2002.

PCC105	PROBABILITY AND STATISTICS					
CORE -V						
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40 Marks	

Course Objectives:	
1	To Understand the Linear Algebra concepts through vector spaces.
2	To learn concepts of probability, discrete and continuous probability distributions
3	To learn the hypotheses testing and acquiring knowledge of basic statistical Inference and its applications.

Course Outcomes:	
On completion of this course, the student will be able to :	
CO-1	Calculate probabilities by applying probability laws and theoretical results, knowledge of important discrete and continuous distributions, the interrelations With real time applications.
CO-2	Understand the use of sample statistics to estimation known parameters.
CO-3	Compute and interpret Correlation Analysis, regression lines and multiple Regression analysis with applications

UNIT– I

Vector Spaces –Vector Spaces and Subspaces-Null Spaces, Column Spaces and Linear Transformations. Linearly Independent Sets -Bases - Coordinate Systems.

UNIT– II

Probability-Basic terminology, Three types of probability, Probability rules, Statistical independence, statistical dependency, Bayes theorem.
Probability Distributions - Random variables, expected values, binomial distribution, Poisson distribution, normal distribution, choosing correct distribution

UNIT– III

Sampling and Sampling Distributions-Random sampling, Non-Random Sampling distributions, operational considerations in sampling.
Estimation-Point estimates, interval estimates, confidence intervals, calculating interval estimates of the mean and proportion, t-distribution, determination of
Sample size in estimation.

UNIT– IV

Testing Hypothesis-one sample tests-Hypothesis testing of mean when the Population standard deviation is known, powers of hypotheses test, hypotheses

testing of proportions, hypotheses testing of means when standard deviation is not known. Testing Hypotheses-Two sample tests-Tests for difference between means- large sample, small sample, with dependent samples, testing for difference between proportions – Large sample.

UNIT–V

Chi-square and Analysis of Variance - chi-square as test of independence, chi-square as a test of goodness off it, analysis of variance, inferences about a population variance, inferences about two population variances.

Regression and Correlation–Simple Regression-Estimation using regression line, correlation analysis, making inferences about population parameters, limitations, errors and cave at sin regression and correlation analysis. Multiple

Regression and correlation analysis. Finding multiple regression equations and making inferences about population parameters.

Suggested Reading:

1	DavidCLay,LinearAlgebraanditsApplications4e.
2	Richard I Levin, David S Rubin - Statistics for Management, Seventh Edition, PHI -1997.
3	RD Sharma—Theory and Problems of Linear Algebra‖, International Publishing House Pvt. Limited, 2011.
4	AK Sharma,—Linear Algebra‖, Discovery Publishing House Ltd., 2019
5	Gilbert Strang, Linear Algebra and its Applications, 2010
6	S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics Sultan Chand & Sons, NewDelhi.

PCC106	MANAGERIAL ECONOMICS AND ACCOUNTANCY					
	CORE -VI					
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To learn important concepts of Managerial Economics and apply them to evaluate Business decisions.
2	To understand various parameters that determine the consumers 'behavior and to Evaluate the factors that affect production
3	To understand the concepts of capital budgeting and payback period and concepts of various book-keeping methods.

Course Outcomes:	
On completion of this course, the student will be able to :	
CO-1	Apply the fundamental concepts of managerial economics to evaluate business decisions Understand types of Demand and factors related to it
CO-2	Identify different types of markets and determine price–output under perfect competition
CO-3	Determine working capital requirement and payback Analyse and interpret financial statements through ratios

UNIT– I

Meaning and Nature of Managerial Economics: Managerial Economics and its usefulness to Engineers, Fundamental Concepts of Managerial Economics - Scarcity, Marginalism, Equi-marginalism, Opportunity costs, Discounting, Time Perspective, Risk and Uncertainty, Profits, Case study method.

UNIT– II

Law of Demand and Supply: Law of Demand, Determinants, Types of Demand; Elasticity of Demand (Price, Income and Cross-Elasticity); Demand Forecasting, Law of Supply and Concept of Equilibrium.

UNIT– III

Theory of Production and Markets :

Production Function, Law of Variable Proportion, ISO-quants, Economics of Scale, Cost of Production (types and their measurement), Concept of Opportunity Cost, Concept of Revenue, Cost–Output relation.

UNIT– IV

Working Capital Management and Capital Budgeting: Concepts, Significance, determination and estimation of fixed and variable, **working capital requirements, sources of capital. Introduction to capital budgeting**, methods –traditional and modern methods with problems.

UNIT–V

Accounting: Meaning – Significance – Principles of double entry bookkeeping, Journal, Ledger accounts, Subsidiary books, Trial Balance, preparation of Final Accounts with simple adjustments, Analysis and interpretation of Financial Statements through Ratios. (Theory questions and numerical problems on preparation of final accounts, cash book, petty cash book, bank reconciliation statement, calculation of some ratios)

Suggested Reading:

1	Mehta P.L., Managerial Economics - Analysis, Problems and Cases, Sultan Chand & Sons Educational Publishers, 2011
2	Maheswari S.N., Introduction to Accountancy, Vikas Publishing House, 2005
3	Pandey I.M., Financial Management, Vikas Publishing House, 2009
4	SP Jain and KL Narang,— Financial Accounting ǁ, Kalyan Publishers, 2018
5	M Hanif and A Mukherjee— Modern Accountancy ǁ, Mc Graw Hill,3rd Edition, 2018

LCC 151	C PROGRAMMING LAB					
Pre-requisites			**L**	**T**	**P**	**C**
			-	-	2	1.5
Evaluation	**SEE**	50 Marks	**CIE**		25Marks	

Course Objectives:	
1	To use tools available under LINUX for C programming
2	To gain hands-on experience on basic constructs of C programming
3	To formulate problems and implement algorithmic solutions in C
4	To write modular programs in C using structure programming techniques and data files.

Course Outcomes:	
On completion of this course, the student will be able to:	
CO-1	Write, compile and debug C programs in Linux environment
CO-2	Write simple programs using control structures, user defined functions and data manipulation using arrays
CO-3	Use standard C library functions to develop modular programs in C

1. Introducing to programming Environment (Linux commands, editing tools such as vi editor, sample program entry, compilation and execution)

2. Write programs using arithmetic, logical, bitwise and ternary operators.

3. **Write programs simple control statements:** Roots of a Quadratic Equation, extracting digits of integers, reversing digits, finding sum of digit ,printing multiplication tables, Armstrong numbers, checking for prime, magic number,

Using C Programming, Solve the following:

4. Sin x and Cos x values using series expansion

5. Recursion: Factorial, Fibonacci, GCD

6. Finding the maximum, minimum, average and standard deviation of given set of numbers using arrays.

7. Reversing an array, removal of duplicates from array

8. Set Union, Intersection, Matrix addition, multiplication and transpose of a square matrix using functions.

9. Functions of string manipulation: inputting and outputting string, using string functions such as strlen ( ), strcat( ), strcpy( )……etc

10. Writing simple programs for strings without using string functions.

11. Finding the No. of characters, words and lines of given text file

12. File handling programs: student memo printing

LCC152	JAVA PROGRAMMING LAB					
	LAB – II					
Pre-requisites			L	T	P	C
			-	-	2	1.5
Evaluation	SEE	50 Marks	CIE		25 Marks	

Course Objectives:	
The course is taught with the objectives of enabling the student to:	
1	Understand and implement basic programs using Java
2	Apply multithreaded concepts in problem solving
3	Implement serialization programs

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	Use appropriate data structure in a given application
CO-2	Implement functions and multithreaded concepts for a given application
CO-3	Write URL class programs
CO-4	Write serialization programs

Programs

1. Write a program to calculate salary of n employees using concept of classes with constructors and methods.
2. Write a program to demonstrate e-commerce website using inheritance, abstract class and dynamic polymorphism.
3. Write a program to demonstrate various arithmetic calculations using packages.
4. Write a program to demonstrate client-server environment using multithreading.
5. Write a program to demonstrate mutual exclusion using thread synchronization.
6. Write a program to demonstrate Linked List class.
7. Write a program to demonstrate Hash Set and Iterator classes.
8. Write a program to demonstrate Enumeration and Comparator interfaces.
9. Write a program to accept data and display output in key, value pair.
10. Write a program to create a registration form with different controls, menus and demonstrate event handling.
11. Write a program to copy data from one file to another file.
12. Write a program to merge contents of two files and display output on console.
13. Write a program to illustrate Serialization.
14. Write a program to retrieve web page using URL class.
15. Write a program to load and display image and perform grayscale.

HSC153	SOFT SKILLS LAB					
	LAB – III					
Pre-requisites			L	T	P	C
			-	-	2	1.5
Evaluation	SEE	50 Marks	CIE		25Marks	

Course Objectives :	
The course is taught with the objectives of enabling The student to:	
1	Conversation skills and reading strategies
2	Time management and stress management

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	Express conversational skills and Specify reading strategies
CO-2	Perform time management and Perform stress management
CO-3	Explore career planning

Activities

1. **Conversation skills, Listening dialogues from TV / Radio / TED Talk / Podcast**
2. Group discussion
3. Interview skills, Making presentation
4. Listening to Lectures and News Programmes, Listening to Talk Show
5. Watching videos on interesting events on YouTube
6. Reading different genres of texts ranging from newspapers to philosophical treatises
7. Reading strategies – Graphic organizers
8. Reading strategies – Summarizing
9. Reading strategies – Interpretation, Reports
10. Cover letter, Resume
11. Writing for publications, Letters, Memos, Emails and Blogs
12. Civil Service (Language related), Verbal ability
13. Motivation, Self-image
14. Goal setting, Managing changes
15. Time management, Stress management
16. Leadership traits
17. Teamwork
18. Career and life planning
19. Multiple intelligences
20. Emotional intelligence
21. Spiritual quotient (Ethics)
22. Intercultural communication
23. Creative and critical thinking
24. Learning styles and strategies

Suggested Readings:

1	Business English Certificate Materials, Cambridge University Press
2	Graded Examinations in Spoken English and Spoken English for Work Downloadable materials from Trinity College, London
3	International English Language Testing System Practice Tests, Cambridge University Press
4	Interactive Multimedia Programs on Managing Time and Stress
5	Personality Development (CD-ROM), Times Multimedia, Mumbai
6	RobertMSherfield—DevelopingSoftSkills‖4thEdition,PearsonEducation,2009.

Web Sources

http://www.slideshare.net/rohitjsh/presentation-on-group-discussion
http://www.washington.edu/doit/TeamN/present_tips.html
http://www.oxforddictionaries.com/words/writing-job-applications
http://www.kent.ac.uk/careers/cv/coveringletters.htm
http://www.mindtools.com/pages/article/newCDV_34.htm

SEMESTER–II

PCC 201	OPERATING SYSTEMS					
	CORE -VII					
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To gain the understanding of operating system and the details of process.
2	To learn the types and architecture of computer memory and file system and its implementation.
3	To realize the operating system concepts into case studies.

Course Outcomes:	
On completion of this course, the student will be able to:	
CO-1	Illustrate the workings of various OS components and process states.
CO-2	Demonstrate paging, demand paging, page replacement, and segmentation with illustrations.
CO-3	Elaborate the file access and allocation methods.

UNIT– I

Unix : Introduction, Commands, File System, Security and File Permission, Regular Expression and grep, Shell Programming, awk

Introduction to Operating Systems: OS Structure and Strategies, Process Concepts, Multithreaded Programming, Process Scheduling, Process Synchronization, Deadlocks

UNIT– II

Memory management strategies with example architectures: Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Virtual Memory Management, Demand Paging, Page Replacement, Thrashing.

UNIT– III

File System Interface: File Concepts, Access Methods and Protection

File System Implementation: File System Structure, Allocation Methods, Directory Implementation of File Systems, Mass Storage Structures, I/O Systems

UNIT– IV

System Protection: Principles and Domain, Access Matrix and Implementation, Access Control and Access Rights, Capability Based Systems, Language Based Protection

System Security: Problem, Program Threats, Cryptography, User Authentication, Implementing Security Defenses, Firewalling, Computer Security Classification.

UNIT–V
Case Studies: The Linux System–Design principles, Kernel modules, Process management, Scheduling, Memory management, File systems, Input and Output, Inter process communication. Windows7 – Design principles, System components, Terminal services and Fast user switching File systems, Networking, Programmer interface.

Suggested Reading:

1	Abraham Silber Schatz, Peter B Galvin, Operating System Concepts, 9th Edition, Wiley, 2016
2	William Stallings, Operating Systems – Internals and Design Principles, 8th Edition, Pearson, 2014
3	Andrew S Tanenbaum, Modern Operating Systems, 4th Edition, Pearson, 2016.

PCC 202	DATABASE MANAGEMENT SYSTEMS					
CORE -VII						
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To learn concepts along with ER modeling and about relational databases
2	To learn SQL query language and advanced SQL
3	To understand the transactions and explore No SQL

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	Explain the concepts and model requirements as ER-model.
CO-2	Suggest relational algebra queries from text specification
CO-3	Elaborate indexing and hashing and describe concurrency control concepts No SQL technology

UNIT– I

Introduction: Database System Applications, Purpose of Database Systems, View of Values, Nested Sub-queries, Complex Queries, Views, Modification of the Database, Joined Relations Data, Database Languages, Relational Databases, Database Design, Object–based and Semi-Structured Databases, Data Storage and Querying, Transaction Management, Data Mining and Analysis, Database Architecture, Database Users and Administrators. Database Design and the **E-R Model**: Over view of the Design Process, The Entity- Relationship Model, Constraints, Entity-Relationship Diagrams, Entity – Relationship Design Issues, Weak Entity Sets, Extended E-R Features, Database Design for Banking Enterprise, Reduction to Relational Schemas, Other Aspects of Database Design

UNIT– II

Relational Model: Structure of Relational Databases, Fundamental Relational-Algebra Operations, Additional Relational–Algebra Operations, Extended Relational-Algebra Operations, Null Values, Modification of the Databases.
Structured Query Language: Data Definition, Basic Structure of SQL Queries, Set Operations, Aggregate Functions, Null

UNIT– III

Advanced SQL: SQL Data Types and Schemas, Integrity Constraints, Authorization, Embedded SQL, Dynamic SQL, Functions and Procedural Constructs, Recursive Queries, Advanced SQL Features. Relational Database Design: Features
Of Good Relational Design, Atomic Domains and First Normal Form,

Functional - Dependency Theory, Decomposition using Functional Dependencies.

UNIT– IV

Indexing and Hashing:

Basic Concepts, Ordered Indices, B+-tree Index Files, B-tree Index Files, Multiple-Key Access, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Index Definition in SQL.

Transactions:

Transaction Concepts, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability.

UNIT–V

Concurrency Control:

Lock-based Protocols, Timestamp-based Protocols, Validation-based Protocols, Multiple Granularity, Multi-version Schemes, Deadlock Handling, Insert and Delete Operations, Weak Levels of Consistency, Concurrency of Index Structures.

Recovery System:

Failure Classification, Storage Structure, Recovery and Atomicity, Log-Based Recovery, Recovery with Concurrent Transactions, Buffer Management, Failure with Loss of Non volatile Storage, Advanced Recovery Techniques, Remote Backup Systems.

NoSQL:

Need for NoSQL, Aggregate Data Models, More Details on Data Models, Distribution Models, Consistency, Version Stamps, Map-Reduce, Key-Value Databases, Document Databases, Column-Family Stores, Graph Databases, Schema Migrations.

Suggested Reading:

1	Abraham Silberschatz, Henry F. Korth, S. Sudarshan, *Database System Concepts*, McGraw-Hill International Edition, 6th Edition, 2010.
2	Ramakrishnan, Gehrke, *Database Management Systems*, McGraw-Hill International Edition, 3rd Edition, 2003.
3	Elmasri, Navathe, Somayajulu, *Fundamentals of Database Systems*, Pearson Education, 4th Edition, 2004.
4	ShashankTiwari,―ProfessionalNoSQL‖,1st Edition,Wileypublishers,2011.

