

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY HYDERABAD
B.TECH. II YEAR
COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

III SEMESTER

R22

Course Code	Title of the Course	L	T	P/D	CH	C
22PC1CS202	Design and Analysis of Algorithms	3	1	0	4	4
22PC1DS201	Mathematical Foundations of Computer Science	3	0	0	3	3
22PC1AM201	Database Management Systems	3	0	0	3	3
22PC1IT201	Object Oriented Programming Through Java	3	0	0	3	3
22HS1MG201	Engineering Economics and Accountancy	3	0	0	3	3
22PC2AM201	Database Management systems Laboratory	0	0	2	2	1
22PC2IT201	Object Oriented Programming Through Java Laboratory	0	0	2	2	1
22ES2DS101	Python Programming Laboratory	0	0	2	2	1
22SD5AM202	Field Project	0	0	2	2	1
22MN6HS103	Happiness and Wellbeing	2	0	0	2	0
Total		17	1	8	26	20

IV SEMESTER

R22

Course Code	Title of the Course	L	T	P/D	CH	C
22BS1MT205	Statistical Inference and Queuing Theory	2	1	0	3	3
22PC1DS202	Digital Logic Design and Computer Organization	3	0	0	3	3
22PC1CS201	Software Engineering	3	0	0	3	3
22PC1AM202	Data Engineering	3	0	0	3	3
22PC1AM203	Automata and Compiler Design	3	0	0	3	3
22PC2AM202	Data Engineering Laboratory	0	0	2	2	1
22PC2AM213	Compiler Design and Software Engineering Laboratory	0	0	2	2	1
22SD5DS201	Data Visualization	0	0	2	2	1
22PW4AM201	Design Thinking	1	0	2	3	2
22MN6HS201	Intellectual Property Rights	2	0	0	2	0
Total		17	1	8	26	20

L – Lecture T – Tutorial P – Practical D – Drawing
C – Credits SE – Sessional Examination CA – Class Assessment
SEE – Semester End Examination D-D – Day to Day Evaluation
CP – Course Project PE – Practical Examination

CH – Contact Hours/Week
ELA – Experiential Learning Assessment
LR – Lab Record

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC1CS202) DESIGN AND ANALYSIS OF ALGORITHMS

TEACHING SCHEME		
L	T/P	C
3	1	4

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE PRE-REQUISITES: Programming, Data Structures, Discrete Mathematics

COURSE OBJECTIVES:

- To reinforce algorithms analysis methods
- To analyse running time of an algorithm
- To understand different algorithm design strategies
- To apply algorithms design techniques to solve computational problems

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Analyse running time of an algorithm using asymptotic analysis

CO-2: Apply appropriate algorithm design techniques to develop efficient algorithms for solving computational problems

CO-3: Compare different algorithms for their efficiency and choose an appropriate data structure

CO-4: Employ graphs to model engineering problems, when appropriate and analyse time complexity

CO-5: Describe and identify the classes P, NP, NP-Hard and NP complete problems and formulate solutions using standard approaches

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	2	2	2	3	-	-	-	2	-	-	-	2	-	3	-
CO-2	3	2	3	3	3	2	2	3	2	1	3	3	3	-	3
CO-3	2	2	2	3	-	-	-	2	-	-	-	2	-	3	-
CO-4	3	3	2	3	1	1	1	2	1	1	1	3	2	3	3
CO-5	3	3	3	3	-	-	-	1	1	1	2	2	-	3	-

UNIT-I:

Performance Analysis: Characteristics of Algorithm. Analysis of Algorithm: Best, Average and Worst-Case behaviour, Asymptotic Analysis– Big oh, Omega, Theta notations and Little oh, Little omega notations, Time and Space Trade-Offs, Analysis of Recursive Algorithms through Recurrence Relations, Substitution Method, Recursion Tree Method and Masters' Theorem.

UNIT-II:

Fundamental Algorithmic Strategies: Divide and conquer General method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication. Greedy method: General method, applications- Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem, Huffman Codes.
Brute force, KMP Pattern Matching Algorithms.

UNIT-III:

Dynamic Programming: General method, Principle of optimality, applications-Multistage graphs, Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

UNIT-IV:

Backtracking Method: Applications- N-Queen problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles. Branch and Bound Method: Applications - Travelling sales person problem, 0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution

UNIT-V:

Graph Algorithms: Depth First Search (DFS) and Breadth First Search (BFS), Topological sorting, Connected and Bi Connected Components, Network Flow Algorithm. Tractable and Intractable Problems: P, NP, NP-complete and NP-hard. Cook's theorem, Randomized Algorithms.

TEXT BOOKS:

1. Fundamental of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, 2nd Edition, Galgotia Publications
2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Lieserson, Ronald L. Rivest and Clifford Stein, 4th Edition, MIT Press/McGraw-Hill

REFERENCES:

1. Algorithm Design, Jon Kleinberg and EvaTardos, 1st Edition, Pearson
2. Algorithm Design: Foundations, Analysis, and Internet Examples, Michael T. Goodrich and Roberto Tamassia, 2nd Edition, Wiley
3. Introduction to the Design and Analysis of Algorithms, Anany Levitin, 3rd Edition, Pearson Publications
4. The Design and Analysis of Computer Algorithms, A. Aho, J. Hopcroft and J. Ullman

ONLINE RESOURCES:

1. <http://nptel.ac.in/courses/106101060/>
2. <https://www.cmi.ac.in/~madhavan/teaching.html>
3. <https://www.ics.uci.edu/~eppstein/161/960312.html>
4. <https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/>

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC1DS201) MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE OBJECTIVES:

- To apply logical reasoning to variety of problems
- To understand the concepts on elementary combinatorics and permutations
- To analyze the properties of graphs and trees
- To evaluate various methods for solving the recurrence relations

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Demonstrate problems using statement calculus, predicate logic and set theory

CO-2: Analyze sets, relations, functions, and discrete structures

CO-3: Apply and calculate permutations and combinations

CO-4: Understand the use of graphs and trees as models

CO-5: Solve various problems using recurrence relation techniques

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	3	1	2	1	2	-	-	-	-	-	1	-	3	2
CO-2	3	3	1	2	1	2	-	-	-	-	-	1	-	3	2
CO-3	3	3	1	2	1	2	-	-	-	-	-	1	-	3	2
CO-4	3	3	1	2	1	2	-	-	-	-	-	1	-	3	2
CO-5	3	-	2	-	2	-	-	-	-	-	-	-	-	2	2

UNIT-I:

Mathematical Logic: Statements and notations, Connectives, Well-formed formulas, Truth Tables, tautology, equivalence implication, Normal forms, Rules of inference, Consistency, proof of contradiction, Automatic Theorem Proving.

Predicates: Quantifiers, Predicative logic, Free & Bound variables.

UNIT-II:

Set Theory: Notations, inclusion and equality sets, operations on sets, Venn diagrams.

Relations: Properties of binary Relations, equivalence, transitive closure, compatibility and partial ordering relations, Hasse diagram.

Functions: Types of Functions, Inverse Function, Composition of functions, recursive Functions.

UNIT-III:

Elementary Combinatorics: Basics of counting, Combinations & Permutations, with repetitions, Constrained repetitions, the principles of Inclusion – Exclusion, Pigeon hole principle, Binomial Coefficients.

UNIT-IV:

Graphs: Basic Concepts, Isomorphisms and Subgraphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler 's Formula, Multigraphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four-Color Problem.

UNIT-V:

Recurrence Relations: Generating Functions, Function of Sequences, Calculating Coefficients of generating functions, Recurrence relations, solving recurrence relation by substitution and Generating functions, the method of Characteristic roots, solution of Inhomogeneous Recurrence Relations.

TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science, J. P. Trembly and R. Manohar, Tata McGraw-Hill, 2019
2. Discrete Mathematics for Computer Scientists & Mathematicians, J. L. Mott, A. Kandel, T. P. Baker, 2nd Edition, PHI, 1986
3. Graph Theory: With Application to Engineering and Computer Science, Narsingh Deo, Prentice Hall of India, 2003

REFERENCES:

1. Elements of Discrete Mathematics, A Computer Oriented Approach, C. L. Liu and D. P. Mohapatra, 3rd Edition, Tata McGraw-Hill, 2017
2. Discrete Mathematics and its Applications, Kenneth H. Rosen, Tata McGraw-Hill, 2003
3. Discrete Mathematics, Norman L. Biggs, 2nd Edition, Oxford University Press, 1989

ONLINE RESOURCES:

1. IIT Discrete Mathematics Lectures – YouTube

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC1AM201) DATABASE MANAGEMENT SYSTEMS

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE PRE-REQUISITES: Basic Computer Programming Skills

COURSE OBJECTIVES:

- To introduction of data base management concepts and to give the description of structure of data base systems
- To understand concepts of ER model and model the data base for the given scenarios and prepare the database through normalization
- To know the features of various models of data and query representations
- To introduce the concepts and protocols related to transaction management and understand the concepts of data storage

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Appreciate the underlying concepts of database system architecture and design E-R models to represent simple database application scenarios

CO-2: Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data

CO-3: Improve the database design by normalization

CO-4: Apply and relate the concept of transaction, concurrency control and recovery in database

CO-5: Familiar with basic database storage structures and access techniques using Indexing, hashing including B tree methods

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	1	3	2	3	2	2	-	1	-	-	-	2	-	2
CO-2	2	2	3	3	3	2	3	-	1	-	-	-	2	2	2
CO-3	2	3	1	3	1	-	2	-	1	-	-	-	2	-	2
CO-4	3	2	1	1	1	-	3	-	-	-	-	-	1	1	1
CO-5	2	2	1	1	1	-	3	-	-	-	-	-	2	2	2

UNIT – I:

Introduction to Databases and Database Management System: Database system Applications, Advantages of DBMS over File System, Data Models, Instances and schema, View of Data, Database Languages –DDL, DML, DCL, Database Users and Administrator, Database System Architecture.

Database Design and ER Diagrams: Attributes and Entity Sets, Relationships and Relationship Sets, Constraints, Keys, Design Issues, Entity-Relationship Diagram, Extended E-R Features, Database Design with ER model, Database Design for a schema.

UNIT – II:

Introduction to the Relational Model: Structure of RDBMS, Integrity Constraints over Relations, Querying Relational Data, Relational Algebra and Relational Calculus.
Introduction to SQL: Data Definition commands, Data Manipulation Commands, Basic Structure, Set operations, Aggregate Operations, Join Operations, Sub queries and correlated queries, views, Triggers, Cursors

UNIT – III:

Functional Dependencies: Introduction, Basic Definitions, Trivial and Non trivial dependencies, closure of a set of dependencies, closure of attributes, irreducible set of dependencies.

