

B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE

Department of Mathematics

Syllabus

Semester: III		
Course: Fourier Transform, Mathematical logic and Advanced Linear Algebra		
Course Code: 24MAC131 (Common to CSE, ISE, AIML)		
L:T:P:J	2:2:0:0	CIA : 50
Credits:	03	SEA : 50
Hours:	40	SEA Duration : 03 Hours
Course Learning Objectives: The students will be able to 1 Have an insight into Fourier series, Fourier transforms. 2 Develop knowledge of Fundamentals of logic and Relations, Vector Spaces, Linear Transformation & Inner product spaces arising in engineering		
Module-1: Fourier Series & Fourier Transforms	No. of hours	Blooms cognitive Levels
<i>Examples from Engineering that require Fourier series and Fourier Transforms.</i> Fourier series: Periodic functions, Introduction to Fourier Series, Dirichlet's condition. Problems on Fourier series over $(-l, l)$. Fourier Transforms: Introduction to infinite Fourier transform, Fourier sine and cosine transform and properties, problems on infinite Fourier transform, Discrete & Fast Fourier transform. <i>Experiential Learning component: Finding the Fourier series and Fourier Transform of a function</i>	L : 04 T : 04	L1 L2 L3
Module-2: Mathematical logic and Boolean Algebra		
<i>Examples from Engineering that require Fundamentals of logic and Relations.</i> Mathematical logic: Basic connectives and truth tables, logic equivalence - the laws of logic, logical implication- rules of inference Boolean Algebra: Boolean functions, Representation of Boolean functions, Logic gates, minimization of circuits. <i>Experiential Learning component: Construction of combinational and sequential circuit.</i>	L : 04 T : 04	L1 L2 L3
Module-3: Vector Spaces		
<i>Examples from Engineering that require vector spaces</i> Recap of system of linear homogenous and non-homogeneous equation and solution sets. Vector spaces, subspaces, linearly independent and dependent, Linear span of a set, Basis and dimension, coordinate vectors. <i>Experiential Learning component: Problems on linearly independent and dependent vectors, basis and dimension of a vector space.</i>	L : 04 T : 04	L1 L2 L3
Module-4: Linear Transformation		
<i>Examples from Engineering that require linear transformation.</i> Linear transformations, algebra of linear transformations, representation of transformations by matrices, Non-singular linear transformation, Inverse of a linear transformation, Range space, Null space and problems on Rank-nullity theorem. <i>Experiential Learning component: Problems on Inverse of a linear transformation and Rank-nullity theorem</i>	L : 04 T : 04	L1 L2 L3
Module-5: Inner Product Spaces		
<i>Examples from Engineering that require Inner product spaces.</i> Inner products Inner product spaces, Orthogonal set, orthogonal projections, orthonormal bases, Gram-Schmidt process, QR-factorization, Recap of Eigen values and Eigen vectors, problems on Singular value decomposition. <i>Experiential Learning component: Problems on QR-factorization and singular value decomposition</i>	L : 04 T : 04	L1 L2 L3

Course Outcomes: After completing the course, the students will be able to	
CO 1:	Apply Fourier series & transform concepts in data visualization and cryptography.
CO 2:	Convert Boolean expressions to logic gates and vice-versa.
CO 3:	Apply the knowledge of vector spaces for solving problems in arising in engineering field
CO 4:	Apply the knowledge of linear transform for solving problems in arising in image processing
CO 5:	Compute orthogonal and orthonormal bases vectors and decomposition of a symmetric matrix using standard technique.

CO - PO Mapping:												
COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO 1	3	2			2							
CO 2	3	2			2							
CO 3	3	2			2							
CO 4	3	2			2							
CO 5	3	2			2							

Reference Books:

1. E. Kreyszig: "Advanced Engineering Mathematics", John Wiley & Sons, 10th Edition (Reprint), 2016.
2. B. S. Grewal: "Higher Engineering Mathematics", Khanna Publishers, 44th Ed., 2017.
3. C. Ray Wylie, Louis C. Barrett : "Advanced Engineering Mathematics", 6th Edition, 2. McGraw-Hill Book Co., New York, 1995.
4. James Stewart : "Calculus —Early Transcendentals", Cengage Learning India Private Ltd., 2017.
5. Srimanta Pal & Subodh C Bhunia: "Engineering Mathematics", Oxford University Press, 3rd Reprint, 2016.
6. David C. Lay, Steven R. Lay and J. J. McDonald "Linear Algebra and its applications", 3rd Edition, Pearson Education Ltd., 2017.
7. Kenneth H Rosen, "Discrete Mathematics and its Applications, Special Indian Edition 2021, McGraw Hill publication (India).
8. Ralph P. Grimaldi, " Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education 2004.

Web links and Video Lectures:

1. <https://nptel.ac.in/courses/111106111>
2. <https://youtu.be/OynpZwylau8>
3. <https://archive.nptel.ac.in/courses/111/106/111106051/>
4. <https://www.youtube.com/watch?v=zvRdbPMEMUI>
5. <https://www.youtube.com/watch?v=PiG2BMkK3s4>
6. https://www.youtube.com/watch?v=ATqV_I8DCh0

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering

Course Name: Computer Organization

Course Code: 24ISE132

SEMESTER-III

L:T:P:J:	3:0:0:0	CIE Marks:	50
Credits:	3	SEA Marks:	50
Total Number of Lecture Hours	40	SEA Duration:	3 Hours

Course objectives:

- To introduce the fundamental subsystems of a computer, including their organization, structure, and operational principles, with a focus on how machine instructions form executable programs.
- To develop a strong conceptual understanding of memory hierarchy and organization, including primary memory, cache memory, and their role in enhancing system performance.
- To equip students with the ability to perform arithmetic and logical operations in digital systems and to understand the functioning and significance of pipelining and multiprocessor architectures in improving computational efficiency.

Module1: Introduction	No. of Hours	Blooms Cognitive Levels
Basic Structure of Computers: Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement. Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes, Assembly Language, Basic Input and Output Operations, Stacks and Queues , Subroutines , Additional Instructions, Encoding of Machine Instructions.	8	Understand
Module 2: Input / Output Organization		
Accessing I/O Devices, Interrupts – Interrupt Hardware, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, USB.	8	Understand
Module 3: Memory System		
Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories Mapping Functions, Replacement Algorithms , Performance Considerations.	8	Apply
Module 4: MIPS Arithmetic operations		
MIPS Addressing for 32-Bit Immediate and Addresses , Parallelism and Instructions: Synchronization, Translating and Starting a Program, Addition and Subtraction (MIPS), Multiplication and Division (MIPS).	8	Apply

Module 5: Pipelining and Multiprocessors	No. of Hours	Blooms Cognitive Levels
Pipelining: Basic concepts, Data Hazards, Instruction hazards, Basic processing unit: some fundamental concepts, execution of complete instruction, multi-bus organization, The structure of general-purposes multiprocessors, Parallel Computer Architecture: Processor Architecture and Technology Trends, Flynn's Taxonomy of Parallel Architectures.	8	Analyze

Course outcomes: The students should be able to:

- **CO1:** Understand the fundamental concepts of computer systems including instruction formats, memory hierarchy, input/output organization, MIPS instruction set, and pipelining in multiprocessors.
- **CO2:** Construct basic assembly language programs using MIPS, and design basic memory and I/O interfacing systems for effective data transfer.
- **CO3:** Analyze the performance of computer systems by evaluating instruction cycles, memory access techniques, pipelining efficiency, and multiprocessor architectures using standard metrics and models.

Reference Books:

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Naraig Manjikian, Computer Organization and Embedded Systems, 6th Edition (Reprint), McGraw Hill Education, 2017.
2. Thomas Rauber, Gudula Rünger, Parallel Programming for Multicore and Cluster Systems, 3rd Edition, Springer, 2023.
3. David A. Patterson, John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, 6th Edition, Morgan Kaufmann / Elsevier, 2021.
4. William Stallings, Computer Organization and Architecture, 11th Edition, Pearson Education, 2021.


24/8/25

Head of the Department
Dept. of Information Science & Engineering
B.N. Institute of Technology
Bangalore - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering

Operating Systems Course Code: 24ISE133

SEMESTER-III

L:T:P:J:	3:0:0:0	CIA Marks:	50
Credits:	3	SEA Marks:	50
Total Number of Lecture Hours	40	SEA Duration:	3 Hours

Course Objectives:

- Understand and analyze the core principles of operating system design and functionality, including process management, CPU scheduling, memory management, and deadlock handling.
- Develop the ability to implement and evaluate synchronization mechanisms and memory management techniques, including page replacement and disk scheduling algorithms, through practical programming exercises.
- Gain proficiency in UNIX system usage, including file system navigation, command-line operations, file permission handling, and shell scripting, to effectively automate tasks and manage system resources.

Module 1: Introduction to Operating System & Process Management	No. of Hours	Blooms Cognitive Levels
Fundamental Concepts of Operating System: Introduction to Operating systems, Operating system functions and services, historical evolution of operating systems, System boot. Process Management: Process abstraction, process address space, process management, system calls, threads. CPU Scheduling: Levels of scheduling, comparative study of scheduling algorithms, Multilevel Queue Scheduling, Multilevel Feedback Queue Scheduling, Multi-processor scheduling. Practical Component: CPU Scheduling Programs not limited to FCFS, SJF, Pre-emptive and non-Pre-emptive.	8	Analyze
Module 2: Process Synchronization and Deadlocks Concurrent Processes: Critical section problem, semaphores, Classical problems of synchronization, monitors, inter-process communication, message passing mechanisms. Deadlocks: Characterization, prevention and avoidance, deadlock detection and recovery. Practical Component: Synchronization Programs not limited to readers writers' problem, bounded buffer problem.	8	Analyze
Module 3: Memory Management Memory Management: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation, virtual memory concept, demand paging, page replacement algorithms, thrashing, Disk Scheduling. Solid State Drives-SSD Architecture, Flash Controller, Practical Component: Disk Scheduling Programs not limited to FCFS, SCAN, LOOK.	8	Analyze

Module 4: UNIX file system		No. of Hours	Blooms Cognitive Levels
Unix files: UNIX Architecture, Naming files. Basic file types/categories. Organization of files. Hidden files. Standard directories. Parent child relationship. The home directory and the HOME variable. Reaching required files- the PATH variable, manipulating the PATH, Relative and absolute pathnames. Directory commands – pwd, cd, mkdir, rmdir commands. The dot (.) and double dots (..) notations to represent present and parent directories and their usage in relative path names. File related commands – cat, mv, rm, cp, wc and od commands. Practical component: Execution of UNIX Shell Commands.		8	Apply
Module 5: File attributes and Shell programming			
File attributes and permissions: The ls command with options. Changing file permissions: the relative and absolute permissions changing methods. Recursively changing file permissions. Directory permissions. The shells interpretive cycle: Wild cards. Removing the special meanings of wild cards. Three standard files and redirection. Connecting commands: Pipe, grep, egrep. Shell programming: Ordinary and environment variables. Read and read-only commands. Command line arguments. Logical operators for conditional execution. The test command and its shortcut. The if, while, for and case control statements. The set and shift commands and handling positional parameters. The here (<<) document. Simple shell program examples. Practical component: Execution of Wildcards & UNIX Shell Programs with respect to conditional statements.		8	Analyze
Course Outcomes: After completing the course, the students will be able to			
CO 1	Understand the fundamental concepts of operating systems, process management, memory organization, and UNIX file system structures.		
CO 2	Apply the concepts of shell scripts, CPU & disc scheduling, process synchronization, and UNIX file operations.		
CO 3	Analyze and compare Shell wild cards, UNIX pattern matching commands, CPU scheduling & Disc Scheduling strategies, deadlock handling techniques, and memory management schemes to improve system performance.		

Reference Books:	
- Sumitabha Das., Unix Concepts and Applications, 4th Edition, Tata McGraw Hill, November 8, 2017 ISBN: 978-0070635463. - Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 10th edition, Wiley-India, April 15, 2018. - William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 9th Edition, 2018. W. Richard Stevens: Advanced Programming in the UNIX Environment, 3rd Edition, Pearson Education, 2013. - Unix System Programming Using C++ - Terrence Chan, 1st Edition, PHI, 1996.	
Web links and Video Lectures:	
https://academicearth.org/ https://archive.nptel.ac.in/courses/106/105/106105214/	


 Head of the Department
 Dept. of Information Science & Engineering
 B.N.M Institute of Technology
 Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

SEMESTER– III

Course Name: Data Structures and its Applications

Course Code: 24ISE134

L:T:P:J:	3:0:2:0	CIE Marks:	50
Credits:	4	SEA Mark33s:	50
Total Number of Lecture Hours	50	SEA Duration:	3 Hours

Course objectives:

1. To introduce fundamental linear data structures such as stacks, queues, and linked lists, along with their operations, implementations, and practical applications in problem-solving.
2. To equip with the understanding and implementation of hierarchical and graph data structures, focusing on their properties, traversals, and real-world use cases.
3. To provide a solid understanding of advanced data structures including heap trees and hashing techniques, covering key operations like insertion, deletion, and collision handling.
4. To introduce efficient file organization and indexing techniques, including primary and secondary indexing, B-trees, and multi-way merging for optimized data access.