PCC 203	DESIGN AND ANALYSIS OF ALGORITHMS					
CORE -IX						
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To learn algorithms time complexity and various data structures.
2	To learn divide and conquer approach and greedy method
3	To learn dynamic programming and backtracking methods

Course Outcomes:	
On completion of this course, the student will be able to:	
CO-1	Understand the algorithm's time complexity and various data structures**.**
CO-2	Apply divide and conquer approach and greedy method based on the applications
CO-3	Analyze the dynamic programming and backtracking techniques

UNIT– I

Introduction to Algorithms: Algorithm Specification, Performance Analysis, Randomized Algorithms.
Elementary Data Structures: Stacks and Queues, Trees, Dictionaries, Priority Queues, Sets and Disjoint Set Union, Graphs.

UNIT– II

Divide and Conquer: Binary Search, Finding the Maximum and Minimum, Merge Sort; Quick Sort, Selection sort, Strassen's Matrix Multiplication, Convex Hull.
TheGreedyMethod:KnapsackProblem,TreeVertexSplitting,JobSequencingwith Deadlines, Minimum-Cost Spanning Trees, Single Source Shortest Paths.

UNIT– III

Dynamic Programming: General Method, Multistage Graphs, All-Pairs Shortest Paths, Single-Source Shortest Paths, Optimal Binary Search Trees, 0/1Knapsack, The Traveling Salesperson Problem.
Basic Traversal and Search Techniques: Techniques for Binary Trees, Techniques for Graphs, Connected Components and Spanning Trees, Biconnected Components and DFS.

UNIT– IV
Back Tracking: General Method, 8-Queens Problem, Sum of Subsets, Graph Coloring, Hamiltonian Cycles, Knapsack Problem. **Branch-Bound**: The Method, 0/1 Knapsack Problem, Traveling Sales Person.

UNIT–V
NP-Hard and NP-Complete Problems: Basic Concepts, Cook's Theorem, NP-Hard. Graph Problems, NP-Hard Scheduling Problems, NP-Hard Code Generation, Some Simplified NP-Hard Problems.

Suggested Reading:

1	E Horowitz, S Sahni, S Rajasekaran, "Fundamentals of Computer Algorithms", Second Edition, Universities Press, 2007.
2	R. Pannerselvam, "Design and Analysis of Algorithms", PHI, 2007.
3	Hari Mohan Pandey, "Design, Analysis and Algorithm", University Science Press, 2009.
4	TH Cormen, CE Leiserson, RL Rivert, CStein, "Introduction to Algorithms‖, Third Edition, PHI, 2010.

PCC 204	DATA ENGINEERING WITH PYTHON					
	CORE -X					
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To learn how to extra craw data and clean data and basics of Python programming
2	To perform transformations on data
3	To load data and visualize the data

Course Out comes:	
On completion of this course the student will be able to :	
CO-1	Understand the python fundamentals and regular expressions
CO-2	Apply relational databases and file operations to extract data.
CO-3	Analyze the tabular numeric data and visualize data using Pyplotlibraries

UNIT– I
Introduction, Parts of Python Programming Language, Control Flow Statements, Functions, Strings, Lists, Dictionaries, Tuples and sets, Files, Regular expressions

UNIT– II
Introduction to Data Science, Data Science: Data Analysis Sequence, Data Acquisition Pipeline, Report Structure **Files and Working with Text Data:** 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. **Working with Text Data:** JSON and XML in Python

UNIT– III
Working with Text Data: Processing HTML Files, Processing Texts in Natural Languages **Regular Expression Operations:** Using Special Characters, Regular Expression Methods, Named Groups in Python Regular Expressions, Regular Expression with *glob* Module **Working with Databases:** Setting Up a MySQL Database, Using a MySQL Database: Command Line, Using a MySQL Database, Taming Document Stores: MongoDB

UNIT– IV
Working with Data Series and Frames: Pandas Data Structures Reshaping Data Handling Missing Data, Combining Data, Ordering and Describing Data, Transforming Data, Taming Pandas File I/O **Plotting**: Basic Plotting with Py Plot, getting to Know Other Plot Types, Mastering Embellishments, Plotting with Pandas

UNIT–V
Probability and Statistics: Reviewing Probability Distributions, Recollecting Statistical measures, Doing Stats the Python way Machine Learning: Designing a Predictive Experiment, Fitting a linear regression, Grouping Data with K- means Clustering. Surviving in Random Decision Forests.

Suggested Readings:

1	Data Science Essentials in Python: Collect, Organize, Explore, Predict, Value. Dmitry Zinoriev, The Pragmatic Programmers LLC, 2016
2	Python for Ever body: Exploring Data Using Python3. Charles R Severance, 2016
3	Python Data Analytics–Data Analysis and Science using Pandas, matplotli band the Python Programming Language. Fabio Nelli, Apress, 2015
4	Web site Scraping with Python. Using Beautiful Soup and Scrapy. Gábor LászlóHajba, A press, 2018
5	Machine Learning with Python Cook book:. Practical Solutions from Preprocessing to Deep Learning. Chris Albon, O'Reilly 2018.

PCC 206	OPERATIONS RESEARCH					
	CORE -XII					
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To learn linear programming and transportation problems
2	To learn classification and assignment problems and its solutions
3	To understand the gaming theory and its applications

Course Outcomes:	
On completion of this course, the student will be able to :	
CO-1	Solve the linear programming problems and transportation problems
CO-2	Analyze the assignment problems and its solutions
CO-3	Explore various gaming theories and its applications

UNIT– I

Linear Programming: Introduction, Concept of Linear Programming Model, Development of LP models, Graphical Method, Linear Programming Methods, Special Cases of Linear Programming, Duality, Sensitivity Analysis.

UNIT– II

Transportation Problem: Introduction, Mathematical Model for 'Transportation Problem, Types of Transportation Problem, Methods to solve Transportation Problem, Transshipment Model.

UNIT– III

Assignment Problem: Introduction, Zero-One Programming Model, Types of Assignment Problem, Hungarian Method, Branch-and-Bound Technique for Assignment Problem.
Integer Programming: Introduction, Integer Programming Formulations, The Cutting-Plane Algorithm, Branch-and-Bound Technique, Zero-One Implicit Enumeration Algorithm.

UNIT– IV

Dynamic Programming: Introduction, Applications of Dynamic Programming, Solution of Linear Programming Problem through Dynamic Programming. Basics of Queuing theory.

UNIT–V

Game Theory: Introduction, Game with Pure Strategies, Game with Mixed Strategies,

Dominance Property, Graphical Method for 2 x norm x 2 Games, Linear Programming Approach for Game Theory.

Suggested Readings:

1	Pannarselvam,"*Operations Research*", 3rd Edition, PHI, 2009
2	Prem Kumar Gupta, DS Hira, "*Problems in Operations Research*", S.Chand, 2010.
3	Rathindra P Sen, "*Operations Research – Algorithm and Application*", PHI,2010.
4	J K Sharma, "*Operations Research*", Fourth Edition, MacMillan, 2009.

PCC 207	DATA STRUCTURES					
	CORE -II					
Pre-requisites			L	T	P	C
			4	-	-	4
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To understand the concepts of data structures and their role in algorithm design.
2	To learn how to implement various linear and non-linear data structures using the C programming language.
3	To analyze the performance of different data structures and algorithms in terms of time and space complexity.
4	To develop problem-solving skills for real-world applications using appropriate data structures.
5	To gain proficiency in memory management techniques, including dynamic memory allocation, for building data structures.

Course Outcomes:	
On completion of this course, the student will be able to :	
CO-1	Apply fundamental concepts of arrays, pointers, and structures in C to represent and manipulate different types of data.
CO-2	Implement linear data structures (stacks, queues, linked lists) and analyze their applications in solving computational problems.
CO-3	Develop and compare tree and graph representations using C, and apply standard traversal/searching techniques.
CO-4	Apply sorting and searching algorithms, evaluate their time and space complexities, and choose appropriate algorithms for problem contexts
CO-5	Design and implement real-world applications by integrating suitable data structures with efficient memory management and error handling in C

UNIT– I

Introduction to Data Structures and Arrays

- **Introduction to Data Structures:**
  - Definition of data structures, algorithms, and abstract data types (ADT).
  - Classification of data structures: linear and non-linear.
  - Need for data structures and their applications.
  - Algorithm analysis: Time and space complexity, Big O notation.
- **Arrays:**
  - Introduction, declaration, and initialization of arrays in C.
  - Operations on arrays: Traversal, insertion, deletion, searching, and sorting.
  - Applications of arrays: Polynomial representation using arrays.
  - Sparse matrices: Introduction and representation.
- **Pointers and Dynamic Memory Allocation:**
  - Pointers in C: Review of pointer concepts.
  - Dynamic memory allocation: malloc(), calloc(), realloc(), and free().
  - Implementing dynamic arrays.

- **Structures and Unions:**
    - User-defined data types in C.
    - Structures: Declaration, initialization, and accessing members.
    - Pointers to structures.
    - Self-referential structures.

UNIT– II

Stacks and Queues

- **Stacks:**
    - Introduction to stacks: LIFO (Last-In, First-Out) principle.
    - Stack as an ADT.
    - Implementation of stacks using arrays and linked lists.
    - Stack operations: push(), pop(), peek(), and is_empty().
    - Applications of stacks:
        - Infix to postfix/prefix conversion.
        - Evaluation of postfix expressions.
        - Checking for balanced parentheses.
- **Queues:**
    - Introduction to queues: FIFO (First-In, First-Out) principle.
    - Queue as an ADT.
    - Implementation of queues using arrays and linked lists.
    - Queue operations: enqueue(), dequeue(), is_empty(), and is_full().
    - Types of queues:
        - Circular queues.
        - Deques (Double-Ended Queues).
        - Priority queues.

UNIT– III

Linked Lists

- **Introduction to Linked Lists:**
    - Concept of a node, data part and next pointer.
    - Advantages and disadvantages over arrays.
    - Types of linked lists.
- **Singly Linked Lists:**
    - Creation of a linked list.
    - Operations:
        - Traversal.
        - Insertion at the beginning, end, and a specific position.
        - Deletion from the beginning, end, and a specific position.
        - Searching for an element.
    - Implementation using C structures and pointers.
- **Other Types of Linked Lists:**
    - **Doubly Linked Lists:** Structure, creation, and operations (insertion, deletion).
    - **Circular Linked Lists:** Structure, creation, and operations.

- **Circular Doubly Linked Lists:** Structure and applications.

UNIT– IV

Trees

- **Introduction to Trees:**
  - Tree terminology: Root, node, leaf, parent, child, sibling, degree, level, height.
  - Types of trees.
- **Binary Trees:**
  - Definition and properties.
  - Binary tree representation using arrays and linked lists.
  - Binary tree traversal techniques:
    - **Inorder Traversal:** Recursive and non-recursive implementation.
    - **Preorder Traversal:** Recursive and non-recursive implementation.
    - **Postorder Traversal:** Recursive and non-recursive implementation.
  - Expression trees.
- **Binary Search Trees (BST):**
  - Definition and properties.
  - Operations on BST: Insertion, deletion, searching.
  - Finding minimum and maximum elements.
  - Deleting a node from a BST (handling different cases).
- **Heaps:**
  - Introduction to heaps.
  - Min-heap and Max-heap.
  - Heap creation and heap sort algorithm.

UNIT–V

Unit 5: Sorting, Searching, and Graphs

- **Sorting Algorithms:**
  - Introduction to sorting.
  - **Simple Sorting Algorithms:** Bubble Sort, Selection Sort, Insertion Sort (with complexity analysis).
  - **Efficient Sorting Algorithms:**
    - Merge Sort.
    - Quick Sort.
    - Heap Sort.
- **Searching Algorithms:**
  - **Linear Search:** Implementation and complexity.
  - **Binary Search:** Prerequisite of a sorted array, implementation, and complexity.
- **Graphs:**
  - Introduction to graphs: Definitions and terminology (vertex, edge, path, cycle, directed/undirected graph).
  - Graph representation:

- Adjacency matrix.
- Adjacency list.

- Graph traversal algorithms:
  - Breadth-First Search (BFS).
  - Depth-First Search (DFS).
- Applications of graphs.

Suggested Reading:

1	"Data Structures Through C in Depth" by S.K. Srivastava and Deepali Srivastava"
2	"Fundamentals of data structure in C" Horowitz, Sahani & Freed, Computer Science Press, 2007
3.	Data Structures Using C , ReemaThareja , Oxford University Press (India) 2nd Edition (Paperback)

LCC 251	OPERATING SYSTEMS LAB					
	LAB – IV					
Pre-requisites			L	T	P	C
			-	-	3	1
Evaluation	SEE	50 Marks	CIE		25Marks	

Course Objectives:	
The course is taught with the objectives of enabling the student to:	
1	Understand and implement shell scripting CPU schedule in algorithms
2	Apply memory management algorithms and synchronization methods
3	Explore file allocation strategies and disk scheduling algorithms

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	Use appropriate CPU scheduling algorithms for a given application
CO-2	Implement memory management algorithms and synchronization methods
CO-3	Write disk scheduling algorithms
CO-4	Write file allocation strategies

Programs

1. Unix Shell Commands
   a) File handling commands
   b) Directory handling commands
   c) General purpose commands
2. Unix Shell Scripts
   a) Print multiplication table of a given number using all loops
   b) Perform all arithmetic operations
   c) Print the type of a file
   d) Rename all files whose names end with .c as .old
   e) Display the number of lines in each text file in a given directory
3. Simulate the following CPU Scheduling Algorithms
   a) FCFS
   b) SJF
   c) Round Robin
   d) Priority
4. Write a C program to simulate the Producer–Consumer problem using Semaphores
5. Write a C program to simulate the concept of Dining Philosophers problem
6. Simulate MVT and MFT

7. Write a C program to simulate the following contiguous memory allocation techniques:
    - Worst Fit
    - Best Fit
    - First Fit
8. Simulate the following Page Replacement Algorithms
    a) FIFO
    b) LRU
    c) OPTIMAL
9. Simulate the following File Organization Techniques
    a) Single-level Directory
    b) Two-level Directory
10. Simulate the following File Allocation Strategies
    a) Sequential
    b) Indexed
    c) Linked
11. Simulate Banker's Algorithm for Deadlock Avoidance
12. Simulate Banker's Algorithm for Deadlock Prevention
13. Write a C program to simulate Disk Scheduling Algorithms:
    a) FCFS
    b) SCAN
    c) C-SCAN

LCC252	DATA ENGINEERING WITH PYTHON LAB					
	LAB – V					
Pre-requisites			L	T	P	C
			-	-	3	1
Evaluation	SEE	50 Marks	CIE		25Marks	

Course Objectives:	
The course is taught with the objectives of enabling the student to:	
1	Understand the process of Importing and Exporting the data.
2	Identify different techniques for data analysis and data visualization
3	Learn how to collect, store and manage data from multiplied at a sources

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	Demonstrate various data types in python and develop programs using files, exception handling, functions, classes in Python.
CO-2	Examine the process for importing and exporting the data.
CO-3	Demonstrated at a visualization techniques for Data Analysis
CO-4	Apply appropriate data collection and pre-processing methods

Programs & Libraries
In this course, students are expected to extract, transform, and load input data that can be text files, CSV files, XML files, JSON, HTML files, SQL databases, NoSQL databases, etc.
For doing this, they should learn the following Python libraries/modules:
pandas, numpy, Beautiful Soup, pymysql, pymongo, nltk, matplotlib

---

Datasets
For this laboratory, appropriate publicly available datasets can be studied and used. Examples:

- MNIST:http://yann.lecun.com/exdb/mnist/

A. Exercises (continued)

B. **Design a relational database** for a small application and populate the database. Using SQL, perform CRUD operations (**Create, Read, Update, Delete**).

C. **Create a Python MongoDB client** using the pymongo module. Using a collection object, practice functions for inserting, searching, removing, updating, replacing, and aggregating documents, as well as for creating indexes.