Schema Refinement in Database Design: Problems Caused by Redundancy, Decompositions – Problem Related to Decomposition, Lossless Join Decomposition, Dependency Preserving Decomposition, FIRST, SECOND, THIRD Normal Forms, BCNF, Multivalued Dependencies, FOURTH Normal Form.

UNIT – IV:

Transaction Management: Transaction state, Implementation of atomicity and Durability, Concurrent executions – Serializability, Recoverability.

Concurrency Control: Lock Based Protocols, Timestamp Based Protocols, Validation Based Protocols, Multiple Granularity, Dead Lock Handling

Recoverability: Failure Classification, Storage Structure, Recovery and Atomicity- Log Based recovery, Recovery with concurrent transactions, Checkpoints.

UNIT – V:

File Organization: Organization of records in file, Data Dictionary Storage.

Indexing and Hashing: Basic Concepts, Ordered Indices, B tree Index files, B+ tree index files, Static Hashing, Dynamic Hashing – Comparison of Indexing with Hashing.

TEXT BOOKS:

1. Database System Concepts, Silberschatz, Korth, Sudarshan, 7th Edition, McGraw-Hill
2. Introduction to Database Systems, C. J. Date, Pearson Education

REFERENCES:

1. Database Management Systems, Raghu Ramakrishnan, Johannes Gehrke, Tata McGraw-Hill
2. Fundamentals of Database Systems, Elmasri Navathe, Pearson Education
3. Database Systems Design, Implementation, and Management, Peter Rob & Carlos Coronel, 7th Edition, Cengage Learning

ONLINE RESOURCES:

1. <https://www.w3schools.com/sql/default.asp>
2. <https://www.javatpoint.com/dbms-tutorial>

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC1IT201) OBJECT ORIENTED PROGRAMMING THROUGH JAVA

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE OBJECTIVES:

- To understand object oriented concepts and problem solving techniques
- To obtain knowledge about the principles of inheritance and polymorphism
- To implement the concept of packages, interfaces, exception handling and concurrency mechanism
- To understand the Java database connectivity architecture

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Solve real-world problems using Object Oriented Programming paradigm

CO-2: Apply code reusability through inheritance, packages and interfaces

CO-3: Solve problems using Java collection framework and I/O classes

CO-4: Develop applications by generics for better performance use multithreading

CO-5: Build applications using the JDBC API to access the database

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	3	2	3	-	2	-	-	1	-	1	2	2	3	2
CO-2	3	2	3	2	-	-	-	1	1	1	-	2	3	3	2
CO-3	2	3	2	3	-	-	-	-	-	-	-	1	2	2	2
CO-4	2	2	3	2	2	2	2	-	-	1	-	2	3	2	2
CO-5	2	2	2	2	2	3	2	1	-	1	1	2	3	2	2

UNIT-I:

Object Oriented Programming Paradigm: Introduction to OOP Paradigm, Evolution of Java, Difference between JDK, JRE, and JVM, Classes, Objects, Data Types, Variables, Comments, Type Conversion and Casting, Operators, Symbolic Constants, Control Statements, Scope of Variables, Constructors, this keyword, Overloading, Command line Arguments, Arrays, String Class, Wrapper classes.

UNIT-II:

Inheritance, Packages, Interfaces and Other Topics: Understanding static, Introducing Nested and Inner classes, Access Modifiers,

Varargs: variable-Length Arguments, Lambda Expressions, garbage collection-finalize.

Inheritance: Basics, Using super, creating a multi level hierarchy, when constructors are executed, method overriding, polymorphism - dynamic method dispatch, using abstract class, final with Inheritance, object class.

Packages and Interfaces: Packages, Access protection, Importing Packages, Interfaces, Default Interfaces, Default interface methods, Use static methods in an Interface,

UNIT-III:

Exception Handling: Exception handling Fundamentals, Exception Types, Using try and catch, throw, throws, finally, Java Custom Exception, Catch Multiple Exceptions, Try-with Resources, Exceptions propagation, Chained Exceptions.

I/O, Generics and Other Topics: The Streams (Byte, Character, and Standard), Output Stream and Input Stream classes, Reading and Writing Files, Random access file operations, Generics: The general form of a generics class, creating a generic method, generics interfaces, Serialization, Annotations,, Auto Boxing.

UNIT-IV:

Multithreaded Programming: Java Thread life cycle model – Thread creation - Thread Exceptions - Thread Priority – Synchronization - Runnable Interface - Interthread Communication - Deadlock - Suspending, Resuming and stopping threads.

Java AWT: AWT Hierarchy, Event Delegation Model, Adapter classes, Listeners, Layout management, AWT Components, Simple UI for Email registration.

UNIT-V:

The Collection Framework: Collection Objects – Sets, Lists, Queues, Maps – ArrayList- LinkedList - Vector– HashSet– LinkedHashSet– TreeSet– HashMap– Hashtable. Retrieving Elements from Collections – Enumeration, Iterator, List Iterator – String Tokenizer and Arrays Class – Sorting using Comparable and Comparator.

Java Database Connectivity: Introduction to JDBC, Types of Drivers, Basic steps in developing JDBC applications, JDBC classes and Interfaces, CRUD operations with JDBC, Transaction Management, Stored Procedures.

TEXT BOOKS:

1. Java The Complete Reference, Herbert Schildt, 9th Edition, McGraw-Hill Education
2. Java How to Program, Paul Dietel, Harvey Dietel, 10th Edition, Pearson Education

REFERENCES:

1. Core Java Volume-1 Fundamentals, Cay S. Horstmann, Pearson Education
2. Java Programming for Core and Advanced Learners, Sagayaraj, Dennis, Karthik and Gajalakshmi, University Press
3. Introduction to Java programming, Y. Daniel Liang, Pearson Education
4. Object Oriented Programming through Java, P. Radha Krishna, University Press

ONLINE RESOURCES:

1. https://www.w3schools.com/java/java_oop.asp
2. <http://peterindia.net/JavaFiles.html>

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22HS1MG201) ENGINEERING ECONOMICS AND ACCOUNTANCY

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE OBJECTIVES:

- To understand the basic concepts of economics and different forms of business organizations
- To create awareness on basics of business economics and to analyse the concepts of demand and supply
- To describe each stage of product life cycle with the help different costs and their role in maintaining optimum cost of production and overall profitability by considering different market competitions
- To acquaint with the basic accounting knowledge and financial accounting process
- To evaluate the performance of the organization using various ratios

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Perform decision making function effectively in an uncertain framework by applying the based concepts of economics and select suitable form of business organization which meets the requirements of business

CO-2: Take the right decisions towards buying and selling of goods and services based on the demand and supply dynamics in the markets

CO-3: Fix the right price based upon production cost which can best meet the pre-determined objectives of the business under different market conditions

CO-4: Prepare book of accounts and understand overall position of the business

CO-5: Interpret the firm's financial performance using various ratios

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	-	-	-	-	-	-	-	1	2	1	3	1	-	-	-
CO-2	-	-	-	-	-	-	-	1	2	1	3	1	-	-	-
CO-3	-	-	-	-	-	-	-	1	2	1	3	1	-	-	-
CO-4	-	-	-	-	-	-	-	1	2	1	3	1	-	-	-
CO-5	-	-	-	-	-	-	-	1	2	1	3	1	-	-	-

UNIT-I:

Introduction to Economics: Definition, nature, scope and types of Economics. National Income (NI) & types of Inflation.

Forms of organizing Private and Public-Sector Business Enterprises

Private Sector Business Enterprises: (i) Sole Proprietorship – Definition, features, merits, limitations & suitability. (ii) Partnership – Definition, Partnership Act, features, types, merits, limitations, suitability. (iii) Joint-Stock Company – Definition, Companies Act, features, types, merits, limitations, suitability.

Public Sector Business Enterprises: Definition, features, objectives, merits, problems

UNIT-II:

Business Economics: Definition, nature and scope, linkages with other disciplines.

Demand Analysis: Law of Demand, Factors affecting demand; Elasticity of Demand-Types Measurement, Factors affecting and Significance,

Demand Forecasting: Characteristics of Good Demand Forecasting, Steps in Demand Forecasting, Methods of Demand Forecasting.

Supply Analysis: Determinants of Supply, Supply function and Law of Supply.

UNIT-III:

Production, Cost, Market Structures & Pricing:

Production Analysis: Factors of Production, Production Function, Production Function with one variable input, two variable inputs, Returns to Scale, Different Types of Production Functions - Cobb-Douglas.

Cost Analysis: Types of Costs, Short run and long run Cost Functions.

Market Structures: Nature of Competition, Features of Perfect competition, Monopoly, Oligopoly, Monopolistic Competition.

Pricing: Types of Pricing, Product Life Cycle based Pricing, Break Even Analysis (Simple problems)

UNIT-IV:

Introduction to Financial Accounting: Definition, basic principles and double-entry book-keeping, practice of accounting process-Journal, ledger, trial balance and final accounts (simple problems)

UNIT-V:

Ratio Analysis: Meaning, computation of ratios (i) Liquidity Ratios: Current Ratio and Quick Ratio, (ii) Solvency Ratios: Interest Coverage Ratio and Debt-Equity Ratio, (iii) Activity Ratios: Stock/Inventory Turnover Ratio and Debt Turnover Ratio, (iv) Profitability Ratios: Gross Profit Ratio, Net Profit Ratio & Earning Per Share (EPS) Ratio

TEXT BOOKS:

1. Managerial Economics, D. M. Mithani, 9th Edition, Himalaya Publishing House, 2022
2. Managerial Economics, Satya P. Das & J. K. Goyal, 2nd Edition, Sage Publications, 2022
3. Financial Accounting, S. N. Maheswari, 6th Edition, Vikas Publications, 2018

REFERENCES:

1. Managerial Economics, Dominick Salvatore, Siddhartha K. Rastogi, 9th Edition, Oxford Publications, 2020
2. Financial Accounting for Management: An Analytical Perspective, Ambrish Gupta, 6th Edition, Pearson Education, 2018
3. Business Economics, H. L. Ahuja, 13th Edition, S. Chand & Co., 2019

4. Principles of Marketing: A South Asian Perspective, Kotler Philip, Gary Armstrong, Prafulla Y. Agnihotri, and Eshan ul Haque, 13th Edition, Pearson Education/ Prentice Hall of India, 2010

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC2AM201) DATABASE MANAGEMENT SYSTEMS LABORATORY

TEACHING SCHEME		
L	T/P	C
0	2	1

EVALUATION SCHEME					
D-D	PE	LR	CP	SEE	TOTAL
10	10	10	10	60	100

COURSE PRE-REQUISITES: Basic Computer Programming Skills

COURSE OBJECTIVES:

- To provide the fundamental concepts of database creation
- To implement the concepts of data manipulation
- To develop procedures for querying multiple tables
- To understand the concepts of PL / SQL

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Understand the given scenario, design it through ER model and normalize the schema.

