VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY HYDERABAD
B.TECH. II YEAR
COMPUTER SCIENCE AND BUSINESS SYSTEMS

III SEMESTER **R22**

Course Code	Title of the Course	L	T	P/D	CH	C
22PC1CB201	Software Engineering	3	0	0	3	3
22PC1CB205	Computer Organization and Architecture	3	1	0	4	4
22PC1CB202	Object Oriented Programming	3	0	0	3	3
22PC1CB204	Computational Statistics	3	0	0	3	3
22PC1CB203	Data Base Management Systems	3	0	0	3	3
22PC2CB202	Object Oriented Programming Laboratory	0	0	2	2	1
22PC2CB203	Data Base Management Systems Laboratory	0	0	2	2	1
22PC2CB201	Software Engineering Laboratory	0	0	2	2	1
22PC2CB204	Computational Statistics Laboratory	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
22PC1CB206	Operating Systems	3	0	0	3	3
22PC1CB207	Design and Analysis of Algorithms	3	0	0	3	3
22PC1CB208	Software Design With UML	3	0	0	3	3
22HS1IE201	Introduction to Innovation, IP Management and Entrepreneurship	3	0	0	3	3
22ES1ME210	Operations Research	3	0	0	3	3
22PC2CB206	Operating Systems (Unix)Laboratory	0	0	2	2	1
22PC2CB207	Design and Analysis of Algorithms Laboratory	0	0	2	2	1
22ES2ME210	Operations Research Laboratory	0	0	2	2	1
22HS2EN204	Business Communication and Value Science – III	0	1	2	3	2
22MN6HS202	Indian Constitution	2	0	0	2	0
Total		**17**	**1**	**8**	**26**	**20**

L – Lecture T – Tutorial P – Practical D – Drawing CH – Contact Hours/Week
C – Credits SE – Sessional Examination CA – Class Assessment ELA – Experiential Learning Assessment
SEE – Semester End Examination D-D – Day to Day Evaluation LR – Lab Record
CP – Course Project PE – Practical Examination

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC1CB201) 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 software development activities and process models
- To understand the importance of project planning and management
- 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 and process models that can be tailored with appropriate methods to manage risk

CO-2: Understand the role of software quality and different maturity models

CO-3: Understand the requirements engineering process and the metrics for different models of the system using the perception of UML

CO-4: Understand the oops concepts to analyze the use cases

CO-5: Understand the role of testing in the software development cycle and be capable of developing a test plan to deliver quality software

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	1	1	3	1	2	1	1	3	3
CO-2	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3
CO-3	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3
CO-4	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3
CO-5	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3

UNIT-I:

Introduction: Programming in the small vs. programming in the large; software project failures; Basic concepts of life cycle models – different models, concepts of feasibility study, software cost estimation models and concepts of software engineering economics; introduction to the concepts of risk and its mitigation;

UNIT-II:

Software Quality and Reliability: Internal and external qualities; process and product quality; principles to achieve software quality; introduction to different software quality models like McCall, Boehm, FURPS / FURPS+, Dromey, ISO – 9126; introduction to Capability Maturity Models (CMM and CMMI); introduction to software reliability, reliability models and estimation.

UNIT-III:

Software Requirements Analysis, Design and Construction: Introduction to Software Requirements Specifications (SRS) and requirement elicitation techniques, techniques for requirement modelling – decision tables, event tables, state transition tables, introduction to UML, introduction to software metrics and metrics-based control methods; measures of code and design quality.

UNIT-IV:

Object Oriented Analysis, Design and Construction: Concepts -- the principles of abstraction, modularity, specification, encapsulation and information hiding; concepts of abstract data type; Class Responsibility Collaborator (CRC) model; quality of design; design measurements; concepts of design patterns; Refactoring; object oriented construction principles; object oriented metrics.

UNIT-V:

Software Testing: Introduction to faults and failures Basic testing concepts; concepts of verification and validation; black box and white box tests; white box test coverage – code coverage, condition coverage, branch coverage; basic concepts of black box tests –equivalence classes, boundary value tests; testing use cases; testing for non-functional requirements – volume, performance and efficiency; concepts of inspection.

TEXT BOOKS:

1. Software Engineering, Ian Sommerville, 9th Edition, Pearson Education, 2015
2. Fundamentals of Software Engineering, Carlo Ghezzi, Jazayeri Mehdi, Mandrioli Dino, 2nd Edition, Pearson Education, 2007
3. Software Requirements and Specification: A Lexicon of Practice, Principles and Prejudices, Michael Jackson, Addison-Wesley, 1995

REFERENCES:

1. Software engineering - A practitioner's Approach, Roger S. Pressman, 6th Edition, McGraw-Hill International Edition, 2001
2. The Unified Software Development Process, Ivar Jacobson, Grady Booch, James Rumbaugh, Pearson Education, 2002
3. Design Patterns: Elements of Object-Oriented Reusable Software, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, 1st Edition, Addison-Wesley, 1994
4. Software Metrics: A Rigorous and Practical Approach, Norman E. Fenton, Shari Lawrence Pfleeger, 2nd Edition, Cengage Learning, 1996

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC1CB205) COMPUTER ORGANIZATION AND ARCHITECTURE

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 functional blocks of computer and instruction set architecture
- To describe micro programming and addressing modes of central processing unit
- To perform arithmetic micro operations on integers and floating-point numbers
- To discuss the different ways of communicating with I/O devices
- To design techniques to enhance the performance using pipelining, parallelism

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Understand the basic components and the design of CPU, Instruction set
CO-2: Understand the micro programming, instruction formats and addressing modes
CO-3: Implement arithmetic operations on integer and floating type values
CO-4: Analyze and understand the input organization and memory organization
CO-5: Explore pipeline processing, vector processing and array processing

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	-	-	-	-	-	2	1	2	3
CO-2	2	3	2	3	2	2	-	-	-	-	2	3	1	3	3
CO-3	3	3	-	3	-	-	-	-	-	-	-	3	1	3	3
CO-4	2	3	-	2	3	2	2	-	-	-	-	3	1	2	3
CO-5	2	3	-	2	3	2	2	-	-	-	3	3	1	3	3

UNIT-I:
Functional Blocks of a Computer: Introduction, Block Diagram, Definition of Computer Organization, Computer Design and Computer Architecture
Instruction Set Architecture: Register Transfer language, Register Transfer, Arithmetic Micro operations, logic micro-operations, shift micro-operations, Arithmetic logic shift unit. Instruction codes, Computer Registers Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Register Reference and Input – Output.

