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**	1.5
Lab -II	LCC 152	Java Programming Lab	-	-	2	25	**50**	1.5
Lab-III	HSC 151	Soft Skills Lab	-	-	2	25	**50**	1.5
TOTAL			**18**	**-**	**6**	**315**	**510**	**22.5**
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 205	Machine Learning	3	-	-	40	60	3
Core – XI	MGC206	Operations Research	3	-	-	40	60	3
Core-XII	PCC 206	Data Structures	3	-	-	40	60	3
Lab-IV	LCC 251	Operating Systems Lab	-	-	2	25	**50**	1.5
Lab-V	LCC 252	Data Engineering with Python Lab	-	-	2	25	**50**	1.5
Lab- VI	LCC 253	Database Management Systems Lab	-	-	2	25	**50**	1.5
Lab-VII	LCC 254	Data Structure Lab	-	-	2	25	**50**	1.5
Mini Project	LCC 254	Mini Project	-	-	8	25	**50**	4
TOTAL			**21**	**-**	**16**	**405**	**670**	**31**
SEMESTER-III								
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	3	-	-	40	60	3
Core -XV	PCC 304	Web Technologies	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	LCC351	Computer Networks Lab	-	-	2	25	50	1.5

Lab-VIII	LCC352	Software Engineering Lab	-	-	2	25	50	1.5
Lab-IX	LCC353	Web Technologies Lab	-	-	2	25	50	1.5
TOTAL			**18**		**6**	**315**	**510**	**22.5**
SEMESTER-IV								
Professional Elective- III	PEC411	Block Chain Technologies	3	-	-	40	60	3
	PEC412	Big Data Analytics						
	PEC413	Cloud Computing						
	PEC414	Deep Learning						
Professional Elective- IV	PEC421	Distributed Database Systems	3	-	-	40	60	3
	PEC422	Digital Forensics						
	PEC423	Optimization Techniques						
	PEC424	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	Proj401	Project Work	-	-	24	50	100	12
TOTAL			**9**	**-**	**24**	**170**	**280**	**21**
GRAND TOTAL			**66**	**-**	**52**	**1240**	**1970**	**98**

PCC101	MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE					
CORE -I						
Pre-requisites			**L**	**T**	**P**	**C**
			4	-	-	4
Evaluation	**SEE**	**60 Marks**	**CIE**		**40**Marks	

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**,PrenticeHallNJ,2nd Edition, 2015
2	P. Tremblay and R Manohar **Discrete Mathematical Structures with Applications to Computer Science**,McGrawHill,1987
3	R.K.Bisht and H.S.Dhami, **Discrete Mathematics** Oxford Higher Education,2015
4	Bhavanari Satyanarayana, Tumurukota Venkata Pradeep Kumar and Shaik Mohiddin Shaw, **Mathematical Foundation of Computer Science,**BSP,2016

PCC102	Programming in C					
CORE -II						
Pre-requisites			L	T	P	C
			4	-	-	4
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, goto: 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		40Marks	

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, BitSet, 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 McGra Hill, 2005.
2	BruceEckel, **Thinking in Java**,4th Edition,Pearson Education,2009
3	Dieteland 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**,TataMcGrawHill,2005

PCC104	COMPUTER ARCHITECTURE					
	CORE -IV					
Pre-requisites			L	T	P	C
			3	1	-	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: Datatypes, Complements, Fixed and Floating Point representations, and Binary codes.
Overview of Computer Function and Interconnections: 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. **Pipe line Processing:** Arithmetic, Instruction and RISC Pipelines. **Assessing and Understanding Performance:** CPU performance and its factors, Evaluating performance.

Suggested Reading:

1	Morris ManoM ,**Computer System Architecture**, Pearson Education India, 3rd Edition, 2007.
2	William Stallings, **Computer Organization and Architecture**, PHI ,7th Edition, 2008.
3	David APatterson,John L Hennessy, **Computer Organization and Design**, Morgan Kaufmann,5thEdition,2013.
4	Carl Hamacher Zvonko Vranesic, SafwatZaky, **Computer Organization,** Tata McGraw-Hill Education,5thEdition,2002.

PCC105	PROBABILITY AND STATISTICS					
CORE -V						
Pre-requisites			L	T	P	C
			3	1	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

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, their inter relations With real time applications.
CO-2	Understand the use of sample statistics to estimate unknown 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 of fit, 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 caveats in regression and correlation analysis. Multiple
Regression and correlation analysis. Finding multiple regression equations and making inferences about population parameters.