CO-2: Use the basics of SQL and construct queries using SQL in database creation and interaction.

CO-3: Apply Integrity constraints for creating consistent RDBMS environment

CO-4: Applying PL/SQL for processing database

CO-5: Develop solutions using database concepts for real time requirements

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	1	3	2	2	1	2	-	1	-	-	1	2	2	3
CO-2	3	2	3	3	2	1	2	-	2	-	-	2	3	2	3
CO-3	3	2	3	3	2	2	2	-	2	-	-	2	3	2	3
CO-4	3	2	3	3	2	1	2	-	2	-	-	2	3	2	3
CO-5	3	2	3	3	2	2	2	-	3	3	2	2	3	3	3

LIST OF PROGRAMS / EXPERIMENTS / EXERCISES:

WEEK 1:

Implement the following using DUAL table: a) Character functions, b) Numeric functions c) Date functions and d) Conversion functions.

WEEK 2:

Practice DDL and DML commands on a basic table without integrity constraints.

WEEK 3:

Practice DDL and DML commands on a Relational Database, specifying the Integrity constraints. (Primary Key, Foreign Key, CHECK, NOT NULL)

WEEK 4:

Apply the concepts of Joins, SET operations and SQL functions on any two relational schemas

WEEK 5-7:

Apply the concepts of Joins, SET operations and SQL functions on the following schema:

a) Employee:

Name	Datatype	width	Constraint	Description
Empno	Integer	4	Primary Key	Employee Number
Ename	Varchar	20		Employee Name
Job	Char	12		Designation
Mgr	Integer	4		Manager Number
Hiredate	Date			
Sal	Number	(8,2)		Salary
Comm	Number	(6,2)		Commission
Deptno	Integer	2	Foreign Key	Department Number

b) Dept:

Name	Datatype	width	Constraint	Description
Deptno	Integer	2	Primary Key	Department Number
Dname	Varchar	12		Department Name
Loc	Char	10		Location

c) Salgrade:

Name	Datatype	width	Constraint	Description
Grade	Integer	1		Grade
Hisal	Integer	4		Upper
Losal	Integer	5		Lower

WEEK 8 – 11:

End to end implementation of a schema for a specific system along with the illustrations of querying.

A system is described by specifying the functional and non-functional requirements. Based on this description, the major entities are identified and modelled. Further the relationships are modelled to form the initial schema. The schema is further refined by removing redundancies through normalization. Also based on the query requirements, the schema is remodeled to facilitate querying. Finally an illustration of various queries to extract required information from the system is shown using SQL / MYSQL.

The five major workflows to be implemented are:

1. System Specification
2. Design of Initial Schema

3. Schema refinement using functional dependencies and normalization
4. Schema refinement using query requirements
5. Illustration of querying the system using SQL / MYSQL.

WEEK 12:

Implementation of PL / SQL concepts

WEEK 13:

Creating and executing CURSORS.

WEEK 14:

Creation and application of TRIGGERS on a Relational schema.

TEXT BOOKS:

1. Database System Concepts, Silberschatz, Korth, Sudarshan, 7th Edition, McGraw-Hill
2. Introduction to Database Systems, C. J. Date, Pearson Education

REFERENCES:

1. Database Management Systems, Raghu Ramakrishnan, Johannes Gehrke, Tata McGraw-Hill
2. Fundamentals of Database Systems, Elmasri Navathe, Pearson Education
3. Database Systems Design, Implementation, and Management, Peter Rob & Carlos Coronel, 7th Edition, Cengage Learning

ONLINE RESOURCES:

1. <https://www.w3schools.com/sql/default.asp>
2. <https://www.javatpoint.com/dbms-tutorial>

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC2IT201) OBJECT ORIENTED PROGRAMMING THROUGH JAVA LABORATORY

TEACHING SCHEME		
L	T/P	C
0	2	1

EVALUATION SCHEME					
D-D	PE	LR	CP	SEE	TOTAL
10	10	10	10	60	100

COURSE OBJECTIVES:

- To understand OOP principles
- To understand the Exception Handling mechanism
- To understand Java collection framework
- To understand multithreaded programming
- To understand Java database connectivity in Java

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Write the programs for solving real world problems using Java OOP principles

CO-2: Write programs using Exceptional Handling approach

CO-3: Write multithreaded applications

CO-4: Build application using Java Collection Framework

CO-5: Develop java application connect database using JDBC

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	3	2	3	-	2	-	-	1	-	1	2	2	3	2
CO-2	3	2	3	2	-	-	-	1	1	1	-	2	3	3	2
CO-3	2	3	2	3	-	-	-	-	-	-	-	1	2	2	2
CO-4	2	2	3	2	2	2	2	-	-	1	-	2	3	2	2
CO-5	2	2	2	2	2	3	2	1	-	1	1	2	3	2	2

LIST OF PROGRAMS / EXPERIMENTS / EXERCISES:

WEEK 1:

- a. Installation of Java software, study of any integrated development environment, Use Eclipse or Netbean platform and acquaint with the various menus. Create a test project, add a test class and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods and classes. Try debug step by step with java program to find prime numbers between 1 to n.
- b. Write a Java program that prints all real solutions to the quadratic equation ax^2+bx+c . Read in a, b, c and use the quadratic formula.
- c. Write a Java program to multiply two given matrices.

WEEK 2:

- a. Write Java program on use of inheritance, preventing inheritance using final, abstract classes.
- b. Write Java program on dynamic binding, differentiating method overloading and overriding.
- c. Develop a java application to implement currency converter (Dollar to INR. EURO to INR, Yen) using Interfaces.

WEEK 3:

- a. Write a Java program to create a package named "com.mycompany.math" that contains a class named "Calculator" with methods to add, subtract, multiply and divide two numbers. Write a test program to use this package.
- b. Create a package named "com.mycompany.util" that contains a class named "StringUtils" with a method named "reverseString" that takes a string as input and returns the reverse of the input string. Write a test program to use this package.

WEEK 4:

- a. Write a Java program to implement user defined exception handling.
- b. Write a Java program to throw an exception "Insufficient Funds" while withdrawing the amount in the user account.
- c. Write a Java program to implement Try-with Resources, Multi-catch Exceptions, and Exception Propagation Concepts?

WEEK 5:

- a. Write a java program to split a given text file into n parts. Name each part as the name of the original file followed by .part where n is the sequence number of the part file.
- b. Write a Java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable. The type of file and the length of the file in bytes.

WEEK 6:

- a. Write a Java program on Random Access File class to perform different read and write operations.
- b. Create a class called Employee with properties name(String), dateofbirth (java.util.Date), department(String), designation(String) and Salary(double). Create respective getter and setter methods and constructors (no-argument constructor and parameterized constructor) for the same. Create an object of the Employee class and save this object in a file called "data" using serialization. Later using deserialization read this object and prints the properties of this object.

WEEK 7:

- a. Create a generic class called Box that can hold any type of object. Implement the following methods: 1) void set(T obj): sets the object stored in the box 2) T get(): retrieves the object stored in the box 3) boolean isEmpty(): returns true if the box is empty, false otherwise
- b. Implement a generic Stack class that can hold any type of object. Implement the following methods: 1) void push(T obj): pushes an object onto the top of the stack ,2) T pop(): removes and returns the object at the top of the stack 3) boolean isEmpty(): returns true if the stack is empty, false otherwise

WEEK 8:

- a. Write a Java program to implement Autoboxing and Unboxing?
- b. Write a Java program to implement Built-In Java Annotations?

WEEK 9:

- a. Write a Java program that creates three threads. First thread displays —Good Morning every one second, the second thread displays —Hello every two seconds and the third thread displays —Welcome every three seconds.
- b. Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.
- c. Create a Email registration Form using Java AWT. The UI should have fields such as name, address, sex, age, email, contact number, etc.,
- d. Demonstrate various Layout Managers in Java AWT. Display the output in card layout wheareas each card exhibits different layout.

WEEK 10:

- a. Write a Java program to create a Vector and add some elements to it. Then get the element at a specific index and print it.
- b. Write a Java program to create a BitSet and set some bits in it. Then perform some bitwise operations on the BitSet and print the result.
- c. Write a Java program to read the time intervals (HH:MM) and to compare system time if the system Time between your time intervals print correct time and exit else try again to repute the same thing. By using String Tokenizer class.

WEEK 11:

- a. Write a Java program to demonstrate the working of different collection classes. [Use package structure to store multiple classes].
- b. Write a Java program to create a TreeMap and add some elements to it. Then get the value associated with a specific key and print it.
- c. Write a Java program to create a PriorityQueue and add some elements to it. Then remove the highest priority element from the PriorityQueue and print the remaining elements.