UNIT-II:
Central Processing Unit: General Register Organization, Instruction Formats, Addressing modes, Data Transfer and Manipulation
Microprogrammed Control: Control memory, Address sequencing, micro program example, design of control unit.

UNIT-III:

Data Representation: Data types, Complements, Fixed Point Representation, Floating Point Representation.

Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, floating – point Arithmetic operations.

UNIT-IV:

Input-Output Organization: Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Program Controlled, Priority, Interrupt, Direct memory Access. Privileged and Non-Privileged Instructions.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate memory, Mapping Techniques, Replacement Algorithms, Cache Memory, Write Policies, Memory interleaving.

UNIT-V:

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor.

TEXT BOOKS:

1. Computer System Architecture, M. Morris Mano, 3rd Edition, Pearson Education, 2007
2. Computer Organization and Embedded Systems, Carl Hamacher, 6th Edition, McGraw-Hill

REFERENCES:

1. Computer Organization and Design: The Hardware/Software Interfaces, David A. Patterson and John L. Hennessy, 5th Edition, Elsevier
2. Computer Architecture and Organization, John P. Hayes, 3rd Edition, WCB/McGraw-Hill
3. Computer Organization and Architecture: Designing for Performance, William Stallings, 10th Edition, Pearson Education
4. Computer System Design and Architecture, Vincent P. Heuring and Harry F. Jordan, 2nd Edition, Pearson Education

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC1CB202) OBJECT ORIENTED PROGRAMMING

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 annotations, java database connectivity architecture

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

CO-1: Solve real-world problems using OOP techniques

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 common applications by annotations; generics for better performance use multithreading

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

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:

Introduction to Object Oriented Programming: Introduction to object-oriented programming concepts (Encapsulation, abstraction, Inheritance, polymorphism, etc.), Difference between JDK, JRE, and JVM, Introduction to Classes, Objects, Class Diagram and Object Diagram, Data Types, Variables, Type Conversion and Casting, Operators, Control Statements: Decision Making, Looping and Branching, Argument Passing Mechanism, Command Line Arguments, Constructors and types, Arrays and types.

UNIT-II:

Inheritance, Packages, Interfaces and Other Topics: Understanding static, Use of "this" keyword, Garbage Collection and finalize () Method, Introduction and Working with Lambda Variables.

Inheritance: Basics, Using super, creating a multi level hierarchy, Constructors in Inheritance, final with Inheritance,

Polymorphism: Introduction and Types, Overloading and Overriding, Dynamic method Dispatch

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

String Handling: Introduction and Types, Operations, Immutable String, Method of String class, String Buffer and String Builder class

UNIT-III:

Exception Handling: Exception handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch clauses, Nested try statements, throw, throws, finally, Java Built-in Exceptions, Creating your own exception subclasses, propagating exceptions Chained Exceptions, Assertions and Localizations Concepts and its working.

I/O Streams: The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and Writing Files, Random access file operations, The Console class, Serialization, Enumerations, Type Wrappers.

UNIT-IV:

Multithreaded Programming, Annotations: The java Thread Model, The main thread, Creating Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization, Interthread Communication, Suspending, resuming and stopping threads, Obtaining a thread state, Using Multithreading.

Annotations: Introduction, Custom Annotations and Applying Annotations.

UNIT-V:

Generics and Collections Framework (Java.Util): Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Hashtable, Properties, Stack, Vector, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter, Scanner, Working with generic class, interfaces.

Database programming: Types of Drivers, JDBC Architecture, Basic steps in developing JDBC applications, JDBC classes and Interfaces, Creating a new database and table 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

(22PC1CB204) COMPUTATIONAL STATISTICS

TEACHING SCHEME		
L	T/P	C
3	0	3

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

COURSE OBJECTIVES:

- To learn multidimensional generalization of a univariate normal random variable
- To learn concept of multivariable linear regression model
- To learn linear discriminant function analysis
- To learn data summarization and data reduction using factor analysis
- To learn grouping the data using cluster analysis

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

CO-1: Model the linear relationship between the explanatory (independent) Variables and response(dependent)variables

CO-2: Make better decisions using linear regression techniques

CO-3: Develop discriminant functions that will discriminate between the categories of the dependent variable in a perfect manner

CO-4: Reduce a large number of variables into fewer numbers of factors using factor analysis

CO-5: Gain the knowledge on different types of clustering

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	-	-
CO-2	3	3	2	1	-	-	-	-	-	-	-	-	2	-	-
CO-3	2	2	1	-	-	-	-	-	-	-	-	-	1	-	-
CO-4	2	2	1	1	-	-	-	-	-	-	-	-	2	1	-
CO-5	2	2	2	1	-	-	-	-	-	-	-	-	2	1	-

UNIT–I:

Multivariate Normal Distribution & Linear Regression Model: Multivariate Normal Distribution Functions, Conditional Distribution and its relation to regression model, Standard multiple regression models with emphasis on detection of collinearity, outliers.

UNIT–II:

Multivariate Regression: Assumptions of Multivariate Regression Models, Parameter estimation, Multivariate Analysis of variance and covariance

UNIT–III:

Discriminant Analysis: Statistical background, linear discriminant function analysis, Estimating linear discriminant functions and their properties.

UNIT–IV:

Factor Analysis: Factor analysis model, Extracting common factors, determining number of factors, Transformation of factor analysis solutions, Factor scores.

UNIT–V:

Cluster Analysis: Introduction, Types of clustering, Correlations and distances, hierarchical clustering, overlapping clustering, clustering by partitioning methods (K-Means clustering).