D. Write programs to create **NumPy arrays** of different shapes and from different sources, reshape and slice arrays, add array indexes, and apply arithmetic, logic, and aggregation functions to some or all array elements.

E. Write programs to use the **pandas data structures (Frames and Series)** as storage

containers and for a variety of data-wrangling operations, such as:

F. Single-level and hierarchical indexing
G. Handling missing data
H. Arithmetic and Boolean operations on entire columns and tables
I. Database-type operations (such as merging and aggregation)
J. Plotting individual columns and whole tables
K. Reading data from files and writing data to files

L. Additional Exercises (for learning and practice)

M. **1. Introduction to Python Programming**

A. Running instructions in Interactive Interpreter and a Python Script

B. Write a program to purposefully raise an **IndentationError** and correct it

C. Write a program to compute distance between two points (taking input from the user)

D. Write a program that takes **2 numbers as command-line arguments** and prints their sum

E. Program to display: **Your Name, Full Address, Mobile Number, College Name, Course Subjects**

F. Write a program for checking whether the given number is even or not

N. **2. Control Structures & Lists**

A. Program to find the largest of three integers using if-else

B. Program that receives a series of positive numbers and displays the numbers in order and their sum

C. Program to find the product of two matrices

D. Program to display two random numbers to be added → allow the student to enter the answer

E. If the answer is correct → display a **"Congratulations"** message

F. If the answer is incorrect → display the **correct answer**

G. Program that prints decimal equivalents of **1/2, 1/3, 1/4 … 1/10**

H. Program using a **while loop** that asks the user for a number and prints a count down from that number to zero

O. **3. Functions and Recursion**

A. Write recursive and non-recursive functions for the following:

P. GCD of two integers

Q. Factorial of a positive integer

R. To print Fibonacci Sequence upto given numbern

S. Todisplayprimenumberfrom2ton.

U. Function that accepts two arguments: a list and a number *n*. It displays all of the numbers in the list that are greater than *n*.

V. Function that accepts a string as an argument and returns the number of vowels and consonants that the string contains.

4. Files, Exceptions, Lists, Sets, Random Numbers

A. Program to write a series of random numbers in a file from 1 to *n* and display.
B. Program to write the content in a file and display it with a line number followed by a colon.
C. Program to display a list of all unique words in a text file.
D. Program to analyze two text files using set operations.
E. Write a program to print each line of a file in reverse order.
F. Write a program to count frequency of characters in a given file. Use character frequency totals to check whether the file is a Python program, a C program, or a plain text file.
G. Write a program to combine lists into a dictionary.

5. Object-Oriented Programming

A. Program to implement inheritance
B. Program to implement polymorphism

6. Data Analysis using NumPy

a. Create an array of 10 zeros
b. Create an array of even integers up to 50
c. Create a 3×3 matrix
d. Generate an array of 25 random numbers sampled from a standard normal distribution
e. Create an array of 20 linearly spaced points between 0 and 1
f. Demonstrate slicing and indexing operations
g. Get the sum of all columns in a matrix

7. Pandas –DataFrames

- Write a Python program to create and combine **student** and **subject** Data Frames.
- Create a DataFrame **Book** containing three vectors [Name, Price, Author]. Convert this DataFrame into a matrix and list the object using the operator as.

8. Exploratory Data Analysis (EDA)

Perform EDA on web-scraped data of **2021–22 NBA player stats**: basketball-reference.com

9. To determine the mean of a set of numbers. To plot the numbers in a bar plot and have a straight line run through the plot at the mean.

 To determine the median of a set of numbers. To plot the numbers in a bar plot and have a straight line run through the plot at the median.

 To determine the standard deviation. To plot the numbers in a bar plot and have a straight line run through the plot at the mean and another straight line run through the plot at mean + standard deviation.

More data set toper form data analysis

SourceoftheData:https://www.kaggle.com/chirin/africa-economic-banking-and-systemic-crisis-data/downloads/africa-economic-banking-and-systemic-crisis-data.zip/1

Dataset: https://www.kaggle.com/khalidative/crimeanalysis

LCC253	DATA BASE MANAGEMENT SYSTEMS LAB					
LAB – VI						
Pre-requisites			L	T	P	C
			-	-	3	1
Evaluation	SEE	50 Marks	CIE		25Marks	

Course Objectives:	
The course is taught with the objectives of enabling the Student to:	
1	Learn SQL queries and PL/SQL stored procedures
2	Learn Triggers and report generation methods
3	Learn data base application creation

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	Write SQL queries and PL/SQL stored procedures
CO-2	Apply Triggers and report generation methods
CO-3	Analyze the database application creation

Programs

Creation of Database (exercising the commands for creation):

1. Simple to complex condition query creation using SQL*Plus
2. Usage of Triggers and Stored Procedures
3. Creation of Forms for Student information, Library information, Payroll etc.
4. Writing PL/SQL procedures for data validation
5. Report generation using SQL reports
6. Creating password and security features for applications
7. Usage of File locking, Table locking facilities in applications
8. Creation of small full-fledged database application spreading over 3 sessions

Note: Thecreationofsampledatabaseforthepurposeoftheexperimentsisexpected to be pre-decided by the instructor.

LCC 254	DATA STRUCTURES LAB					
LAB – I						
Pre-requisites			L	T	P	C
			-	-	3	1
Evaluation	SEE	50 Marks	CIE		25Marks	

Course Objectives:	
The course is taught with the objectives of enabling the student to:	
1	Understand implement basic data structures using C
2	Apply linear and non-linear data structures in problem solving
3	Implement searching and sorting algorithms

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	Use appropriate linear data structure in a given application
CO-2	Implement functions and recursive functions in C fora given application
CO-3	Use a different search trees for practical problems
CO-4	Application string matching algorithms in different domains

List of Experiments

1. Implementation of Stacks, Queues (using both arrays and linked lists).
2. Implementation of circular queue using arrays.
3. Implementation of double ended queue (de queue) using arrays.
4. Implement a program to evaluate a given postfix expression using stacks.
5. Implement a program to convert a given infix expression to postfix form using stacks.
6. Implement the following operations on singly linked list: i) Creation ii) Insertion iii) Deletion iv) Traversal
7. Implement the following operations on doubly linked list: i) Creation ii) Insertion iii) Deletion iv) Traversal
8. Implement the following operations on circular linked list: i) Creation ii) Insertion iii) Deletion iv) Traversal
9. Implementation of Polynomial arithmetic using linked list.
10. Implementation of recursive and iterative traversals on binary tree.
11. Implementation of operations on binary tree (delete entire tree, copy entire tree, mirror image, level order, search for a node etc.)
12. Implementation of the following operations on binary search tree (BST): (a) Minimum key (b) Maximum key (c) Search for a given key (d) Delete a node with given key
13. Implement the following sorting algorithms: a) Bubble sort b) Selection sort c) Insertion sort (d) Merge sort (e) Quick sort (f) Heap sort
14. Implementation of recursive and non-recursive functions to perform the following searching operations for a key value in a given list of integers: i) Linear search ii) Binary search
15. Implementation of hashing with (a) Separate Chaining and (b) Open addressing methods.

LCC 255	MINI PROJECT					
MINI PROJECT						
Prerequisites			**L**	**T**	**P**	**C**
			-	**-**	**4**	**2**
Evaluation	**SEE**	**50 Marks**	**CIE**		**25 Marks**	

Course Objectives

1	To develop capability to analyze and solve real world problems with an emphasis on Applying / integrating knowledge acquired.
2.	To learn the communication and presentation of the project work

Course Outcomes

After completion of the course, Student will be able to	
CO-1	Analyze and solve real world problems.
CO-2	Implement the system using SQL, data structures, C/C++,JAVA, Python and different software engineering models.

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 mini project 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 Two (2) weeks of 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 noticeboard.
2. Givea30 minutes presentation followed by10minutesdiscussion.
3. Submit a technical write-up on the talk.

At least two teachers will be associated with the Mini Project 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

* * * * *

M.C.A (Master of Computer Applications) *with effect from academic year 2025-26*

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, U.C.E., O.U

MASTER OF COMPUTER APPLICATIONS (MCA)

SEMESTER-III

Type of course	Course Code	Course Name	Contact hours per week			Scheme of Evaluation		Credits
			L	T	P	CIE	SEE	
Core-XII	PCC 301	Software Engineering	3	-	-	40	60	3
Core-XIII	PCC 302	Computer Networks	3	-	-	40	60	3
Core- XIV	PCC 303	Artificial Intelligence and Machine Learning	3	-	-	40	60	3
Core -XV	PCC 304	Web Technologies	3	-	-	40	60	3
Core –XVI	PCC 305	Advanced Operating Systems	3	-	-	40	60	3
Professional Elective- I	PEC 311	Software Quality & Testing	3	-	-	40	60	3
	PEC 312	Distributed Systems						
	PEC 313	Internet of Things						
	PEC 314	Image Processing						
Professional Elective-II	PEC 321	Network Security	3	-	-	**40**	**60**	3
	PEC 322	Cyber Security						
	PEC 323	Information Retrieval System						
	PEC 324	Natural Language Processing						
Lab-VII	LCC 351	Computer Networks Lab	-	-	2	25	50	1
Lab-VIII	LCC 352	Software Engineering Lab	-	-	2	25	50	1
Lab-IX	LCC 353	Web Technologies Lab	-	-	2	25	50	1
TOTAL			**21**		**6**	**355**	**570**	**24**
SEMESTER-IV								
Core-XVII	PCC 306	Image Processing and Computer Vision	3	-	-	40	60	3
Core-XVIII	PCC 307	Generative AI	3	-	-	40	60	3
Professional Elective- III	PEC 411	Block Chain Technologies	3	-	-	40	60	3
	PEC 412	Big Data Analytics						
	PEC 413	Cloud Computing						
	PEC 414	Deep Learning						
Professional Elective- IV	PEC 421	Distributed Database Systems	3	-	-	40	60	3
	PEC 422	Digital Forensics						
	PEC 423	Optimization Techniques						
	PEC 424	Enterprise Architecture						
Open Elective	OE 431	Professional Ethics	3	-	-	40	60	3
	OE 432	Constitution of India						
	OE 433	Disaster Management						
	OE 434	Organization Behavior						
	OE 435	Intellectual Property & Cyber Law						
	OE 436	Environmental Science						
Project	Proj 401	Project Work	-	-	24	50	100	12
TOTAL			**15**	**-**	**24**	**250**	**400**	**27**
GRAND TOTAL			**72**	**-**	**52**	**1285**	**2090**	**98**

PCC301 **SOFTWARE ENGINEERING**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn the software problem and addressing it through various software processes
2. Study the SRS and software architecture
3. Understand planning and designing a software project
4. Comprehend the testing strategies and the need for performing testing
5. Learn how to carry out reengineering to the system and maintain it

Course Outcomes – Students will learn to

1. Apply software processes to solve software problem
2. Create SRS document and software architecture
3. Perform software planning in terms of staffing and scheduling
4. Create test cases and procedures
5. Re-engineer the developed software

UNIT I
The software Problem: Cost, Schedule and Quality, Scale and change,
Software Processes: Process and project, Component Software Processes, Software Development Process Models, Project management Process.

UNIT II
Software Requirements Analysis and Specification: Value of a good SRS, Requirements Process, Requirements Specification, Functional Specification with Use Cases, Other approaches for analysis.
Software Architecture: Role of Software Architecture Views, Component and connector view, Architectural styles for C & C view, Documenting Architecture Design, Evaluating Architectures.

UNIT III
Planning a Software Project: Effort Estimation, Project Schedule and staffing, Quality Planning, Risk Management Planning, Project Monitoring Plan, Detailed Scheduling. **Design**: Design concepts, Function oriented Design, Object Oriented Design, Detailed Design, Verification, Metrics.

UNIT IV
Coding and Unit Testing: Programming Principles and Guidelines, incrementally developing code, managing evolving code, unit testing, code inspection, Metrics. **Testing**: Testing Concepts, Testing Process, Black Box testing, White box testing, Metrics.

UNIT V

Maintenance and Re-engineering: Software Maintenance, supportability, Reengineering, Business process Reengineering, Software reengineering, Reverse engineering; Restructuring, Forward engineering, Economics of Reengineering.

Software Process Improvement: Introduction, SPI process, CMMI, PCMM, Other SPI Frameworks, SPI return on investment, SPI Trends.

Suggested Reading

1. Pankaj Jalote, "Software Engineering- A Precise Approach", Wiley India, 2010.
2. Roger. S.Pressman , "Software Engineering - A Practitioner's Approach", 7th Edition, McGraw Hill Higher Education, 2010.
3. Deepak Jain, "Software Engineering", Oxford University Press, 2008.
4. Rajib Mall, "Fundamentals of Software Engineering", 4th Edition, PHI Learning, 2014.
5. Ian Sommerville, "Software Engineering", 10th Edition, Addison Wesley, 2015.

PCC302 **COMPUTER NETWORKS**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Comprehend the fundamentals of computer networks
2. Learn the aspects relevant to physical and datalink layer
3. Understand network layer and its significance and functionality
4. Study transport layer and its operations
5. Learn the protocols implemented at application layer

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

1. Elaborate the network model
2. Explain transmission media and functions of datalink layer
3. Create routing tables based on DVR and LSR
4. Describe TCP and UDP protocols
5. Explain application layer protocols

UNIT I

Data Communications: Components - Direction of Data flow - networks - Components and Categories - types of connections - Topologies -Protocols and Standards - ISO/OSI model, TCP/IP. **Transmission Media** - Coaxial Cable - Fiber Optics - Line Coding - Modems - RS232 Interfacing.

UNIT II

Datalink Layer: Error detection and correction, CRC, Hamming code,

Flow Control and Error control , Stop and Wait protocol, Sliding Window protocol -go back-N ARQ - selective repeat ARQ .

MAC Layer: LAN - Pure and Slotted ALOHA, Ethernet IEEE 802.3 LAN Ethernet Efficiency Calculation, Bridges. ARP, RARP

UNIT III

Network Layer: - Distance Vector Routing, Link State Routing,

IP v4 addressing, Subnetting, CIDR., Introduction to IPv6 ICMP , IGMP,OSPF and BGP.

UNIT IV

Transport Layer: Services of transport layer, Multiplexing. Transmission Control Protocol (TCP) Congestion Control, timer management, Quality of services (QOS) and User Datagram Protocol (UDP)

UNIT V

Socket Programming: Primitive and Advance System calls, Iterative and concurrent client server programs

Application Layer: Domain Name Space (DNS) - SMTP - FTP - HTTP

Suggested Readings

1. Andrew S. Tanenbaum, "Computer Networks", Pearson Education; Fourth Edition, 2008.
2. Behrouz A. Forouzan, "Data communication and Networking", Tata McGraw-Hill, 2009.
3. James F. Kurose and Keith W. Ross, "Computer Networking: A Top-Down Approach Featuring the Internet", Pearson Education, 2006.
4. W Richard Stevents, Unix Network Programming, PHI,2003

PCC303	ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING					
CORE-IV						
Pre-requisites			**L**	**T**	**P**	**C**
			3	-	-	3
Evaluation	**SEE**	60 Marks	**CIE**		40 Marks	

Course Objectives :	
1	To familiarize the principles of Artificial Intelligence
2	To study the techniques for knowledge representation and inference
3	To learn the techniques involved in the creation of intelligent systems
4	To study different applications like Game Playing Expert Systems, machine learning and natural language processing

Course Outcomes :	
On completion of this course, the student will be able to:	
CO-1	Use different logical systems for inference over formal domain representations.
CO-2	Formalize a given problem in the language/framework of different AI methods
CO-3	Design and perform an empirical evaluation of different algorithms on a problem formalisation

UNIT– I

Overview of Artificial Intelligence: Introduction. The Turing Test, Strong AI versus Weak AI, Heuristics, Identifying Problems Suitable for AI, Applications and Methods, Early History of AI, Recent History of AI to the Present, AI in the New Millennium

Uninformed Search: Introduction: Search in Intelligent Systems, State-Space Graphs, Generate-and-Test Paradigm, Blind Search Algorithms, Implementing and Comparing Blind Search Algorithms

Informed Search: Introduction, Heuristics, Informed Search Algorithms – Finding Any Solution, The Best-First Search, The Beam Search, Additional Metrics for Search Algorithms, Informed Search – Finding An Optimal Solution, Informed Search – Advanced Search Algorithms.