WEEK 12:

- a. Deve1op a Java application to establish a JDBC connection, create a table student with properties name, register number, mark 1, mark2, mark3. Insert the values into the table by using the java and display the information of the students at font end.
- b. Write a program to perform CRUD operations on the student table in a database using JDBC

TEXT BOOKS:

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition, Pearson Education
2. Thinking in Java, Bruce Eckel, Pearson Education
3. Understanding Object-Oriented Programming with Java, T. Budd, Pearson Education

REFERENCES:

1. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning
2. Core Java, Volume 1, Cay S. Horstmann and G. Cornell, 9th Edition, Pearson
3. Programming in Java, S. Malhotra, S. Choudhary, 2nd Edition, Oxford University Press

4. Java Programming and Object-Oriented Application Development, R. A. Johnson, Cengage Learning

ONLINE RESOURCES:

1. https://www.w3schools.com/java/java_oop.asp
2. <http://peterindia.net/JavaFiles.html>

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22ES2DS101) PYTHON PROGRAMMING LABORATORY

TEACHING SCHEME		
L	T/P	C
1	2	2

EVALUATION SCHEME					
D-D	PE	LR	CP	SEE	TOTAL
10	10	10	10	60	100

COURSE OBJECTIVES:

- To install and run the Python interpreter
- To learn control structures
- To understand Lists, Dictionaries in Python
- To handle Strings and Files in Python
- To acquire programming skills in core Python

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Develop the application specific codes using Python

CO-2: Understand Strings, Lists, Tuples and Dictionaries in Python

CO-3: Verify programs using modular approach, file I/O, Python standard library

CO-4: Implement digital systems using Python

CO-5: Develop the skill of designing Graphical User Interfaces in Python

COURSE ARTICULATION MATRIX:

(Define Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	2	3	3	2	3	2	2	1	2	2	2	3	3	3	2
CO-2	3	2	3	3	2	3	3	2	1	2	3	3	-	-	-
CO-3	3	2	3	1	2	3	2	3	2	1	2	3	-	-	-
CO-4	3	2	3	2	3	2	3	2	2	2	3	3	3	3	2
CO-5	3	2	3	3	3	3	3	2	3	3	2	3	3	3	2

LIST OF PROGRAMS / EXERCISES:

WEEK-1: Basics

1. i) Use a web browser to go to the Python website <http://python.org>. This page contains information about Python and links to Python-related pages, and it gives you the ability to search the Python documentation.
ii) Start the Python interpreter and type help() to start the online help utility.
2. Start a Python interpreter and use it as a Calculator.
3. Write a program to purposefully raise Indentation Error and correct it
4. i) Write a program to calculate compound interest when principal, rate and number of periods are given.
ii) Given coordinates (x1, y1), (x2, y2) find the distance between two points

5. Read name, address, email and phone number of a person through keyboard and print the details.

WEEK-2: Operations

1. Print the below triangle using for loop.

5

4 4

3 3 3

2 2 2 2

1 1 1 1 1

2. Write a program to check whether the given input is digit or lowercase character or uppercase

character or a special character (use 'if-else-if' ladder)

3. Python Program to Print the Fibonacci sequence using while loop

4. Python program to print all prime numbers in a given interval (use break)

5. Write a program to compute LCM of two numbers by taking input from the user

6. Write a program add.py that takes 2 numbers as command line arguments and prints its sum

WEEK-3: Lists & Tuples

1. i) Write a program to convert a list and tuple into arrays.

ii) Write a program to find common values between two arrays.

2. Write a function called gcd that takes parameters a and b and returns their greatest common divisor.

3. Write a function called palindrome that takes a string argument and returns True if it is a palindrome and False otherwise. Remember that you can use the built-in function len to check the length of a string.

4. Find mean, median, mode for the given set of numbers in a list.

5. Write a Python program to create a tuple.

6. Write a Python program to create a tuple with different data types.

7. Write a Python program to check whether an element exists within a tuple.

WEEK-4: Sets, Dictionaries and Strings

1. Write a function called is_sorted that takes a list as a parameter and returns True if the list is sorted in ascending order and False otherwise.

2. Write a function called has_duplicates that takes a list and returns True if there is any element that appears more than once. It should not modify the original list.

i). Write a function called remove_duplicates that takes a list and returns a new list with only the unique elements from the original. Hint: they don't have to be in the same order.

ii). The wordlist I provided, words.txt, doesn't contain single letter words. So you might want to add

"I", "a", and the empty string.

iii). Write a python code to read dictionary values from the user. Construct a function to invert its

content. i.e., keys should be values and values should be keys.

3. i) Add a comma between the characters. If the given word is 'Apple', it should become 'A,p,p,l,e'

ii) Remove the given word in all the places in a string?

iii) Write a function that takes a sentence as an input parameter and replaces the first letter of every

word with the corresponding upper case letter and the rest of the letters in the word by

corresponding letters in lower case without using a built-in function?

4. Write a recursive function that generates all binary strings of n-bit length

5. Write a Python program to implement all set operations

6. Write a program to check whether a string is palindrome or not

WEEK-5: Functions and Multi-D Lists

1. i) Write a python program that defines a matrix and prints

ii) Write a python program to perform addition of two square matrices

iii) Write a python program to perform multiplication of two square matrices

2. Simple Calculator program by making use of functions

3. Find the factorial of a number using recursion

4. Write a function cumulative_product to compute cumulative product of a list of numbers.

5. Write a function reverse to print the given list in the reverse order.

WEEK-6: Exceptions in Python

1. Write a program that detects an Exception

2. Write a program that raise an Exception (divide by zero error,voter's age validity

3. Write a program that raise an Exception as string(), student mark range validation)

4. Use the structure of exception handling all general purpose exceptions.

5. Write a python code to read a phone number and email-id from the user and validate it for correctness.

WEEK-7: Modules and Inheritance

1. How do you make a module? Give an example of construction of a module using different geometrical shapes and operations on them as its functions.

2. a. Write a function called draw_rectangle that takes a Canvas and a Rectangle as arguments and draws a representation of the Rectangle on the Canvas.

b. Add an attribute named color to your Rectangle objects and modify draw_rectangle so that it uses the color attribute as the fill color.

c. Write a function called draw_point that takes a Canvas and a Point as arguments and draws a representation of the Point on the Canvas.

d. Define a new class called Circle with appropriate attributes and instantiate a few Circle objects. Write a function called draw_circle that draws circles on the canvas.

3. Write a Python program to demonstrate the usage of Method Resolution Order (MRO) in multiple levels of Inheritance.

WEEK-8: Files

1. Write a Python code to merge two given file contents into a third file.
2. Write a Python code to open a given file and construct a function to check for given words present in it and display on found.
3. Write a Python code to Read text from a text file, find the word with most number of occurrences
4. Write a function that reads a file file1 and displays the number of words, number of vowels, blank spaces, lower case letters and uppercase letters.
5. Write a program to print each line of a file in reverse order.

WEEK-9: Exploration of NumPy Package

1. Import numpy, and explore their functionalities.
2. a) Install NumPy package with pip and explore it.
3. Write a program for slicing arrays using numpy
4. Write a program for Math operations on array using numpy
5. Write a program for searching
6. Write a program for sorting

WEEK-10: Exploration of Pandas Package

1. Import Pandas and Plotpy and explore their functionalities
2. Python Data Frame
3. Python series

WEEK-11: Exploration of SciPy and GUI

1. Import SciPy and explore their functionalities
2. Write a GUI program to create a window wizard having two text labels, two text fields and two buttons as Submit and Reset.

WEEK-12: Digital Logic Systems

1. Write a program to implement Digital Logic Gates – AND, OR, NOT, EX-OR
2. Write a program to implement Half Adder, Full Adder, and Parallel Adder

TEXT BOOKS:

1. Supercharged Python: Take Your Code To The Next Level, Overland
2. Learning Python, Mark Lutz, O'Reilly
3. Python for Data Analysis, 2nd Edition, Wes McKinney, O'Reilly

REFERENCES:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Python Programming A Modular Approach with Graphics, Database, Mobile, and Web Applications, Sheetal Taneja, Naveen Kumar, Pearson
3. Programming with Python, A User's Book, Michael Dawson, Cengage Learning
4. Think Python, Allen Downey, Green Tea Press
5. Core Python Programming, W. Chun, Pearson

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22SD5AM202) FIELD PROJECT

TEACHING SCHEME		
L	T/P	C
0	2	1

EVALUATION SCHEME		
CIE	SEE	TOTAL
50	-	50

COURSE OBJECTIVES:

- To identify, analyze and solve industry / technical / societal problems creatively through sustained critical investigation
- To practice the skills, elegance and commitment to excellence needed to engage in lifelong learning
- To demonstrate an awareness and application of appropriate personal, social and professional ethical standards

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Understand the formulated industry / technical / societal problems

CO-2: Apply fundamental and disciplinary concepts and methods in ways appropriate to their principal areas of study

CO-3: Demonstrate skills and knowledge of current information, technological tools and techniques specific to the professional field of study

CO-4: Analyze and / or develop models for providing solution to industry / technical / societal problems

CO-5: Use effectively oral, written and visual communication

COURSE ARTICULATION MATRIX:

(Define Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	2	-	-	-	-	3	3	-	3	-	1	3	-	-	-
CO-2	3	3	2	2	3	2	-	-	3	-	2	3	-	-	-
CO-3	-	-	3	-	3	2	-	2	3	3	3	3	-	-	-
CO-4	2	3	3	3	3	3	3	3	3	-	3	3	-	-	-
CO-5	-	-	-	-	2	-	-	3	3	3	-	3	-	-	-

COURSE OUTLINE:

Filed project-based learning offers students real world opportunities to research issues, think critically, gain new perspectives, solve problems and develop written and oral communication skills all within the framework of a team environment and guided by engaged and involved faculty

- A student shall undergo a one credit Field Project course in II year.

- It shall be a project based course involving the student to undertake issues for industries, companies, and any organizations which they encounter in their day-to-day work.
- Evaluation of the field project shall consist of Continuous Internal Evaluation (CIE) only for 50 marks.
- CIE shall be done by a Project Review Committee (PRC) consisting of the Head of the Department, faculty supervisor and a senior faculty member of the specialization / department.
- The internal evaluation shall be on the basis of two seminars for 50 marks one before SE-I and the other before SE-II as per the calendar dates and evaluation format.
- CIE shall be carried out for 50 marks on the basis of review presentation as per the calendar dates and evaluation format.
- The field project report shall be accepted for submission to the PRC only upon meeting the prescribed similarity index of less than 25%.