Module 1: Stacks and Queues	No. of Hours	Blooms Cognitive Levels
Data Structures: Introduction to Structures, Classification (Primitive & Non-primitive), Operations, Pattern Matching Algorithms (Brute force, KMP) Stacks: Definition, Operations, Implementation using arrays, Applications of Stacks – Infix to Postfix Conversion and Postfix Expression Evaluation. Queues: Definition, Operations, Implementation, Applications, Circular Queue (Message queue using Circular queue), Doubly Ended Queue, Priority Queue. Practical Component: Tracing Exercises, Analysis Exercises, Exercises to Modify programs. Programs not limited to: 1. Implement KMP pattern matching algorithm for a given main string and pattern. 2. Implement Boyer-Moore pattern matching algorithm for a given main string and pattern. 3. Program to implement stack operations (push, pop, and display) using static array and also on dynamic array and compare the performance. 4. Program to convert infix expression into postfix. 5. Program to evaluate given postfix expression. 6. Program to implement queue operations using static arrays and on dynamic arrays and compare the performance. 7. Program to implement circular queue operations using arrays and on dynamic arrays and compare the performance.	10	Apply

Module 2: Linked List	No. of Hours	Blooms Cognitive Levels
Linked Lists: Definition, Create, Insert, Delete, Update, Traverse, and Position-based Operations, Concatenate, Merge, and Reverse Lists, Doubly Linked List Implementation and Operations, Circular Linked List Implementation and Operations, Applications of Lists (Polynomial addition). Implementation of stacks and queues using Linked List. Practical Component: Tracing Exercises, Analysis Exercises, Exercises to Modify programs. Programs not limited to: 1. Program to create single linked list and implement its operations with and without header nodes: i. Insert (front and rear end) ii. Delete. (Front and rear end) iii. Search. iv. Reverse. 2. Implement DLL as a stack and also as a queue. 3. Create 2 CLL of sparse polynomials and perform addition on these 2 polynomials. Represent the resulting polynomial in a CLL. 4. Implement multiple stacks and multiple queues in a singly linked list.	10	Apply
Module 3: Trees		
Concept Learning: Trees: General Tree Representation, Traversals, Applications. Binary Trees: Definition, Properties, Traversals, Applications. Binary Search Tree: Definition, Implementation, Search, Insert, Delete operations. Building and Evaluating Binary Expression Tree. Concept Learning Graphs: Adjacency Lists, and Traversal of Graphs (BFS and DFS) Practical Component: Tracing Exercises, Analysis Exercises, Exercises to Modify programs. Programs not limited to: 1. Implement the following operations on Binary tree: - A. Count the number of nodes in the binary tree B. Count the leaf nodes C. Check if 2 trees are equal D. Perform Tree Traversals namely in-order, pre-order, post-order and level order. 2. Create a Binary Search Tree for a set of integer values, perform deletion of a key and searching for a key. 3. For a given input graph check whether the nodes are connected or not using BFS and DFS method. 4. Create an expression tree to evaluate the given expression using binary tree.	10	Apply
Module 4: Advanced Trees & Hashing		
Concept Learning: Heap Tree: Definition, Implementation, Insert, Delete, Peek operations. Hashing: Hash Table, Hash Functions, Collision Handling by Open Addressing, Chaining.	10	Apply

Practical Component: Tracing Exercises, Analysis Exercises, Exercises to Modify programs. Programs not limited to: 1. Create a max heap and min heap of integers, display the values and perform deletion operations. 2. Program to implement hashing with collision resolution.		
Module 5: File Structures	No. of Hours	Blooms Cognitive Levels
File Structures: Primary Indexing, Secondary Indexing, B-Trees, Co sequential processing, K way merge . Practical Component: Tracing Exercises, Analysis Exercises, Exercises to Modify programs. Programs not limited to: 1. Write a program to implement simple index on primary key for a file. Implement add (), search (), delete () using the index. 2. Write a program to read two lists of names and then match the names in the two lists using Consequential Match based on a single loop. Output the names common to both the lists. 3. Write a program to read k Lists of names and merge them using k-way merge algorithm with k = 8.	10	Apply

Course outcomes: The students should be able to:

1. Understand the fundamental concepts of linear and non-linear data structures, their types, operations, and applications in problem-solving, along with file structures like indexing, B-trees, and k-way merge.
2. Apply various linear and non-linear data structures such as stacks, queues, linked lists, trees, graphs, and file structures to develop efficient solutions for real-world problems.
3. Apply static and dynamic memory techniques to implement linear and non-linear data structures such as stacks, queues, linked lists, trees, graphs, heaps, and file structures effectively to solve computational problems.
4. Apply pattern matching, expression evaluation, traversals, hashing, and file operations to ensure correct and efficient solutions.

Reference Books:

1. Data Structures Using C – Reema Thareja, 3rd Edition, Oxford Higher Education, 2023, ISBN-13: 978-9354977190.
2. Introduction to Algorithms – T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, 4th Edition, MIT Press, MA, 2022.
3. Data Structures and Algorithm Analysis in C++ – Mark Allen Weiss, 4th Revised Edition, Addison-Wesley, 2014, ISBN-13: 978-0132847377.

Weblinks:

1. <https://youtube.com/playlist?list=PLBF3763AF2E1C572F&feature=shared>
2. <https://youtu.be/gJ6f3qGwGAU?feature=shared>


 Head of the Department
 Dept. of Information Science & Engineering
 B.N.M Institute of Technology
 Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

SEMESTER– III

Course Name: Object Oriented Programming using Java

Course Code: 24ISE135

L: T:P: J	3:0:2:0	CIA Marks	50
Credits	04	SEA Marks	50
Total Number of Contact Hours	50	SEA Duration	3 Hours

Course Learning Objectives: This course will enable students to:

- Show competence in the use of the Java programming language in the development of small to medium-sized application programs that demonstrate professionally acceptable coding and performance standard.
- Understand the basic principles of the object-oriented programming.
- Demonstrate an introductory understanding of graphical user interfaces, multithreaded programming, and event-driven programming.

Module – 1	No. of Hours	Blooms Cognitive Levels
Basics of Java: Features of Java, Byte Code and Java Virtual Machine, JDK, Data types, Operator, Control Statements – If , else, nested if, if-else ladders, Switch, while, do-while, for, for-each, break, continue. Array : Single and Multidimensional Array, Classes, Objects and Methods: Class, Object, Object reference, Constructor, Constructor Overloading, Method Overloading, Recursion, Passing and Returning object form Method, new operator, this and static keyword, finalize() method , Access control, modifiers, Nested class, Inner class, Anonymous inner class, Abstract class. Programs: 1. Write a program to convert rupees to dollar. 60 rupees=1 dollar. 2. Write a program that calculate the percentage marks of the student if marks of 6 subjects are given. 3. Write a program to enter two numbers and perform mathematical operations on them. 4. Write an interactive program to print a diamond shape. 5. Program to define class Box and constructors. Demonstrate constructors. 6. Program to define class Addition, methods and objects. Demonstrate method overloading to add 2 integers, 3 floating numbers and 2 strings.	12	Apply
Module – 2		
String: String class, StringBuffer class, Operations on string, Command line argument, Use of Wrapper Class. Inheritance: Use of Inheritance, Inheriting Data members and Methods, constructor in inheritance, Multilevel Inheritance – method overriding Handle multilevel constructors – super keyword, Stop Inheritance - Final keywords, Dynamic method dispatch, Understanding of Java Object Class. Package: Use of Package, CLASSPATH, Import statement, Static import, Access control.		

	No. of Hours	Blooms Cognitive Levels
Programs: 1. Write a program to find length of string and print second half of the string. 2. Write a program to accept a line and check how many consonants and vowels are there in line. 3. Write a program to count the number of words that start with capital letters. 4. Write a program to find that given number or string is palindrome or not. 5. Create a class which asks the user to enter a sentence, and it should display count of each vowel type in the sentence. The program should continue till user enters a word "quit". Display the total count of each vowel for all sentences. 6. Write an interactive program to print a string entered in a pyramid form. 7. Program to define inheritance and show method overriding - Program to define base class Shapes, with methods and create derived classes Square, Triangle, Rectangle and Circle. Create objects and show method overriding for calculating area of Square, Triangle, Rectangle and Circle.	10	Apply
Module – 3		
Interfaces : Creation and Implementation of an interface, Interface reference, Interface inheritance, Nesting Interfaces, variables in Interfaces, Comparison between Abstract Class and Interface. Exception Handling: Exception and Error, Use of try, catch, throw, throws and finally, Built in Exception, Custom exception, Throwable Class. Programs: 1. Create an Interface Stack with methods push(), pop() and show(). Create class IntStack which implements Stack. Demonstrate the working of this class. 2. Create an Interface Queue with methods insert(), delete() and display(). Create class StringQueue which implements Queue. Demonstrate the working of this class. 3. Analyze how a programmer must handle an error when an invalid mobile number is entered (lesser than 10 digits or greater than 10 digits; with alphabets; starting with 0) is entered while taking the details of a student. The Student class must contain the following: Name; USN; Branch and Mobile Number. Write a program using a custom exception type. 4. Create an exception called Non Int Result Exception, which is generated when the result of dividing two integer values produces a result with a fractional component. Non Int Result Exception has two fields that hold integer values; a constructor; and an override of the toString() method, allowing a friendlier description of the exception to be displayed using println(). 5. Create a Counter class with a private count instance variable and two methods. The first method: synchronized void increment() – tries to increment count by 1. If count is already at its maximum of 3, then it waits until count is less than 3 before incrementing it. The other method is synchronized void decrement() – attempts to decrement count by 1. If count already at a minimum of 0, then it waits until count is greater than 0 before decrementing it. Every time either method has to wait, it displays a statement as to why it is waiting. Also every time an increment or decrement occurs, the Counter displays a statement that says what occurred and shows count's new value. 6. Create a Bank class with withdrawal and deposit methods. Demonstrate the use of notify() and wait().	10	Apply

Module – 4	No. of Hours	Blooms Cognitive Levels
Multithreaded Programming: Use of Multithread programming, Thread class and Runnable interface , Thread priority, Thread synchronization, Thread communication, Deadlock. Swings: Swings: The origins of Swing; Two key Swing features; Components and Containers; The Swing Packages; A simple Swing Application; Create a Swing J Applet; J label and Image Icon; J Text Field; The Swing Buttons; J Tabbed pane; J Scroll Pane; JList; J Combo Box; JTable. Programs: 1. Write a JAVA program which creates two threads, one thread displays “JAVA” 10 times, and another thread displays “PROGRAMMING LANGUAGE” 15 times continuously? 2. Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number and prints; third thread will print the value of cube of the number. 3. Write a program that prints the Multiplication Table for numbers 2- 10. At a time the tables must print completely for one number before printing for the next number. 4. Create a GUI for a business of your choice using swings. Demonstrate the use of Button, Radio Button, Combo box, List, Textfield etc.	10	Apply
Module – 5 Servlet : The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A simple Servlet; The Servlet API; The Javax. servlet Package; Reading Servlet Parameter; The Javax. servlet.http package; Handling HTTP Requests and Responses; Using Cookies; Session Tracking. Java Server Pages (JSP): JSP, JSP Tags, Tomcat, Request String, User Sessions, Cookies, Session Objects. Programs: 1. Write Servlet application to print current date &time. 2. Write Servlet application to demonstrate session tracking. 3. Implement Program 1 using JSP. 4. Implement Program 2 using JSP.	8	Apply

Course outcomes: The students should be able to:

- CO1:** Demonstrate a solid understanding of object-oriented programming (OOP) principles and effectively apply them using Java.
- CO2:** Utilize core OOP constructs such as classes, inheritance, polymorphism, interfaces, and packages to develop modular and reusable code.
- CO3:** Implement robust exception handling and multithreading techniques to create efficient and error-resilient applications.
- CO4:** Design and develop Java-based GUI and web applications, incorporating servlets and event-driven programming to simulate real-world scenarios.
- CO5:** Apply Java syntax and semantics proficiently to write clean, maintainable, and scalable object-oriented programs.

Reference Books

1. Java: The Complete Reference, Herbert Schildt, 13th Edition, 2023, McGraw Hill Education, ISBN-13: 978-1265058431.
2. Java Programming: A Comprehensive Introduction, Herbert Schildt, Dale Skrien, 1st Edition, 2013, McGraw Hill Education, ISBN-13: 978-0078022074.
3. Programming with Java: A Primer, E. Balagurusamy, 5th Edition, 2014, McGraw Hill Education, ISBN-13: 978-9351343202.
4. Core Java Volume I – Fundamentals, Cay S. Horstmann, 13th Edition, 2024, Pearson Education, ISBN-13: 978-0138051808.
5. Head First Java: A Brain-Friendly Guide, Kathy Sierra, Bert Bates, Trisha Gee, 3rd Edition, 2021, O'Reilly Media, ISBN-13: 978-1491910771.


 Head of the Department
 Dept. of Information Science & Engineering
 B.N.M Institute of Technology
 Bangalore - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering SEMESTER-III

Course Name: Web Technology

Course Code: 24ISE136

L: T: P: J:	0:0:2:2	CIA:	50
Credit:	02	SEA:	50
Hours:	25	SEA Duration:	3 HOURS

Course Learning Objectives: The students will be able to

1. Understand the structure and semantics of HTML and create well-structured web pages.
2. Learn and apply core JavaScript concepts, including DOM manipulation and ES6 features, for interactive web applications.
3. Implement client-side scripting with JavaScript, event handling, and JSON for dynamic user interfaces.
4. Understand server-side programming using Java Servlets and JDBC for data handling and session management.
5. Analyze and apply React concepts and component-based architecture for building modern, scalable web applications.