TEXT BOOKS:

1. An Introduction to Multivariate Statistical Analysis, T. W. Anderson, 3rd Edition, Wiley, 2009
2. Applied Multivariate Data Analysis, Vol. II, J. D. Jobson, 2nd Reprint, Springer, 1994
3. Statistical Tests for Multivariate Analysis, H. Kris

REFERENCES:

1. Regression Diagnostics, Identifying Influential Data and Sources of Collinearity, D. A. Belsey, E. Kuh and R. E . Welsch, Wiley-Interscience, 2013
2. Applied Linear Regression Models, J. Neter, W. Wasserman and M. H. Kutner, 4th Edition, McGraw-Hill, 2004
3. Python for Data Analysis, Wes Mc Kinney, 3rd Edition, O'Reilly Media, 2022
4. Introduction to Linear Regression Analysis, D. C. Montgomery and E. A. Peck, 3rd Edition, Wily, 2006

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

((22PC1CB203) DATATBASE MANGEMENT SYSTEM CONCEPTS

TEACHING SCHEME		
L	T/P	C
3	0	3

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

COURSE OBJECTIVES:

- To introduction of data base management systems concepts and to give the description of structure of data base systems
- To know the features of various models of data and query representations
- To prepare the database through normalization, query optimization and understand the concepts of data storage
- To introduce the concepts and protocols related to transaction management, database recovery and security

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

CO-1: Appreciate the underlying concepts of database system architecture and technologies

CO-2: Query the database using the relevant programming language

CO-3: Develop database schema for a given scenario and implement optimization techniques

CO-4: Design schedules using multiple transactions, implement data base recovery and security

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	-	1	-	-	3	-	-	3	-	-	2	-
CO-2	1	1	1	1	1	-	3	3	2	3	3	2	2	2	2
CO-3	1	1	1	1	1	3	3	3	2	2	2	1	1	1	1
CO-4	1	1	1	1	1	3	3	3	2	2	2	1	2	1	1
CO-5	3	1	3	-	1	-	-	3	-	-	3	-	-	2	-

UNIT – I:

Introduction: Introduction to Database. Hierarchical, Network and Relational Models. Database system architecture: Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). Data models: Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

UNIT – II:

Relational Query Languages: Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server.

UNIT – III:

Relational Database Design: Domain and data dependency, Armstrong's axioms, Functional Dependencies, Normal forms, Dependency preservation, Lossless design. Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms. Storage strategies: Indices, B-trees, Hashing.

UNIT – IV:

Transaction Processing: Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.

UNIT – V:

Database Security: Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection. Advanced topics: Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.

TEXT BOOKS:

1. Database System Concepts. Abraham Silberschatz, Henry F. Korth and S. Sudarshan, 6th Edition, McGraw-Hill, 2010
2. An Introduction to Database Systems, C. J. Date, 8th Edition, Pearson Education
3. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition, McGraw-Hill

REFERENCES:

1. Principles of Database and Knowledge – Base Systems, Vol. 1, J. D. Ullman
2. Fundamentals of Database Systems. R. Elmasri and S. Navathe, 7th Edition, Pearson Education, 2017
3. Foundations of Databases, Serge Abiteboul, Richard Hull, Victor Vianu, Pearson Education, 1994
4. Introduction to Database Management Systems, Kahate, 1st Edition, Pearson Education, 2004
5. An Introduction to Database Systems, Bipin Desai, Galgotia Publication, 2010

ONLINE RESOURCES:

1. https://www.digimat.in/nptel/courses/video/106105175/L01.html
2. https://onlinecourses.nptel.ac.in/noc21_cs04/preview

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC2CB202) OBJECT ORIENTED PROGRAMMING 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 ax2+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 tile exists, whether the file is readable, or writable. The type of tile 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.

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. Chudhary, 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

(22PC2CB203) DATATBASE MANGEMENT SYSTEM 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 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: Implement SQL functions using the DUAL table
CO-2: Apply integrity constraints for creating consistent RDBMS environment
CO-3: Create, maintain and manipulate the data through SQL commands
CO-4: Develop triggers, query through PL /SQL structures

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	2	-	-	-	1	2	3	3	-	2	-	1	3	-	3
CO-3	-	1	1	1	1	2	-	3	3	2	3	1	2	2	3
CO-4	-	2	1	1	1	2	-	3	2	2	3	1	1	1	2
CO-5	2	1	1	1	1	2	-	2	2	2	3	1	1	1	2

WEEK 1: Implement the following using DUAL table:
a) Character functions.
b) Numeric functions.
c) Date functions.
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 Scale of salary
Losal	Integer	5		Lower Scale of salary

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 remodelled 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: implementing functions and procedures.

WEEK 14: Creating and executing Cursors, sequences and clusters.

WEEK 15: Creation and application of TRIGGERS on a Relational schema.

WEEK 16: implementation of locking and partitioning

TEXT BOOKS:

1. Database System Concepts. Abraham Silberschatz, Henry F. Korth and S. Sudarshan, 6th Edition, McGraw-Hill, 2010
2. An Introduction to Database Systems, C. J. Date, 8th Edition, Pearson Education
3. Database Management Systems, Raghu Rama Krishnan and Johannes Gehrke, 3rd Edition, McGraw-Hill

REFERENCES:

1. Principles of Database and Knowledge – Base Systems, Vol. 1, J. D. Ullman
2. Fundamentals of Database Systems. R. Elmasri and S. Navathe, 7th Edition, Pearson Education, 2017
3. Foundations of Databases, Serge Abiteboul, Richard Hull, Victor Vianu, Pearson Education, 1994
4. Introduction to Database Management Systems, Kahate, 1st Edition, Pearson Education, 2004
5. An Introduction to Database Systems, Bipin Desai, Galgotia Publication, 2010

ONLINE RESOURCES:

1. https://www.digimat.in/nptel/courses/video/106105175/L01.html
2. https://onlinecourses.nptel.ac.in/noc21_cs04/preview

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC2CB201) 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 understand the role of formal specifications in project design and be able to develop such specifications
- To be able to design an interface and develop a prototype for a complex software system
- To understand the role of testing in the software development cycle and be capable of developing a test plan
- To be aware of and able to use Computer Aided Software Engineering (CASE) tool

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Develop requirement specifications for a software problem in hand
CO-2: Perform functional oriented and object oriented design
CO-3: Implement the concepts of object oriented to develop a real world application
CO-4: Prepare test cases to rigorously test the application for ensuring quality

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	1	1	3	1	2	1	1	3	3
CO-2	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3
CO-3	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3
CO-4	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3

LIST OF PROGRAMS / EXPERIMENTS / EXERCISES:

WEEK 1 & 2:
Phases in software development project, overview and need. Understand problems in existing systems and perform system analysis: Requirement analysis, SRS

WEEK 3:
To perform the function oriented design: Data flow diagrams and Structured chart

WEEK 4:
To perform the user's view analysis: Use case diagram

WEEK 5:
To draw the structural view diagram: Class diagram, object diagram.