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22MN6HS103) HAPPINESS AND WELLBEING

TEACHING SCHEME		
L	T/P	C
2	0	0

EVALUATION SCHEME			
SE-I	SE-II	SEE	TOTAL
50	50	-	100

COURSE OBJECTIVES:

- To learn sustainable strategies to develop positive attitude and happy heart
- To develop self-awareness and self-discipline to meet the needs of happiness
- To practice good health & mindfulness for wellbeing
- To adapt personality attributes of happiness and success strategies
- To nature happiness development index for better living

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Recognize what is happiness in life and how to sustain it

CO-2: Focus on interpersonal skills for a mindful approach

CO-3: Develop to mindfulness to handle challenging situations

CO-4: Recognize the importance of positive attitude for personal and professional development

CO-5: Interpret the need for nurturing happiness development index through Indicators

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	-	-	-	-	-	3	-	2	1	-	-	3	-	-	-
CO-2	-	-	-	-	-	3	-	3	1	-	-	3	-	-	-
CO-3	-	-	-	-	-	1	-	1	1	-	-	3	-	-	-
CO-4	-	-	-	-	-	2	-	2	1	-	-	3	-	-	-
CO-5	-	-	-	-	-	3	-	1	1	-	-	3	-	-	-

UNIT-I:

Introduction to Happiness: Definition & theories of happiness: Hedonism theory, Desire theory, Objective list theory. Identifying potential barriers of happiness: Devaluing happiness, chasing superiority, being needy, being overly control-seeking, distrusting others, distrusting life, and ignoring the source within. Strategies for overcoming the potential barriers

UNIT – II:

Power of Emotions & Relationships: Role of emotional intelligence, self-awareness, and empathy in creating harmonious relationship with ourselves and others. Balancing emotions. Hormones that promote happiness. The importance of social connections for happiness. Role of share & care, gratitude, forgiveness & kindness in building relationships

UNIT – III:

Health and Wellbeing: The link between health & happiness-exercise regularly, eat a healthy diet, get enough sleep for physical fitness. Mental wellbeing-Take notice, keep learning, stay connected with nature, and financial wellbeing. The practice of mindfulness and its benefits for mental and physical health. Moving from restlessness to restfulness- meditation and yoga to increase awareness and reduce stress

UNIT – IV:

Re-wirement for Wellbeing: Abundance in life, freedom of choice, accepting change, ways of implementation for wellbeing: practicing habits-be proactive, begin with end-in-mind, put-first things-first, think win-win, seek first to understand then to be understood, synergize, sharpen the saw, and effectiveness to greatness

UNIT – V:

Nurturing Happiness Development Index: Exploring the sources of temporary joy and lasting happiness. Acceptance, Appreciation, forgiveness, gracefulness, and creative procrastination. Time management with four D's (delete, delay, delegate, do). Developing happiness index-track changes in happiness levels over time and identify the indicators

TEXT BOOKS:

1. The How of Happiness: A Scientific Approach to Getting the Life You Want, Sonja Lyubomirsky, Penguin Books, 2008
2. Authentic Happiness: Using the New Positive Psychology to Realize Your Potential for Lasting Fulfillment, Martin Seligman, Atria Books, 2004
3. The Book of Joy: Lasting Happiness in a Changing World, Dalai Lama, Desmond Tutu, and Douglas Abrams, Avery, 2016

REFERENCES:

1. 7-Habits of Highly Successful People, Stephen Covey, Simon & Schuster, 2020
2. Mindfulness Book of Happiness: Mindfulness and Meditation, Aimen Eman, Publish Drive Edition, 2018
3. Mindfulness at Work: How to Avoid Stress, Achieve More and Enjoy Life, Dr. Stephen McKenzie, Exisle Publishing, 2014
4. The 8th Habit: From Effectiveness to Greatness, Stephen R. Covey, Free Press, 2004

ONLINE RESOURCES:

1. Life of Happiness and Fulfillment, Indian School of Business, Coursera <https://in.coursera.org/learn/happiness>
2. Science of Wellbeing, Yale University, Coursera, <https://www.coursera.org/>

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22BS1MT205) STATISTICAL INFERENCE AND QUEUING THEORY

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE PRE-REQUISITES: Statistical Methods for Data Analysis

COURSE OBJECTIVES:

- To learn the concept of sampling distribution
- To learn the various methods to test the hypothesis for large sample
- To learn the various methods to test the hypothesis for small sample
- To learn the linear and polynomial relationship between the given data set
- To learn the basic concepts of queuing theory

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Evaluate sampling distribution of means, variance and confidence interval

CO-2: Apply the knowledge to test the hypothesis for large sample

CO-3: Apply the knowledge about small sample tests based on Chi-square, t and F distributions

CO-4: Use and fit a linear and polynomial regression model to data and use it for prediction

CO-5: Apply the knowledge of different probability distributions to solve problems in queuing theory

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	2	1	1	1	-	-	-	-	-	-	-	2	-	1
CO-2	3	3	2	2	2	-	-	-	-	-	-	-	1	-	1
CO-3	3	2	1	1	-	-	-	-	-	-	-	-	2	-	1
CO-4	3	3	2	2	2	-	-	-	-	-	-	-	2	-	1
CO-5	3	2	-	-	-	-	-	-	-	-	-	-	1	-	1

UNIT- I:

Sampling Distributions: Definition of population, sampling, statistic, parameter, Types of sampling, sample mean and Variance, sampling distribution, standard error, sampling distributions of means and variance, Estimation, interval estimation, point estimation and confidence interval for the mean and proportions.

UNIT- II:

Testing of Hypothesis - Large Samples: Central limit theorem, Tests of hypothesis - null hypothesis, alternate hypothesis, type I, type II errors, critical region. Inferences

concerning means and proportions- Large samples- test of hypothesis for single mean and difference between the means. Test of hypothesis for the proportions- single and difference between the proportions.

UNIT- III:

Testing of Hypothesis - Small Samples: Test of significance-t distribution, confidence interval for the t- distribution, F-distribution and Chi square distribution, Analysis of variance (ANOVA).

UNIT- IV:

Correlation and Regression: Correlation, Correlation coefficient and its properties, rank correlation coefficient, Multiple and partial correlation, Introduction to Linear Regression, Estimation of parameters of β_0 and β_1 , Multiple linear regression, Inferences for Linear Regression, Regression of second degree polynomial by least square method.

UNIT -V:

Queuing Theory: Queuing theory -Arrival process and service process- Pure birth and death process, M/M/1 model with finite and infinite capacities, M/M/C model with infinite capacity.

TEXT BOOKS:

1. Applied Probability, I. N. Blake, 9th Edition, John Wiley & Sons, 1979
2. Introductory Statistics, Thomas H. Wonnacott & Ronald J. Wonnacot, John Wiley & Sons, 1969
3. The Single Server Queue, J. W. Cohen, Wiley Interscience, 1969

REFERENCES:

1. Applied Statistics and Probability for Engineers, Douglas C. Montgomery, George C. Runger, 3rd Edition, John Wiley & Sons, 2003
2. Probability and Statistics for Engineers, Richard A. Johanson, 5th Edition, Prentice Hall, 1995
3. Applied Statistics for Engineers, Jay L. Devore, Nicholas R. Famum, Jimmy A. Doi, 3rd Edition, Cengage Learning
4. Some Problems in the Theory of Queues, D. G. Kendall, Journal of the Royal Statistical Society, Series B, 13, 151–185, 1951

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PC1DS202) DIGITAL LOGIC DESIGN AND COMPUTER ORGANIZATION

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE PRE-REQUISITES: IT Workshop

COURSE OBJECTIVES:

- To learn about the functional blocks and data representation of computer system
- To understand the design principles of processor
- To understand the organization and management of memory
- To understand the management of peripheral devices

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Evaluate digital number systems and use Boolean algebra theorems, properties and Canonical forms for digital logic circuit design

CO-2: Design combinational logic circuits and Sequential logic circuits

CO-3: Understand computer architecture and data representation to perform computer arithmetic operations

CO-4: Illustrate the design principles of control unit and pipelining

CO-5: Analyze the memory hierarchy in a computer system

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	3	3	2	2	1	1	-	3	-	-	3	1	3	3
CO-2	3	3	3	3	3	3	3	-	3	-	-	3	3	2	3
CO-3	3	3	3	3	3	3	3	-	3	-	-	3	3	3	3
CO-4	3	3	3	3	3	3	3	-	3	-	-	3	3	2	3
CO-5	3	3	3	3	3	3	3	-	3	-	-	3	3	3	2

UNIT – I:

Number Systems, Logic Gates and Boolean Algebra:

Number Systems: Binary, Octal, Decimal, Hexadecimal Number Systems. Conversion of Numbers from One, Radix to another Radix, r's Complement and (r-1)'s Complements, Binary Codes, Basic Gates: NOT, AND, OR, Universal Gates: NAND, NOR, Special Gates: Ex-OR and Ex-NOR Gates

Boolean Algebra: Fundamental postulates of Boolean algebra, Basic theorems and properties, Complement and Dual of Logical Expressions, SOP, POS, Minimizations of Logic Functions Using Boolean Theorems, Karnaugh Map Method (K-Map): Minimization of Boolean Functions maximum up to Four Variables, Simplifications with Don't Care Conditions Using K-Map.

UNIT-II:

Combinational Logic Circuits & Sequential Logic Circuits: Design of Half Adder, Full Adder, Half Subtractor, Full Subtractor, Ripple Carry Adder. Design of Decoders, Encoders, Multiplexers, De-multiplexers, Priority Encoder,

Sequential Logic Circuits: Latch and Flip-Flop, RS- Latch Using NAND and NOR Gates, RS, JK, T and D Flip-flops, Truth and Excitation Tables, Conversion of Flip Flops, Master-Slave Flip-flops. Registers and Counters

UNIT – III:

Functional Blocks of a Computer: CPU, Memory, Input-Output Subsystems, Control Unit. Instruction Set Architecture of A CPU–Registers, Instruction Execution Cycle, RTL Interpretation of Instructions, Addressing Modes.

Data Representation: Fixed and Floating-Point Representations. Computer Arithmetic – Integer Addition and Subtraction, Multiplication – Shift-And Add, Booth Multiplier.

UNIT – IV:

CPU Control Unit Design: Hardwired and micro-programmed design approaches.

Pipelining: Basic concepts of pipelining, throughput and speedup, pipeline hazards.

UNIT-V:

Memory System Design: Memory hierarchy, Cache Memory and Main Memory.

Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers–program controlled, interrupt driven and DMA.

TEXTBOOKS:

1. Digital Design, M. Morris Mano, Michael D. Ciletti, 4th Edition, Pearson Education, 2008
2. Switching and Finite Automata Theory, Zvi Kohavi, Tata McGraw-Hill
3. Computer Organization and Design: The Hardware/Software Interfaces, David A. Patterson and John L. Hennessy, 5th Edition, Elsevier
4. Computer Organization and Embedded Systems, Carl Hamacher, 6th Edition, McGraw-Hill

REFERENCES:

1. Fundamentals of Logic Design, Roth, 5th Edition, Thomson
2. Switching and Logic Design, C. V. S. Rao, Pearson Education
3. Digital Principles and Design, Donald D. Givone, Tata McGraw-Hill
4. Computer System Architecture, M. Morris Mano, 3rd Edition
5. Computer Architecture and Organization, John P. Hayes, 3rd Edition, WCB/McGraw-Hill

ONLINE RESOURCES:

1. <https://nptel.ac.in/courses/108106177>
2. <https://nptel.ac.in/courses/106105163>

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PC1CS201) SOFTWARE ENGINEERING

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE OBJECTIVES:

- To identify the importance of software engineering principles and software process framework
- To understand contemporary approaches for design models and requirements validation
- To explore various metrics and quality assurance strategies
- To analyze different strategies for testing and risk management

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Analyze software engineering framework activities that can be tailored with appropriate methods for developing the projects

CO-2: Design relevant software system models from the available software requirements and validate desired user models with realistic constraints

CO-3: Understand the Requirements Engineering Process and model the system using the perception of UML

CO-4: Deliver quality software products by applying software testing strategies and product metrics over the entire system life cycle

CO-5: Specify contemporary issues of handling risk management and quality management in software development

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	3	2	2	2	2	1	1	1	1	2	2	1	3	3
CO-2	3	3	3	2	3	2	1	1	1	1	2	3	1	3	3
CO-3	3	3	2	2	2	2	1	1	1	1	2	3	1	3	3
CO-4	3	3	2	2	3	2	3	1	1	1	2	3	1	3	3
CO-5	3	3	2	2	3	2	3	1	1	1	2	3	1	3	3

UNIT – I:

Introduction to Software Engineering: Software characteristics, changing nature of software, software myths.