PART A

Module-1	No. of Hours	Blooms Cognitive Levels
Introduction to HTML, what is HTML and Where did it come from? HTML Syntax, Semantic Markup, Structure of HTML Documents, HTML Elements, HTML5 Semantic Structure Elements. HTML Tables and Forms, Introducing Tables, Styling Tables, Introducing Forms, Form Control Elements, Table and Form Accessibility. Build a Personal Portfolio Webpage, Create a Responsive Form, Style and Enhance an HTML Table.	5	Apply
Module-2		
JavaScript: What is JavaScript and What can it do? JavaScript Design Principles, where does JavaScript Go? Syntax, JavaScript Objects, The Document Object Model (DOM), ES6 features, form validation and basic animation ES6 features, form validation and basic animation Explore ES6 Features, Form Validation Exercise, Simple Animation with JavaScript.	5	Apply
Module-3		
Client Side Programming Java Script: An introduction to JavaScript- JavaScript DOM Model-Exception HandlingValidation- Built-in objects-Event Handling- DHTML with JavaScript- JSON introduction -Syntax - Function Files. Event-Driven Web App, Parse and Display JSON Data, DHTML Project.	5	Apply
Module-4		
Server Side Programming Servlets: Java Servlet Architecture- Servlet Life Cycle- Form GET and POST actions- Session Handling- Understanding Cookies- DATABASE CONNECTIVITY: JDBC. Self- Build a Login System Using Servlets, Form Handling with GET and POST, JDBC CRUD Operation	5	Apply

Module – 4	No. of Hours	Blooms Cognitive Levels
Multithreaded Programming: Use of Multithread programming, Thread class and Runnable interface , Thread priority, Thread synchronization, Thread communication, Deadlock. Swings: Swings: The origins of Swing; Two key Swing features; Components and Containers; The Swing Packages; A simple Swing Application; Create a Swing J Applet; J label and Image Icon; J Text Field; The Swing Buttons; J Tabbed pane; J Scroll Pane; JList; J Combo Box; JTable. Programs: 1. Write a JAVA program which creates two threads, one thread displays “JAVA” 10 times, and another thread displays “PROGRAMMING LANGUAGE” 15 times continuously? 2. Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number and prints; third thread will print the value of cube of the number. 3. Write a program that prints the Multiplication Table for numbers 2- 10. At a time the tables must print completely for one number before printing for the next number. 4. Create a GUI for a business of your choice using swings. Demonstrate the use of Button, Radio Button, Combo box, List, Textfield etc.	10	Apply
Module – 5		
Servlet : The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A simple Servlet; The Servlet API; The Javax. servlet Package; Reading Servlet Parameter; The Javax. servlet.http package; Handling HTTP Requests and Responses; Using Cookies; Session Tracking. Java Server Pages (JSP): JSP, JSP Tags, Tomcat, Request String, User Sessions, Cookies, Session Objects. Programs: 1. Write Servlet application to print current date &time. 2. Write Servlet application to demonstrate session tracking. 3. Implement Program 1 using JSP. 4. Implement Program 2 using JSP.	8	Apply

Course outcomes: The students should be able to:

- CO1:** Demonstrate a solid understanding of object-oriented programming (OOP) principles and effectively apply them using Java.
- CO2:** Utilize core OOP constructs such as classes, inheritance, polymorphism, interfaces, and packages to develop modular and reusable code.
- CO3:** Implement robust exception handling and multithreading techniques to create efficient and error-resilient applications.
- CO4:** Design and develop Java-based GUI and web applications, incorporating servlets and event-driven programming to simulate real-world scenarios.
- CO5:** Apply Java syntax and semantics proficiently to write clean, maintainable, and scalable object-oriented programs.

Reference Books

1. Java: The Complete Reference, Herbert Schildt, 13th Edition, 2023, McGraw Hill Education, ISBN-13: 978-1265058431.
2. Java Programming: A Comprehensive Introduction, Herbert Schildt, Dale Skrien, 1st Edition, 2013, McGraw Hill Education, ISBN-13: 978-0078022074.
3. Programming with Java: A Primer, E. Balagurusamy, 5th Edition, 2014, McGraw Hill Education, ISBN-13: 978-9351343202.
4. Core Java Volume I – Fundamentals, Cay S. Horstmann, 13th Edition, 2024, Pearson Education, ISBN-13: 978-0138051808.
5. Head First Java: A Brain-Friendly Guide, Kathy Sierra, Bert Bates, Trisha Gee, 3rd Edition, 2021, O'Reilly Media, ISBN-13: 978-1491910771.

Course Outcomes: After completing the course, the students will be able to

CO 1	Understand the HTML semantics, JavaScript concepts, client-side scripting, server-side programming with servlets and JDBC, and the core principles of building modular front-end applications using React.
CO 2	Apply web development technologies including HTML5, JavaScript, DHTML, Servlets, JDBC, and React to design, validate the web applications with interactive and dynamic user interfaces.
CO 3	Design and develop the web applications by applying HTML semantics, implementing client-side functionalities using JavaScript and JSON, managing server-side operations with Java Servlets and JDBC, and building modern front-end interfaces using React.
CO 4	Analyse and implement server-side applications using Java Servlets and JDBC for session, cookie, and database interaction.

Reference Books:

1. Programming the World Wide Web, 8th Edition, Robert W. Sebesta, Pearson Education, ISBN: 978-0133775983, 2014.
2. Introduction to React, Cory Gackenheimer, 1st Edition, Apress, ISBN: 978-1484212462, September 2015.
3. Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics" by Jennifer Niederst Robbins, 5th Edition published in May 2018.
4. Learning React: Modern Patterns for Developing React Apps" by Alex Banks and Eve Porcello, Alex Banks and Eve Porcello, 2020
5. Learning React: A Hands-On Guide to Building Web Applications Using React and Redux, Kirupa Chinnathambi, 2nd Edition, Addison-Wesley Professional, ISBN: 978-0134843551, 2018.

Web links and Video Lectures:

1. <https://academicearth.org/>
2. https://onlinecourses.nptel.ac.in/noc20_ee98/preview
3. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_014157697606279168211/o_verview


24/6/25

Head of the Department
Dept. of Information Science & Engineering
B.N.M. Institute of Technology
Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE

Department of Mathematics

Syllabus

Semester: IV

Course: Statistics, Probability and Graph theory
Course Code: 24MAC141 (Common to CSE, ISE, AIML)

L:T:P:J	2:2:0:0	CIA: 50
Credits:	03	SEA: 50
Hours:	40	SEA Duration: 03 Hours

Course Learning Objectives: The students will be able to

- 1 Provide an insight into applications of Graph Theory, Curve fitting & Statistical methods.
- 2 Develop the knowledge of probability, joint probability distribution and Queuing theory occurring in digital signal processing, design engineering and micro wave engineering.

Module-1: Curve fitting & Statistical methods		No. of hours	Blooms cognitive Levels
<i>Examples from Engineering that require curve fitting and statistical methods.</i> Curve Fitting: Curve fitting by the method of least squares-fitting the curves of the form: $y = ax+b$, $y = ax^b$ and $y = ax^2 + bx + c$. Statistical methods: Introduction to Moments, Skewness, Kurtosis and problems. Karl Pearson's coefficient of correlation and lines of regression. <i>Experiential Learning component: Problems on curve fitting and statistical methods</i>		L: 04 T: 04	L1 L2 L3
Module-2: Probability distributions & Joint probability distribution			
<i>Examples from Engineering that require Probability and Joint probability distribution</i> Probability distributions: Review of basic probability theory. Discrete and continuous Random variables, probability mass/density functions (definitions only). Binomial, Poisson, exponential and normal distributions (without proof). Joint probability distribution: Joint Probability distribution for two discrete random variables, expectation, covariance and correlation. <i>Experiential Learning component: Problems on Binomial, Poisson, Exponential and Normal distributions</i>		L: 04 T: 04	L1 L2 L3
Module-3: Markov chain & Sampling theory			
<i>Examples from Engineering that require Markov Chain and Sampling Theory</i> Markov chain: Introduction to Stochastic process, Probability vectors, Stochastic matrices, Regular stochastic matrices, Markov Chains, Higher transition probabilities, Stationary distribution of Regular Markov chains and absorbing states, Markovian processes. Sampling theory: Introduction to sampling theory, testing of hypothesis, level of significance, confidence limits, test of significance of mean and difference of means for large samples-z-test, test of significance of small Samples-Student's t- distribution. <i>Experiential Learning component: Problems on Markovian processes and, Sampling Theory</i>		L: 04 T: 04	L1 L2 L3
Module-4: Queuing theory			
<i>Examples from Engineering that require queueing theory</i> Introduction, birth and death process, Kendall's Notation, Symbolic representation of a queueing model, single server Poisson queueing model with infinite capacity (M/M/1: ∞ /FCFS), when $\lambda_n = \lambda$ and $\mu_n = \mu$ ($\lambda < \mu$), Multiple server Poisson queueing model with infinite capacity (M/M/S: ∞ /FCFS), when $\lambda_n = \lambda$ for all n , ($\lambda < S\mu$), <i>Experiential Learning component: Problems on (M/M/1: ∞/FCFS) and (M/M/S: ∞/FCFS) queueing models</i>		L: 04 T: 04	L1 L2 L3
Module-5: Graph theory			
<i>Examples from Engineering that require graph theory</i> Basic concepts, types of graphs, order and size of a graph, in-degree and out-degree, bipartite-graphs, connected and disconnected graphs, Eulerian graph, Hamiltonian graphs, sub-graphs, isomorphic graphs. Matrix representation of graphs, adjacency matrix, incidence matrix. Planar graphs: definition, characterization of planar graphs, Kuratowski's theorem, Euler's formula and consequences. <i>Experiential Learning component: Problems on detection of planar and non-planar graphs</i>		L: 04 T: 04	L1 L2 L3

Course Outcomes: After completing the course, the students will be able to

- CO 1: Make use of correlation and regression analysis to fit a suitable mathematical model for the statistical data.
- CO 2: Apply discrete and continuous probability and joint probability distributions in analyzing the probability models arising in engineering field.
- CO 3: Use Markov chain in prediction of future events and demonstrate the validity of testing the hypothesis.
- CO 4: Acquire skills in analyzing queuing models.
- CO 5: Apply the knowledge of Graph Theory in Network modeling, electrical network and computational algorithms.

CO - PO Mapping:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO 1	3	2			2							
CO 2	3	2			2							
CO 3	3	2			2							
CO 4	3	2			2							
CO 5	3	2			2							

Reference Books:

1. E. Kreyszig: "Advanced Engineering Mathematics", John Wiley & Sons, 10th Edition(Reprint), 2016.
2. B. S. Grewal: "Higher Engineering Mathematics", Khanna Publishers, 44th Edition, 2017.
3. S. D. Sharma : "Operations Research", Kedar Nath Ram Nath & Co. Meerut, 2014.
4. T. Veerarajan : Probability, Statistics and Random processes, McGraw Hill Education(India) Private Limited, Third edition, Nineteenth reprint 2017.
5. C. Ray Wylie, Louis C. Barrett : "Advanced Engineering Mathematics", 6th Edition, McGraw-Hill Book Co., New York, 1995.
6. B. V. Ramana: "Higher Engineering Mathematics" 11th Edition, Tata McGraw-Hill, 2010.
7. Srimanta Pal & Subodh C. Bhunia: "Engineering Mathematics", Oxford University Press, 3rd Reprint, 2016.
8. Narsingh Deo, "Graph Theory with Applications to Engineering and Computer Science", Prentice Hall of India, 2000.

Web links and Video Lectures:

1. <https://nptel.ac.in/courses/111104098>
2. <https://www.youtube.com/watch?v=1YkfeR05YXY>
3. <https://archive.nptel.ac.in/courses/111/104/111104079/>
4. <https://www.youtube.com/watch?v=xGkpXk-AnWU>
5. <https://archive.nptel.ac.in/courses/106/104/106104170/>

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

SEMESTER-IV

Course Name: MICROCONTROLLER AND EMBEDDED SYSTEM

Course Code: 24ISE142

L:T:P:J	3:1:1:0	CIA:	50
Credits:	04	SEA:	50
Hours:	50	SEA Duration:	3 HOURS

Course Learning Objectives: The students will be able to

1. Understand the fundamentals of ARM-based embedded systems, including architecture, instruction set, hardware components, selection criteria, and core attributes of embedded system design.
2. Develop proficiency in programming the ARM7TDMI/LPC2148 microcontroller, using Embedded C and Keil μ Vision-4 tools, and to implement and test embedded applications using appropriate instructions and interfacing techniques.
3. Explore the applicability and role of real-time operating systems (RTOS) in embedded system design and evaluate system-level integration through experiments on ARM-based development boards.

Module-1	No. of Hours	Blooms cognitive Levels
Microprocessors versus Microcontrollers, ARM7 Embedded Systems: The RISC design philosophy, The CISC design philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software. ARM Processor Fundamentals: Registers, Current Program Status Register, Pipeline, Exceptions, Interrupts, and the Vector Table, Core Extensions.	10	Apply
Module-2		
Introduction to the ARM7 Instruction Set: Data Processing Instructions. Branch Instructions, Load-Store Instructions, Software Interrupt Instructions, Program Status Register Instructions, Loading Constants. Writing and Optimizing ARM Assembly Code: Writing Assembly code, Profiling and cycle counting, Instruction scheduling, Register Allocation, Conditional Execution, Looping Constructs. List of programs: 1. Write a program to add an array of 16 bit numbers & store the 32 bit result in internal RAM 2. Write a program to find the square of a number (1 to 10) using look-up table. 3. Write a program to find the largest number in an array of 32 numbers. 4. Write a program to arrange a series of 32 bit numbers in ascending order. 5. Write a program to count the number of ones and zeros in two consecutive memory locations.	10	Apply

Module-3		
Embedded System Components: Embedded Vs General computing system, History of embedded systems, Classification of Embedded systems, Major applications areas of embedded systems. Typical Embedded System: Core of an Embedded System, Memory, Sensors and Actuators (LED, 7segment LED display, stepper motor, Keyboard, Push button switch), Communication Interface (on board Communication Interface), Embedded firmware, Other system components. List of programs: 6. Display "Hello World" message using Internal UART. 7. Interface a Stepper motor and rotate it in clock wise and anti-clock wise direction.	10	Apply
Module-4		
Characteristics and Quality Attributes of an Embedded Systems: Characteristics of an Embedded Systems, Quality Attributes of an Embedded Systems. Embedded Systems-Application and Domain Specific: Washing Machine-Application Specific Embedded System. Hardware and Software Co-Design and Program Modelling: Fundamental Issues in Hardware Software Co-Design, Computational Models in Embedded Design, Introduction to Unified Modelling Language(UML). List of programs: 8. Interface a Stepper motors and rotate in clockwise and anti-clock wise directions. 9. Interface DAC and generate Triangular wave forms.	10	Apply
Module-5		
RTOS and IDE for Embedded System Design: How to choose an RTOS, Integration and testing of Embedded hard ware and firmware, Embedded system Development Environment-Block diagram (excluding Keil), Disassembler / decompiler, simulator, emulator and debugging techniques, target hardware debugging, Magnifying Glass, Multimeter , Digital CRO, Logic Analyzer, Function Generator , boundary scan. List of programs: 10. Demonstrate the use of an external interrupt to toggle an LED On/Off. 11. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.	10	Apply

Course Outcomes: After completing the course, the students will be able to	
CO 1	Identify the architectural features and instruction set of the ARM7 microcontroller, and describe the characteristics, components, and design principles of embedded systems.
CO 2	Develop and test embedded applications by programming the ARM7 microcontroller (LPC2148), and interface it with external devices using appropriate I/O techniques.
CO 3	Demonstrate hardware/software co-design, implement firmware design approaches, and illustrate the role and need of real-time operating systems in embedded system applications.