UNIT– II

Search Using Games: Introduction, Game Trees and Minimax Evaluation, Minimax with Alpha-Beta Pruning, Variations and Improvements To Minimax, Games of Chance and the Expect mini max Algorithm, Game Theory

Logic in Artificial Intelligence: Introduction, Logic and Representation, Propositional Logic, Predicate Logic – Introduction, Several Other Logics

Knowledge Representation: Introduction, Graphical Sketches and the Human Window, Graphs and the Bridges of Königsberg Problem, Search Trees, Representational Choices, Production Systems, Object Orientation, Frames, Scripts and the Conceptual Dependency System, Semantic Networks, Associations, More Recent Approaches, Agents: Intelligent or Otherwise.

UNIT – III
Production Systems: Introduction, Background, Basic Examples, Production Systems and Inference Methods, Production Systems and Cellular Automata, Stochastic Processes and Markov Chains **Uncertainty in AI:** Introduction, Fuzzy Sets, Fuzzy Logic, Fuzzy Inferences, Probability Theory and Uncertainty **Expert Systems:** Introduction, Background, Characteristics of Expert Systems, Knowledge Engineering, Knowledge Acquisition, Case-Based Reasoning, More Recent Expert Systems

UNIT– IV
Machine Learning : Introduction, Machine Learning: A Brief Overview, The Role of Feedback in Machine Learning Systems, Inductive Learning, Learning With Decision Trees, Problems Suitable for Decision Trees, Entropy, Constructing A Decision Tree With ID3, Issues Remaining **Machine Learning :**Neural Networks Introduction, Rudiments of Artificial Neural Networks, McCulloch-Pitts Network, The Perceptron Learning Rule, The Delta Rule, Backpropagation, Implementation Concerns, Discrete Hopfield Networks, Application Areas

UNIT –V
Natural Language Understanding: Introduction, History of Natural Language Processing, Syntax and Formal Grammars, Semantic Analysis and Extended Grammars, Statistical Methods in NLP, Probabilistic Models for Statistical NLP, Linguistic Data Collections for Statistical NLP.

Suggested Reading:

1	Stephen Lucci, Danny Kopec. Artificial Intelligence iMercury Learning and Information. 2nd Edition. 2016
2	Russell, Norvig: Artificial Intelligence, A Modern Approach, Pearson Education, Second Edition. 2004
3	Rich, Knight, Nair: Artificial Intelligence, Tata McGraw Hill, Third Edition 2009
4	Saroj Kaushik. Artificial Intelligence. Cengage Learning. 2011

PCC304 **WEB TECHNOLOGIES**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn basics of HTML and DHTML
2. Understand the workings of event model
3. Study the java scripting language
4. Learn the VB scripts
5. Comprehend the active server pages

Course Outcomes

1. Write HTML and DHTML programs
2. Create programs on event models
3. Implement java script programs
4. Write VB script programs
5. Create ASP programs

UNIT I

HTML: Markup languages, common tags, header, test styling, linking images Formatting text, Unordered lists, nested and ordered list, Tabs-and formatting, Basic forms; Complex forms linking, Meta Tags. **Dynamic HTML**: Cascading style sheets in line styles, style element, External Style sheet, text flow and Box model, user style sheets.

UNIT II

Object model and collections: Object referencing, collections all, children frames, navigator object. **Event model**: ONCLICK, ONLOAD, Error Handling, ON ERRORS ONMOUSEMOVE, ONMOUSE OVER, ONMOUSEOUT, ONFOCUS, ONBLUR, ONSUBMIT. **Dynamic HTML**: Filters and transitions, Data binding with Tabular data control binding to IMO, TABLE, Structured graphics, Active controls.

UNIT III

Introduction to scripting, Java Script, Data types, Arithmetic's Equality relational, assignment increment, decrement operators, Java Script Control Structures- if, if-else, while. Java Script **Control Structures**: For, Switch, Do/while, break.

Programming modules, recursion, recursion vs iteration global functions arrays, using arrays, Reference and reference parameters, passing arrays to functions, multiple subscripted arrays, objects-math, string. Boolean and number.

UNIT IV

Client side scripting with VB Script, operations, Data types and control structures, Functions, Arrays, String manipulations, classes and objects. **Web Servers**: Personal Web server, Internet information server, Apache Web Server, Installation of a Web Server.

UNIT V

Active Sever Pages, Client side Scripting vs Server side Scripting, Server side Active X Component, ADO, file system objects, Session tracking, CGI and PERL5, String Processing and Regular Expressions, Server side includes, Cookies and PERL XML Document Type Definition, XML Parsers, Using XML with HTML.

Suggested Readings

1 Deiterl, Deitel & NIETO, "Internet & World Wide Web - How to Program", Pearson Education, Third Edition, 2004.

2 Steven Holzner, "HTML Black Book - Comprehensive Problem Server", Dream Tech Press, 2000.

3 B Sosinsky, V Hilley, "Programming the Web - An Introduction", MGH, 2004.

PCC305	ADVANCED OPERATING SYSTEMS					
CORE-V						
Pre-requisites			**L**	**T**	**P**	**C**
			3	-	-	3
Evaluation	**SEE**	60 Marks	**CIE**		40 Marks	

Course Objectives :	
1	Understand global view of distributed operating systems and provides theoretical foundation for distributed systems.
2	Study the characteristics of OS for Multiprocessor and Multicomputer.
3	Learn the issues related to designing OS.
4	Understand Security & protection in computer systems and mechanisms used in building multiprocessor operating systems.
5	Explore management of different resources in distributed systems.

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	Understand the concept of distributed system and foundations.
CO-2	Familiarize with advanced paradigms, architectures & protocols necessary in solve the challenges in design of advanced operating systems.
CO-3	Analysis of efficiency and proofs of correctness for multiple aspects in design of Advanced Operating Systems

UNIT – I

Architecture of Distributed Systems: Types, Distributed Operating System, Issues in Distributed Operating Systems, Theoretical Foundations: Global Clock, Lamport's Logical Clock, Vector Clocks, Global State, and Termination Detection.

UNIT – II

Distributed Mutual Exclusion: Classification, requirement, performance, non-token based algorithms, Lamport's algorithm, the Richart-Agarwala algorithm, token-based algorithm-Suzuki liasamil's broadcast algorithm, Singhals heuristic algorithm.

Deadlock Detection: Resource Vs Communication deadlock, A graph- theoretic model, prevention, avoidance, detection, control organization, centralized deadlock-detection algorithm, the completely centralized algorithm, the HO-Ramamoorthy algorithm. Distributed deadlock detection algorithm - path - pushing, edge-chasing, hierarchical deadlock detection algorithm, menace-muntz and Ho-Ramamoorthy algorithm. Agreement Protocols: The system model, the Byzantine agreement, and the consensus problem.

UNIT– III
Distributed File System: Mechanisms, Design Issues - Andrew File System. Design and implementation of a log structured file system.**Distributed Shared Memory:** Algorithms for Implementing DSM, Memory Coherence, Coherence Protocols, Design Issues. **Distributed Scheduling :** Issues in Load Distribution, Components of Algorithm, Stability Load Distributing Algorithm, Performance.

UNIT – IV
Failure Recovery: Backward, Forward Error Recovery in Concurrent Systems, Consistent Set of Check Points, Synchronous and Asynchronous Check Pointing and Recovery. **Fault Tolerance:** Commit Protocols, Non-Blocking Commit Protocols, Voting Protocols. **Protection and Security:** Access Matrix, Private Key, Public key, and Kerberos System.

UNIT –V
Multiprocessor Operating Systems: Motivation, Basic Multiprocessor System Architecture, Interconnection Networks for Multiprocessor Systems, Caching, Hypercube Architecture. Threads, Process Synchronization, Processor Scheduling, and Memory Management. **Database Operating System:** Concurrence Control, Distributed Databases, and Concurrency Control Algorithms.

Suggested Reading:

1	Singhal M, Shivaratri N.G, Advanced Concepts in Operating Systems, McGraw-Hill Intl., 1994.
2	Pradeep K Sinha, Distributed Operating Systems Concepts and Design, PHI, First Edition, 2002.
3	Andrew S. Tanenbaum, Distributed Operating Systems, Pearson Education India, First Edition, 2011.

PEC311 **SOFTWARE QUALITY AND TESTING**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn the essentials of software quality
2. Study methods to integrate software quality activities in the project
3. Understand the software quality metrics
4. Learn building software testing strategy
5. Comprehend testing various artifacts of a software project

Course Outcomes

1. Explain the essentials of software quality
2. Elaborate the methods to integrate software quality activities in the project
3. Describe the software quality metrics
4. Discuss building software testing strategy
5. Perform testing various artifacts of a software project

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 Reading

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

PEC312 **DISTRIBUTED SYSTEMS**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Understand the architecture, processes and communication of distributed system
2. Learn the naming and synchronization strategies
3. Study fault tolerance, and distributed object based system
4. Learn distributed file system and distributed web based system
5. Comprehend the distributed coordination based system and map reduce

Course Outcomes

1. Explain the architecture, processes and communication of distributed system
2. Elaborate the naming and synchronization strategies
3. Describe the fault tolerance and distributed object based system
4. Discuss the distributed file system and distributed web based system
5. Explain distributed coordination based system and map reduce

UNIT I

Introduction: Goals and Types of Distributed Systems

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

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

Communication: Fundamentals, Remote Procedure Call, 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 Positioning of Nodes, and Election Algorithms. **Consistency and Replication:** Introduction, Data-Centric Consistency Models, Client-Centric Consistency 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: Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security.

UNIT IV

Distributed File Systems: Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security.

Distributed Web-Based Systems: Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security.

UNIT V

Distributed Coordination-Based Systems: Introduction to Coordination Models, Architecture, Processes, Communication, Naming, Synchronization, Consistency and Replication, Fault Tolerance, and Security.

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. R.Hill, L.Hirsch, P.Lake, S.Moshiri, ― Guide to Cloud Computing, Principles and Practice‖, Springer, 2013.
3. R.Buyya, J.Borberg, A. Goscinski,‖ Cloud Computing-Principles and Paradigms‖, Wiley 2013.

PEC313 **INTERNET OF THINGS**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Discuss fundamentals of IoT and its applications and requisite infrastructure
2. Describe Internet principles and communication technologies relevant to IoT
3. Discuss hardware and software aspects of designing an IoT system
4. Describe concepts of cloud computing and Data Analytics
5. Discuss business models and manufacturing strategies of IoT products

Course Outcomes

Student will be able to

1. Understand the various applications of IoT and other enabling technologies.
2. Comprehend various protocols and communication technologies used in IoT
3. Design simple IoT systems with requisite hardware and C programming software
4. Understand the relevance of cloud computing and data analytics to IoT
5. Comprehend the business model of IoT from developing a prototype to launching a product.

UNIT- I

Introduction to Internet of Things

IOT vision, Strategic research and innnovation directions, IoT Applications, Related future technologies, Infrastructure, Networks and communications, Processes, Data Management, Security, Device level energy issues.

UNIT- II

Internet Principles and communication technology

Internet Communications: An Overview – IP, TCP, IP protocol Suite, UDP. IP addresses – DNS, Static and Dynamic IP addresses, MAC Addressess, TCP and UDP Ports, Application Layer Protocols HTTP, HTTPS, Cost Vs Ease of Production, Prototypes and Production, Open Source Vs Closed Source.

UNIT- III

Prototyping and programming for IoT

Prototyping Embedded Devices – Sensors, Actuators, Microcontrollers, SoC, Choosing a platform, Prototyping, Hardware platforms – Arduino, Raspberry Pi. Prototyping the physical design – Laser Cutting, 3D printing, CNC Milling.

Techniques for writing embedded C code: Integer data types in C, Manipulating bits - AND, OR, XOR, NOT, Reading and writing from I/ O ports. Simple Embedded C programs for LED Blinking, Control of motor using switch and temperature sensor for arduino board.

UNIT- IV

Cloud computing and Data analytics

Introduction to Cloud storage models -SAAS, PAAS, IAAS. Communication APIs, Amazon webservices for IoT, Skynet IoT Messaging Platform.

Introduction to Data Analytics for IoT - Apache hadoop- Map reduce job execution workflow.

UNIT- V

IoT Product Manufacturing - From prototype to reality

Business model for IoT product manufacturing, Business models canvas, funding an IoT Startup, Mass manufacturing - designing kits, designing PCB,3D printing, certification, Scaling up software, Ethical issues in IoT- Privacy, Control, Environment, solutions to ethical issues.

Suggested Readings

1. Internet of Things - Converging Technologies for smart environments and Integrated ecosystems, River Publishers.
2. Designing the Internet of Things, Adrian McEwen, Hakim Cassimally. Wiley India Publishers
3. Fundamentals of embedded software: where C meets assembly by Daneil W lewies, Pearson.
4. Internet of things -A hands on Approach, Arshdeep Bahga, Universities press.

EC314 **IMAGE PROCESSING**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Understand image processing fundamentals
2. Understand image transforms
3. Understand image enhancement
4. Understand image restoration and feature extraction
5. Understand image reconstruction

Course Outcomes

1. Learn image processing fundamentals
2. Learn image transforms
3. Learn image enhancement
4. Learn image restoration and feature extraction
5. Learn image reconstruction

UNIT I

Fundamentals- Need for DIP- Fundamental steps in DIP – Elements of visual perception -Image sensing and Acquisition – Image Sampling and Quantization – Imaging geometry, discrete image mathematical characterization.

UNIT II

Image Transforms - Two dimensional Fourier Transform - Properties – Fast Fourier Transform – Inverse FFT, Discrete cosine transform and KL transform. - Discrete Short time Fourier Transform- Wavelet Transform- Discrete wavelet Transform- and its application in Compression.

UNIT III

Image Enhancement - Spatial Domain: Basic relationship between pixels- Basic Gray level Transformations – Histogram Processing – Smoothing spatial filters- Sharpening spatial filters. **Frequency Domain**: Smoothing frequency domain filters- sharpening frequency domain filters Homomorphic filtering.

UNIT IV

Image Restoration:- Overview of Degradation models –Unconstrained and constrained restorations- Inverse Filtering ,Wiener Filter.

Feature Extraction: - Detection of discontinuities – Edge linking and Boundary detection- Thresholding- -Edge based segmentation-Region based Segmentation- matching-Advanced optimal border and surface detection- Use of motion in segmentation. Image Morphology – Boundary descriptors- Regional descriptors.

UNIT V

Image Reconstruction from Projections: - Need- Radon Transform – Back projection operator- Projection Theorem- Inverse Radon Transform.

Suggested Reading

1. Rafael C.Gonzalez & Richard E.Woods – Digital Image Processing – Pearson Education- 2/e – 2004.
2. Anil. K. Jain – Fundamentals of Digital Image Processing- Pearson Education-2003.
3. B. Chanda & D. Dutta Majumder – Digital Image Processing and Analysis – Prentice Hall of India – 2002
4. William K. Pratt – Digital Image Processing – John Wiley & Sons-2/e, 2004

PEC321 **NETWORK SECURITY**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Understand the significant aspects of network security
2. Comprehend secret and public key cryptography
3. Learn hash functions and digital signatures
4. Study the digital signatures and smart cards
5. Comprehend the applications of network applications

Course Outcomes

1. Explain the fundamentals of network security
2. Elaborate the concepts secret and public key cryptography
3. Elucidate the hash functions digital signatures
4. Describe the digital signatures and smart cards
5. Explain the applications of network security

UNIT-I

Introduction: Attributes of Security, Integrity, Authenticity, Non-repudiation, Confidentiality Authorization, Anonymity, Types of Attacks, DoS, IP Spoofing, Replay, Man-in-the-Middle attacks General Threats to Computer Network, Worms, Viruses, -Trojans

UNIT-II

Secret Key Cryptography: DES, Triple DES, AES, Key distribution, Attacks

Public Key Cryptography: RSA, ECC, Key Exchange (Diffie-Hellman), Java Cryptography Extensions, Attacks

UNIT-III

Integrity, Authentication and Non-Repudiation : Hash Function (MD5, SHA5), Message Authentication Code (MAC), Digital Signature (RSA, DSA Signatures), Biometric Authentication.