A Generic View of Process: Software engineering-A layered technology, process framework, The Capability Maturity Model Integration (CMMI)

UNIT – II:

Process Models: The waterfall model, spiral model, Incremental process model, evolutionary process model, agile process

Software Requirements: Functional and non-functional requirements, the software requirements document.

Requirements Engineering Process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management, and forward and reverse engineering.

UNIT – III:

Modeling with UML: Modeling Concepts and Diagrams - Use Case Diagrams - Class Diagrams - Interaction Diagrams - State chart Diagrams – Activity Diagrams - Package Diagrams - Component Diagrams – Deployment Diagrams -Diagram Organization-Diagram Extensions.

Design Engineering: The design process and design quality, design concepts, and design model.

UNIT – IV:

Testing Strategies: A strategic approach to software testing, verification and validation, Testing Strategies, Black box, and White box testing, and deployment.

Product Metrics: Metrics for analysis model, Metrics for design model, Metrics for source code, Metrics for testing, Metrics for maintenance

Metrics for Process and Projects: Software measurement, Metrics for software quality

UNIT – V:

Risk Management: Reactive vs. Proactive risk strategies, Software risks, Risk identification, Risk projection, RMMM plan

Quality Management: Quality concepts, Software quality assurance, Formal technical reviews, ISO 9000 Quality standards.

TEXT BOOKS:

1. Software Engineering - A Practitioner's Approach, Roger S. Pressman, 6th Edition, McGraw-Hill, 2001
2. Software Engineering, Ian Sommerville, 7th Edition, Pearson Education Asia, 2000
3. The Unified Modeling Language User Guide, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education

REFERENCES:

1. An Integrated Approach to Software Engineering, Pankaj Jalote, Springer Verlag, 1997
2. Software Engineering – An Engineering Approach, James F. Peters and Witold Pedrycz, John Wiley and Sons, 2000
3. Software Engineering Fundamentals, Ali Behforooz and Frederick J. Hudson, Oxford University Press, 1996

ONLINE RESOURCES:

1. <https://www.guru99.com/what-is-software-engineering.html>
2. https://www.tutorialspoint.com/software_engineering/index.htm

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PC1AM202) DATA ENGINEERING

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE OBJECTIVES:

- To explore data preprocessing techniques
- To learn the techniques related to feature engineering
- To exploit the statistics for data engineering
- To explore the Hadoop environment and its framework activities
- To understand, explore various databases for big data analytics

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Perform data preprocessing and apply the related techniques

CO-2: Demonstrate feature engineering techniques and handle the issues related to high dimensionality

CO-3: Analyse the statistics required for data engineering

CO-4: Illustrate Hadoop and its framework activities to deal with the data process analytics

CO-5: Work with databases for big data analytics

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	3	2	3	1	1	-	-	-	-	-	1	3	1	3
CO-2	3	3	2	2	2	1	-	-	-	-	-	1	3	1	3
CO-3	3	3	3	3	3	1	-	-	-	-	-	1	3	1	3
CO-4	3	3	3	3	3	1	-	-	-	-	-	1	3	1	3
CO-5	3	3	3	3	3	1	-	-	-	-	-	1	3	1	3

UNIT – I:

Data Preprocessing: Types of data, exploring structure of data: Exploring and Plotting numerical data, categorical data and relationship between variables, data quality and remediation, data pre-processing: Dimensionality reduction and feature selection.

UNIT – II:

Feature Engineering: Feature, feature engineering, feature transformation: Feature construction, Feature extraction, Feature subset selection: issues in high dimensional data, key drivers of feature selection – feature relevance and redundancy, measures of feature relevance and redundancy, overall feature selection process, feature selection approaches

UNIT – III:

Statistics for Data Engineering: Importance of statistical tools for handling data, concept of probability- frequentist and Bayesian interpretation, review of probability theory, random variables-discrete random variables, continuous random variables, common discrete distributions-Bernoulli distributions, Binomial distribution, multinomial and multinoulli distributions, Poisson distribution, common continuous distributions-uniform distribution, Gaussian distribution, Laplace distribution. Multiple random variables-Bivariate random variables, joint distribution functions, joint probability mass functions, joint probability density functions, conditional distributions, covariance and correlation, central limit theorem, Sampling distributions – Sampling with replacement, sampling without replacement, Mean and variance of sample, hypothesis testing, Monte Carlo approximation.

UNIT – IV:

Types of Digital Data, Introduction to Big Data: Characteristics of Data, Evolution of Big Data and Challenges with Big Data, Big Data, Terminologies used in Big Data Environment.

Introduction to Hadoop: Features of Hadoop, Why Hadoop, RDBMS vs Hadoop, Hadoop Overview, HDFS, Processing Data with Hadoop.

UNIT – V:

NoSQL: Basics of NoSQL - uses, Types of NoSQL databases, significance of NoSQL. Advantages of NoSQL, SQL vs NoSQL.

MongoDB: uses and need of MongoDB, MongoDB Query Language: Insert, Save, Update, Remove, Find methods, Dealing with NULL values, Count, Limit, Sort and Skip, Arrays, Aggregate Functions and MapReduce Functions.

TEXT BOOKS:

1. Machine Learning, Saikat Dutt, Subramanian Chandramouli, Amit Kumar Das, Pearson India
2. Big Data and Analytics, Seema Acharya, Subhasinin Chellappan, Wiley
3. Machine Learning, Tom M. Mitchell, McGraw-Hill Education

REFERENCES:

1. Introduction to Data Mining, Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Pearson
2. Hadoop: The Definitive Guide, Tom White, 3rd Edition, O'reilly Media, 2012

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PC1AM203) AUTOMATA AND COMPILER DESIGN

TEACHING SCHEME		
L	T/P	C
3	0	3

EVALUATION SCHEME				
SE	CA	ELA	SEE	TOTAL
30	5	5	60	100

COURSE OBJECTIVES:

- To explain the relationships between languages and machines, the inherent efficiency of solving problems using FA
- To convert among equivalently powerful notations for a language, including DFAs, NFAs, and regular expressions
- To outline the usage of different phases of compiler and understand the various techniques of parsing in a compilation process
- To explain the different representations of intermediate code
- To explain the code generator for the optimized code

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Design solutions for problems related to FA, RE, and CFG

CO-2: Construct the parse tree for checking the grammatical errors in programming Languages

CO-3: Design the solutions for the problems related to different types of parsing techniques

CO-4: Analyze different representations of intermediate code

CO-5: Apply optimization techniques to optimize the code and generate object code

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	3	3	3	3	2	-	-	1	-	1	1	3	2	3
CO-2	3	3	3	3	3	2	-	-	1	-	1	1	3	2	3
CO-3	3	3	3	3	3	2	-	-	1	-	1	1	2	2	-
CO-4	3	3	3	3	3	2	-	-	1	-	1	1	2	2	-
CO-5	3	3	3	3	3	2	-	-	1	-	1	1	3	2	-

UNIT – I:

Formal Languages and Regular Expressions: Definition of Languages, Finite Automata – DFA, NFA, regular expressions, Conversion of regular expression to NFA, NFA to DFA, Pumping Lemma for regular languages, lex tools.

UNIT – II:

Overview of Compilation: Phases of Compilation – Lexical Analysis, Pass and Phases of translation, interpretation, bootstrapping, data structures in compilation.

Context-free Grammars and Parsing: Context free grammars, derivation, parse trees, ambiguity, LL(K) grammars and LL (1) parsing, bottom-up parsing, handle pruning, LR Grammar Parsing, LALR parsing, YACC programming specification.

UNIT – III:

Semantics: Syntax directed translation, S-attributed and L-attributed grammars, Intermediate code – abstract syntax tree, translation of simple Assignments statements and control flow statements.

UNIT – IV:

Run Time Environments: Storage organization, storage allocation strategies, access to non-local names, language facilities for dynamics storage allocation.

Code Optimization: Principal sources of optimization, Optimization of basic blocks, peephole optimization, flow graphs, optimization techniques.

UNIT – V:

Code Generation: Machine dependent code generation, object code forms, generic code generation algorithm, Register allocation and assignment. Using DAG representation of Block.

TEXTBOOKS:

1. Introduction to Automata Theory Languages & Computation, John E. Hopcroft, Rajeev M. & J. D. Ullman, 3rd Edition, Pearson Education, 2007
2. Compilers: Principles, Techniques and Tools, A. V. Aho, Ravi Sethi, J. D. Ullman, 2nd Edition, Pearson Education, 2009

REFERENCES:

1. The Theory & Practice of Compiler writing, Tremblay J. P., Sorenson G. P., 1st Edition, BSP Publication, 2010
2. Modern Compiler Implementation in C, Andrew W. Appel, 1st Edition, Cambridge University Press, 2003
3. Compiler Construction, Principles & Practice, Loudon, 1st Edition, Thomson Press, 2006
4. Introduction to Theory of Computation, Sipser Michael, 1st Edition, Thomson, 2009

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PC2AM202) DATA ENGINEERING LABORATORY

TEACHING SCHEME		
L	T/P	C
0	2	1

EVALUATION SCHEME					
D-D	PE	LR	CP	SEE	TOTAL
10	10	10	10	60	100

COURSE OBJECTIVES:

- To explore and understand various data management and handling methods
- To understand the concept of data engineering
- To explore Hadoop framework and its components
- To use Big Data tools and techniques for data processing

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Implement data management and handling methods

CO-2: Implement data engineering methods

CO-3: Acquire the skills to work with Hadoop framework activities

CO-4: Acquire the skills to work with NOSQL

CO-5: Use various Big Data tools and techniques for data management

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	3	3	3	3	2	-	-	1	-	1	1	3	2	3
CO-2	3	3	3	3	3	2	-	-	1	-	1	1	3	2	3
CO-3	3	3	3	3	3	2	-	-	1	-	1	1	3	2	3
CO-4	3	3	3	3	3	2	-	-	1	-	1	1	3	2	3
CO-5	3	3	3	3	3	2	-	-	1	-	1	1	3	2	3

LIST OF PROGRAMS / EXPERIMENTS / EXERCISES:

WEEK-1:

Basic Data Handling Commands:

1. Read data from the csv file
2. Dimension of the data
3. Display data (top 5 rows and total data)
4. List the column names of a data frame
5. Change columns of a data frame
6. Display specific single column or multiple columns of a data frame
7. Bind sets of rows of dataframes
8. Bind sets of columns of data frames.
9. Find missing values in the dataset

WEEK-2:

1. Measures of central tendency – mean, median, mode
2. Measures of data spread
3. Dispersion of data – variance, standard deviation
4. Position of the different data values – quartiles, inter-quartile range (IQR).