Reference Books:

1. ARM System Developer's Guide: Designing and Optimizing System Software, Andrew N. Sloss, Dominic Symes, Chris Wright, 1st Edition, Elsevier/Morgan Kaufmann, 2005, ISBN-13: 978-8181476463.
2. Introduction to Embedded Systems, K.V. Shibu, 2nd Edition, McGraw Hill Education, 2016, ISBN-13: 978-9339219680.
3. Microcontroller (ARM) and Embedded Systems, Raghunandan G.H., 1st Edition, Cengage Learning India Pvt. Ltd., 2020, ISBN-13: 978-9353504106.
4. The Insider's Guide to the Philips ARM7-Based Microcontrollers: An Engineer's Introduction to the LPC2100 Series, Trevor Martin, 1st Edition, Hitex, 2005, ISBN-13: 978-0954998813.
5. ARM System-on-Chip Architecture, Steve Furber, 2nd Edition, Pearson Education, 2000, ISBN-13: 978-0201675191.
6. Embedded Systems: Architecture, Programming & Design", published in 2014 by McGraw-Hill Education, ISBN: 9789332901506.

Web links and Video Lectures:

1. <https://academicearth.org/>
2. https://onlinecourses.nptel.ac.in/noc20_ee98/preview



Head of the Department
Dept. of Information Science & Engineering
B.N.M. Institute of Technology
Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering
SEMESTER-IV

Course Name: Database Management System
Course Code: 24ISE143

L: T: P: J	3:0 :1:1	CIA Marks: 50
Credits:	4	SEA Marks: 50
Hours/Week (Total)	5	SEA Duration: 03 Hours

Course Learning Objectives: The students will be able to

- 1 Understand fundamental concepts, terminology and application of databases, SQL and NoSQL
- 2 Design concepts and creation of relational databases using relation algebra.
- 3 Practice SQL programming through a variety of database problems.
- 4 Demonstrate the use of Normalization, concurrency and transactions in database.

Module-1: Database System Concepts, Data Modeling	No. of Hours	Blooms cognitive Levels
Databases and Databases Users: Characteristics of database Approach, Advantages of using the DBMS Approach. Database System Concepts and Architecture: Data Models-Schemas and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces. Data Modeling Using the Entity-Relationship (ER) Model: Entity Types-Entity sets- Attributes and Keys, Relationship types – Relationship Sets – Roles and structural Constraints, Weak Entity Types. Practical component: Draw ER Diagram for the following Databases using GitMind software. Order Database Library Database Bank Database	10	Apply

Module-2: Relational Data Model and Relational Algebra

Concept of relations, schema-instance distinction, keys, referential integrity and foreign keys, relational algebra operators: selection, projection, cross product, various types of joins, division, example queries, converting the database specification in E/R notation to the relational schema Practical component: Create Schema, insert at least 5 records in each table and add appropriate constraints for the following Library Database using ORACLE or MySQL DBMS under LINUX/Windows environment BOOK (Book_id, Title, Publisher_Name, Pub_Year) BOOK_AUTHORS (Book_id, Author_Name) PUBLISHER (Name, Address, Phone) BOOK_COPIES (Book_id, Branch_id, No-of_Copies) BOOK_LENDING (Book_id, Br_id, Card_No, Date_Out, Due_Date) LIBRARY_BRANCH (Branch_id, Branch_Name, Address) Write SQL queries to 1. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each branch, etc. 2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2020 to Jun 2022. 3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.	10	Apply
--	----	-------

Module-3: SQL

Basic SQL: SQL Data Definition and Data Types, Specifying Constraints in SQL, Basic Retrieval Queries in SQL, INSERT – DELETE and UPDATE Statements in SQL, Additional features in SQL

More SQL: Complex Queries, Triggers, Views and Schema Modification: Complex SQL Retrieval Queries, Specifying Constraints as Assertions and actions as Triggers, Views (Virtual Tables) in SQL.

Practical component:

Create Schema, insert at least 5 records for each table and add appropriate constraints for the following Order Database using ORACLE or MySQL DBMS under LINUX/Windows environment.

SALESMAN (Salesman_id, Name, City, Commission)

CUSTOMER (C_id, Cust_Name, City, Grade, Salesman_id)

ORDERS (Ord_No, Purchase_Amt, Ord_Date, C_id, S_id)

Write SQL queries to

1. Count the customers with grades above Bangalore's average.
2. Find the name and numbers of all salesman who had more than one customer.
3. List all the salesman and indicate those who have and don't have customers in their cities (Use UNION operation.)
4. Create a view that finds the salesman who has the customer with the highest order.

10

Apply

Module-4: Functional Dependencies and Normalization

Basics of Functional Dependencies and Normalization for Relational Database: Functional Dependencies, Armstrong's axioms for FD's, Equivalent Decompositions, closure of a set of FDs, minimal covers, Normal forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce- Codd Normal Forms [BCNF]

Practical component:

Create Schema, insert at least 5 records for each table and add appropriate constraints for the following Company Database using ORACLE or MySQL DBMS under LINUX/Windows environment.

EMPLOYEE (SSN, Name, Address, Sex, Salary, Super SSN, D No)

DEPARTMENT (D No, D Name, Mgr. SSN, Mgr. Start Date)

DLOCATION(D No,D Loc)

PROJECT (P No, P Name, P Location, D No)

WORKS_ON(SSN, P No, Hours)

Write SQL queries to

1. Make a list of all project numbers for projects that involve an employee whose last name is 'Scott', either as a worker or as a manager of the department that controls the project.
2. Show the resulting salaries if every employee working on the 'IoT' project is given a 10 percent raise.
3. Find the sum of the salaries of all employees of the 'Accounts' department, as well as the maximum salary and the average salary in this department.

10

Analyze

Module-5: Transaction Processing, Concurrency Control, NoSQL		
Introduction to Transaction Processing –Introduction to Transaction Processing, Desirable Properties on Transactions (ACID)	10	Analyze
Concurrency Control Techniques: Transactions and Schedules, Serializability, Precedence Graphs, Concurrency, Lock Based Protocols: 2PL, Strict 2PL Protocols, Deadlocks - Detection and Prevention		
NoSQL: SQL v/s NoSQL, The Emergence of NoSQL, BASE Properties, Data Models: Relationships, Graph Database, Schema less Database.		

Course Outcomes: After completing the course, the students will be able to	
24ISE143.1	Apply DBMS architecture principles, ER modeling, relational algebra, SQL queries, normalization techniques, and transaction management concepts to design and implement complete database solutions for real-world applications.
24ISE143.2	Apply conceptual data modeling, relational schema design, advanced SQL operations, normalization rules, and concurrency control mechanisms to create consistent, efficient, and reliable database systems.
24ISE143.3	Analyze database requirements using ER models, relational mappings, SQL query behavior, functional dependencies, and transaction schedules to identify design flaws and optimize database performance.
24ISE143.4	Analyze relational databases by integrating ER modeling, SQL processing, normalization theory, transaction processing, and NoSQL characteristics to ensure data integrity, scalability, and consistency.
24ISE143.5	Analyze and evaluate database system designs by combining data modeling, relational and non-relational query techniques, normalization strategies, concurrency control protocols, and transaction management for appropriate real-world deployment.

Reference Books
1. Fundamentals of Database Systems, Ramez Elmasri, Shamkant B. Navathe, 8th Edition, 2023, Pearson Education, ISBN-13: 978-0137340530. 2. Database System Concepts, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 8th Edition, 2020, McGraw-Hill Education, ISBN-13: 978-9390319108. 3. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pramod J. Sadalage, Martin Fowler, 2nd Edition, 2019, Addison-Wesley Professional, ISBN-13: 978-0135182646.


 Head of the Department
 Dept. of Information Science & Engineering
B.N.M. Institute of Technology
 Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering
SEMESTER- IV

Course Name: Design and Analysis of Algorithms

Course Code: 24ISE144

L:T:P:J	3:0:2:0	CIE Marks:	50
Credits:	4	SEE Marks:	50
Total Number of Lecture Hours	50	SEE Duration:	3 Hours

Course objectives: This course will enable students to:

1. Apply appropriate algorithm design paradigms such as brute force, divide and conquer, greedy, dynamic programming, backtracking, and branch and bound to solve standard computational problems across different domains.
2. Implement and analyze classical algorithms for searching, sorting, graph processing, and optimization problems, and inspect their time and space efficiency using asymptotic notations, recurrence relations, and empirical performance analysis.
3. Compare and evaluate multiple algorithmic solutions for the same problem to determine their efficiency, scalability, and suitability under practical constraints.
4. Evaluate algorithmic approaches for computationally hard problems by understanding complexity classes such as P, NP, NP-Complete, and NP-Hard, and by selecting feasible solution strategies.

Module 1

Teaching
Hours

Blooms
Cognitive
Levels

Introduction: Notion of algorithm, Fundamentals of Algorithmic Problem Solving, Fundamentals of the Analysis of Algorithmic Efficiency: Analysis frame work, Asymptotic Notations and Basic Efficiency Classes, Mathematical Analysis of Non-recursive and Recursive Algorithms.

Practical Programs not limited to :

1. Implement the following concepts:
 - a. Linear Search
 - b. To find the maximum and minimum from a given list of n elements using Brute Force Method.
2. There are 5 books in the shelf, find the number of ways to select 3 books from 5 books on the shelf using the NCR with recursion.
3. Find the next three terms of the sequence 15, 23, 38, 61, ... Fibonacci series of the given number using recursion.

10

Apply

Module 2

Brute Force: Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching

Divide and Conquer: General method, Binary search, Recurrence equation for divide and conquer, Finding the Maximum and Minimum, Merge sort, Quick sort, Strassen's matrix multiplication. Decrease and Conquer Approach: Topological Sort.

Practical Programs:

- 1. Implement the Selection sort algorithm.**
2. Implement Bubble sort algorithm.
3. Implement the Sequential Search algorithm.
4. Write a program to search a key in a given set of elements using Binary search method and find the time required to find the key.
5. Write a program to find Maximum and Minimum using divide and conquer technique and find the time required to find the elements.
- 6. Sort a given set of elements using Merge Sort method and determine the time required sort the elements. Plot a graph of number of elements versus time taken. Specify the time efficiency class of this algorithm.**
7. Sort a given set of elements using Quick Sort method and determine the time required sort the elements. Plot a graph of number of elements versus time taken. Specify the time efficiency class of this algorithm.
- 8. Implement Topological sort using source removal method find the time required to sort the elements.**

Understand and explain fundamental algorithmic paradigms such as brute force, divide and conquer, greedy method, dynamic programming, and backtracking.

2. Analyze the efficiency of algorithms using asymptotic notations and mathematical techniques for both recursive and non-recursive solutions.

3. Apply algorithm design strategies to solve classical computational problems including searching, sorting, shortest paths, minimum spanning trees, and optimization problems.

10

Apply

4. Implement algorithmic solutions for real-world scenarios and evaluate their time and space complexity using experimental and theoretical methods. 5. Demonstrate the ability to design and develop efficient algorithms for NP-complete problems using techniques like backtracking and branch-and-bound, while understanding the complexity classes P, NP, NP-complete, and NP-hard.		
Module 3		
Greedy Method: General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines. Minimum cost spanning trees: Prim's Algorithm, Kruskal's Algorithm. Single source shortestpaths: Dijkstra's Algorithm. Optimal Tree problem: Huffman Trees and Codes. Practical Programs: 1. Implement Coin Changing problem method and find the time required. 2. Implement Job Sequence problem using Greedy method. 3. Implement Prim's algorithm and Find Minimum Cost Spanning Tree of a given connected undirected graph. 4. Implement Kruskal's algorithm and Find Minimum Cost Spanning Tree of a given connected undirected graph. 5. Implement Dijkstra's algorithm find shortest paths to other vertices from a given vertex in a weighted connected graph.	10	Analyze
Module 4		
Dynamic Programming: General method with Examples, Multistage Graphs. Transitive Closure: Warshall's Algorithm, All Pairs Shortest Paths: Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem, Bellman-Ford Algorithm, Travelling Sales Person problem, Reliability design. Practical Programs: 1. Implement all-pairs shortest paths problem using Floyd's algorithm. 2. Implement all-pairs shortest paths problem using Warshal's algorithm. 3. Implement 0/1 Knapsack using Dynamic Programming. 4. Implementation of Bellman Ford Algorithm using a directed graph. 5. Implement Travelling Sales man problem using Dynamic Programming.	10	Apply
Module 5		
Backtracking: General method (T2:7.1), N-Queens problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles. Branch and Bound: Assignment Problem, Travelling Sales Person problem, 0/1 Knapsack problem: LC Branch and Bound solution, FIFO Branch and Bound solution. NP- Complete and NP-Hard problems: Basic concepts, nondeterministic algorithms, P, NP, NP-Complete and NP-Hard classes	10	Analyze

Practical Programs:

1. Implementation of N Queen Problem using Backtracking technique.
2. **Implementation of SUM-SUBSET Problem.**
3. Design and implement to find all Hamiltonian Cycles in a connected undirected Graph G of n vertices using the backtracking principle.
4. **Implementation Assignment Problem using Backtracking technique.**
5. **Implementation of Travelling Sales Man Problem using Branch and Bound method.**
6. **Implementation of Knapsack problem using Branch and Bound method.**