Suggested Reading:

1	David CLay,Linear Algebra and its Applications 4e.
2	Richard I Levin, David S Rubin - Statistics for Management, Seventh Edition, PHI -1997.
3	RD Sharma—Theory and Problems of Linear Algebra Ⅱ,International Publishing House Pvt.Limited,2011.
4	AK Sharma,—Linear Algebra‖, Discovery Publishing HouseLtd.,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, New Delhi.

PCC106	MANAGERIAL ECONOMICS AND ACCOUNTANCY					
CORE -VI						
Pre-requisites			L	T	P	C
			3	1	-	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	MehtaP.L.,ManagerialEconomics—Analysis,ProblemsandCases,SultanChand &Sons Educational Publishers, 2011
2	Maheswari S.N., Introduction to Accountancy,VikasPublishingHouse,2005
3	Pandey I .M., Financial Management, Vikas Publishing House, 2009
4	SPJainandKLNarang,—FinancialAccounting‖,KalyanPublishers,2018
5	M Hanif and A Mukherjee—Modern Accountancy ‖,McGraw Hill,3rd Edition, 2018

	C Programming Lab					
Pre-requisites			**L**	**T**	**P**	**C**
			-	-	2	1
Evaluation	**SEE**	50 Marks	**CIE**		25 Marks	

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
			-	-	3	2
Evaluation	SEE	50 Marks	CIE		25Marks	

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 LinkedList class.
7. Write a program to demonstrate HashSet 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	SOFTSKILLSLAB					
	LAB – III					
Pre-requisites			L	T	P	C
			-	-	3	2
Evaluation	SEE	50 Marks	CIE		25Marks	

Course Objectives :	
Thecourseistaught 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	Robert M Sherfield—Developing Soft Skills ‖4th Edition, Pearson Education,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	1	-	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 Filesystems, Networking, Programmer interface.

Suggested Reading:

1	AbrahamSilberschatz,PeterBGalvin,OperatingSystemConcepts,9th Edition, Wiley,2016
2	WilliamStallings,OperatingSystems-InternalsandDesignPrinciples,8thEdition, Pearson, 2014
3	Andrew S Tanenbaum, Modern OperatingSystems,4thEdition, Pearson,2016.

PCC 202	DATABASE MANAGEMENT SYSTEM					
	CORE -VII					
Pre-requisites			L	T	P	C
			3	1	-	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 NoSQL

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 NoSQL 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, RecursiveQueries,AdvancedSQLFeatures.RelationalDatabaseDesign: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	1	-	4
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.
The Greedy Method : Knapsack Problem ,Tree Vertex Splitting ,Job Sequencing with 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 SearchTrees,0/1Knapsack, The Traveling Salesperson Problem.
BasicTraversalandSearchTechniques:TechniquesforBinaryTrees,TechniquesforGraphs,ConnectedComponentsandSpanningTrees,Biconnected Components and DFS.

UNIT– IV

BackTracking:GeneralMethod,8-QueensProblem,SumofSubsets,GraphColoring,HamiltonianCycles,KnapsackProblem.**Branch-Bound**:TheMethod, 0/1KnapsackProblem, 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	T H Cormen,CELeiserson,RL Rivert,C Stein,"Introduction to Algorithms Ⅱ,Third Edition,PHI,2010.

PCC 204	DATA ENGINEERING WITH PYTHON					
CORE -X						
Pre-requisites			L	T	P	C
			3	1	-	4
Evaluation	SEE	60 Marks	CIE		40Marks	

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

Course Outcomes:	
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 Pyplot libraries

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 Text sin 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 PyPlot , Getting to Know Other Plot Types, Mastering Embellishments, Plotting with Pandas

UNIT–V
ProbabilityandStatistics: ReviewingProbabilityDistributions,RecollectingStatistical 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	PythonforEverybody:ExploringDataUsingPython3.CharlesRSeverance,2016
3	Python Data Analytics–Data Analysis and Science using Pandas, matplotlib and the Python ProgrammingLanguage.FabioNelli,Apress,2015
4	Website Scraping with Python. Using BeautifulSoup andScrapy. GáborLászlóHajba, Apress,2018
5	MachineLearningwithPythonCookbook:.PracticalSolutionsfromPreprocessingto DeepLearning.ChrisAlbon,O‘Reilly2018.