WEEK 6:
To draw the behavioral view diagram: Sequence diagram, Collaboration diagram

WEEK 7:
To draw the behavioral view diagram: State-chart diagram, Activity diagram

WEEK 8:
To draw the implementation view diagram: Component diagram, deployment diagram.

WEEK 9:
Implement any one design pattern for a relevant context

WEEK 10:
To perform various techniques for testing using the testing tool: unit testing, integration.

WEEK 11 & 12:
Use a Software Configuration Management tool for tracking and controlling changes in the software.

WEEK 13:
Identify and refactor the code snippets in a given code

WEEK 14:
Draw all UML diagrams diagrams for ATM simulation

TEXT BOOKS:

1. Software Engineering, Ian Sommerville, 10th Edition, Pearson Education, 2017
2. Fundamentals of Software Engineering, Carlo Ghezzi, Jazayeri Mehdi, Mandrioli Dino, 2nd Edition, Pearson Education, 2007
3. Software Requirements and Specification: A Lexicon of Practice, Principles and Prejudices, Michael Jackson, Addison-Wesley, 1995

REFERENCES:

1. The Unified Software Development Process, Ivar Jacobson, Grady Booch, James Rumbaugh, Pearson Education, 2002
2. Design Patterns: Elements of Object-Oriented Reusable Software, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, 1st Edition, Addison-Wesley, 1994
3. Software Metrics: A Rigorous and Practical Approach, Norman E. Fenton, Shari Lawrence Pfleeger, 2nd Edition, Cengage Learning, 1996

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester

(22PC2CB204) COMPUTATIONAL STATISTICS 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 learn object oriented concepts in Python
- To reading and writing data from different sources
- To learn the available libraries in Python
- To relate statistical methods to real time applications

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Able to understand various data structures in Python
CO-2: Apply predefined libraries for various applications
CO-3: Analyze the results by applying various statistical methods
CO-4: Plot graphs for the given data sets

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	1	-	1	1	-	-	-	-	-	1	1	2	-	-
CO-2	1	1	-	1	1	-	-	-	-	-	1	1	2	-	-
CO-3	1	1	1	1	1	-	-	-	-	-	1	1	2	-	-
CO-4	1	1	-	1	1	-	-	-	-	-	1	1	2	-	-
CO-5	1	1	1	1	1	-	-	-	-	-	1	1	2	-	-

WEEK 1, 2, 3, 4:
Python Concepts, Data Structures, Classes: Interpreter, Program Execution, Statements, Expressions, Flow Controls, Functions, Numeric Types, Sequences, lists, dictionaries, Text & Binary Files - Reading and Writing

WEEK 5, 6, 7, 8, 9:
Visualization in Python: Matplotlib package, Plotting Graphs, Controlling Graph, Adding Text, More Graph Types, Getting and setting values

WEEK 10, 11, 12:
Multivariate data analysis: Multiple regression, multi variate regression, cluster analysis with various algorithms, factor analysis, PCA and linear discriminant analysis. Various datasets should be used for each topic

TEXT BOOKS:

1. An Introduction to Multivariate Statistical Analysis, T. W. Anderson, 3rd Edition, Wily, 2009
2. Applied Multivariate Data Analysis, Vol I & II, J. D. Jobson, 2nd Reprint, Springer, 1994
3. Statistical Tests for Multivariate Analysis, H. Kris

REFERENCES:

1. Regression Diagnostics, Identifying Influential Data and Sources of Collinearity, D. A. Belsey, E. Kuh and R. E. Welsch, Wiley-Interscience, 2013
2. Introduction to Linear Regression Analysis, D. C. Montgomery and E. A. Peck, 3rd Edition, Wily, 2006
3. Programming Python, Mark Lutz, 4th Edition, O'Reilly Media, 2011
4. Python 3 for Absolute Beginners, Tim Hall and J. P. Stacey, Apress, 2009
5. Python for Data Analysis, Wes McKinney, 3rd Edition, O'Reilly Media, 2022

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 Fulfilment, 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

(22PC1CB206) OPERATING SYSTEMS

TEACHING SCHEME		
L	T/P	C
3	0	3

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

COURSE PREREQUISITES: Basics of Computer Organization

COURSE OBJECTIVES:
- To understand operating system overview, functions and services
- To analyze process concept, inter process communication and synchronization
- To understand deadlock and memory management
- To understand file system interface and mass storage management
- To understand protection, security and i/o control

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Understand operating system overview, functions and services
CO-2: Analyze process concept, inter process communication and synchronization
CO-3: Understand deadlock and memory management
CO-4: Understand file system interface and mass storage management
CO-5: Understand protection, security and i/o control

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	-	1	-	-	-	-	-	1	-	1	3	1	-
CO-2	3	1	-	1	-	-	-	-	-	1	-	1	3	1	-
CO-3	3	1	-	1	-	-	-	-	-	1	-	1	3	1	-
CO-4	3	1	-	1	-	-	-	-	-	1	-	1	3	1	-
CO-5	3	1	-	1	-	-	-	-	-	1	-	1	3	1	-

UNIT-I:
Operating System Overview: Overview of Computer Operating Systems, Operating System Functions, Operating System Services and Systems Calls.
Process Concept: Process Concepts, process states.Threads, multithreading.
UNix case study: Process API, Thread API in Unix

UNIT-II:
Process Management: Scheduling-Criteria, CPU Scheduling, Algorithms Evaluation, Thread Scheduling.
Inter Process Communication: message passing, shared memory,
Unix Case study: IPC in unix-message queue, shared memory, semaphore and sockets.

UNIT-III:

Process Synchronization: Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization examples, Atomic Transactions.

Deadlock System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery from Deadlock.