COs	Statement	Bloom's Cognitive level	POs/PSOs
24ISE144.1	Apply suitable algorithmic paradigms to solve problems related to sorting, searching, graph processing, greedy, divide and conquer, dynamic programming techniques and optimization using structured programmatic solutions.	Apply	PO 1,2,3,4,5,11 PSO 1,2
24ISE144.2	Analyze the computational complexity of algorithms using asymptotic notations, recurrence relations, and experimental analysis, and classify them into appropriate efficiency classes.	Analyze	PO 1,2,4,5,11 PSO 1,2
24ISE144.3	Compare alternative algorithmic strategies for the same problem and evaluate their trade-offs in terms of time complexity, space usage, and practical applicability	Evaluate	PO 1,2,3,4,5,11 PSO 1,2
24ISE144.4	Evaluate algorithmic solutions for NP-Complete and NP-Hard problems and justify the use of approximation, backtracking, or branch and bound techniques based on problem constraints.	Evaluate	PO 1,2,3,4,5,11 PSO 1,2

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 4th Edition, MIT Press, April 2022.
2. Introduction to the design and analysis of algorithms, by Anany Levitin, 3rd Edition, Pearson Education, 2021.
3. Fundamentals of Computer Algorithms, Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, 2nd Edition, Universities Press, 2008
4. The Art of Computer Programming, Donald E. Knuth, Addison-Wesley, Volume 4B (2023)
5. Algorithms Unlocked, Thomas H. Cormen, MIT Press, 2013

Web Links:

<https://www.cs.duke.edu/courses/fall08/cps230/Book.pdf>

https://www.tutorialspoint.com/design_and_analysis_of_algorithms/design_and_analysis_of_algorithms_tutorial.pdf


 Head of Department
 Dept. of Information Science & Engineering
 B.N.M Institute of Technology
 Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science & Engineering
SEMESTER: IV

Course Name: Robotic Process Automation

Course Code: 24ISE145

L:T:P:J	0:0:2:2	CIA	: 50
Credits:	02	SEA	: 50
Hours:	24	SEA Duration	: 03 Hours

Course Learning Objectives: The learners will be able to

1	Apply UiPath RPA tools to develop and execute end-to-end automation workflows for data processing, UI interaction, and resource management tasks.
2	Apply sequences, flowcharts, control flow structures, debugging tools, and exception-handling techniques to implement structured and reliable automation solutions.
3	Apply variables, collections, data tables, string manipulation methods, and regular expressions to perform effective data processing within UiPath workflows.
4	Apply selectors, descriptors, synchronization activities, and dynamic targeting techniques to execute dependable and scalable UI automation tasks.
5	Apply version control systems, modular workflow design, and UiPath Orchestrator features to manage, organize, and deploy automation projects following standard RPA development practices.

Part A

Module-1:	No. of hours	Blooms Cognitive Levels
Introduction to Robotic Process Automation (RPA) concepts, tools and fundamentals of implementation Describe Robotic Process Automation (RPA) and its benefits • Know UiPath Business Automation Platform • Describe how the UiPath core RPA components (Studio, Orchestrator and Robot with Assistant) work together • Differentiate between the two types of UiPath robots—attended and unattended • Locate and describe the key components of the UiPath Studio user interface • Differentiate between modern vs classic design • Explain and create and configure variables in an automation project • List the most common data types used in UiPath Studio and explain the conversion methods used for them • Explain and create arguments in an automation project • Use the Invoke Workflow File Activity to chain workflow execution and pass data through arguments • Apply Automation best practices for when working with variables, arguments, and workflow files • Explain and create global constants and global variables in your automation projects • Explain what arrays are and why they are useful	05	Apply

Module-2:	No. of hours	Blooms Cognitive Levels
Control flow, common RPA implementations, Exception handling and Debugging Determine when to use Sequences and Flowcharts • Explain and work with control flow statements for decisions and iterations • Apply Excel and Workbook activities to build simple automations • Differentiate between different types of exceptions and apply best practices for error and exception handling • Use file and folder activities to perform key tasks like selecting, creating, deleting, moving and renaming files and folders • Use the UiPath Studio integration capabilities to access a Gmail account, retrieve, and work with emails • Use Microsoft Office 365 activities to build an automation process in Studio • Use the appropriate activities, from the Package, to extract large text segments or single piece of information from PDF files • Understand the available debugging actions	05	Apply
Module-3:		
Data Manipulation Explain and apply data manipulation • Use the most common VB.NET methods to manipulate string variables • Use the RegEx builder in UiPath Studio to perform complex string manipulation • Use string activities in Studio to perform string manipulation Explain the basic concepts of Lists and Perform data manipulation on Lists • List the features of Invoke and String Conversion methods • Explain the differences between Arrays and Lists • Explain dictionary variables and perform data manipulation using dictionary variables • Explain Data table variable and use most common activities and methods to create data tables in Studio • Differentiate between a worksheet and a data table	05	Apply
Module-4:		
UI Automation, Descriptors and Selectors Explain the importance of synchronizing activities in automation workflows • Use the Check App State activity and Verify Execution feature • Use the Pick Branch activity to create branching logic in workflows • Explain the different targeting methods used in UI automation and understand their characteristics • Validate target elements to ensure the accuracy of descriptors • Fine-tune descriptors by utilizing the 'Dynamic Text Target' option, wildcards, variables, and making adjustments to enhance image accuracy • Explain the structure and type of selectors in the context of web development • Demonstrate the functionality and purpose of the Property Explorer tool during editing selectors.	05	Apply
Module-5:		
Version Control System, Orchestrator resources and Project organization Describe the Orchestrator capabilities and entities • Differentiate between the tenant context and the folder context Use Orchestrators resources directly in Studio • Explain the benefits and challenges of using version control systems, Explain and use basic GIT features for version control • Choose the suitable project layout for an automation process • Break down complex automation project into functional workflows that can be developed separately • Identify the benefits of utilizing best practices for project organization	04	Apply

Choose the suitable project layout for an automation process • Break down complex automation project into functional workflows that can be developed separately • Identify the benefits of utilizing best practices for project organization		
--	--	--

PART – B (Mini projects)

The List of Possible Projects

1. MOVING FILES FROM SOURCE FOLDER TO DESTINATION FOLDER

Task: The aim is to automate the process of moving files from the source folder to the destination folder.

2. WEB AUTOMATION

Task: The aim is to scrape data from a website and store it in a .csv File.

3. EMAIL AUTOMATION

Task: The aim is to save the attachments of the top 30 emails which have a keyword in the subject line. The attachments will be stored in a particular folder.

4. EXCEL AUTOMATION

Task: The aim is to fill a google form from the data stored in a .csv file automatically.

5. PDF AUTOMATION

Task: The aim is to extract text and images from a PDF File and store the output in a Message Box/Textile.

6. IMAGE & TEXT AUTOMATION

Task: Consider an application is running in a virtual environment. Now, once we enter the account number, amount, choose transaction type, a transaction ID is automatically generated. Our task is to display the Transaction ID in a message box once the transaction is successfully completed

7. AUTOMATED CUSTOMER SUPPORT EMAILS (WITH INSTRUCTIONS)

Task: The aim is to send automated replies to emails that have text in their subject line. For example, if a subject line has the word 'refund,' your solution would send an automated reply to it accordingly.

8. TO FIND OUT WHETHER A NUMBER IS EVEN OR ODD.

Task: The aim is to find out whether a number is even or odd using the different components of UiPath Studio and Dashboard of UiPath

9. EXTRACTING DATA FROM AN E-COMMERCE WEBSITE

Task: Is to design automation which reads the subject line of an email and performs a custom search. So, For example, if my subject line says LG TV then, the designed automation must search for LG TVs' in Flipkart and extract the name, URL, and the price.

10. HOW TO AUTOMATE TASKS IN A VIRTUAL ENVIRONMENT?

Task: Automating Task on a Desktop Application:

Consider a scenario where you want to open a Notepad in the virtual environment and want to type something in it. How will you automate this task?

****Projects are not limited**

Course Outcomes:

After studying these laboratory programs using UiPath, students will be able to

CO1: Apply UiPath RPA tools to design and execute complete automation workflows involving data processing, UI interaction, and resource management.

CO2: Apply sequences, flowcharts, control flow activities, debugging tools, and exception-handling techniques to implement automation processes in UiPath.

CO3: Apply variables, collections, data tables, and advanced string and regular expression operations within UiPath workflows to manipulate and transform data.

CO4: Execute UI automation tasks by applying selectors, descriptors, synchronization activities, and dynamic targeting methods to build reliable automation solutions.

CO5: Implement version control, project organization, and deployment of automation solutions using Git, Orchestrator, modular workflows, and standard RPA development practices.

Reference Books:

1. The Robotic Process Automation Handbook: A Guide to Implementing RPA Systems, Tom Taulli, 1st Edition, February 2020, Apress, ISBN-13: 978-1484257296.
2. Learning Robotic Process Automation: Create Software Robots and Automate Business Processes with the Leading RPA Tool – UiPath, Alok Mani Tripathi, Second Edition, 2018, Packet Publishing, ISBN-13: 978-1838988341.
3. Introduction to Robotic Process Automation: A Primer, Frank Casale, Rebecca Dilla, Heidi Jaynes, Lauren Livingston, 1st Edition, 2015, Institute of Robotic Process Automation, ISBN: Not available.
4. Robotic Process Automation: Guide to Building Software Robots, Automate Repetitive Tasks & Become an RPA Consultant, Richard Murdoch, 1st Edition, 2018, Independently Published, ISBN-13: 978-1983036835.
5. Robotic Process Automation Tools, Process Automation and Their Benefits: Understanding RPA and Intelligent Automation, Srikanth Merianda, 1st Edition, 2018, CreateSpace Independent Publishing Platform, ISBN-13: 978-1720626077.

Web links

- <https://docs.uipath.com/studio/standalone/2023.10/user-guide/install-studio>
- <https://docs.uipath.com/>
- <https://community.uipath.com/>
- <https://forum.uipath.com/>
- [https://academy.uipath.com/learning-plans/automation-developer-associate-training-\(v2023.10\)](https://academy.uipath.com/learning-plans/automation-developer-associate-training-(v2023.10))
- <https://www.uipath.com/learning/certification>


Head of the Department
Dept. of Information Science & Engineering
B.N.M Institute of Technology
Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Semester: III/IV

COURSE: CONSTITUTION OF INDIA, PROFESSIONAL ETHICS, IKS and UHV

Course Code: 24CIP47	L:T:P:J: 0:2:0:0	CIA Marks: 50
Credits:	1	SEA Marks: 50
Hours:	15 hrs	SEA Duration: 2Hr

Course Learning Objectives: The students will be able to

1	know the fundamental political codes, structure, procedures, powers, and duties of Indian government institutions, fundamental rights, directive principles, and the duties of citizens
2	know the Indian top civil service positions and the exams conducted by UPSC and SPSC for the same
3	Understand engineering ethics and their responsibilities; identify their individual roles and ethical responsibilities towards society.

MODULE 1: Foundations of the Indian Constitution and Governance	RBT	Hrs
--	------------	------------

Necessity, relevance, and historical background of the Constitution; Constituent Assembly – role and functioning; Preamble – vision, philosophy, and objectives; Salient features of the Indian Constitution; Parliamentary vs. Presidential systems
Federal features of Indian polity

1,2,3

3

MODULE 2: Rights, Duties, and Policy Framework	RBT	Hrs
---	------------	------------

Fundamental Rights – scope, reasonable restrictions, and landmark cases; Directive Principles of State Policy – classification and implementation; Fundamental Duties – legal status and significance; Union Executive: President, Vice-President, Prime Minister & Cabinet; State Executive: Governor, Chief Minister & Cabinet; Bicameral and unicameral legislatures.

1,2,3

3

MODULE 3: Legislature, Judiciary, and Emergency Provisions	RBT	Hrs
---	------------	------------

Parliament: Lok Sabha & Rajya Sabha – composition, powers, sessions; Legislative procedure: quorum, language, joint session; Judiciary – Supreme Court & High Courts: structure, powers, and jurisdiction; Judicial review and judicial activism; Constitutional Amendments – methods and significant amendments; Emergency provisions – types, impact, and case studies.

1,2,3

3

MODULE 4: Elections, Institutions, and Public Administration	RBT	Hrs
---	------------	------------

Election Commission of India – structure, powers, conduct of elections; Electoral reforms and use of EVMs; Constitutional bodies: UPSC, SPSC, GST Council – functions; Non-Constitutional bodies: CIC, SIC – role and relevance; Role of democratic and public institutions in a constitutional democracy.

1,2,3

3

N. Shrinivas
Professor & HOD
Department of Humanities
B.N.M. Institute of Technology
Bangalore-560 070

G. S. 109
Additional Director & Principal
BNM Institute of Technology
Bangalore-560 070

MODULE 5: Professional Ethics, Universal Human Values, and Indian Knowledge System	RBT	Hrs
Scope and relevance of Engineering and Professional Ethics; Responsibilities of engineers – legal, moral, and social; Risk, safety, liability, and ethical dilemmas; Intellectual Property Rights (IPR) – overview; Indian Knowledge System – components, contributions to science and sustainability; Universal Human Values – truth, peace, non-violence, harmony; Role of education in value inculcation Integrating values in professional and personal life Case studies from daily life and professions.	1,2,3	3

Course outcome: On completion of this course, students will be able to,
CO1: Have constitutional knowledge and legal literacy.

CO2: Have knowledge on All India Services and State Civil Services.

CO3: Understand Engineering and Professional Ethics and responsibilities of Engineers.

CO4: Have knowledge on Universal Human Values

CO5: Understand Indian Knowledge System

Reference Books

Suggested Learning Resources:

1. Title of the Book - Indian Polity

Name of the Author - M Lakshmikanth

Name of the Publisher-Mc Graw Hill Education

Edition and Year- 2019

2. Title of the Book - Engineering Ethics

Name of the Authors - M. Govindarajan, S.Natarajan, V.S. Senthilkumar

Name of the Publisher- Prentice-Hall

Edition and Year-2004

3. Durga Das Basu (DD Basu): "Introduction to the Constitution on India", (Students Edition.)

Prentice –Hall EEE, 19th / 20th Edn., (Latest Edition) or 2008.