UNIT-IV

PKI Interface: Digital Certificates, Certifying Authorities, POP Key Interface, System Security using Firewalls and VPN's.

Smart Cards: Application Security using Smart Cards, Zero Knowledge Protocols and their use in Smart Cards, Attacks on Smart Cards

UNIT-V

Applications: Kerberos, Web Security Protocols (SSL), IPSec, Electronic Payments, E-cash, Secure Electronic Transaction (SET), Micro Payments, Case Studies of Enterprise Security (.NET and J2EE)

Suggested Reading

1. William Stallings, Cryptography and Network Security, 4th Edition. Pearson,. 2009.

2. Behrouz A Forouzan, Cryptography and Network Security, TMH, 2009

3. Joseph MiggaKizza, A Guide to Computer Network Security, Springer, 2010

4. Dario Cataiano, Contemporary Cryptology, Springer, 2010.

PEC322 **CYBER SECURITY**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Understand the policies and security evolution
2. Learn cyber security objectives and guidance
3. Study policy catalog and issues
4. Comprehend cyber management and infrastructure issues
5. Learn the cyber security case studies

Course Outcomes

1. Explain the policies and security evolution
2. Describe cyber security objectives and guidance
3. Discuss policy catalog and issues
4. Elaborate cyber management and infrastructure issues
5. Elucidate the case studies on cyber security

UNIT I: Policies and Security Evolution

Introduction - Cyber Security, Cyber Security policy, Domain of Cyber Security

Policy, Laws and Regulations

Cyber Security Evolution - Enterprise Policy, Technology Operations, Technology

Configuration, Strategy Versus, Policy, Cyber Security Evolution, Productivity,

Internet, E-Commerce, Counter Measures, Challenges.

UNIT II: Cyber Security Objectives and Guidance

Security Objectives - Cyber Security Metrics, Security Management Goals, Counting Vulnerabilities, Security Frameworks, E-Commerce Systems, Industrial Control Systems, Personal Mobile Devices, Security Policy Objectives, Guidance for Decision Makers, Tone at the Top, Policy as a Project.

Catalog Approach - Cyber Security Management, Arriving at Goals, Cyber Security Documentation, the Catalog Approach, Catalog Format, Cyber Security Policy Taxonomy

UNIT III: Policy Catalog and Issues

Cyber Security Policy Catalog - Cyber Governance Issues, Net Neutrality, Internet Names and Numbers, Copyright and Trademarks, Email and Messaging, Cyber User Issues, Malvertising, Impersonation.

Cyber user and conflict Issues - Appropriate Use, Cyber Crime, Geo location, Privacy, Cyber Conflict Issues, Intellectual property Theft, Cyber Espionage, Cyber Sabotage, Cyber Welfare.

UNIT IV: Cyber Management and Infrastructures Issues

Cyber Management Issues - Fiduciary Responsibility – Risk Management – Professional Certification – Supply Chain – Security

Cyber Infrastructure Issues - Principles – Research and Development – Cyber Infrastructure Issue – Banking and finance – Health care – Industrial Control systems.

UNIT V: Case Study

Government's Approach to Cyber Security Policy - Cyber security strategy-Brief history-Public policy development in the U.S Federal Government.

Espionage - The rise of cyber crime- Espionage and Nation-state Actions-Policy response to growing Espionage threats-Congressional Action.

Suggested Readings:

1. Jennifer L. Bayuk, J. Healey, P. Rohmeyer, Marcus Sachs, Jeffrey Schmidt, Joseph Weiss "Cyber Security Policy Guidebook" John Wiley & Sons 2012.
2. Rick Howard "Cyber Security Essentials" Auerbach Publications 2011.
3. Richard A. Clarke, Robert Knake "Cyberwar: The Next Threat to National Security & What to Do About It" Ecco 2010
   Dan Shoemaker "Cyber security The Essential Body of Knowledge", 1st edition, Cengage Learning 2011.

PEC323 **INFORMATION RETRIEVAL SYSTEM**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Understand IR strategies
2. Study basic retrieval utilities
3. Learn cross language IR
4. Comprehend efficiency aspects
5. Learn distributed IR

Course Outcomes

1. Explain IR strategies
2. Elucidate basic retrieval utilities
3. Discuss cross language IR
4. Describe efficiency aspects
5. Elaborate distributed IR

UNIT-I

Introduction to Retrieval. Strategies: Vector Space model, Probabilistic Retrieval.

Strategies Language Models: Simple Term Weights, Non Binary Independence Model.

UNIT-II

Retrieval Utilities: Relevance Feedback, Clustering, N-grams, Regression Analysis, Thesauri.

UNIT-III

Retrieval Utilities: Semantic Networks, Parsing, Cross-Language Information Retrieval:

Introduction, Crossing the Language Barrier.

UNIT-IV

Efficiency: Inverted Index, Query Processing, Signature Files, Duplicate Document Detection.

UNIT - V

Integrating Structured Data and Text: A Historical Progression, Information Retrieval as a Relational Application, Semi-Structured Search using a Relational Schema.

Distributed Information Retrieval: A Theoretical Model of Distributed Retrieval, Web Search.

Suggested Reading:

1. David A. Grossman, Ophir Frieder. “Information Retrieval - Algorithms and Heuristics”, Springer, 2nd Edition (Distributed by Universities Press), 2004.

2. Gerald J Kowalski, Mark T Maybury. “Information Storage and Retrieval Systems”, Springer, 2000.

3. Soumen Chakrabarti, “Mining the Web: Discovering Knowledge. from Hypertext Data", Morgan-Kaufmann Publishers, 2002.

4. Christopher D. Manning, Prabhakar Raghavan, Hinrich SchGtze, “An Introduction to Information Retrieval”, Cambridge University Press, Cambridge, England,-2009.

PEC324 **NATURAL LANGUAGE PROCESSING**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn elementary probability and information theory
2. Study the linguistic essentials
3. Comprehend statistical inference and word sense disambiguation
4. Understand evaluation measures and markov models
5. Learn probabilistic context free grammars

Course Outcomes – Learners on completion of the course, be able to

1. Explain elementary probability and information theory
2. Discuss the linguistic essentials
3. Describe statistical inference and word sense disambiguation
4. Elaborate evaluation measures and markov models
5. Elucidate probabilistic context free grammars

UNIT I

Introduction of Elementary Probability Theory, Essential Information Theory. Linguistic Essentials Corpus-Based Work Collocations.

UNIT II

Statistical Inference: Bins: Forming Equivalence Classes, Reliability vs. Discrimination, n-gram models, Building ngram models, An Information Theoretic Approach.

UNIT III

Word Sense Disambiguation: Methodological Preliminaries, Supervised and unsupervised learning, Pseudo words, Upper and lower bounds on performance, Supervised Disambiguation, Bayesian classification.

UNIT IV

Evaluation Measures, Markov Models: Hidden Markov Models, Use, General form of an HMM Part-of-Speech Tagging

UNIT-V

Probabilistic Context Free Grammars: Introduction of Clustering **Information Retrieval:** Background, The Vector Space Model.

Suggested Reading

1. Christopher D. Manning, HinrichSchutze, Foundations of Statistical Natural Language Processing, MIT Press, 1999.

2. James Allan, Natural Language Understanding, Pearson Education, 1994.

3. Tanveer Siddiqui, US Tiwary, Natural Language Processing and Information Retrieval, Oxford University Press, 2008.

LCC351 **COMPUTER NETWORKS LAB**

Credits : 1

Instruction	3P hrs per week	Duration of SEE	3 hours
CIE	25 marks	SEE	50 marks

Course Objectives

1. Understand basic commands of networks
2. Learn socket program implementation
3. Understand connection oriented socket programs
4. Learn connectionless socket programs
5. Understand DNS implementation

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

1. Execute basic commands of networks
2. Implement socket program implementation
3. Execute connection oriented socket programs
4. Implement connection less socket programs
5. Execute DNS implementation

Programs to be written on the following concepts using any programming language like Python, C, C++, Java.

1. Understanding and using of commands like ifconfig, netstat, ping, arp, telnet, ftp, finger, traceroute, whois.
2. Socket Programming: Implementation of Connection-Oriented Service using standard ports.
3. Implementation of Connection-Less Service using standard ports.
4. Implementation of Connection-Oriented Iterative Echo-Server, date and time, character generation using user-defined ports.
5. Implementation of Connectionless Iterative Echo-server, date and time, character generation using user-defined ports.
6. Implementation of Connection-Oriented Concurrent Echo-server, date and time, character generation using user-defined ports.
7. Program for connection-oriented Iterative Service in which server reverses the string sent by the client and sends it back.
8. Program for connection-oriented Iterative service in which server changes the case of the strings sent by the client and sends back (Case Server).
9. Program for Connection-Oriented Iterative service in which server calculates the net-salary of an employee based on the following details sent by the client

i) basic ii) hra iii) da iv) pt v) epf vi) net-salary=basic+hra+da-pt-epf).

10. Program for file access using sockets.
11. Program for Remote Command Execution using sockets .
12. Implementation of DNS.

LCC352 **SOFTWARE ENGINEERING LAB**

Credits : 1

Instruction	3P hrs per week	Duration of SEE	3 hours
CIE	25 marks	SEE	50 marks

Course Objectives

1. Learn use case diagram
2. Learn class and object diagram
3. Understand sequence and collaboration diagrams
4. Study state-chart and activity diagrams
5. Comprehend component and deployment diagrams

Course Outcomes

1. Apply use case diagram
2. Apply class and object diagram
3. Apply sequence and collaboration diagrams
4. Apply state-chart and activity diagrams
5. Apply component and deployment diagrams

1. Phases in software development project, overview, need, coverage of topics

2. To assign the requirement engineering tasks

3. To perform the system analysis: Requirement analysis, SRS

4. To perform the function-oriented diagram: DFD and Structured chart

5. To perform the user's view analysis: Use case diagram

6. To draw the structural view diagram: Class diagram, object diagram

7. To draw the behavioral view diagram: Sequence diagram, Collaboration diagram

8. To draw the behavioral view diagram: State-chart diagram, Activity diagram

9. To draw the implementation view diagram: Component diagram

10. To draw the environmental view diagram: Deployment diagram

11. To perform various testing using the testing tool unit testing, integration testing

Draw UML diagrams for the following system

1. ATM application
2. Library management system
3. Railway reservation
4. E-Commerce System
5. Banking System

Perform the following tasks

Background: Software has made the world a global village today. The impact of software spans across almost all aspect of human life. All organizations, Institutions and companies are leveraging the potentials of software in automating the critical functions and eliminating manual interventions. Software is also a predominant area for trade and export especially for the countries like India. Domains like health care, Airlines, financial Services, Insurance, retails, Education, and many more have exploited software and still there a lot of the scope for software to create impact and add values in multiple dimensions.

Problem Description: In the context of this background, identify the areas (or application or systems) how software has been leveraged extensively in the following domains

1. Health Care 2. Airlines 3. Banking Insurance

4. Retail 5. Education

Background: In the early years of computers applications, the focus of the development and innovation were on hardware. Software was largely views as an afterthought. Computer programming was an art. Programmers did not follow any disciplined or formalized approaches. This way of doing things was adequate for a while, until the sophisticated of computer applications outgrow. Software soon took over and more functions which were done manually. A software houses begin to develop for widespread distribution. Software development projects produced thousands of source program statement. With the increase in the size and complexity of the software, following situation resulted is collectively termed as software crisis.

1. Time Slippage

2. Cost Slippage

3. Failure at customer Site

4. Intractable Error after delivery

Problem Description: In the context of this background, for each of the scenario mentioned below, identify the most appropriate problem related to software crisis and mention the same in the table provided.

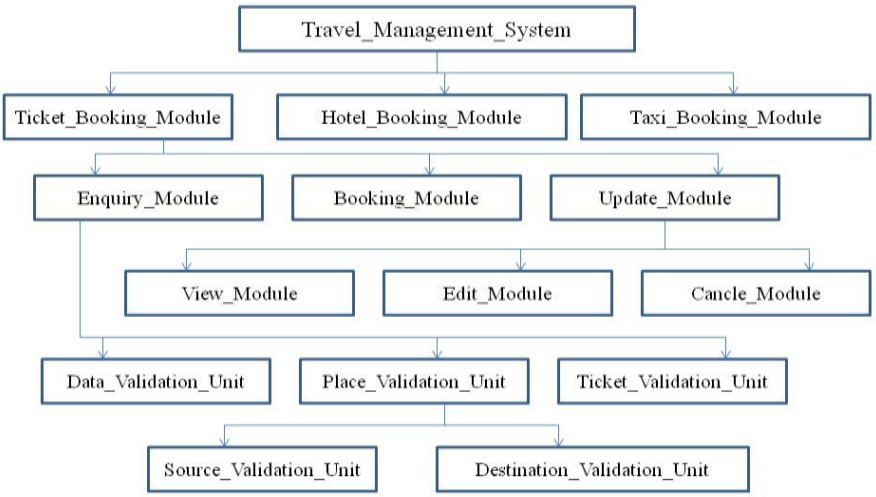
Scenario A: Railways reservation software was delivered to the customer and was installed in one of the metro station at 12.00 AM (mid-night) as per the plan. The system worked quite fine till the next day 12.00 PM (noon). The system crashed at 12.00 PM and the railways authorities could not continue using software for reservation till 02.00 PM. It took two hours to fix the defect in the software in the software.

Scenario B: A polar satellite launch vehicle was scheduled for the launch on August 15th. The auto-pilot of the rocket to be delivered for integration of the rocket on May 15th. The design and development of the software for the auto-pilot more effort because of which the auto-pilot was delivered for the integration on June 15th (delayed by a month). The rocket was launched on Sep 15th (delayed by a month).

Scenario C: Software for financial systems was delivered to the customer. Customer informed the development team about a mal-function in the system. As the software was huge and complex, the development team could not identify the defect in the software.

INTEGRATION TESTING

Background: Integration testing is carried out after the completion of unit testing and before the software is delivered for system testing. In top down integration testing, dummy stubs are required for bottom level modules. Similarly, in bottom up testing, dummy drivers are required for top level modules



Problem Description: Consider the scenario of development of software for Travel, Management System (TMS) is in progress. The TMS software has 3 major modules namely Ticket_Booking_Module, Hotel_Booking_Module and Taxi_Booking_Module. The Ticket_Booking_Module has 3 sub modules namely Enquiry_Module, Booking_Module and Update_Module. The enquiry module uses Date_Validation_Unit, Ticket_Validation_Unit and Place_Validation_Unit.

In the context of the given scenario, identify the usage of stub or driver for the following situations.

1. Except the Ticket_validation_Unit, the coding and unit testing of all other modules, sub modules and units of TMS are completed. The top-down integration is in progress for the TMS software. To carry out the integration testing, which among the following is necessary?

2. The coding and unit testing of all the module, sub modules and units of TMS are completed except the Update_Module (coding and testing for Edit_Module, Cancel_Module and View_Module are also completed). The bottom-up integration is to be started for the TMS software. Mention any stub or driver needed to carry out the integration testing?

3. Except the Taxi_Booking_Module, the coding and unit testing of all other modules, sub modules and units of TMS are completed. The top-down integration is to be started for the TMS software. Mention any stub or driver needed to carry out the integration testing.

Background: Performance testing tests the non-functional requirements of the system. The different types of performance testing are load testing, stress testing, endurance testing and spike testing.