WEEK-3:

Basic Plots for Data Exploration (Use Iris dataset):

1. Generate box plot for each of the four predictors.
2. Generate box plot for a specific feature
3. Generate histogram for a specific feature
4. Generate Scatter plot of petal length vs. sepal length

WEEK-4:

Data Pre-Processing Methods: Use AutoMPG Dataset and Perform the Following Tasks

1. Removing outliers / missing values.
2. Imputing standard values
3. Capping of values

WEEK-5:

Feature Construction: (Use packages that are applicable)

1. Dummy coding categorical(nominal) variables.
2. Encoding categorical(ordinal) variables.
3. Transforming numeric(continuous)features to categorical features

WEEK-6:

Feature Extraction: (Use packages that are applicable)

1. Principal Component Analysis (PCA)
2. Singular Value Decomposition (SVD)
3. Linear Discriminant Analysis (LDA)
4. Feature Subset Selection

WEEK-7:

HDFS(Storage)

A. Hadoop Storage File System

Your first objective is to create a directory structure in HDFS using HDFS commands. Create the local files using Linux commands and move the files to HDFS directory and vice versa.

- i. Write a command to create the directory structure in HDFS.
- ii. Write a Command to move file from local unix/linux machine to HDFS.

B. Viewing Data Contents, Files and Directory

Try to perform these simple steps:

- i. Write HDFS command to see the contents of files in HDFS.
- ii. Write HDFS command to see contents of files which are present in HDFS.

C. Getting Files data from the HDFS to Local Disk

- i. Write a HDFS command to copy the file from HDFS to local file system.

To process any data first move the data in HDFS. All files stored in HDFS can be accessed using HDFS commands.

Lab Instructions:

i. Your objective is to use HDFS commands to move data to HDFS for processing data.

WEEK-8 & 9:

Map Reduce Programming (Processing data).

Hadoop Map-Reduce framework is developed using Java, but the framework allows you to write programs in other languages as well.

Word Count

The word count problem is the most famous using map reduce program. Same thing we can do with java but takes lot of time with huge file, in MR it will process less time even with huge and distributed files. The objective is to count the frequency of words of a large text.

Lab Instructions:

Develop MapReduce example program in a MapReduce environment to find out the number of occurrences of each word in a text file

WEEK-10 & 11:

Data Processing Tool – Hive (NoSQL query based language)

Hive command line tool allows you to submit jobs via bash scripts.

Identifying properties of a data set:

We have a table 'user data' that contains the following fields: data_date: string
user_id: string
properties: string

The properties field is formatted as a series of attribute=value pairs. Ex: Age=21; state=CA; gender=M;

Lab Instructions:

1. Create the table in HIVE using hive nosql based query.
2. Fill the table with sample data by using some sample data bases.
3. Write a program that produces a list of properties with minimum value (min_value), largest value (max_value) and number of unique values. Before you start, execute the prepare step to load the data into HDFS.
4. Generate a count per state.
5. Now that extracted the properties, calculate the number of records per state.

Lab Instructions:

1. Write a program that lists the states and their count from the data input.

WEEK-12 & 13:

MongoDB:

Create a database and explore basic database commands. Explore data types in MongoDB

Create a database with sample dataset and work with MongoDB Query Language

WEEK-14: Internal Lab Exam

TEXT BOOKS:

1. Machine Learning, Saikar Dutt, Subramanian Chandramouli, Amit Kumar Das, Pearson India
2. Big Data and Analytics, Seema Acharya, Subhasini Chellappan, Wiley Publications
3. Machine Learning, Tom M. Mitchell, McGraw-Hill Education

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PC2AM213) COMPILER DESIGN AND SOFTWARE ENGINEERING LABORATORY

TEACHING SCHEME		
L	T/P	C
0	2	1

EVALUATION SCHEME					
D-D	PE	LR	CP	SEE	TOTAL
10	10	10	10	60	100

COURSE OBJECTIVES:

- To implement Lexical Analyzer using Lex tool and Syntax Analyzer using YACC tool
- To implement front end of compiler by means of generating intermediate codes
- To impart technical knowledge of software engineering principles
- To explore and demonstrate design considerations and testing phase of software development lifecycle phase

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Design Lexical Analyzer for a given language using C and Lex tool

CO-2: Design various parsers and generate intermediate code for given Language

CO-3: Translate end-user requirements into system and software requirements

CO-4: Generate a high-level design of the system from the software requirements

CO-5: Explore various testing strategies and able to generate testing reports for sample case studies

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	3	3	3	2	3	-	-	-	-	-	-	1	-	-	-
CO-2	3	3	3	2	3	-	-	-	-	-	-	1	-	-	-
CO-3	2	-	3	-	2	-	1	-	1	2	3	-	2	3	3
CO-4	2	-	3	-	2	-	1	-	1	2	2	-	2	3	3
CO-5	1	-	3	-	2	-	1	-	1	2	2	-	1	3	3

LIST OF PROGRAMS / EXERCISES:

COMPILER DESIGN LABORATORY:

WEEK-1:

Design a Lexical Analyzer for C language. The Lexical Analyzer should ignore redundant spaces, tabs and newlines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value.

WEEK-2:

Implement the Lexical Analyzer using Lex.

WEEK-3:

Design predictive parser for a given language.

WEEK-4:

Design LALR bottom up parser for a given language.

WEEK-5:

Convert the BNF rules into YACC form and Write code to generate abstract syntax tree.

WEEK-6:

Construct YACC code to perform arithmetic operations.

WEEK-7:

Write a program to generate intermediate code.

SOFTWARE ENGINEERING LABORATORY:**WEEK-8:**

Development of problem statement.

WEEK-9:

Preparation of Software Requirement Specification Document

WEEK-10:

Implementing Structural modelling design phase using case tools.

WEEK- 11 & 12:

Implementing Behavioral and Architectural I modelling design phase using case tools.

WEEK-13:

Develop test cases for unit testing and integration testing

WEEK-14:

Develop test cases for various white box and black box testing techniques.

TEXT BOOKS:

1. Introduction to Automata Theory Languages & Computation, John E. Hopcroft, Rajeev M. & J. D. Ullman, 3rd Edition, Pearson Education, 2007
2. Compilers: Principles, Techniques and Tools, A. V. Aho, Ravi Sethi, J. D. Ullman, 2nd Edition, Pearson Education, 2009
3. Software Engineering, A Practitioner's Approach, Roger S. Pressman, 6th Edition, McGraw-Hill

REFERENCES:

1. The Unified Modeling Language User Guide, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education
2. The Theory & Practice of Compiler writing, Tremblay J. P., Sorenson G. P., 1st Edition, BSP Publication, 2010
3. Modern Compiler Implementation in C, Andrew W. Appel, 1st Edition, Cambridge University Press, 2003
4. Compiler Construction, Principles & Practice, Loudon, 1st Edition, Thomson Press, 2006
5. Introduction to Theory of Computation, Sipser Michael, 1st Edition, Thomson, 2009

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22SD5DS201) DATA VISUALIZATION

TEACHING SCHEME		
L	T/P	C
0	2	1

EVALUATION SCHEME						
D-D	PE	LR	CP	VV	SEE	TOTAL
10	10	10	10	10	-	50

COURSE OBJECTIVES:

- To install and run the R studio for data analysis
- To understand principles and techniques for data visualization
- To visualize the data that can improve comprehension, communication, and decision making
- To implement various tools and methods for easy interpretation of data

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Understand the importance of data visualization in analytics

CO-2: Gain knowledge in the principles of data visualization

CO-3: Apply the principles of data visualization on toy datasets using R

CO-4: Analyze data towards decision making using visualization

CO-5: Identify appropriate/suitable visualization for particular requirements imposed by the data type and analytics algorithms

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	2	2	2	3	3	-	-	-	-	-	-	2	3	2	-
CO-2	1	1	2	2	3	-	-	-	-	-	-	1	2	2	-
CO-3	2	2	1	1	3	-	-	-	-	-	1	1	3	2	2
CO-4	2	2	1	1	2	-	-	-	-	-	1	1	2	3	3
CO-5	1	1	2	2	1	-	-	-	-	-	2	1	1	2	2

EXERCISE 1: Basics

Introduction to basic components of R programming, overview of visualization, data types, basics of plotting graphs, different types of graphs in analytics

EXERCISE 2: Importance of Visualizations

Principles of communicating data, Principles of encoding data to make visualizations, Importance of color in visualizations

EXERCISE 3: Plots using Basic R

EXERCISE 3.1: Plots with One Categorical, Continuous Variable

Functions in R for plotting, plots with one categorical variable, plots with one continuous variable, plots with one categorical and one continuous variable

EXERCISE 3.2: Plots with 2 Continuous Variables

Plots with two continuous variables, controlling various aesthetics of the graph.

EXERCISE 4: ggplot2 in R

Group manipulation and data reshaping in R, understanding the philosophy of ggplot2, bar plot, pie chart, histogram, boxplot, scatter plot and regression plots

EXERCISE 5: ggplot2 in R

Controlling aesthetics like colour, size, legend and facets.