4. Shubham Singles, Charles E. Haries, and Et al : "Constitution of India and Professional Ethics" byCengage Learning India Private Limited, Latest Edition – 2018.

5. M.Govindarajan, S.Natarajan, V.S.Senthilkumar, "Engineering Ethics", Prentice –Hall of IndiaPvt. Ltd. New Delhi, 2004

6. M.V.Pylee, "An Introduction to Constitution of India", Vikas Publishing, 2002.

7. Latest Publications of NHRC - Indian Institute of Human Rights, New Delhi.

N. Senthil Kumar
Professor & HOD
Department of Humanities
BNT Institute of Technology
070

[Signature]

[Signature]
Additional Director & Principal
BNM Institute of Technology
Bangalore-560 070

Web Links and Video Lectures

www.unacademy.com/lesson/future-perfect-tense/YQ9NSNQZ <https://successesacademy>

Question paper pattern for SEA and CIA.

- The SEA question paper will be set for 50 marks and the pattern of the question paper will be objective type (MCQ). Students should score 40% marks to pass.
- The CIA question paper will be set for 50 marks and the pattern of the question paper will be objective type (MCQ). Students should score 40% marks to pass.

Final Marks = CIA + SEA = 50+50 = 100 Marks

Class Internal Assessment

IA1	Objective type questions 50Marks	Sum of IA1 and SEA= 100 Marks
SEA	Objective type questions 50Marks	
	Total CIA	100 Marks

Final Marks = CIA + SEA = 50+50 = 100 Marks

W. Shreeharsha

Professor & HOD
Department of Humanities
BNM Institute of Technology
Bangalore-560 070

[Signature]

[Signature]
01/09

Additional Director & Principal
BNM Institute of Technology
Bangalore-560 070

For Internal Communication

Syllabus

Course Name: Soft Skill – II

Sub Code: 24SFT148

Semester: IV Semester

Year of Study: 2025-26

Total Number of Contact Hours: 30

Credits: 1

Total marks (CIA only): 100

Course Objectives: This course will enable students to

- understand the techniques to solve aptitude problems under logical & quantitative sections.
- develop thinking ability to solve complex problems.
- to solve company simulated aptitude papers related to campus recruitments.

Module 1 (Logical Reasoning)	Ages & Blood Relation - Generation Tree, Family Tree Problems, Statement Based Questions, Coded Blood Relation Question, Blood Relation Based Puzzles.
	Seating Arrangement - Single or Double rows facing each other or away from each other in the same direction Circular Seating Arrangement · Uni- & Bi-directional problems on · Circular, Square, Rectangular, Hexagonal tables
	Distance & Direction - Distance and Displacement between any two points as well as puzzles based on that, Concept of Shadows.
	Clocks & Calendars - Understanding concepts and basic formula along with solving different types of problems.
	Probability - Single event probability, multi event probability, independent events and dependent events, mutually exclusive events, non-mutually exclusive events, combination method for finding the outcomes.
	Analogies - Drawing similarity between different but sufficiently similar events, situations, or circumstances using tips and tricks, find the odd one out, rule detection,
	Ratio and Proportion · Simple Ratios, Compound Ratios · Comprehend and Dividend · Direct & Indirect Proportions · Problems on ages,

Module 2 (Quantitative Aptitude)	Mixtures & Alligation – technique, general representation, the straight-line approach, application of weighted allegation methods in problems involving mixtures, application of allegation on situations other than mixtures problems & Coin problems.
	Data Interpretation – Simple arithmetic, rules for comparing fractions, calculating (approximation) fractions, shortcut ways to find the percentages, Classification of data tables, Bar Graph, Tabular Form, Line Chart, case let Form. Pie Chart, Radar/Web, combination of graphs, combination of graphs and tables. and Missing Data Interpretation.
	Simple interest and Compound Interest - Simple Interest, Basic Difference b/w both the Interests. Mixed interest, CI with a Fractional Rate, to find Installments of both SI and CI.
	- Relative speed, Average speed, Problems on Races when somebody gives a start to other Person. Persons moving opposite & same direction in circular track, problems on train, Boat & Stream,
	Permutation & Combination - Arrangement, law of addition & multiplication, factorial Speed, Time & Distance function, concept of step arrangement, permutation of similar things, circular things, find the rank of word in dictionary. Concepts of selections using combination.
	Time & Work - Partial work done problems Work done on Alternate days. Problems related to efficiency. Division of Wages According to Work. Work & wages, pipes & cisterns

Course Outcome: (CO)

By end of the course the students will be able to:

1. identify and categorize problems under various sections of aptitude, reasoning & quantitative.
2. apply the appropriate techniques to solve reasoning related problems.
3. apply the appropriate techniques to solve quantitative related problems.
4. analyze the problem and solve it within the allocated time span.
5. implement out of the box solutions for complex problems.
6. answer company simulated aptitude papers related to campus recruitments.

CO-PO/PSO Mapping:

CO No.	Statement	Bloom's Cognitive level	POs
1	identify and categorize problems under various sections of aptitude, reasoning & quantitative	Analyze	PO1, PO2 & PO11
2	apply the appropriate techniques to solve reasoning related problems	Apply	PO1 & PO2
3	apply the appropriate techniques to solve quantitative related problems	Apply	PO1 & PO2

4	analyze the problem and solve it within the allocated time span	Analyze	PO1 & PO2
5	implement out of the box solutions for complex problems	Apply	PO1, PO2 & PO11
6	answer company simulated aptitude papers related to campus recruitments	Apply	PO1 & PO2

Strength of CO Mapping to PO/PSOs:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
1	2	2									2
2	2	2									
3	2	2									
4	2	2									
5	2	2									2
6	2	2									

List of Innovative teaching methods (if any):

- Quiz – logical questions, GK, crossword, etc.
- Problems based learning through discussion.
- Enquiry and evaluation-based learning.
- Blended learning (Classroom teaching, Presentation based teaching and activity-based learning)

Teaching methodologies:

- Classroom Teaching
- Quizzes and Discussions
- Mock Sessions
- Content related activities
- Organizing group wise discussions connecting to placement activities.
- Hands-on practice on topics.

Assessment processes:

CIA (100)	Components	Description	Maximum Marks to be Scored	Minimum Marks to be Scored
	Daily Assignments	Daily assignments should be conducted on the topics completed every day. After training weekly tests to be given	50	20

	Final Assessment	Final MCQ's Test to be conducted in College	50	20
	Total Marks for the Course		100	40

Assessment plan:

CO (Cognitive Level)	Daily Assessments	Final Assessment
CO-1 (Analyze)	Analyze	Analyze
CO-2 (Apply)	Apply	Apply
CO-3 (Apply)	Apply	Apply
CO-4 (Analyze)	Analyze	Analyze
CO-5 (Apply)	Apply	Apply
CO-6 (Apply)		Apply

Reference Books:

1. Quantitative aptitude by R S Agarwal, S Chand Publications
2. Verbal and nonverbal Reasoning by R S Agarwal, S Chand publications
3. Aptitude Book by Rakesh Sharma

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science & Engineering

SEMESTER – V

Course Name: Software Engineering Project Management and Finance

Course Code: 24ISE151

L:T:P:J	2:2:0:0	CIA Marks	50
Credits:	3	SEA Marks	50
Hours:	40	Exam Hours	03

Course Learning Objectives:

This course will enable students to

- Apply fundamental software engineering concepts, including software processes, requirements engineering techniques, and professional ethical principles, in the development and evaluation of software systems.
- Apply standard models and techniques such as UML, software design principles, testing strategies, agile methodologies, and project planning in real-world software development contexts.
- Apply project management techniques and modern tools to plan and monitor software development activities, including risk identification and cost estimation.

	Number of Hours	Bloom's Level
Module-1		
Introduction: Software Crisis, Need for Software Engineering. Professional Software Development, Software Engineering Ethics. Case Studies. Software Processes: Models: Waterfall Model, Incremental Model and Spiral Model, Process activities. Requirements Engineering: Requirements Engineering Processes, Functional and non-functional requirements. The software Requirements Document, Requirements Specification, Requirements validation, Requirements Management Self-Study Component: Comparative Study, Research Paper Review	8	Apply
Module-2		
System Models: Structural models, Behavioral models, UML modeling using StarUml tool. Design and Implementation: Introduction to RUP, Design Principles Software Testing: Development testing, Test-driven development, Release testing. User testing. Self-Study Component: UML Modeling Exercise, Tool Exploration, Test Plan Creation	8	Apply
Module-3		
Project management: Risk management, Managing People, Teamwork. Project Planning: Software pricing, Plan-driven development, Project scheduling: Estimation techniques, Quality management: Software quality, Reviews and inspections, Software measurement and metrics, Software standards Self-Study Component: Case Study Analysis, Estimation Practice, Metric Evaluation	8	Apply
Module-4		
Agile Software Development: Coping with Change, The Agile Manifesto: Values and Principles. Agile methods: SCRUM (Ref —The SCRUM Primer, Ver 20.) and Extreme Programming. Plan-driven and agile development. Agile project management, Scaling agile methods. Self-Study Component: Agile vs. Plan-driven Essay, SCRUM Simulation, XP Practice.	8	Apply

Module-5

How to Manage Project Finances-Cost estimating: Work Breakdown Structure, Cost budgeting: Cost Aggregation, Reserve Analysis, Parametric estimating, Infrastructure and overheads, Cost control: Change Control, Resource Management
Performance Measurement and Analysis- Cost Variance, Earned Value, Schedule Variance, Cost Performance Index, Schedule Performance Index.
Forecasting, Introduction of Tools to manage project Finances- TouchBase Project Financials
Self-Study Component: Budget Planning Exercise, Performance Dashboard, Tool Demo Report.

8

Apply

Course outcomes:

CO1:Apply software engineering principles, software process models, and requirements engineering techniques to analyze user needs and prepare software requirement specifications for real-world applications.

CO2:Apply UML modeling, design principles, and software testing methodologies to design, implement, and validate software systems using appropriate development tools and techniques.

CO3:Apply project planning, estimation, risk management, quality management, and software metric: to develop effective software project plans and monitor project performance.

CO4

Apply Agile and Scrum practices along with project financial management techniques to manage software development projects, track progress, control costs, and support continuous improvement.

Reference Books:

1. *Software Engineering* by Ian Sommerville, published by Pearson Education, is available in its 9th Edition released in 2012.
2. *Software Engineering: A Practitioner's Approach* by Roger S. Pressman is published by Tata McGraw-Hill and is available in its 7th Edition.
3. *An Integrated Approach to Software Engineering* by Pankaj Jalote is published by Wiley India; the specific edition and year were not mentioned.
4. *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)*, published by the Project Management Institute, is now available in its 7th Edition, released in August 2021.

Web links and Video Lectures:

- <https://nptel.ac.in/courses/106105182>
- <https://nptel.ac.in/courses/106105218>
- <https://nptel.ac.in/courses/110107430>



Head of the Department
Dept. of Information Science & Engineering
B.S.M. Institute of Technology
Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science & Engineering

SEMESTER: V

Course Name: Automata Theory and Computability

Course Code: 24ISE152

L:T:P:J	2:2:0:0	CIA	:	50
Credits:	03	SEA	:	50
Total Number of Lecture Hours	40	SEA Duration	:	03 Hours

Course Learning Objectives: The students will be able to

1	Apply the fundamental concepts of automata theory, formal languages, and computational models such as finite automata, pushdown automata, and Turing machines.
2	Develop grammars and recognizers for various classes of formal languages, and to analyze their properties and relationships using formal mathematical tools.
3	Analyze how Nondeterministic Pushdown Automata process strings, and differentiate between the formal mechanisms of acceptance by final state versus acceptance by empty stack to determine their equivalence.
4	Analyze the ability to evaluate the decidability and complexity of computational problems, and to prove key theoretical results in automata and computation theory.

Module-1: Introduction to the Theory of Computation:

**No. of
hours**

**Blooms
cognitive
Levels**

Three Basic Concepts: Languages, Grammars and Automata, Some applications. Finite Automata: Deterministic Finite Accepters, Deterministic Accepters and Transition Graphs, Languages and DFA's Regular Languages, Nondeterministic Finite Accepters: Definition of a Nondeterministic Acceptor, Why Nondeterminism ? Equivalence of Deterministic and Nondeterministic Finite Accepters

8

Apply

Module-2: Regular Languages and Regular Grammars

Regular Expressions: Formal Definition of a Regular Expression, Languages Associated with Regular Expressions, Connection Between Regular Expressions and Regular Languages, Regular Expressions Denote Regular Languages, Regular Expressions for Regular Languages, Regular Expressions for Describing Simple Patterns.
Properties of Regular Languages: Closure Properties of Regular Languages Closure under Simple Set Operations, Closure under Other Operations,
Identifying Non-Regular Languages, Using the Pigeonhole Principle A Pumping Lemma.

8

Apply

Module-3: Context-Free Languages:

Context-Free Grammars, Examples of Context-Free Languages Leftmost and Rightmost Derivations, Derivation Trees, Relation Between Sentential Forms and Derivation Trees, Parsing and Ambiguity, Parsing and Membership, Ambiguity in Grammars and Languages, Simplification of Context-Free Grammars and Normal Forms: Methods for Transforming Grammars, A Useful Substitution Rule, Removing Useless Productions, Removing λ -Productions, Removing Unit-Productions, Two Important Normal Forms, Chomsky Normal Form, Definition of Greibach Normal Form.