PCC 205	MACHINE LEARNING					
	CORE -XI					
Pre-requisites			L	T	P	C
			3	-	-	3
Evaluation	SEE	60 Marks	CIE		40Marks	

Course Objectives:	
1	To learn regression techniques and dimensionality reduction methods
2	To learn classification and Clustering methods
3	To understand the valuation metrics

Course Outcomes:	
On completion of this course ,the student will be able to :	
CO-1	Solve the regression problems and dimensionality reduction methods
CO-2	Analyze the clustering mechanisms
CO-3	Explore Various classification methods and evaluation metrics

UNIT– I

BasicMaths:Probability,LinearAlgebra,ConvexOptimization**Background**:Statistical DecisionTheory,BayesianLearning(ML,MAP,Bayesestimates,Conjugatepriors)

UNIT– II

Regression: Linear Regression ,Ridge Regression, Lasso **Dimensionality Reduction**: Principal Component Analysis, Partial Least Square

UNIT– III

Classification: Linear Classification, Logistic Regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis ,Perceptron ,Support Vector Machines +Kernels ,Artificial Neural Networks + Back Propagation, Decision Trees, Bayes Optimal Classifier, Naive Bayes.

UNIT– IV

Evaluationmeasures:Hypothesistesting,EnsembleMethods,Bagging,AdaboostGradient Boosting,Clustering,K-means,K-medoids,Density-basedHierarchical,Spectral

UNIT–V

Expectation Maximization ,GMMs ,Learning theory ,Introduction to Reinforcement Learning **Graphical Models** : Bayesian Networks.

Suggested Readings:

1	EthemAlpaydin. Introduction to Machine Learning e(Adaptive Computation and Machine Learning Series).TheMITPress,2004.
2	TomM.Mitchell,MachineLearningMcGrawHillEducation,2013

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, DevelopmentofLPmodels,GraphicalMethod,LinearProgrammingMethods,Special Cases of Linear Programming, Duality, Sensitivity Analysis.

UNIT– II

Transportation Problem: Introduction, Mathematical Model for 'Transportation Problem,TypesofTransportationProblem,MethodstosolveTransportationProblem, Transshipment Model.

UNIT– III

AssignmentProblem: 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-PlaneAlgorithm,Branch-and-BoundTechnique,Zero-OneImplicitEnumerationAlgorithm.

UNIT– IV

DynamicProgramming:Introduction,ApplicationsofDynamicProgramming,Solutionof LinearProgrammingProblemthroughDynamicProgramming.BasicsofQueuingtheory.

UNIT–V

GameTheory:Introduction,GamewithPureStrategies,GamewithMixedStrategies,

DominanceProperty, Graphical Methodfor2xnormx2Games,Linear Programming Approach for Game Theory.

Suggested Readings:

1	Pannarselvam,"*OperationsResearch*",3rdEdition,PHI,2009
2	Prem Kumar Gupta, DS Hira ,"*Problems in Operations Research*",S.Chand,2010.
3	RathindraPSen,"*OperationsResearch-AlgorithmandApplication*",PHI,2010.
4	J K Sharma," *Operations Research*", Fourth Edition ,MacMillan, 2009.

PCC102	DATA STRUCTURES USING C					
	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 , Reema Thareja , Oxford University Press (India) 2nd Edition (Paperback)

LCC251	OPERATING SYSTEMS LAB					
	LAB – IV					
Pre-requisites			L	T	P	C
			-	-	3	2
Evaluation	SEE	50 Marks	CIE		25Marks	

Course Objectives:	
Thecourseistaught with the objectives of enabling the student to:	
1	Understand and implement shell scripting and CPU scheduling 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	2
Evaluation	SEE	50 Marks	CIE		25Marks	

Course Objectives:	
Thecourseistaught 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 multiple data sources

Course Outcomes :	
On completion of this course, the student will be able to :	
CO-1	Demonstratevariousdatatypesinpythonanddevelopprogramsusingfiles, exception handling, functions, classes in Python.
CO-2	Examine the process for importing and exporting the data.
CO-3	Demonstrate data 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, BeautifulSoup, 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 countdown

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 up to given number n

S. To display prime number from 2 to n.

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** DataFrames.
- 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 to perform 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

Data set: https://www.kaggle.com/khalidative/crimeanalysis

LCC253	DATABASE MANAGEMENT SYSTEMS LAB					
LAB – VI						
Pre-requisites			L	T	P	C
			-	-	3	2
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 database 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 data base 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: The creation of sample database for the purpose of the experiments is expected to be pre-decided by the instructor.

LCC151	DATA STRUCTURES USING C LAB					
	LAB – I					
Pre-requisites			L	T	P	C
			-	-	3	2
Evaluation	SEE	50 Marks	CIE		25Marks	

Course Objectives :	
Thecourseistaught 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 for a 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.

LCC254	MINI PROJECT					
Mini Project						
Prerequisites			L	T	P	C
			-	-	4	2
Evaluation	SEE	50 Marks	CIE		25 Marks	

Course Objectives

1	Todevelopcapabilitytoanalyzeandsolverealworldproblemswithanemphasison 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

****