UNIT-IV:

Memory Management: Swapping, Contiguous Memory Allocation, Paging, Page-Table Structure, Segmentation, Virtual Memory, Demand Paging, Page-Replacement Algorithms, Frames Allocation, Thrashing.

File System Interface: File Concepts, Access Methods and Directory Structure, File System Mounting, File Sharing and Protection, Allocation Methods, Free-Space Management, Efficiency and Performance

Unix Case Study: File API,inodes

UNIT-V:

Mass Storage Overview Mass: Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, Swap-Space Management, RAID Structure

Protection Goals of Protection, Principles of Protection, Domain of Protection Access Matrix, Implementation of Access Matrix,

Security Security Problem, Program Threats, System and Network Threats

I/O Hardware: I/O devices, Device controllers, Direct Memory Access

TEXT BOOKS:

1. Operating System Concepts, Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 7th Edition, John Wiley
2. Operating Systems-A Concept based Approach, D. M. Dhamdhere, 2nd Edition, Tata McGraw-Hill

REFERENCES:

1. UNIX: Concepts and Applications, Sumitabha Das, 4th Edition, Tata McGraw-Hill
2. Operating System: A Design-Oriented Approach, Charles Patrick Crowley, McGraw-Hill, 2017
3. Operating Systems: A Modern Perspective, Gary J. Nutt, Pearson Education,1997
4. Design of the Unix Operating Systems, Maurice J. Bach

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PC1CB207) DESIGN AND ANALYSIS OF ALGORITHMS

TEACHING SCHEME		
L	T/P	C
3	0	3

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

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

COURSE OBJECTIVES:
- To learn reinforce algorithms analysis methods
- To analyze 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: Calculate complexity measure and analyse the time complexity
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

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	-	-	1	1	1	1	-	1	3	2	-
CO-2	3	2	3	3	2	2	1	2	3	2	2	2	3	2	2
CO-3	2	3	-	2	-	-	1	1	1	1	-	1	3	2	-
CO-4	3	3	3	3	2	2	1	2	2	2	2	2	3	2	-
CO-5	3	3	3	3	-	1	1	2	2	2	2	2	3	2	-

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.

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, Approximation Algorithms, Randomized Algorithms, Introduction to Quantum Algorithms.

TEXT BOOKS:

1. Fundamental of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, 2nd Edition, Galgotia Publication
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 Eva Tardos, 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. IV Semester

(22PC1CB208) SOFTWARE DESIGN WITH UML

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 how and why models are used during software development and maintenance
- To identify the different perspectives from which software can be modeled
- To distinguish between static and dynamic models and identify the view or perspective provided by each type of model described
- To be able to create and interpret examples of each type of model described here

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 process models for developing the projects
CO-2: Model and analyse software requirements (use case modeling) using UML
CO-3: Model and analyse software requirements (Interaction Modeling) using UML
CO-4: Apply appropriate UML design to model static structure diagrams
CO-5: Understand the role of package diagrams and dynamic models

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	1	1	3	1	2	1	1	3	3
CO-2	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3
CO-3	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3
CO-4	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3
CO-5	3	3	3	2	3	1	1	1	3	1	2	1	1	3	3

UNIT-I:

Software Development Process Models: The Waterfall Model vs. The Spiral Model, Agile Model, The Software Crisis, Classes, inheritance, Quality software characteristics, Description of the Object Oriented Analysis process vs. the Structure Analysis Model.

UNIT-II:

Introduction to UML Language: Standards, Elements of the language, General description of various models, The process of Object-Oriented software development, Description of Design Patterns, Technological Description of Distributed Systems. Use Case Diagrams, Use Case Relationships.

UNIT-III:

Transfer from Analysis to Design in the Characterization Stage: Interaction Diagrams: Defining UML Method, Operation, Sequence Diagram, finding objects from Flow of Events, Describing the process of finding objects using a Sequence Diagram, Describing the process of finding objects using a Collaboration Diagram.

UNIT-IV:

The Logical View Design Stage: The Static Structure Diagrams: The Class Diagram Model, Attributes descriptions, Operations descriptions, Connections descriptions in the Static Model, Association, Generalization, Aggregation, Dependency, Multiplicity.

UNIT-V:

Package Diagram Model: Description of the model, Connections between packages, interfaces, Create Package Diagram, Drill Down. Dynamic Model: State Diagram / Activity Diagram: Description of the State Diagram, Events Handling, Description of the Activity Diagram.

TEXT BOOK:

1. Object-Oriented Software Engineering: Using UML, Patterns, and Java, Bernd Bruegge and Allen H. Dutoit

REFERENCE:

1. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, and John M. Vlissides

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22HS1IE201) INTRODUCTION TO INNOVATION, IP MANAGEMENT AND ENTREPRENEURSHIP

TEACHING SCHEME		
L	T/P	C
2	1	3

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

COURSE OBJECTIVES:

- To understand the context of innovation in the light of sustainable development and Industry 4.0
- To acquire fundamental knowledge of innovation, intellectual property management, and entrepreneurship
- To acquaint themselves with challenges in starting new ventures and the need for innovation in business
- To prepare business plans with understanding of innovation, entrepreneurship, IP, finance, and marketing use IPR to protect innovations and intangible assets from exploitation

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

CO-1: Innovate in the backdrop of sustainable development and Industry 4.0 and understand innovation in business and technology

CO-2: Consider innovation and act on it in building and managing a startup

CO-3: Understand entrepreneurship and the role of innovation, IP, and management, and market research

CO-4: Learn the bases for developing a business plan and financial plan for a startup

CO-5: Identify various types of intellectual property and associated IP rights as well as IPR management, in the context of innovation and entrepreneurship

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	-	2	1	1	-	-	-	-	-	1	3	2
CO-2	1	1	1	-	1	1	1	-	-	-	-	-	3	-	-
CO-3	-	-	-	-	-	1	2	-	-	-	-	-	3	3	2
CO-4	1	1	1	-	1	1	1	-	-	-	-	-	1	2	1
CO-5	1	1	1	-	1	1	1	-	-	-	-	-	1	2	1

UNIT-I:

Innovation Overview: Industry 4.0, Sustainable Development Goals, and Need for Innovation, Innovation in the context of Business, Engineering, and Computer Science; Sources of innovation, Knowledge push vs. Need pull innovations; Radical Innovation, Incremental Innovation, Disruptive Innovation, Open Innovation, Jugaad,

Collaborating for Innovation; Three lenses of innovation – Desirability, Feasibility, and Viability

UNIT-II:

Building an Innovative Organization: Innovation in the context of entrepreneurship and intellectual property, Innovation as a core business process, creating new products and services, Ideation, Use of innovation for starting a new venture

UNIT-III:

Entrepreneurship: Opportunity recognition and entry strategies, Understanding the market through research and segmentation, Entrepreneurship as a style of management, Competitive Advantage, Role of IP, Scalability

UNIT-IV:

Financial Planning: Financial Projections and Valuation, Stages of Financing, Debt, Venture Capital, Other forms of financing, Introduction to various keywords pertaining to financial management and investment

UNIT-V:

Intellectual Property: Introduction, Economics behind development of IPR, Business and Marketing Perspectives of IP and IPR, IPR in the context of India and the world, IP Management Definitions, Application Process, Protection and Penalty for Infringement for different types of IPR: Patents, Trademarks, Geographical Indications, Copyrights, Industrial Designs

Suggested Topics for Class Discussion: Major Court bottles regarding violation of patents between corporate companies Is innovation manageable or just a random gambling activity?

Innovation: Co-operating across networks vs. 'go-it-alone' approach

TEXTBOOKS:

1. Managing Innovation: Integrating Technological, Market and Organizational Change, Joe Tidd, John Bessant

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22ES1ME210) OPERATIONS RESEARCH

TEACHING SCHEME		
L	T/P	C
3	0	3

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

COURSE OBJECTIVES:

- To introduce to use quantitative methods and techniques for effective decisions–making
- To develop operational research models from the verbal description of the real system
- To model formulation and applications that are used in solving business decision problems
- To understand the mathematical tools that are needed to solve optimization problems

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Evaluate the problems using linear programming
CO-2: Build and solve transportation models and assignment models
CO-3: Design new simple models, like CPM, PERT to improve decision – making and develop critical thinking and objective analysis of decision problems
CO-4: Build and solve the problems using Inventory, queuing theory and simulation problems

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	3	1	-	3	1	2	1	3	3	1
CO-2	3	3	3	3	3	3	1	-	3	1	2	1	3	3	1
CO-3	3	3	3	3	3	3	1	-	3	1	2	1	3	3	1
CO-4	3	3	3	3	3	3	1	-	3	1	2	1	3	3	1

UNIT – I:

Introduction to OR: Origin of OR and its definition. Concept of optimizing performance measure, Types of OR problems, Deterministic vs. Stochastic optimization, Phases of OR problem approach – problem formulation, building mathematical model, deriving solutions, validating model, controlling and implementing solution.

UNIT – II:

Linear Programming: Linear programming–Examples from industrial cases, formulation.
Geometric Method: 2-variable case, Special cases – infeasibility, unboundedness, redundancy &degeneracy.
Simplex Algorithm: slack, surplus & artificial variables, computational details, big-M method, identification and resolution of special cases through simplex iterations.

Duality – formulation, results, fundamental theorem of duality, dual-simplex and primal-dual algorithms.

UNIT – III:

Transportation and Assignment Problems: TP - Examples, Definitions – decision variables, supply & demand constraints, formulation, Balanced & unbalanced situations, Solution methods – NWCR, minimum cost and VAM, test for optimality (MODI method), degeneracy and its resolution. AP - Examples, Definitions – decision variables, constraints, formulation, Balanced &unbalanced situations, Solution method – Hungarian, test for optimality (MODI method).

UNIT – IV:

PERT – CPM: Project definition, Project scheduling techniques – Gantt chart, PERT & CPM, Determination of critical paths, Estimation of Project time and its variance in PERT using statistical principles, Concept of project crashing/time-cost trade-off.

Inventory Control: Functions of inventory and its disadvantages, ABC analysis, Concept of inventory costs, Basics of inventory policy (order, lead time, types), Fixed order-quantity models – EOQ, POQ & Quantity discount models.

UNIT – V:

Queuing Theory: Definitions – queue (waiting line), waiting costs, characteristics (arrival, queue, service discipline) of queuing system, queue types (channel vs. phase). Kendall's notation, Little's law, steady state behaviour, Poisson's Process & queue, Models with examples - M/M/1 and its performance measures; M/M/m and its performance measures; brief description about some special models.

Simulation Methodology: Definition and steps of simulation, random number, random number generator, Discrete Event System Simulation – clock, event list, Application in Scheduling, Queuing systems and Inventory systems.

TEXT BOOKS:

1. Operations Research, J. K. Sharma, McMillan
2. Operations Research, R. Pannerselvam, Prentice Hall International

REFERENCES:

1. Operations Research, A. M. Natarajan, P. Balasubramani, A. Tamilarasi, Pearson Education
2. Operations Research: Methods and Problems, Maurice Saseini, Arthur Yaspan and Lawrence Friedman
3. Introduction to OR, Taha, Prentice Hall International

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PC2CB206) OPERATING 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 OBJECTIVES:

- To understand the basic system calls related to process
- To implement various CPU scheduling algorithms
- To implement memory management and virtual memory techniques
- To implement file management and disk scheduling algorithms

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Know the basic system calls related to process
CO-2: Simulate CPU scheduling algorithms
CO-3: Understand memory management and virtual memory techniques
CO-4: Analyze the various disk scheduling algorithms
CO-5: Understand file management 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	2	1	1	1	-	-	-	3	1	1	-	3	-	1
CO-2	3	2	1	1	1	-	-	-	3	1	1	-	3	-	1
CO-3	3	2	1	1	1	-	-	-	3	1	1	-	3	-	1
CO-4	3	2	1	1	1	-	-	-	3	1	1	-	3	-	1
CO-5	3	2	1	1	1	-	-	-	3	1	1	-	3	-	1

LIST OF PROGRAMS / EXPERIMENTS / EXERCISES:

1. Write programs using the following system calls of UNIX operating system fork(), exec(), getpid(), exit(), wait(), close().
2. Simulate the following CPU scheduling algorithms
   a) FCFS b) SJF c) Priority d) Round robin
3. Implement the classic problem of synchronization using semaphore and monitor
4. Implement the Following IPC Mechanisms
   a) Pipes b) shared Memory c) Message queues
5. Implement the Client server communication using Internet Domain Sockets
6. Simulate Bankers Algorithm for Deadlock Avoidance and Deadlock Detection Algorithm
7. Implementation of the following Memory Allocation Methods for fixed partition
   a) First Fit b) Worst Fit c) Best Fit
8. Implementation of Paging Technique of Memory Management

9. Implementation of the following Page Replacement Algorithms
   a) FIFO b) optimal c) LRU d) LFU
10. Simulate frame allocation Methods.
   a) minimum Number of frames b) equal allocation c) proportional allocation
11. Simulate the following File Organization Techniques
   a) Single level directory b) Two level Directory
12. Simulate Disk scheduling algorithms
   a) FCFS b) SSTF c) SCAN d) C-SCAN e) LOOK.
13. Simulate all file allocation strategies.
   a) Sequential b) Indexed c) Linked.

TEXT BOOKS:

1. UNIX: Concepts and Applications ,4th Edition, Sumitabha Das, Tate McGraw-Hill
2. Let us C, Yeshvant Kanetkar, 16th Edition, BPB Publications, 2016

REFERENCES:

1. Operating Systems: A Modern Perspective, Gary J. Nutt
2. Operating System Concepts-Abraham Silberchatz, Peter B. Galvin, Greg Gagne, 7th Edition, John Wiley

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22PC2CB207) DESIGN AND ANALYSIS OF ALGORITHMS 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 analyze the asymptotic performance of algorithms
- To demonstrate a relationship between major algorithms and data structures
- To apply important algorithmic design paradigms for solving computational problems
- To implement problems using different algorithm design paradigms and analyze their time complexity

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

CO-1: Analyse running time complexity of non-recursive and recursive algorithm using asymptotic analysis

CO-2: Implement the problems using divide and conquer, greedy methods and analyse their time complexity

CO-3: Implement the problems using dynamic programming methods and analyse their time complexity

CO-4: Implement the problems using backtracking methods and analyse their time complexity

CO-5: Implement the graph problems and analyse time their time complexity

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	-	-	1	1	1	1	-	1	3	2	-
CO-2	3	2	3	3	2	2	1	2	3	2	1	2	3	3	-
CO-3	3	2	3	3	2	2	1	2	3	2	1	2	3	3	-
CO-4	3	2	3	3	2	2	1	2	3	2	1	2	3	3	-
CO-5	3	2	3	3	2	2	1	2	3	2	1	2	3	3	-

LIST OF PROGRAMS / EXPERIMENTS / EXERCISES:

WEEK 1:

- Implement and analyze time complexity of Binary Search (Recursive & Non-Recursive), Merge and Quick Sort

WEEK 2:

- Implement and analyze time complexity in best & worst case for Stassen Matrix Multiplication

WEEK 3:
- Implement and analyze time complexity of Greedy Application Problems: Knapsack, Job Secluding, Huffman Tree generation

WEEK 4:
- Implement and analyze time complexity of Greedy Application Problems: Prims and Kruskals.

WEEK 5:
- Implement and analyze time complexity of Dynamic Programming Application Problems: Matrix chain multiplication, Optimal binary search trees, Travelling Salesperson (TSP)

WEEK 6:
- Implement and analyze time complexity of Backtracking Application Problems: Multistage Graph, TSP.

WEEK 7:
- Implement and analyze time complexity of Backtracking Application Problems: N-Queens, Sum of Subsets, Graph Coloring, Hamiltonian Cycles.

WEEK 8:
- Implement Shortest path algorithm and Topological Sorting.

WEEK 9:
- Implement and analyze time complexity of Dijkstra and Floyd Warshall Algorithms

WEEK 10:
- Implement and analyze time complexity Randomized Quick Sort and Ford Fulkerson Algorithm.

WEEK 11 (Case Study):
- Develop an algorithm to generate Fibonacci numbers with improved time complexity compared to the naive recursive approach

WEEK 12 (Case Study):
- Create a problem that solves popular puzzle like SUDOKO, TOWERS OF HANOII using efficient algorithms.

WEEK 13:
- Internal lab Exam

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 Eva Tardos, 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

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22ES2ME210) OPERATIONS RESEARCH 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 the concept of graphical analysis of LPP
- To identify a transportation problem and implement various transportation models
- To derive the solution of an assignment problem and integer programming problems using various methods
- To understand classification of queuing models (finite and infinite models) and the simulation models

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

CO-1: Give solution of any LPP by using simplex algorithm

CO-2: Implement feasible, basic, non-degenerate solutions of a transportation problem

CO-3: Implement assignment problem that will be balanced and also solve integer Programming problem

CO-4: Implement queuing theory techniques and simulation models.

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	3	1	-	3	1	2	1	3	3	1
CO-2	3	3	3	3	3	3	1	-	3	1	2	1	3	3	1
CO-3	3	3	3	3	3	3	1	-	3	1	2	1	3	3	1
CO-4	3	3	3	3	3	3	1	-	3	1	2	1	3	3	1

WEEK 1, 2, 3:

Experiment 1: To solve Linear Programming Problem using Graphical Method with

I. Unbounded solution

II. Infeasible solution

III. Alternative or multiple solutions.

WEEK 4 ,5, 6:

Experiment 2: Solution of LPP with simplex methods.

WEEK 7, 8 ,9:

Experiment 3: Solving transportation problems.

WEEK 10, 11,12:

Experiment 4: solving assignment problems.