8

Apply

Module-4: Pushdown Automata Nondeterministic Pushdown Automata: Definition of a Pushdown Automaton, The Language Accepted by a Pushdown Automaton, Pushdown Automata and Context-Free Languages: Pushdown Automata for Context-Free Languages, Context-Free Grammars for Pushdown Automata, A Pumping Lemma For Context-Free Language	8	Analyze
Module-5: Turing Machines and Undecidability Turing Machines: The Standard Turing Machine, Turing Machines as Language Accepters, Turing Machine with more complex storage: Multitape Turing Machine, Multidimensional Turing Machine, Nondeterministic Turing Machine, Universal Turing Machine, Turing Thesis, Linear Bounded Automata. Hierarchy of Formal Languages and Automata: Recursive and Recursively Enumerable Languages, the Chomsky Hierarchy. Limits of Algorithmic Computation: Some Problems that cannot be solved by Turing, Undecidable Problem for Recursively Enumerable Languages.	8	Analyze

Course Outcomes: After completing the course, the students will be able to

CO1: Apply the fundamental concepts of formal languages, grammar, finite automata, and Turing machines, and describe their role in recognizing different classes of languages.

CO2: Apply deterministic and nondeterministic automata, formulate regular expressions, and grammar, for given language descriptions, and implement Turing machines for specific language recognition tasks.

CO3: Analyze the internal mechanics of Nondeterministic Pushdown Automata (NPDA) by mathematically differentiating between acceptance by final state and acceptance by empty stack, and prove the structural equivalence of both acceptance criteria

CO4: Analyze and differentiate between language classes and computational models and examine the limits of computation through undecidability and the Chomsky hierarchy.

Reference Books:

1. *An Introduction to Formal Languages and Automata* by Peter Linz, 7th Edition, published by Jones & Bartlett Learning in 2022.
2. *Introduction to Automata Theory, Languages, and Computation* by John E. Hopcroft, Rajeev Motwani, and Jeffrey D. Ullman, 3rd Edition, published by Pearson Education in 2006.
3. *Introduction to the Theory of Computation* by Michael Sipser, 3rd Edition, published by Cengage Learning in 2012.
4. *Introduction to Languages and the Theory of Computation* by John C. Martin, 4th Edition, published by McGraw-Hill Education in 2010.
5. *Formal Languages and Automata Theory* by Basavaraj S. Anami and Karibasappa K. G., published by Wiley India in 2012.
6. *Formal Languages and Automata Theory* by C. K. Nagpal, published by Oxford University Press in 2012.

Web links and Video Lectures:

- <https://archive.nptel.ac.in/courses/106/106/106106049/>
- <https://archive.nptel.ac.in/courses/111/103/111103016/>
- <https://www.csa.iisc.ac.in/~dee`pakd/atc-2021/>


Head of the Department
Dept. of Information Science & Engineering
B.N.M Institute of Technology
Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

SEMESTER: V

Course Name: Data Science using Python

Course Code: 24ISE153

L:T:P:J	2:1:1:0	CIA : 50
Credits:	03	SEA : 50
Hours:	50	SEA Duration : 03 Hours

Course Learning Objectives: The students will be able to

1	To introduce the fundamentals of Python programming and essential data science libraries for performing data manipulation, analysis, and visualization.
2	To develop an understanding of descriptive statistics, data preparation, and exploratory data analysis techniques for gaining insights from data.
3	To enable students to implement supervised learning models and understand concepts like model validation, learning curves, and performance evaluation.
4	To familiarize students with regression and clustering techniques, including their application to real-world datasets and evaluation using statistical metrics.
5	To provide knowledge about recommender systems, including how to model user preferences and evaluate the effectiveness of recommendations.

Module-1:	No. of hours	Blooms cognitive Levels
Introduction, Toolboxes: Python, fundamental libraries for data Scientists. Integrated development environment (IDE). Data operations: Reading, selecting, filtering, manipulating, sorting, grouping, rearranging, ranking, and plotting. Program: 1,2,3	10	Apply
Module-2:		
Descriptive statistics, data preparation. Exploratory Data Analysis data summarization, data distribution, Outlier Treatment Programs: 4,5	10	Analyze
Module-3:		
Supervised Learning: First step, learning curves, training-validation and test. Learning models generalities, support vector machines, random forest. Examples Programs: 6,7	10	Apply
Module-4:		
Regression analysis, Regression: linear regression simple linear regression, multiple & Polynomial regression, Logistic Regression. Unsupervised learning, clustering, similarity and distances, quality measures of clustering, case study. Programs: 8,9,10	10	Apply

Module-5:		
Recommender Systems: Introduction, How do recommender system works?, Modelling user preferences, Evaluating recommenders, Practical case. Programs: 11	10	Apply

Course Outcomes: After completing the course, the students will be able to

CO1: Apply the foundational concepts of data science including data manipulation techniques, descriptive statistics, supervised and unsupervised learning algorithms, and recommender systems, along with their applications using appropriate Python libraries and tools.

CO2: Apply data manipulation techniques, exploratory data analysis, and statistical methods using Python to prepare, clean, and visualize datasets for machine learning tasks and predictive modeling.

CO3: Implement supervised and unsupervised learning algorithms including classification, regression, clustering, and recommender systems using Python libraries to solve real-world data science problems.

CO4: Analyze complex datasets by integrating data operations, statistical methods, machine learning models, regression techniques, and recommender systems to evaluate performance, identify insights, and support data-driven decision making.

Reference Books:

1. Introduction to Data Science: A Python Approach to Concepts, Techniques and Applications by **Laura Igual and Santi Seguí**, published by **Springer in 2024**.
2. Doing Data Science: Straight Talk from the Frontline by **Cathy O'Neil and Rachel Schutt**, published by **O'Reilly Media in 2013**.
3. The Data Visualization Workshop: A Self-Paced, Practical Approach to Transforming Your Complex Data into Compelling, Captivating Graphics by **Mario Döbler and Tim Großmann**, 1st Edition, published by **Packt Publishing in 2020**.
4. Mining of Massive Datasets by **Jure Leskovec, Anand Rajaraman, and Jeffrey D. Ullman**, 3rd Edition, published by **Cambridge University Press in 2020**.
5. Data Science from Scratch: First Principles with Python by **Joel Grus**, 2nd Edition, published by **O'Reilly Media in 2019**.
6. Data Visualisation: A Handbook for Data Driven Design by **Andy Kirk**, 3rd Edition, published by **SAGE Publications in 2024**.

Web links and Video Lectures:

- <https://www.knowledgehut.com/blog/data-science/eda-data-science>
- <https://towardsdatascience.com/data-exploration-understanding-and-visualization-72657f5eac41>
- <https://matplotlib.org/>


 Head of the Department
 Dept. of Information Science & Engineering
B.N.M. Institute of Technology
 Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science & Engineering

SEMESTER: V

Course Name: Computer Networks

Course Code: 24ISE154

L: T:P: J	2:2:2:0	CIA	: 50
Credits:	04	SEA	: 50
Total Number of Lecture Hours	50	SEA Duration	: 03 Hours

Course Learning Objectives: The students will be able to

1. Apply fundamental networking concepts and protocol architectures to model and interpret layered communication systems such as OSI and TCP/IP.
2. Apply error detection, data link control, and medium access control techniques to design and analyze reliable data transmission over communication networks.
3. Analyze network layer concepts and routing algorithms to configure logical addressing, routing, and packet forwarding in IP-based networks.
4. Apply transport and application layer protocols to evaluate end-to-end communication mechanisms and analyze performance aspects such as flow control, congestion control, and service reliability.

Module-1:	No. of hours	Blooms cognitive Levels
Introduction: Data Communications, Networks, Network Types, Networks Models: Protocol Layering, TCP/IP Protocol suite, OSI model. Introduction to Physical Layer: Data Rate Limits, Performance Digital to Digital Conversion- Line coding: polar, unipolar, Switching – Circuit switching, packet switching.	10	Apply
Module-2:		
Error Detection and Correction: Introduction, Block coding, Cyclic codes Data Link Layer: Data Link Control –DLC services: Framing, Flow and Error Control, Data link layer protocols: HDLC, Point to Point Protocol. Media Access Control – Random Access: CSMA/CD, CSMA/CA.	10	Apply
Module-3:		
Network Layer: IPv4Addresses:0 Address space, Classful Addressing, Classless Addressing, Dynamic Host Configuration Protocol (DHCP), Network Address Translation (NAT). IPv6 Addresses, Routing Algorithms: Link-State (LS) Routing Algorithm, The Distance-Vector (DV) Routing Algorithm.	10	Apply
Module-4:		
Transport layer: Multiplexing and Demultiplexing, Connectionless Transport UDP: UDP Segment Structure, UDP Checksum, Connection- Oriented Transport-TCP: The TCP Connection, TCP Segment Structure, Round-Trip Time Estimation and Timeout, Flow Control, TCP Connection Management, TCP congestion control.	10	Analyze

Module-5:			
Application Layer: The Web and HTTP: Overview of HTTP, Non- Persistent and Persistent Connections, HTTP Message Format, User- Server Interaction Cookies, Web Caching, The Conditional GET. Electronic Mail in the Internet: SMTP, Comparison with HTTP, Mail Access Protocols. DNS—The Internet's Directory Service: Services Provided by DNS, Overview of How DNS Works, DNS Records and Messages,		10	Analyze

Course Outcomes: After completing the course, the students will be able to	
CO 1:	Apply fundamental concepts of data communication, network models, and physical layer techniques to design and interpret basic communication systems and transmission mechanisms.
CO 2:	Apply error detection, correction, and data link layer protocols to design reliable communication systems using techniques such as CRC, framing, flow control, and MAC protocols.
CO 3:	Apply network layer and transport layer concepts to implement and simulate routing, addressing, congestion control, and reliable data transfer mechanisms in communication networks.
CO 4:	Analyze the behavior and performance of application layer protocols and network systems using simulation tools and packet analysis techniques to evaluate efficiency, congestion, and real-time network behavior.

Reference Books:	
1. Data Communication and Networking, Behrouz A. Forouzan, McGraw Hill, 5th Edition, 2013. 2. James F. Kurose and Keith W. Ross: Computer Networking: A TopDown Approach, 8th edition, Addison-Wesley, 2021. 3. Data and Computer Communication, William Stallings, 10th Edition, Pearson Education, 2013. 4. Introduction to Data Communications and Networking – Wayne Tomasi, Pearson Education, 5th Edition, 2011. 5. Larry L. Peterson and Bruce S Davie: Computer Networks: A Systems Approach, Fifth Edition, Elsevier, 2011. 6. Tanenbaum: Computer Networks, 5th Edition, Pearson Education/PHI, 2010	
Web links and Video Lectures:	
• https://archive.nptel.ac.in/courses/106/105/106105183/ • https://www.wireshark.org/docs/wsug_html_chunked/ • https://www.softwaretestinghelp.com/computer-networking-basics/ • https://ns3tutorial.com/ns2-ns3/	

List of Exercises	
Possible list of practical programs:	
Implement the following in C/C++/Java.	
1. Write a program for error detecting code using CRC-CCITT (16- bits). 2. Write a program to find the shortest path between vertices using bellman-ford algorithm 3. Write a program for congestion control using leaky bucket	
algorithm. Analyze the following in Wireshark.	
1. TCP 2. UDP 3. HTTP 4. FTP 5. DNS Server	

6. DHCP

Implement the following in NS3 or any other suitable simulator.

1. Implement three nodes point – to – point network with duplex links between them. Set the queue size, vary the bandwidth and find the number of packets dropped.
2. Implement transmission of ping messages/trace route over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.
3. Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.



Head of the Department
Dept. of Information Science & Engineering
B.M. Institute of Technology
Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science & Engineering
SEMESTER: V

Course Name: Computer Graphics & Animation
Course Code:24ISE155

L:T:P:J	0:0:2:2	CIA:	50
Credits:	02	SEA:	50
Total Number of Lecture Hours :	25	SEA Duration:	3 HOURS

Course Learning Objectives: The students will be able to

1. Apply the basic architecture and applications of computer graphics and the structure of OpenGL programs.
2. Apply 2D geometric transformations and clipping algorithms to graphical objects in a raster environment.
3. Apply 3D geometric transformations and model basic 3D objects such as colored cubes using color representations.
4. Analyze interactive input mechanisms and program event-driven interfaces using menus and motion events.
5. Analyze and implement animation techniques for images and objects using OpenGL features like texture mapping.

Module-1:	No. of Hours	Blooms cognitive Levels
Introduction: Application of Computer Graphics, Architecture. Graphics Programing: Introduction to OpenGL, The OpenGL API, Primitives and Attributes, Viewing, The Gasket Program. Self-Study Component: Explore OpenGL Setup, Study Real-Time Applications of Computer Graphics, Implement the Gasket Program	5	Apply
Module-2:		
2D Transformations: Basic Transformations-Translation, Scaling, Rotation. Clipping in Raster World: Rectangular Clipping Windows. Self-Study Component: Visualize 2D Transformations, Compare Clipping Algorithms, Practice Coordinate Geometry.	5	Apply
Module-3:		
Three Dimensional Geometric Transformations: Translation, Scaling, Rotation. Modelling a Colored Cube, Color Representation. Self-Study Component: Build and Rotate a 3D Cube, Study Color Models, Watch a 3D Modeling Tutorial.	5	Apply
Module-4:		
Input and Interaction: Interaction, Programming Event Driven Input, Menus, Motion Event. Self-Study Component: Menu Programming Practice, Learn About Event-Driven Programming, Motion Event Demo.	5	Apply
Module-5:		
Animation Technique: Animation of an Image (Using Texture Mapping Technique), Animation of an object. Self-Study Component: Texture Mapping Demo, Basic Object Animation, Explore Animation in Games/Movies	5	Apply

Course Outcomes: After completing the course, the students will be able to	
CO 1	Apply the concepts of computer graphics, graphics pipeline architecture, and OpenGL programming fundamentals to develop basic graphics applications.
CO 2	Apply 2D and 3D geometric transformations (translation, scaling, rotation) to graphical objects using OpenGL.
CO 3	Apply graphics programming techniques to implement drawing primitives, attributes, and clipping operations in raster-based environments.
CO 4	Apply event-driven programming concepts, menus, motion events, and input handling mechanisms to develop interactive graphics applications.
CO 5	Apply texture mapping and animation techniques in OpenGL to create and animate graphical objects and scenes.