Problem Description: Identify the type of performance testing for the following:

1. A space craft is expected to function for nearly 8 years in space. The orbit control system of the spacecraft is a real-time embedded system. Before the launch, the embedded software is to be tested to ensure that it is capable of working for 8 years in the space. Identify the suitable performance testing category to be carried out to ensure that the space craft will be functioning for 8 years in the space as required.

2. Global Education Centre (GEC) at Infosys Mysore provides the training for fresh entrants. GEC uses an automated tool for conducting objective type test for the trainees. At a time, a maximum of 2000 trainees are expected to take the test. Before the tool is deployed, testing of the tool was carried out to ensure that it is capable of supporting 2000 simultaneous users. Indicate the performance testing category?

3. A university uses its web-based portal for publishing the results of the students. When the results of an examination were announced on the website recently on a pre-planned date, the web site crashed. Which type of performance testing should have been done during web-site development to avoid this unpleasant situation?

4. During unexpected terrorist attack, one of the popular websites crashed as many people logged into the web-site in a short span of time to know the consequences of terrorist attack and for immediate guidelines from the security personnel. After analyzing the situation, the maintenance team of that website came to know that it was the consequences of unexpected load on the system which had never happened previously. Which type of performance testing should have been done during web-site development to avoid this unpleasant situation?

Background: Enhancements are introduction of new features to the software and might be released in different versions. Whenever a version is released, regression testing should be done on the system to ensure that the existing features have not been disturbed.

Problem Description: Consider the scenario of development of software for Travel Management System (TMS) discussed in previous assignment. TMS has been developed by Infosys and released to its customer Advance Travel Solutions Ltd. (ATSL). Integration testing, system testing and acceptance testing were carried out before releasing the final build to the customer. However, as per the customer feedback during the first month of usage of the software, some minor changes are required in the Enquiry Module of the TMS. The customer has approached Infosys with the minor changes for upgrading the software. The development team of Infosys has incorporated. Those changes, and delivered the software to testing team to test the upgraded software. Which among the following statement is true?

a. Since minor changes are there, integration of the Enquiry Module and quick system testing on Enquiry module should be done.

b. The incorporation of minor changes would have introduced new bugs into other modules, so regression testing should be carried out.

c. Since the acceptance testing is already carried out, it is enough if the team performs sanity testing on the Enquire module.

d. No need of testing any module.

Background: There are some metrics which are fundamental and the rest can be derived from these. Examples of basic (fundamental) measures are size, effort, defect, and schedule. If the fundamental measures are known, then we can derive others. For example if size and effort are known, we can get Productivity (=size/effort). If the total numbers of defects are known we can get the Quality (=defect/size) and so on.

Problem Description: Online loan system has two modules for the two basic services, namely Car loan service and House loan service.

The two modules have been named as Car_Loan_Module and House_Loan_Module. Car_Loan_Module has 2000 lines of uncommented source code. House_Loan_Module has 3000 lines of uncommented source code. Car_Loan_Module was completely implemented by Mike. House_Loan_Module was completely implemented by John. Mike took 100 person hours to implement Car_Loan_Module. John took 200 person hours to implement

House_Loan_Module. Mike‟s module had 5 defects. Jonh's module had 6 defects. With respect to the context given, which among the following is an INCORRECT statement?

Choose one:

1. John‟s quality is better than Mike.

2. John‟s productivity is more than Mike.

3. John introduced more defects than Mike.

4. John‟s effort is more than Mike.

LCC353 **WEB TECHNOLOGIES LAB**

Credits : 1

Instruction	3P hrs per week	Duration of SEE	3 hours
CIE	25 marks	SEE	50 marks

1. Develop College Website using XHTML and CSS.
2. Develop HTML form with client validations using Java Script.
3. Publishing XML document using XSLT
4. XML document processing using SAX and DOM.
5. Text processing using Regular expressions and pattern matching.
6. Develop form processing application using CGI.pm
7. Develop CGI-Perl Web application with State and Session Tracking.
8. Develop a simple Java servlet application.
9. Develop Java servlet application with session tracking
10. Develop a simple JSP application.
11. Creation of an application access database with JDBC
12. Develop full-fledged web application with database access spreading over to 3 sessions.

SEMESTER – IV

PCC 306	**IMAGE PROCESSING AND COMPUTER VISION**					
Pre-requisites			**L**	**T**	**P**	**C**
			3	-	-	3
Evaluation	**SEE**	60 Marks	**CIE**		40 Marks	

Course Objectives :	
1	To study elements of visual perception , intensity transformations and spatial filtering and smoothing techniques
2	To introduce the concepts of filtering in frequency domain and image restoration
3	To learn the concepts of color image processing and multi-resolution processing
4	To introduce concepts Image formation from geometrical perspective, 3D reconstruction
5	To study statistical modeling techniques and inverse problems in vision

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	apply intensity transformations and Spatial filters on digital images
CO-2	use frequency domain filtering techniques Image Smoothing and restoration
CO-3	compress and segment color images, and use wavelet transforms in multi-resolution processing
CO-4	understand Pinhole Camera model and Multi-view stereo with N- cameras
CO-5	Use statistical modeling techniques and restore blurred images

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 Transform of 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

Color Image Processing: Color fundamentals, Color models, Pseudocolor Image Processing, Basics of Full-color Image Processing, Color Transformations, Smoothing and Sharpening, Colorbased Image Segmentation, Noise in Color Images, Color 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 formation from geometrical perspective: Pinhole Camera model, Epipolar Geometry, Camera parameters, Essential matrix, Fundamental Matrix, Camera Calibration (calibration from predefined pattern, calibration from scene structure)

Multiview 3D reconstruction: Stereo (depth from binocular view, rectification, one (or two) disparity map estimation technique(s)), Multi-view stereo (depth from 3 cameras, N number of cameras).

UNIT –V

Statistical Modelling of images: Markov Random field, Conditional random field, Gibb's sampling, Loopy Belief propagation based approximation.

3D reconstruction from single image: Shape from Shading, Depth from Defocus.
Structure from Motion: Triangulation, Two-frame structure from motion, Factorization, Bundle adjustment, Constrained structure and motion

Inverse problems in CV: Image restoration for images blurred by non-uniform motion. Super resolution (image registration and interpolation techniques)

Lightfield Photography: Definition, Capturing techniques, Fourier Slicing and Digital Re focusing
Computational Photography: HDR imaging, Super slo-mo video capturing.

Suggested Reading:

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.
3	*Multiple View Geometry in Computer Vision*. Second Edition. Richard *Hartley*. Australian
	National University, Canberra, Australia. Andrew *Zisserman*, 2004.
4	Anand Rangarajan, Rama Chellappa, ― Markov random field models in image processing‖ , MIT Press, 1995
5	Andrew Blake , Pushmeet Kohli, Carsten Rother, ―Markove Random Fields for Vision and Image Processing‖ , The MIT Press,2011
6	Ali Mohammed DJafari ―Inverse Problems in Vision and 3D Tomography ―, Wiley ,2010

PCC307	**GENERATIVE AI**					
CORE						
Pre-requisites	Deep Learning		**L**	**T**	**P**	**C**
			3	-	-	3
Evaluation	**SEE**	60 Marks	**CIE**		40 Marks	

Course Objectives :	
1	Understand and implement modern generative models for text, images, and other modalities
2	Adapt foundation models using prompting and fine-tuning techniques
3	Analyze scaling laws, attention mechanisms, and diffusion processes
4	Develop applied generative AI solutions with real-world impacts
5	Explore ethical considerations, risks, and interpretability challenges of generative AI

Course Outcomes :	
On completion of this course, the student will be able to:	
CO-1	Build and evaluate generative models like RNNs, Transformers, GANs, and VAEs
CO-2	Apply in-context learning, parameter-efficient tuning, and reinforcement learning from human feedback (RLHF)
CO-3	Analyze the architecture and optimization of large foundation models for diverse modalities
CO-4	Explore cutting-edge applications such as text-to-image generation, code generation, and autonomous agents
CO-5	Evaluate ethical, safety, and interpretability issues in generative AI systems

UNIT– I

Text Generation & Language Models :

- Introduction to RNN, LSTM, and Transformer-based language models, - Decoding strategies (sampling, beam search), Pre-training & fine-tuning, Foundation models (e.g., GPT, T5, BERT), Applications: Chatbots, text completion, summarization, LLM Agents & Tools:LangChain, AutoGen, ReAct framework — for building agents using LLM, Evaluation Techniques : BLEU, ROUGE, BERTScore, and newer factuality/hallucination metrics, Industry Use Cases:Customer support, document automation, voice assistants (integrate APIs like OpenAI, Cohere, Mistral)

UNIT – II

Generative Models for Images & Diffusion :

- CNNs and Vision Transformers, Generative Adversarial Networks (GANs), Diffusion models: Denoising Score Matching, DDPM , Variational Autoencoders (VAEs), Applications: Text-to-image (DALL·E), image inpainting, **Stable Diffusion XL (2023)** and **SD Turbo (2024)**: more efficient, widely used in industry, **Video Generation Models**: Sora by OpenAI, Runway Gen-3, Pika — include short case studies or lab demos, **Comparative Evaluation**: FID, CLIPScore, human preference.

UNIT – III

Adaptation & Control of Generative Models : In-context learning, Prompt engineering and Prompt-to-Prompt, Fine-tuning: LoRA, Adapter tuning, Reinforcement Learning from Human Feedback (RLHF), Applications: Instruction tuning, controlled generation, Direct Preference Optimization

DPO) and ORPO — for safety-aligned model fine-tuning, Function Calling / Tool Use, How models use APIs, tools, and planners (e.g., function-calling in OpenAI GPT-4o.

UNIT– IV
Scaling Laws & Efficient Training : Scaling laws in deep learning, Mixture-of-Experts (MoE), Efficient attention: Flash Attention, Long former, Parallel and distributed training, Applications: Efficient deployment of large models, A**gentic RAG**: planning, reflection, and tool use in retrieval-augmented systems, **Speculative Decoding**, **Token Merging**, **Quantization** — for faster, cheaper inference, **Mamba, RWKV** – efficient attention-free architectures

UNIT –V
Multimodal, Ethical & Emerging Applications - Multimodal models: CLIP, Flamingo, Video Generation, Generative models for code (Codex), agents (AutoGPT), Interpretability and hallucinations , AI alignment, safety, and bias mitigation, Multimodal Agent Systems (e.g., GPT-4o, Gemini, Claude 3) — integration of vision + language + tools, Synthetic Data Generation for model training, Legal & Policy Updates : AI Bill of Rights (US), EU AI Act, India's Digital India guidelines on AI, Open-source Trends: Mistral, LLaMA 3, Phi-3, TinyML for generative AI.

Suggested Reading:

1	Vaswani et al. (2017), Radford et al. (2019) for Unit-I topics
2	Goodfellow et al. (2014), Ho et al. (2020), Kingma& Welling (2014) for Unit-II topics
3	Ouyang et al. (InstructGPT), DPO (2023) for Unit-III topics
4	Kaplan et al. (2020), Shazeer et al. (MoE), DAO modelsfor Unit-IV topics
5	OpenAI Codex, DeepMind’s Flamingo, Survey on Hallucination in LLMsfor Unit-V topics
6	Transformers for Natural Language Processing by Denis Rothman
7	Toolkits- LangChain, AutoGen, LlamaIndex, Diffusers (by Hugging Face)

PEC411 **BLOCK CHAIN TECHNOLOGIES**

Credits : 3

Instruction 3L hrs per week Duration of SEE 3 hours

CIE 40 marks SEE 60 marks

Course Objectives

1. Learn the basic concept of Cryptographic Hash Functions, Hash Pointers
2. Study Elliptic Curve Digital Signature Algorithm.
3. A technical overview of decentralized digital currencies like Bitcoin, as well as their broader economic, legal and financial context.
4. To get an insight into the working of the Bitcoin network Wallet
5. Comprehend Bitcoin mining and distributed consensus for reliability.

Course Outcomes

1. Learn the basics of hash functions
2. Learn the importance of digital signature
3. Understand the structure of a blockchain.
4. Learn different ways of storing Bitcoin keys, security measures.
5. Learn how Bitcoin relies on mining.

UNIT – I

Introduction to Cryptography: Cryptographic Hash Functions, SHA-256, Hash Pointers and Data Structures, Merkle tree.

UNIT – II

Digital Signatures: Elliptic Curve Digital Signature Algorithm (ECDSA), Public Keys as identities, A Simple Cryptocurrency.

UNIT – III

Centralization vs Decentralization, Distributed consensus, Consensus without identity using a block chain, Incentives and proof of work.

Mechanics of Bitcoin: Bitcoin Transactions, Bitcoin Scripts, Applications of Bitcoin Scripts, Bitcoin Blocks, The Bitcoin Network.

UNIT – IV

Storage and Usage of Bitcoins: Simple Local Storage, Hot and Cold Storage, Splitting and Sharing Keys, Online Wallets and Exchanges, Payment Services, Transaction Fees, Currency

Exchange Markets.

UNIT – V

Bitcoin Mining: The Task of Bitcoin miners, Mining Hardware, Mining pools,

Mining incentives and strategies.

Bitcoin and Anonymity: Anonymity Basics, Mixing, Zerocoin and Zerocash.

Applications of Block Chain Technologies.

Suggested Reading

1. Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction by Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, Princeton Press, 2016.

2. Mastering Bitcoin: Programming the Open Block chain by Andreas M. Antonopoulos Shroff, O'Reilly; 2nd Edition, 2017.

PEC412 **BIG DATA ANALYTICS**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Understand big data fundamentals
2. Understand Learn hadoop ecosystem
3. Understand map reduce and hbase fundamentals
4. Understand database concepts related to big data
5. Understand NoSQL fundamentals

Course Outcomes

1. Learn how to handle big data
2. Learn hadoop ecosystem
3. Learn mapreduce and hbase fundamentals
4. Learn database concepts related to big data
5. Learn NoSQL fundamentals

UNIT– I

Getting an overview of Big Data: Introduction to Big Data, Structuring Big Data, Types of Data, Elements of Big Data, Big Data Analytics, Advantages of Big Data Analytics.

Introducing Technologies for Handling Big Data: Distributed and Parallel Computing for Big Data, Cloud Computing and Big Data, Features of Cloud Computing, Cloud Deployment Models, Cloud Services for Big Data, Cloud Providers in Big Data Market.

UNIT– II

Understanding Hadoop Ecosystem: Introducing Hadoop, HDFS and Map Reduce, Hadoop functions, Hadoop Ecosystem.

Hadoop Distributed File System-HDFS Architecture, Concept of Blocks in HDFS Architecture, Name nodes and Data nodes, Features of HDFS. Map Reduce.

Introducing HBase- HBase Architecture, Regions, Storing Big Data with HBase, Combining HBase and HDFS, Features of HBase, Hive, Pig and Pig Latin, Sqoop, Zoo Keeper, Flume, Oozie.

UNIT-III

Understanding Map Reduce Fundamentals and HBase: The Map Reduce Framework, Exploring the features of Map Reduce, Working of Map Reduce, Techniques to optimize Map Reduce Jobs,

Hardware/Network Topology, Synchronization, File system, Uses of Map Reduce, Role of HBase in Big Data Processing-Characteristics of HBase.

Understanding Big Data Technology Foundations: Exploring the Big Data Stack, Data Sources Layer, Ingestion Layer, Storage Layer, Physical Infrastructure Layer, Platform Management Layer, Security Layer, Monitoring Layer, Visualization Layer.

UNIT– IV

Storing Data in Databases and Data Ware houses: RDBMS and Big Data, Issues with Relational Model, Non–Relational Database, Issues with Non-Relational Database, Polyglot Persistence, Integrating Big Data with Traditional Data Warehouse, Big Data Analysis and Data Warehouse.

UNIT–V

NoSQL Data Management: Introduction to NoSQL, Characteristics of NoSQL, History of NoSQL, Types of NoSQL Data Models-Key Value Data Model, Column Oriented Data Model, Document Data Model, Graph Databases, Schema-Less Databases, Materialized Views, CAP Theorem.