WEEK 13, 14, 15, 16:
Experiment 5: Solving queuing and simulation models

TEXT BOOKS:

1. Operations Research: An Introduction, H. A. Taha, 10th Edition, Pearson Education, 2019
2. Principles of OR with Application to Managerial Decisions, H. M. Wagner, 2nd Edition, Prentice Hall of India, 1980
3. Introduction to Operations Research, F. S. Hiller and G. J. Lieberman, 10th Edition, McGraw-Hill, 2017

REFERENCES:

1. Linear Programming, K. G. Murthy, 1st Edition, Wiley, 1983
2. Elements of Queuing Theory, Thomas L. Saaty, Dover Publications, 1983
3. Operations Research and Management Science, Hand Book: Edited By A. Ravi Ravindran, 1st Edition, CRC Press, 2007
4. Management Guide to PERT/CPM, Wiest & Levy, Prentice-Hall, 1970
5. Modern Inventory Management, J. W. Prichard and R. H. Eagle, John Wiley, 1965

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22HS2EN204) BUSINESS COMMUNICATION AND VALUE SCIENCE -III

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-REQUISITE(S):
- Basic Knowledge of English (verbal and written)

COURSE OBJECTIVES:
- To create clear, accurate, and succinct content to write business letters, resume, sop, proposals and technical reports for academics as well as for workplace
- To introduce self-analysis techniques like swot & tows
- To introduce to key concepts of pluralism, cultural spaces and cross-cultural communication
- To understand the importance of science for nation building

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Identify the best practices of technical writing and understand, apply &analyze the tools of technical writing
CO-2: Apply & analyze the basic principles of SWOT & life positions
CO-3: Identify & respect pluralism in cultural spaces- Analyze cross cultural communication and apply the concepts of Global, and translocational and Recognize the roles and relations of different genders
CO-4: Apply the science of Nation building
CO-5: Apply the basic communication practices

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	1	1	1	2	2	2	2	3	2	2	-	1	-
CO-2	1	1	1	1	1	2	2	2	2	3	3	3	-	1	-
CO-3	-	-	1	1	1	2	2	2	1	3	2	3	-	1	-
CO-4	-	1	1	1	-	2	2	2	2	2	2	3	-	-	-
CO-5	-	1		1	-	2	2	3	2	3	2	3	-	-	-

UNIT – I:
Basic Principles of SWOT and Life Positions
a) SWOT and Life Positions
b) Apply SWOT in real life scenarios, Create SWOT
c) SWOT Vs. TOWS-The Balancing Act
d) Importance of Motivation in real life
e) Leverage motivation in real-life scenarios.

UNIT – II:

Pluralism in Cultural Spaces

a) Awareness and respect for pluralism in cultural spaces
b) Rhythms of India (Cultures in India)
c) Define and Differentiate -Global, glocal, translocational
d) Cross-cultural communication- Culture shock
e) Gender awareness

UNIT – III:

Role of Science in Nation Building

a) Role of scientists and mathematicians from ancient India.
b) Role of science post- independence
c) Inventions –Inventors-Institutes-Information technology
d) Introduction to technical writing
e) Basic rules of technical writing

UNIT – IV:

Artificial Intelligence – Voice of the Future

a) Artificial intelligence in Everyday Life
b) Communicating with machines
c) Applying technical writing in profession
d) Scenario-based technical writing
e) Best practices of Technical writing

UNIT – V:

Technical Writing

a) Summarizing & Synthesizing
b) Abstract Writing
c) Report Writing
d) Product Description and Description of a mechanism
e) Project Work:

Visit rural area/ underprivileged parts of city to address some of the local issues; if relevant, suggest a practical technology solution to the issues.

TEXT BOOK:

There are no prescribed texts for Semester 4 – there will be handouts and reference links.

REFERENCES:

1. Effective Technical Communication, Ashraf Rizvi M., 2nd Edition, Tata McGraw-Hill, 2005
2. Technical Communication: A Reader-Centered Approach, Anderson, Paul V. Reports In Paul V. Anderson's Boston, 9th Edition, Heinle, 2003
3. Technical Communication: A Practical Approach, William S. Pfeiffer, 8th Edition, Pearson, 2012
4. Technical Communication, Burnett Rebecca, 6th Edition, Cengage Learning, 2001

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester

(22MN6HS202) INDIAN CONSTITUTION

TEACHING SCHEME		
L	T/P	C
2	0	0

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

COURSE PRE-REQUISITES: Basic knowledge of Panchayat Raj and human rights studied at schooling level

COURSE OBJECTIVE:

- To develop constitutional awareness
- To understand democracy at grass root level
- To familiarization of human rights and duties among students
- To inculcate responsibilities towards nation building through technology

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

CO-1: Understand constitutional creation its process and the process of Panchayat Raj and its working

CO-2: Get familiar with the science of Nation building through constitutional process of India

CO-3: Understand human rights, responsibilities and recognize the responsibilities for societal well-being

CO-4: Recognize the roles of constitution for corporate culture building

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	1	1	1	2	2	2	2	3	2	2	-	1	-
CO-2	1	1	1	1	1	2	2	2	2	3	3	3	-	1	-
CO-3	-	-	1	1	1	2	2	2	1	3	2	3	-	1	-
CO-4	-	1	1	1	-	2	2	2	2	2	2	3	-	-	-
CO-5	-	1		1	-	2	2	3	2	3	2	3	-	-	-

UNIT – I:

(Union Government and its Administration-Part 1)

Constitution' meaning of the term, Indian Constitution: Sources and constitutional history, Features: Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy,

UNIT – II:

(Union Government and its Administration-Part 2)

Structure of the Indian Union: Federalism, Centre- State relationship, President: Role, power and position, PM and Council of ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha.

UNIT – III:
(State Government and its Administration-Part 1)
Governor: Role and Position, CM and Council of ministers, State Secretariat: Organization, Structure and Functions.

UNIT – IV:
(State Government and its Administration-Part 2)
Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation, Pachayati raj: Introduction, PRI: Zila Pachayat, Elected officials and their roles, CEO Zila Pachayat: Position and role, Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy.

UNIT – V:
(Duties and Fundamental Rights-Part 1)
Features of fundamental rights, laws inconsistent with fundamental rights, right to equality, right to freedom, right against exploitation, right to freedom of religion, cultural and educational rights, right to constitutional remedies, criticism of fundamental rights, significance of fundamental rights.
Swaran Singh's Committees recommendation, list of fundamental duties, features of fundamental duties, Protection of Human Rights Act, 1993.

REFERENCE BOOKS:

1. Indian Polity, Laxmikanth, McGraw Hill
2. Indian Constitution, M. V. Pylee
3. Human Rights in Constitutional Law, Durgadas Basu
4. Indian Constitution, Upkar Publication