EXERCISE 6: Data Visualization

Importing the data, Dimensions and measures, color code for various types of variables

EXERCISE 7: Working with sheets

Understanding the worksheet, row and column shelves, show me card, filter and pages shelf

EXERCISE 8: Calculations

Working with different measures, creating new calculated fields, Quick table calculations, parameters and groups

EXERCISE 9: Graphs for Analytics

Calculate Proportions and percentages, Comparing current to historical and Actual to Target

EXERCISE 10: Normal Distribution Variation

Calculate Mean and Median – Normal Distribution Variation and Uncertainty

EXERCISE 11: Reporting and Visualizing Variation

Reporting and Visualizing variation, Control Charts Multiple Quantities – Scatter Plots, Stacked bars, Regressions and trend lines

EXERCISE 12: Depicting Changes Over Time

Depicting changes over time, Line Chart, Dual Axis Line Chart, scatterplot

EXERCISE 13: Reporting

Introduction to dashboard, use of filters in dashboard, Imbedding pictures, Insert live webpage, story

TEXT BOOKS:

1. Microsoft Power BI, Marco Russo
2. R for everyone, Jarad P. Lander
3. Statistics: An Introduction using R, Michael J. Crawley

REFERENCES:

1. Think Python, Allen Downey, Green Tea Press
2. Core Python Programming, W. Chun, Pearson
3. Introduction to Python, Kenneth A. Lambert, Cengage

ONLINE RESOURCES:

1. <https://www.tableau.com/learn>
2. <https://tableauacademy.substack.com/p/tableau-training-and-learning-2021>

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PW4AM201) DESIGN THINKING

TEACHING SCHEME		
L	T/P	C
1	2	2

EVALUATION SCHEME		
CIE	SEE	TOTAL
40	60	100

COURSE OBJECTIVES:

- To instill a sense of significance towards applying creativity to product and service design
- To teach a systematic approach to identifying and defining a problem before brainstorming for a solution
- To inculcate core design principles and applied creativity to develop innovative strategies that better connect engineers and technologies with their end users
- To build a mindset leading to flow of creative ideas, validating those ideas and prioritizing the best ones among them
- To motivate students to apply design thinking while implementing projects focusing on local, regional or global societal problems

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Demonstrate the understanding of design principles from a technology perspective

CO-2: Validate problem statements through user empathisation with societal, cultural, global and Environmental consciousness

CO-3: Use specific and relevant ideation and brainstorming techniques to find innovative solutions

CO-4: Prototype a solution to address user challenges

CO-5: Investigate the cultural, emotional, Environmental, technological and business factors relevant to developing new product or service design concept

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	-	-	2	-	-	2	-	-	3	3	-	-	-	2	1
CO-2	-	-	3	2	1	3	-	-	3	2	-	-	2	1	3
CO-3	-	-	2	-	-	2	-	-	2	2	-	-	-	-	2
CO-4	-	-	3	-	2		-	-	2	3	-	-	-	1	2
CO-5	-	-	-	3	-	3	1	-	1	-	-	-	-	-	2

UNIT-I:

Design Overview and Doing Design: Various perspectives of design; Good and Bad Design; Introduction to the Design Double Diamond: Discover-Define-Develop-Deliver; Discover Phase- Looking for problems; Identifying Stakeholders and Defining User Personas; User Empathization; Data collection, creating and conducting surveys

and Empathy Tools – What/How/Why, Five Why method, Empathy Maps, AEIOU method, Story Share and Capture

UNIT-II:

Need Analysis: Types of Users, Types of Needs; Market Size; Value Proposition to the Users; Identifying Addressable Needs and Touch points; Structuring Need Statements; Customer Experience (CX) Design; Service Design and Development Process; Customer Journey Map (CJM), Service Experience Cycle.

UNIT-III:

Ideation Process: Introduction to creativity and closed-world solutions, Idea generation techniques: Brainstorming, Mind Maps, SCAMPER, Systematic Inventive Thinking methods (Subtraction, Multiplication, Division, Task Unification and Attribute Dependency)

Strategic Innovation for Competition in Future: Linear Innovation vs. Non-linear innovation, Understanding and identifying weak signals, 3-box thinking, 3-Box framework and Box-3 ideation, Four-Action Framework (Eliminate-Reduce-Raise-Create, or ERRC Matrix).

UNIT-IV:

Building Prototypes: Building Conceptual model of product/service using various prototype methods, test a business model or business case to support the viability of the solution using MVP.

Design for Sustainability: Concern for Environment and Sustainability in Design, Case Studies to understand good Design For Environment (DFE) Decisions; Sustainable Design Approaches in the five stages of the Product Life Cycle.

UNIT-V:

Capstone Project (Interdisciplinary): Applying design thinking principles and methods for problem definition, ideation, prototyping, testing, refining and taking the solution to the users, using visual representation tools to indicate problem, User persona, needs, empathisation, ideas and prototype that leads to chosen solution, creating presentation.

TEXT BOOKS:

1. Change by Design, Tim Brown, Harper Business, 2012
2. The Design of Everyday Things, Donald A. Norman, MIT Press, 2013

REFERENCES:

1. The Art of Innovation, Tom Kelly, Jonathan Littman, Harper Collins Business, 2002
2. Design Thinking: Integrating Innovation, Customer Experience, and Brand Value, Thomas Lockwood, Allworth Press, 2009
3. Design Thinking for Start-ups: A Handbook for Readers and Workbook for Practitioners, Jimmy Jain, Notion Press, 2018

ONLINE RESOURCES:

1. <https://www.ideo.com/pages/design-thinking>
2. <https://www.ibm.com/design/thinking/page/framework>
3. https://onlinecourses.nptel.ac.in/noc20_mg38/preview
4. <https://nptel.ac.in/courses/110106124>

5. <https://www.interaction-design.org/literature/article/5-stages-in-the-design-thinking-process>

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22MN6HS201) INTELLECTUAL PROPERTY RIGHTS

TEACHING SCHEME

L	T/P	C
2	0	0

EVALUATION SCHEME

SE-I	SE-II	SEE	TOTAL
50	50	-	100

COURSE OBJECTIVES:

- To familiarize students with the nuances of Intellectual Property Rights (IPR) to help them integrate the IPR process in their research activities
- To make the students capable of identifying their own protectable innovations and realizing the process of taking it from bench to market

COURSE OUTCOMES: After completing this course the student should be able to

CO-1: Get an adequate knowledge on patent and copyright for their innovative research works and academic projects

CO-2: Understand and acquire the knowledge of trademarks and registration aspects

CO-3: Interpret various forms of Intellectual Property on Design, Geographical Indication (GI), Plant Variety and Layout Design Protection and their registration aspects

CO-4: Obtain useful insights from the information in patent documents, especially on novelty of their idea from state-of-the art search, during their research career. This provides further way for developing their idea or innovations

CO-5: Get awareness about current trends in IPR and Govt. steps in fostering IPR

COURSE ARTICULATION MATRIX:

(Correlation of Course Outcomes with Program Outcomes and Program Specific Outcomes using mapping levels 1 = Slight, 2 = Moderate and 3 = Substantial)

CO	PROGRAM OUTCOMES (PO)												PROGRAM SPECIFIC OUTCOMES (PSO)		
	PO-1	PO-2	PO-3	PO-4	PO-5	PO-6	PO-7	PO-8	PO-9	PO-10	PO-11	PO-12	PSO-1	PSO-2	PSO-3
CO-1	-	-	-	-	-	2	-	3	-	-	-	2	-	-	-
CO-2	-	-	-	-	-	2	-	3	-	-	-	2	-	-	-
CO-3	-	-	--	-	-	2	-	3	-	-	-	2	-	-	-
CO-4	-	-	-	-	-	2	-	3	-	-	-	2	-	-	-
CO-5	-	-	-	-	-	2	-	3	-	-	-	2	-	-	-

UNIT – I:

Overview of Intellectual Property: Introduction and the need for Intellectual Property Right (IPR) - Kinds of Intellectual Property Rights: Patent, Copyright, Trade Mark, Design, Geographical Indication, Plant Varieties and Layout Design – Genetic Resources and Traditional Knowledge – Trade Secret - IPR in India : Genesis and development – IPR in abroad - Major International Instruments concerning Intellectual Property Rights: Paris Convention, 1883, the Berne Convention, 1886, the Universal Copyright Convention, 1952, the WIPO Convention, 1967, the Patent Co-operation Treaty, 1970, the TRIPS Agreement, 1994

UNIT – II:

Patents: Patents - Elements of Patentability: Novelty, Non-Obviousness (Inventive Steps), Industrial Application - Non - Patentable Subject Matter - Registration Procedure, Rights and Duties of Patentee, Assignment and licence, Restoration of lapsed Patents, Surrender and Revocation of Patents, Infringement, Remedies & Penalties - Patent office and Appellate Board

UNIT – III:

Copyrights: Nature of Copyright - Subject matter of copyright: original literary, dramatic, musical, artistic works; cinematograph films and sound recordings - Registration Procedure, Term of protection, Ownership of copyright, Assignment and licence of copyright - Infringement, Remedies & Penalties – Related Rights - Distinction between related rights and copyrights

UNIT – IV:

Trademarks:

Concept of Trademarks - Different kinds of marks (brand names, logos, signatures, symbols, well known marks, certification marks and service marks) - Non Registrable Trademarks - Registration of Trademarks - Rights of holder and assignment and licensing of marks - Infringement, Remedies & Penalties - Trademarks registry and appellate board

UNIT – V:

Design: meaning and concept of novel and original - Procedure for registration, effect of registration and term of protection

Geographical Indication (GI): meaning, and difference between GI and trademarks - Procedure for registration, effect of registration and term of protection

Plant Variety Protection: meaning and benefit sharing and farmers' rights – Procedure for registration, effect of registration and term of protection

Layout Design Protection: meaning – Procedure for registration, effect of registration and term of protection

Current Contour: India's New National IP Policy, 2016 – Govt. of India step towards promoting IPR – Govt. Schemes in IPR – Career Opportunities in IP - IPR in current scenario with case studies

TEXTBOOKS:

1. Intellectual Property Rights: Protection and Management Nithyananda, K V, India, IN: Cengage Learning India Private Limited, 2019
2. Intellectual Property Rights, Neeraj, P., & Khusdeep, D, India, IN: PHI learning Private Limited, 2014
3. Intellectual property right, Deborah, E. Bouchoux, 4th Edition, Cengage learning

REFERENCE:

1. Law relating to Intellectual Property Rights, Ahuja, V K, India, IN: Lexis Nexis, 2017

ONLINE RESOURCES:

1. Intellectual Property Rights – An Overview, Subramanian, N., & Sundararaman, M. Retrieved from <http://www.bdu.ac.in/cells/ipr/docs/ipr-eng-ebook.pdf>, 2018
2. WIPO Intellectual property Handbook, World Intellectual Property Organisation, Retrieved from https://www.wipo.int/edocs/pubdocs/en/intproperty/489/wipo_pub_489.pdf, 2004