Suggested Reading

1. BIG DATA, Black Book TM, Dream Tech Press, 2016 Edition.
2. Seema Acharya, Subhasni Chellappan, “BIG DATA and ANALYTICS”, Wiley publications, 2016
3. Nathan Marzand James Warren, “BIG DATA-Principles and Best Practices of Scalable Real-Time Systems”, 2010

PEC413	CLOUD COMPUTING		
	Credits : 3		
Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn the cloud computing services including resource virtualization
2. Study the scaling, planning and file system and storage
3. Understand database technology and security issues
4. Comprehend portability issues and programming model case study
5. Learn the enterprise architecture and its related information

Course Outcomes

1. **Elaborate the cloud computing services and resource virtualization**
2. Explain the scaling, planning and file system and storage
3. Describe the database technology and security issues
4. Elucidate portability issues and programming model case study
5. Discuss the enterprise architecture and its related information

UNIT-I

Introduction, Benefits and challenges, Cloud computing services, Resource Virtualization, Resource pooling sharing and provisioning, Case study of Iaas, Paas and Saas

UNIT-II

Scaling in the Cloud, Capacity Planning, Load Balancing, File System and Storage, Containers

UNIT-III

Multi-tenant Software, Data in Cloud, Database Technology, Content Delivery Network, Security Reference Model, Security Issues, Privacy and Compliance Issues

UNIT-IV

Portability and Interoperability Issues, Cloud Management and a Programming Model Case Study, Popular Cloud Services

UNIT-V

Enterprise architecture and SOA, Enterprise Software, Enterprise Custom Applications, Work flow and Business Processes, Enterprise Analytics and Search, Enterprise Cloud Computing Eco system.

Suggested Reading

1. Cloud Computing-Sandeep Bhowmik, Cambridge University Press, 2017.
2. Enterprise Cloud Computing-Technology, Architecture, Applications by Gautam Shroff, Cambridge University Press, 2016.
3. Kai Hwang, Geoffrey C.Fox, Jack J.Dongarra, “Distributed and Cloud Computing From Parallel Processing to the Internet of Things”, Elsevier, 2012.

PEC414 **DEEP LEARNING**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn deep learning basics and optimization algorithms
2. Understand deep learning computation, CNNs and modersn CNNs
3. Study recurrent neural networks and its modern versions
4. Learn computer vision
5. Comprehend GANs

Course Outcomes

1. Learn deep learning basics and optimization algorithms
2. Understand deep learning computation, CNNs and modersn CNNs
3. Study recurrent neural networks and its modern versions
4. Learn computer vision
5. Comprehend GANs

UNIT I

Introduction – A motivating example, the key components – data, models, and algorithms, kinds of machine learning, the road to deep learning, success stories, Linear regression, multilayer perceptrons, model selection, under fitting and overfitting, weight decay, dropout, forward propagation, backward propagation and computational graphs, numerical stability and initialization, considering the environment, predicting house prices on Kaggle

Optimization algorithms – optimization and deep learning, convexity, gradient descent, momentum, adagrad, RMSProp, Adadelta, Adam, learning rate scheduling

UNIT II

Deep learning computation – layers and blocks, parameter management, deferred initialization, custom layers, file I/O, GPUs

Convolutional neural networks – from dense layers to convolutions, convolutions for images, padding and stride, multiple input and output channels, pooling, convolutional neural networks (Le Net)

Modern Convolutional neural networks – deep Convolutional neural networks (Alex Net), Networks using blocks (VGG), Networks with parallel concatenations (Google Net), batch normalization, residual networks (Res Net), densely connected networks (Dense Net)

UNIT III

Recurrent neural networks – sequence models, text processing, language models and the dataset, recurrent neural networks, back propagation through time

Modern Recurrent neural networks – gated recurrent units (GRU), long short term memory (LSTM), deep recurrent neural networks, bidirectional recurrent neural networks, machine translation and the dataset, encoder-decoder architecture, sequence to sequence

Attention mechanisms – attention mechanism, sequence to sequence with attention mechanism, transformer

UNIT IV

Computer vision – Image augmentation fine tuning, object detection and bounding boxes, anchor boxes, multiscale object detection, the object detection dataset, single shot multibox detection (SSD), region based CNNs (R-CNNs), semantic segmentation and the dataset, transposed convolution, fully convolutional networks (FCN), neural style transfer, image classification(CIFAR-10) on kaggle, dog breed identification (Imagenet dogs) on kaggle

UNIT V

Generative adversarial networks – Generative adversarial networks, deep convolutional generative adversarial networks

Tools for deep learning – using jupyter, using amazon sagemaker, using AWS EC2 instances, using google colab, selecting servers and GPUs

Suggested Reading

1. Ian good fellow, Yoshuabengio, Aaron courville, “Deep learning”
   Zhang, Aston, et al. "Dive into deep learning." arXiv preprint arXiv:2106.11342 (2021).

PEC421 **DISTRIBUTED DATABASE SYSTEMS**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objects:

The aim of the course is to

- Enhance the previous knowledge of database systems by deepening the understanding of the theoretical and practical aspects of the database technologies, and showing the need for distributed database technology to tackle deficiencies of the centralized database systems;
- Introduce basic principles and implementation techniques of distributed database systems,
- Expose active and emerging research issues in distributed database systems and application development,
- Apply theory to practice by building and delivering a distributed database query engine, subject to remote Web service calls.

Course Outcomes:

After the completion of the course, the students are expected to

1. Get familiar with the currently available models, technologies for and approaches to building distributed database systems and services;
2. Have developed practical skills in the use of these models and approaches to be able to select and apply the appropriate methods for a particular case;
3. Be aware of the current research directions in the field and their possible outcomes;
4. Be able to carry out research on a relevant topic, identify primary references, analyze them, and come up with meaningful conclusions
5. Be able to apply learned skills to solving practical database related tasks.

UNIT- I

Introduction: Database-System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Object-Based and Semistructured Databases, Data Storage and Querying, Transaction Management, Data Mining and Analysis, Database Architecture, Database Users and Administrators, History of Database Systems. **Relational Model**: Structure of Relational Databases, Fundamental Relational-Algebra Operations, Additional

Relational-Algebra Operations, Extended Relational-Algebra Operations, Null Values, Modification of the Database.

UNIT-II

Query Processing: Overview, Measures of query cost, Selection operation, sorting, Join operation, other operations, Evaluation of Expressions.

Query Optimization: Overview, Transformation of Relational expressions, Estimating statistics of expression results, Choice of evaluation plans, Materialized views.

UNIT-III

Parallel Systems: Speedup and Scaleup, Interconnection Networks, Parallel Database Architectures.

Parallel Databases: Introduction, I/O Parallelism, Interquery Parallelism, Intraquery Parallelism, Interoperation Parallelism, Intraoperation Parallelism, Design of Parallel Systems.

UNIT-IV

Distributed Databases: Reference architecture for DDB, Types of Data Fragmentation, Distribution Transparency for Read-only applications, Distribution Transparency for Update applications, Distributed

Database Access Primitives, Integrity Constraints in DDB.

Distributed Database Design: A frame work for Distributed Database Design, The design of Database fragmentation, The allocation of fragmentation.

UNIT-V

Translation of Global Queries to Fragment Queries: Equivalence transformations for queries, Transforming global queries into fragment queries, Distributed grouping and aggregate function evaluation, Parametric queries.

Optimization of Access Strategies: Access Control Models, Database Security, A framework for query optimization, Join queries, General queries.

Suggested Reading:

1. Silberschatz A, Korth HF, Sudarshan S, *Database System Concepts*, McGraw-Hill International Edition, 5th Edition, 2006.

2. Ceri S, Pelagatti G, *Distributed Databases: Principles and Systems*, McGraw-Hill International Edition, 1984.

PEC422 **DIGITAL FORENSICS**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	41 marks	SEE	60 marks

Course Objectives

1. Understand the basic digital forensics and techniques for conducting the forensic examination on different digital devices.
2. Understand how to examine computing investigations
3. Understand data acquisition
4. Understand processing crimes
5. Understand forensics tools

Course Outcomes

1. Know how to apply forensic analysis tools to recover important evidence for identifying computer crime.
2. To be well-trained as next-generation computer crime investigators.
3. Learn data acquisition
4. Learn processing crimes
5. Learn forensics tools

UNIT -I

Computer forensics fundamentals, Benefits of forensics, computer crimes, computer forensics evidence and courts, legal concerns and private issues.

UNIT- II

Understanding Computing Investigations – Procedure for corporate High-Tech investigations, understanding data recovery work station and software, conducting and investigations.

UNIT-III

Data acquisition- understanding storage formats and digital evidence, determining the best acquisition method, acquisition tools, validating data acquisitions, performing RAID data acquisitions, remote network acquisition tools, other forensics acquisitions tools.

UNIT-IV

Processing crimes and incident scenes, securing a computer incident or crime, seizing digital evidence at scene, storing digital evidence, obtaining digital hash, reviewing case.

UNIT-V

Current computer forensics tools- software, hardware tools, validating and testing forensic software, addressing data-hiding techniques, performing remote acquisitions, E-Mail investigations-investigating email crime and violations, understanding E-Mail servers, specialized E-Mail forensics tool.

Suggested Readings:

1. Warren G. Kruse II and Jay G Heiser, "Computer Forensics: Incident Response Essentials", Addison Wesley, 2002.

2. Nelson, B, Phillips, A, Enfinger, F, Stuart, C., "Guide to Computer Forensics and Investigations, 2nd ed., Thomson Course Technology, 2006, ISBN: 0-619-21706-5.

3. Vacca, J, *Computer Forensics, Computer Crime Scene Investigation*, 2nd Ed, Charles River Media, 2005, ISBN: 1-58450-389.

PEC423 **OPTIMIZATION TECHNIQUES**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Understand the optimization basics
2. Understand optimization using calculus
3. Understand dynamic programming and its applications
4. Understand integer programming
5. Understand advanced optimization techniques

Course Outcomes

1. Learn the optimization basics
2. Learn optimization using calculus
3. Learn dynamic programming and its applications
4. Learn integer programming
5. Learn advanced optimization techniques

UNIT I

Introduction and Basic Concepts

Historical Development; Engineering applications of Optimization; Art of Modeling Objective function; Constraints and Constraint surface; Formulation of design problems as Optimization techniques –classical and advanced techniques

UNIT II

Optimization using Calculus

Stationary points; Functions of single and two variables; Global Optimum Convexity and concavity of functions of one and two variables Optimization of function of one variable and multiple variables; Gradient vectors; Examples Optimization of function of multiple variables subject to equality constraints; Lagrangian function Optimization of function of multiple variables subject to equality constraints; Hessian matrix formulation; Eigen values Kuhn-Tucker Conditions; Examples

UNIT III

Dynamic Programming

Sequential optimization; Representation of multistage decision process; Types of multistage decision problems; Concept of sub optimization and the principle of Optimality Recursive equations –Forward

and backward recursions; Computational procedure in dynamic programming (DP) Discrete versus continuous dynamic programming; Multiple state variables; curse of dimensionality in DP

UNIT IV

Dynamic Programming Applications

Problem formulation and application in Design of continuous beam and Optimal geometric layout of a truss Water allocation as a sequential process Capacity expansion and Reservoir operation

Integer Programming

Integer linear programming; Concept of cutting plane method Mixed integer programming; Solution algorithms; Examples

UNIT V

Advanced Topics in Optimization

Piecewise linear approximation of a nonlinear function Multi objective optimization –Weighted and constrained methods; Multi level optimization Direct and indirect search Methods Evolutionary algorithms for optimization and search Applications in civil engineering

Suggested Reading:

1. S.S. Rao, "Engineering Optimization: Theory and Practice", New Age International P. Ltd., New Delhi, 2000.
2. G. Hadley, "Linear programming", Narosa Publishing House, New Delhi, 1990.
3. H.A. Taha, "Operations Research: An Introduction", 5th Edition, Macmillan, New York, 1992.
4. K. Deb, "Optimization for Engineering Design-Algorithms and Examples", Prentice-Hall of India Pvt. Ltd., New Delhi, 1995.

K. Srinivasa Raju and D. Nagesh Kumar, "Multicriterion Analysis in Engineering and Management", PHI Learning Pvt. Ltd., New Delhi, India, ISBN 978-81-203-3976-7, pp.288, 2010.

PEC424 **ENTERPRISE ARCHITECTURE**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn the fundamentals of EA
2. Study the business architecture
3. Understand the organizational structure of EA
4. Comprehend enterprise engineering
5. Gain insights into cloud computing opportunities for EA

Course Outcomes

1. Learn the fundamentals of EA
2. Study the business architecture
3. Understand the organizational structure of EA
4. Comprehend enterprise engineering
5. Gain insights into cloud computing opportunities for EA

UNIT I

Introduction to EA -System analysis, general system theory, definitions and objectives of considerations, Properties of EA, system approach to EA development, principle definitions

UNIT II

Business architecture, definition and features, BSC – balanced score card basics and its reflection in EA, Strategic governance, Event Causality effects in EA under scope of BSC

UNIT III

Organizational structure of EA and basic models, Information and technology architecture basics, Introduction to EA structuring and modeling, Business architecture (inc. business process modeling, IBM Component business model), Information architecture, Technology architecture and integration between the layers model

UNIT IV

Introduction to enterprise engineering (EE), Enterprise transformations (waterfall and agile), EAP, EA methodologies: PRISM, ARIS Framework, Zachmann Framework , FEAF, DODAF and TOGAF,

Introduction to Service orientation in Enterprise Engineering (SOA, SoEA), Technological infrastructure for Big Data handling in EA

UNIT-V

Cloud Computing Opportunities for EA, Flexible (agile) business and information architectures (SoEA).

Introduction to Spark, Spark Data Frames, SQL, Datasets through worked examples.

Spark's low level APIs, RDDS, execution of SQL & Data Frames.

How Spark Runs on a Cluster.

Structured Streaming, Spark's Stream – Processing Engine.

Suggested Reading

1. Designing Enterprise Architecture Frameworks: Integrating Business Processes with IT Infrastructure by N Zarvić, R Wieringa. Apple Academic Press (19 April 2016), 360 p. URL: https://doi.org/10.1201/b16417
2. Neubauer M., Stary CH., S-BPM in the Production Industry. Stakeholder approach, Springer Open, 2017. URL: https://www.springer.com/gp/book/9783319484655
3. A systematic literature review on Enterprise Architecture Implementation Methodologies by Babak D., Mohd N. Elsevier (June 2015), p. 1-20. URL: https://doi.org/10.1016/j.infsof.2015.01.012
5. Spark : The Definite Guide – Bill Chambers, MateiZaharia, 2018.

OE431 **PROFESSIONAL ETHICS**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn the developments of legal profession in India
2. Study the seven lamps of advocacy
3. Understand disciplinary proceedings
4. Comprehend the accountancy for lawyers
5. Gain insights into safety and risk

Course Outcomes

1. Explain the developments of legal profession in India
2. Describe the seven lamps of advocacy
3. Elaborate disciplinary proceedings
4. Elucidate the accountancy for lawyers
5. Discuss insights into safety and risk

UNIT-I

Development of Legal Profession in India — Advocates Act, 1961 — Right to Practice — a right or privilege? -- Constitutional guarantee under Article 19(1) (g) and its scope — Enrolment and Practice — Regulation governing enrolment and practice — Practice of Law — Solicitors firm — Elements of Advocacy.

UNIT-II

Seven lamps of advocacy — Advocates duties towards public, clients, court, and other advocates and legal aid ; Bar Council Code of Ethics.

UNIT-III

Disciplinary proceedings --- Professional misconduct — Disqualifications — Functions of Bar Council of India/State Bar Councils in dealing with the disciplinary proceedings — Disciplinary Committees -- Powers and functions - Disqualification and removal from rolls.

UNIT-IV

Accountancy for Lawyers - Nature and functions of accounting — Important branches of accounting — Accounting and Law – Bar - Bench Relations.

UNIT- V

Safety and Risk – Assessment of Safety and Risk – Risk Benefit Analysis and Reducing Risk – Respect for Authority – Collective Bargaining – Confidentiality – Conflicts of Interest – Occupational Crime – Professional Rights – Employee Rights – Intellectual Property Rights (IPR) – Discrimination.

Suggested Readings

1. Myneni S.R, Professional Ethics, Accountancy for Lawyers and Bench-Bar Relation, Asia

2. Gupta S.P, Professional Ethics, Accountancy for Lawyers and Bench-Bar Relation, Asia Law House, Hyderabad.

3. Kailash Rai, Professional Ethics, Accountancy for Lawyers and Bench-Bar Relation, Allahabad Law Agency.

4. Siroh, Professional Ethics, Central Law Publications, Allahabad.

5. Ramachandra Jha, Selected Judgements on Professional Ethics, Bar Council of India Trust.

OE432 **CONSTITUTION OF INDIA**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn the basics of the constitution
2. Understand the structure of the union government
3. Comprehend the state government structure
4. Gain insights into local administration
5. Study about the election commission

Course Outcomes

1. Explain the basics of the constitution
2. Elucidate the structure of the union government
3. Elaborate the state government structure
4. Describe the local administration
5. Discuss the election commission

UNIT I – The Constitution - Introduction

• The History of the Making of the Indian Constitution

• Preamble and the Basic Structure, and its interpretation

• Fundamental Rights and Duties and their interpretation

• State Policy Principles

UNIT II – Union Government

• Structure of the Indian Union

• President – Role and Power

• Prime Minister and Council of Ministers

• Lok Sabha and Rajya Sabha

UNIT III– State Government

• Governor – Role and Power

• Chief Minister and Council of Ministers

• State Secretariat

UNIT IV – Local Administration

• District Administration

• Municipal Corporation

• Zila Panchayat

UNIT V – Election Commission

• Role and Functioning

• Chief Election Commissioner

• State Election Commission

Suggested Readings

1. Ethics and Politics of the Indian Constitution Rajeev Bhargava Oxford University Press, New Delhi, 2008
2. The Constitution of India B.L. Fadia Sahitya Bhawan; New edition (2017)
3. Introduction to the Constitution of India DD Basu Lexis Nexis; Twenty-Third 2018 edition

Suggested Software/Learning Websites

1. https://www.constitution.org/cons/india/const.html
2. http://www.legislative.gov.in/constitution-of-india
3. https://www.sci.gov.in/constitution
4. https://www.toppr.com/guides/civics/the-indian-constitution/the-constitution-of-india/

OE433 **DISASTER MANAGEMENT**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. To learn about various types of natural and man-made disasters.
2. To know pre- and post-disaster management for some of the disasters.
3. To know about various information and organisations in disaster management in India.
4. To get exposed to technological tools and their role in disaster management.

Course Outcomes

After competing this course, student will be

1. Acquainted with basic information on various types of disasters
2. Knowing the precautions and awareness regarding various disasters
3. Decide first action to be taken under various disasters
4. Familiarised with organisation in India which are dealing with disasters
5. Able to select IT tools to help in disaster management

UNIT – I: Understanding Disaster

Understanding the Concepts and definitions of Disaster, Hazard, Vulnerability, Risk, Capacity, Disaster and Development, and disaster management.

UNIT – II: Types, Trends, Causes, Consequences and Control of Disasters

Geological Disasters (earthquakes, landslides, tsunami, mining); Hydro-Meteorological Disasters (floods, cyclones, lightning, thunder-storms, hail storms, avalanches, droughts, cold

and heat waves) Biological Disasters (epidemics, pest attacks, forest fire);

Technological Disasters (chemical, industrial, radiological, nuclear) and Manmade Disasters

(building collapse, rural and urban fire, road and rail accidents, nuclear, radiological, chemicals and biological disasters) Global Disaster Trends – Emerging Risks of Disasters – Climate Change and Urban Disasters.

UNIT- III: Disaster Management Cycle and Framework

Disaster Management Cycle – Paradigm Shift in Disaster Management.

Pre-Disaster – Risk Assessment and Analysis, Risk Mapping, zonation and Microzonation,

Prevention and Mitigation of Disasters, Early Warning System; Preparedness, Capacity Development; Awareness. During Disaster – Evacuation – Disaster Communication – Search and Rescue – Emergency Operation Centre – Incident Command System – Relief and Rehabilitation –Post-disaster – Damage and Needs Assessment, Restoration of Critical Infrastructure – Early Recovery – Reconstruction and Redevelopment; IDNDR, Yokohama Stretegy, Hyogo Framework of Action.

UNIT– IV: Disaster Management in India

Disaster Profile of India – Mega Disasters of India and Lessons Learnt.

Disaster Management Act 2005 – Institutional and Financial Mechanism,

National Policy on Disaster Management, National Guidelines and Plans on Disaster Management; Role of Government (local, state and national), Non-Government and Inter Governmental Agencies

UNIT– V: Applications of Science and Technology for Disaster Management

Geo-informatics in Disaster Management (RS, GIS, GPS and RS). Disaster Communication System (Early Warning and Its Dissemination). Land Use Planning and Development Regulations, Disaster Safe Designs and Constructions, Structural and Non Structural Mitigation of Disasters

S&T Institutions for Disaster Management in India

Suggested Readings

1. Publications of National Disaster Management Authority (NDMA) on Various Templates and Guidelines for Disaster Management
2. Bhandani, R. K., An overview on natural & man-made disasters and their reduction, CSIR, New Delhi
3. Srivastava, H. N., and Gupta G. D., Management of Natural Disasters in developing countries, Daya Publishers, Delhi
4. Alexander, David, Natural Disasters, Kluwer Academic London
5. Ghosh, G. K., Disaster Management, A P H Publishing Corporation
6. Murthy, D. B. N., Disaster Management: Text & Case Studies, Deep & Deep Pvt. Ltd.

OE434 **ORGANIZATION BEHAVIOUR**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn management process and functions
2. Study decision making and negotiations
3. Comprehend psychological contract
4. Understand models of organization behaviour
5. Gain insights into organization design

Course Outcomes

1. Explain management process and functions
2. Discuss decision making and negotiations
3. Describe psychological contract
4. Elaborate models of organization behaviour
5. Elucidate the organization design

UNIT I

Management Process and Functions, Scientific and Modern Management, 3D Model of Managerial Behavior - MBO - MBWA - Line and Staff - The Peter's Principle - Parkinson's Law - Approaches to Organization Structure-Management - Classical, Human Relations, Systems and Contingency Approaches, Hawthorne's Experiments - Human Engineering.

UNIT II

Decision Making and Negotiations: Approaches to Decision making - Rational, Behavioral, Practical, and Personal Approaches - Open and Closed Models of Decision Making, Types and steps in planning, Authority, Responsibility, Centralization, Decentralization and Recentralization, Bureaucracy.

UNIT III

Psychological contract - Personality Traits, Big 5 personality traits, MBTI inventory, the Process of Perception - Perceptual distortions and errors, Kelly's personal construct Theory, Motivation-Content Theories: Maslow, Alderfer, Herzberg, McCleland. Process Theories: Vroom, Potter and Lawler, Equity Theory - Goal Theory - Attribution Theory.

UNIT IV

Models of Organization Behavior - Autocratic, Custodial, Supportive, Collegial and System Models, Transactional Analysis, Johari Window. Group Dynamics: Typology of Groups - Conflicts in groups - The nature, of conflict - Reactions to conflict - A model of conflict. Trait and Behavioral Approaches to Leadership, Managerial Grid, Path-Goal Theory, Vroom's Decision Tree Approach to Leadership - Hersey and Blanchard Model.

UNIT V

Organization Design, Organization culture and organization climate, Stress Management and Counseling, Management of change and organization development. Communication - Emerging aspects of OB.

Suggested Readings

1. Harold Koontz and Heinz Weihrich, Essentials of Management, 9th Edition, McGraw Hill Education, 2015.
2. Curtis W. Cook and Phillip L. Hunsaker, Management and Organizational Behavior, 3rd Edition, McGraw-Hill,2010.

OE435 INTELLECTUAL PROPERTY AND CYBER LAW

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn the fundamentals of intellectual property
2. Study the basics of international instruments of IPR
3. Understand the laws concerning copyright in India
4. Comprehend the IP in trademarks
5. Gain insights into the concept of patent

Course Outcomes

1. Explain the fundamentals of intellectual property
2. Elaborate the basics of international instruments of IPR
3. Describe the laws concerning copyright in India
4. Discuss the IP in trademarks
5. Explain the concept of patent

UNIT-I

Meaning, Nature, Classification and protection of Intellectual Property, The main forms of Intellectual Property, Copyright, Trademarks, Patents, Designs (Industrial and Layout), Geographical Indications Plant Varieties Protection and Biotechnology

UNIT-II

Introduction to the leading International instruments concerning Intellectual Property Rights, The Berne Convention, Universal Copyright Convention, The Paris Union, Patent Cooperation Treaty, The World Intellectual Property Organization (WIPO) and the UNEESCO, International Trade Agreements concerning IPR, WTO, TRIPS.

UNIT-III

Select aspects of the Law of Copyright in India The Copy Right Act,1957 Historical evolution, Meaning of copyright, Copyright in literary, dramatic and musical works, computer programmes and cinematograph films, Neighbouring rights, Rights of performers and broadcasters, etc., Ownership and Assignment of copyright, Author's special rights, Notion of infringement, Criteria of infringement

Infringement of copyright in films, literary and dramatic works, Authorities under the Act, Remedies for infringement of copyright.

UNIT-IV

Intellectual Property in Trademarks and the rationale of their protection - The Trade Marks Act. 1999 —Definition of Trademarks — Distinction between Trademark and Property Mark - Registration — Passing off — Infringement of Trademark — Criteria of Infringement — Remedies. The Designs Act, 2000 -- Definition and characteristics of Design — Law in India — Protection and rights of design holders -- Copyright in design — Registration — Remedies for infringement.

UNIT-V

Patents — Concept of Patent — Historical overview of the Patents Law in India — Patentable Inventions —Kinds of Patents — Procedure for obtaining patent — The Patents Act, 1970 — Rights and obligations of a patentee — Term of patent protection — Use and exercise of rights — Exclusive Marketing Rights — Right to Secrecy — The notion of 'abuse' of patent rights — Infringement of patent rights and remedies available.

Suggested Readings

1. P. Narayanan, Patent Law, Eastern Law House, 1995.

2. Roy Chowdhary, S.K. & Other, Law of Trademark, Copyrights, Patents and Designs, Kamal Law House, 1999.

3. Dr. G.B. Reddy, Intellectual Property Rights and the Law ,5th Edition, Gogia Law Agency, 2005.

OE436 **ENVIRONMENTAL SCIENCE**

Credits : 3

Instruction	3L hrs per week	Duration of SEE	3 hours
CIE	40 marks	SEE	60 marks

Course Objectives

1. Learn the scope and importance of environmental studies
2. Study about the environment and natural resources
3. Understand the environmental pollution
4. Comprehend the regional and sectoral issues concerning environment
5. Gain insights into social issues and the environment

Course Outcomes

1. Explain the scope and importance of environmental studies
2. Elaborate the environment and natural resources
3. Describe the environmental pollution
4. Discuss the regional and sectoral issues concerning environment
5. Explain the social issues and the environment

UNIT-I

Environmental Studies: Introduction - Definition, Scope and Importance - Basic principle of ecosystem functioning - Concept of ecosystem, structure and functioning of ecosystem, introduction and characteristic features, structures and functions, different ecosystems.

Biodiversity and its conservation: Introduction - Bio-geographical classification of India. Value of biodiversity - consumptive and predictive use, social, ethical and optional values. Biodiversity - Global, National and local levels. Hot spots of biodiversity - Threats to biodiversity - Endangered and endemic species of India - Conservation of biodiversity - In-situ and Ex-situ conservant.

UNIT-II

Environmental and Natural Resources: Forest resources - Use and over-exploitation, Deforestation, Timber extraction, Mining and dams - their effects on forests and tribal' people. Water resources - Use and over-utilization of surface and ground water, floods, droughts, conflicts over water, dams - effects of extracting and using mineral resources. Food resources - World food problems - change caused by agricultural and overgrazing, effects of modern agricultural fertilizer pesticide problems, water logging and salinity.

Environmental Valuation: Welfare measure and environmental values, definition and classification of environmental values, valuation methods. Environmental Economics: Economic approach to environmental preservation and conservation, property rights and externalities, management of natural resources.

UNIT-III

Environmental Pollution: Causes, effects and control measures of air pollution, water pollution, soil pollution, marine pollution, noise pollution.

Environmental Problems in India: Effects of human activities on the quality of life, Water and River, Ground water, Wasteland reclamation.

UNIT-IV

Regional and Sectoral Issues: Urbanization, Agro-forestry, Dry lands, Goods and services, Mountain development, River basin water resources management. sustainable tourism, and Costal zone management. Environment and Development: The economy and environment interaction, State of the Environment - Economics of development; Preservation and conservation.

Sustainability: Theory and Practice, Equitable use of resources for sustainable life styles - Role of an individual in prevention of pollution.

Human Population and the Environment: Population growth and environment - Human Rights.

UNIT-V

Social Issues and the Environment: Sustainable Development - Resettlement and rehabilitation of people and its problems and concerns.

Environmental ethics: Issues and possible solutions-Consumerism and waste products - Public awareness. Sustainable resources management. Design of Environmental Policy -- Direct regulation by Government - Command and control instrumentation.

Suggested Readings

1. B. Sudhakara Reddy, T. Sivaji Rao, U. Tataji& K. Purushottam Reddy, An Introduction to Environmental Studies, Maruti Publications.
2. C.Manohar Chary and P.Jayaram Reddy, Principles of Environmental Studies, B.S. Publications, Hyderabad.
3. Y.Anjaneyulu, Introduction to Environmental Science, B.S. Publications, Hyderabad.

Proj401 **PROJECT WORK**

Credits : 12

Instruction	24 hrs per week	Duration of SEE	3 hours
CIE	50 marks	SEE	100 marks

Project has to be carried out by each student individually in a period of 15 weeks of duration. Students should submit a synopsis at the end of 2nd week in consultation with the Project Guide. The synopsis should consist of definition of the problem, scope of the problem and plan of action. After completion of eight weeks students are required to present a Project Seminar on the topic covering the aspects of analysis, design and implementation of the project work.

At the end of the semester the students are required to present themselves for a University Viva-voce examination. Evaluation guidelines for the award of SEE marks are mentioned in the Rules and Regulations book.

A committee consisting of two faculty members of the respective college along with a guide will evaluate the project and award CIE marks.

EACH STUDENT WILL BE REQUIRED TO:

1. Submit one page of synopsis on the project work for display on notice board.
2. Give a 20 minutes presentation followed by10 minutes discussion.
3. Submit a technical write-upon the project.

At least two teachers will be associated with the Project Seminar to evaluate students for the award of CIE marks which will be on the basis of performance in all the 3 items stated above.

The project seminar presentation should include the following components of the project:

- Problem definition and specification.
- Literature survey, familiarity with research journals.
- Broad knowledge of available techniques to solve a particular problem.
- Planning of the work, preparation of bar (activity) charts, Presentation both or a land written.

Course Objectives:

The aim of the course is to

- Enhance the previous knowledge of database systems by deepening the understanding of the theoretical and practical aspects of the database technologies, and showing the need for distributed database technology to tackle deficiencies of the centralized database systems; Introduce basic principles and implementation techniques of distributed database systems

- Expose active and emerging research issues in distributed database systems and application
- development, apply theory to practice by building and delivering a distributed database query engine,
- subject to remote Web service calls. Course Outcomes: After the completion of the course, the students are expected to 1. Get familiar with the currently available models, technologies for and approaches to building distributed database systems and services; 2. Have developed practical skills in the use of these models and approaches to be able to select and apply the appropriate methods for a particular case; 3. Be aware of the current research directions in the field and their possible outcomes; 4. Be able to carry out research on a relevant topic, identify primary references, analyze them, and come up with meaningful conclusions 5. Be able to apply learned skills to solving practical database related tasks.

* * * * *