Reference Books:
1. Meenakshi M. Raikar, Computer Graphics with OpenGL, CENGAGE, 2019 2. Donald D Hearn, M Pauline Baker and WarrenCarithers: Computer Graphics with OpenGL 4th Edition, Pearson, 2014 3. S. Sridhar, Digital Image Processing, second edition, Oxford University press 2016.An Integrated Approach to Software Engineering Pankaj Jalote Wiley India. 4. Edward Angel: Interactive Computer Graphics- A Top Down approach with OpenGL, 5th edition. Pearson Education, 2008 5. James D Foley, Andries Van Dam, Steven K Feiner, John F Huges Computer graphics with OpenGL: Pearson education
Programs:
1. Program to display an empty Window. 2. Program to display a triangle. 3. Program to generate the 2D Sierpinski gasket using random points. 4. Create and rotate a triangle about the origin and a fixed point. 5. Program to implement Liang-Barsky Line Clipping Algorithm. 6. Draw a color cube and spin it using Open GL transformation matrices. 7. Program to display a sphere, with keyboard interaction. 8. Program to display a pop-up menu to change the color of object square. 9. Program to animate an image.
Project List:
2DGame 3D House Aeroplane Crash Archery Bike Simulation Bull's Eye Car Parking Catch Me Taj Mahal Maze Game Walking Robot Flowing Fountain


Head of the Department
Dept. of Information Science & Engineering
B.N.M Institute of Technology
Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science & Engineering
SEMESTER: VI

Course Name: Cryptography and Information Security

Course Code: 24ISE161

L:T:P:J	2:2:0:0	CIA	:	50
Credits:	03	SEA	:	50
Total Number of Lecture Hours	40	SEA Duration	:	03 Hours

Course Learning Objectives: The students will be able to

- 1 To apply Cryptographical concepts, and Algorithms.
- 2 To plan necessary Approaches and Techniques to build protection mechanisms in order to secure computer networks.
- 3 To analyze Information Security Models, threats, and attacks.
- 4 To compare the legal, ethical and professional issues in Information Security.
- 5 To analyze the security aspects of Information.

Module-1:	No. of hours	Blooms cognitive Levels
Introduction: Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security. Classical Encryption Techniques: Symmetric Cipher Model, Substitution Techniques – Caesar Cipher, Monoalphabetic Ciphers, Playfair Cipher, Hill Cipher, Polyalphabetic Ciphers, One-Time Pad, Transposition Techniques. Steganography- Foundations of modern cryptography: Perfect security – Information theory – Product cryptosystem – Cryptanalysis.	8	Apply
Module-2:		
Mathematics of Symmetric Key Cryptography: Algebraic structures - Modular arithmetic- Euclid's algorithm- Congruence and matrices - Groups, Rings, Fields- Finite fields- Traditional Block Cipher Structure – Stream Ciphers and Block Ciphers, Motivation for the Feistel Cipher Structure, The Feistel Cipher, Block Cipher Design Principles Symmetric key Cipher: DES, Advanced Encryption Standard - Stream Ciphers, RC4.	8	Apply

Module-3:		
Mathematics of Asymmetric Key Cryptography: Primes – Primality Testing – Factorization – Euler’s totient function, Fermat’s and Euler’s Theorem - Chinese Remainder Theorem – Exponentiation and logarithm – Asymmetric Key Ciphers: RSA cryptosystem – Key distribution – Key management – Diffie Hellman key exchange, Introduction to ElGamal cryptosystem. Cryptographic Hash Functions: Secure Hash Algorithm (SHA) – SHA-512 Logic, SHA-512 Round Function. Digital Signature Standard (DSS).	8	Apply
Module-4:		
Introduction to Information Security: Introduction, The history of Information Security, what is security? Critical characteristics of Information, CNSS security model, Components of an Information System - The security systems development life cycle. The need for security: Threats and Attacks, Professional, Legal, Ethical Issues in information Security.	8	Analyze
Module-5:		
Security Analysis: Risk Management: Identifying and Assessing Risk, Assessing and Controlling Risk. Security Technologies: Introduction, Access Control, Firewalls, Protecting Remote Connections, Intrusion Detection and Prevention Systems – Honeypots, Honeynets and Padded Cell Systems – Scanning and Analysis Tools – Bio metric Access Controls.	8	Analyze

Course Outcomes: After completing the course, the students will be able to

- CO1: Apply and understand the fundamental concepts, structures, characteristics, and approaches of encryption techniques, cryptographic functions, key management, and IP security to address the need for effective information security.
- CO2: Apply public key and private key cryptosystems, hash functions and key distribution techniques in real-time applications
- CO3: Analyze information security risk management frameworks, threat and vulnerability assessment processes, and associated security technologies..

Reference Books:

1. Cryptography and Network Security – Principles and Practice by William Stallings, Person, 8th Edition, 2022.
2. Principles of Information Security, 6th edition, Michael E Whittman, Herbert J Mattord, CENGAGE Learning, 2021.
3. Network Security Essentials Applications and Standards, William Stallings, Pearson, 6th Edition, 2016.
4. Network Security Private Communication in a Public world, Charlie Kaufman, Radia Perlman and Mike Speciner, 3rd Edition, PHI, 2019.
5. Network Security and Management, Brijendra Singh, 3rd Edition, PHI, 2013.

Web links and Video Lectures:

1. <https://dl.hiva-network.com/Library/security/Cryptography-and-network-securityprinciples-and-practice.pdf>.
2. <https://imcs.dvfu.ru/lib.int/docs/Networks/Security/Network%20Security%20Foundati>

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science & Engineering
SEMESTER: VI

Course Name: Cloud Computing and Its Applications
Course Code: 24ISE162

L:T:P:J	2:0:1:0	CIA	:	50
Credits:	03	SEA	:	100
Total Number of Lecture Hours	40	SEA Duration	:	03 Hours

Course Learning Objectives: The students will be able to

1	To introduce the fundamental concepts of cloud computing, its characteristics, service models, and the evolution from distributed and parallel computing
2	To provide an understanding of cloud-enabling technologies, particularly virtualization, and their role in building and managing scalable cloud infrastructure.
3	To explore cloud service models (IaaS, PaaS, SaaS), cloud storage technologies, and architectural design principles.
4	To impart knowledge on resource management and cloud security mechanisms, and simulate cloud scenarios using appropriate tools.
5	To familiarize students with industry-standard cloud platforms such as AWS, Google App Engine, and Aneka, and enable them to develop and deploy cloud-based applications.

Module-1: INTRODUCTION	No. of hours	Blooms cognitive Levels
Introduction to Cloud Computing – Definition of Cloud – Evolution of Cloud Computing –Underlying Principles of Parallel and Distributed Computing – Cloud Characteristics – Elasticity in Cloud – On-demand Provisioning.	8	Apply
Module-2: CLOUD-ENABLING TECHNOLOGIES		
Service Oriented Architecture – REST and Systems of Systems – Web Services – Publish? Subscribe Model – Basics of Virtualization – Types of Virtualization – Implementation Levels of Virtualization – Virtualization Structures – Tools and Mechanisms – Virtualization of CPU –Memory – I/O Devices – Virtualization Support and Disaster Recovery. List of programs: • Install Virtual box/VMware Workstation with different flavors of Linux or Windows OS on top of windows7 or 8. • Install a C compiler in the virtual machine created using a virtual box and execute Simple Programs.	8	Apply
Module-3: CLOUD ARCHITECTURE, SERVICES AND STORAGE		
Layered Cloud Architecture Design – NIST Cloud Computing Reference Architecture – Public, Private and Hybrid Clouds – IaaS – PaaS – SaaS – Architectural Design Challenges – Cloud Storage – Storage-as-a-Service – Advantages of Cloud Storage – Cloud Storage Providers – S3. List of programs: • Install Google App Engine. Create hello world app and other simple web applications using python/java. • Use GAE launcher to launch the web applications.	8	Apply

Module-4: RESOURCE MANAGEMENT AND SECURITY IN CLOUD		
Inter Cloud Resource Management – Resource Provisioning and Resource Provisioning Methods – Global Exchange of Cloud Resources – Security Overview – Cloud Security Challenges –Software-as-a-Service Security – Security Governance – Virtual Machine Security – IAM –Security Standards. List of programs: Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim. Find a procedure to transfer the files from one virtual machine to another virtual machine.	8	Apply

Module-5: CLOUD TECHNOLOGIES AND ADVANCEMENTS		
CLOUD APPLICATION PROGRAMMING AND THE ANEKA PLATFORM: Framework overview, Anatomy of the Aneka container, Building Aneka clouds, Cloud programming and management INDUSTRIAL PLATFORMS AND NEW DEVELOPMENTS: Amazon web services. Google AppEngine, Microsoft Azure. Lab Component: - Design and deploy a web application in a paas - Working with Mangra soft Aneka Software - AWS lab components are included	8	Apply

Course Outcomes: After completing the course, the students will be able to	
CO 1:	Understand the fundamentals of cloud computing, including its evolution, characteristics, and key enabling technologies such as virtualization and service-oriented architecture
CO 2:	Install and configure virtualization platforms and cloud services, and execute basic cloud applications using tools like VirtualBox, Google App Engine, and Aneka.
CO 3:	Describe cloud architecture models (IaaS, PaaS, SaaS), and analyze their design challenges and storage solutions.
CO 4:	Apply resource management and security concepts in cloud environments, including simulation with CloudSim and implementing cloud security practices.
CO 5:	Analyze advanced cloud platforms and technologies such as AWS, Microsoft Azure, and Aneka, and design cloud-based applications using these industrial platforms.

Reference Books:
- Kai Hwang, Geoffrey C. Fox, Jack J. Dongarra, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, 2nd Edition, Morgan Kaufmann, 2018. - John W. Rittinghouse, James F. Ransome, Cloud Computing: Implementation, Management, and Security, 3rd Edition, CRC Press, 2016. - Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, Mastering Cloud Computing, 1st Edition, Tata McGraw Hill Education, 2013. - Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing: A Practical Approach, 1st Edition, Tata McGraw Hill Education, 2010. - George Reese, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, 1st Edition, O'Reilly Media, 2009.
Web links and Video Lectures:
https://archive.nptel.ac.in/courses/106/105/106105167/ https://www.youtube.com/watch?v=EN4fEbcFZ_E https://www.guru99.com/cloud-computing-for-beginners.html https://www.simplilearn.com/tutorials/cloud-computing-tutorial https://www.mygreatlearning.com/cloud-computing/tutorials

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering
SEMESTER – VI

Course Name: Artificial Intelligence and Machine Learning
Course Code: 24ISE163

L: T:P:J	2:1:1:0	CIA: 50
Credits:	04	SEA: 50
Total Number of Lecture Hours	50	SEE Duration: 03

Course Learning Objectives: This course will enable students to:

1. Understand and implement foundational machine learning models including concept learning, decision trees, and inductive learning strategies.
2. Apply probabilistic reasoning and Bayesian learning methods to address uncertainty in intelligent systems.
3. Explore the functioning of neural networks and reinforcement learning algorithms for data-driven decision-making.
4. Analyze search algorithms and constraint satisfaction techniques used for solving structured AI problems.
5. Understand the principles of Artificial Intelligence, including intelligent agents and rational behavior in various environments.

Module – 1	Teaching Hours	Blooms cognitive Levels
What is artificial intelligence (AI)? The foundations of AI, The history of AI, The state of the Art, Intelligent Agents: Agents and Environments, Good Behavior: The concept of Rationality, The nature of the environment, The structure of Agents. Self Study: Implement different types of agents Programs:1	10	Apply
Module – 2		
Solving Problems by Searching: Problem Solving Agents, Example problems, searching for solutions, Uninformed search strategies, Informed search strategies, Heuristic functions, Constraint satisfaction problems: Defining constraint satisfaction problems, Constraint propagation: Inference in CSPs, Local search for CSPs, The structure of problems. Self Study: Different types of Constraint satisfaction problems Programs:2,3,4	10	Apply
Module-3		
Introduction: What is machine learning, Types of Machine learning systems, Well Posed Learning problem, Designing Learning systems, Perspectives and Issues in	10	Apply

machine learning, Concept Learning: Concept learning task, Find-S algorithm, Candidate Elimination Algorithm, Inductive bias of Candidate Elimination Algorithm. Decision Tree Learning: Introduction, Decision tree representation, Appropriate problems, ID3 algorithm. Self study: Factors for Performance analysis of different algorithms Programs:5,6		
Module-4		
Bayesian Learning: Introduction, Bayes Theorem, Bayes Theorem and Concept Learning, Maximum Likelihood and least squared error hypotheses, Minimum Description Length Principle, Bayes Optimal Classifier, and Naive Bayes Classifier, Bayesian Belief network, EM Algorithm- General Statements of EM Algorithm. Self study: Deep Dive into Probabilistic Models and Applications Programs:7,8,9	10	Apply
Module-5		
Neural Networks: Introduction, Neural Network Representations, Appropriate problems for Neural Networks, Perceptron's, Multilayer Networks and Back Propagation Algorithms. Reinforcement Learning: Introduction, The learning task, Q-Learning, Nondeterministic rewards and actions, and Temporal difference learning. Self study: Implementation and Applications of Neural and Reinforcement Learning Models Programs:10,11	10	Apply
List of Programs		
1) Implement a program to demonstrate simple reflex agents. 2) Implement Best first search. 3) Implement A* Algorithm. 4) Implement Constraint satisfaction problem (CSP) 5) For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples. 6) Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample. 7) Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets. 8) Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API. 9) Implement EM Algorithm. 10) Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets. 11) Implement Q-learning algorithm.		

Course outcomes: The students will be able to:	
CO1	Understand the fundamental concepts of Artificial Intelligence, including intelligent agents, search strategies, machine learning, probabilistic reasoning, and neural networks.
CO2	Apply various AI algorithms and models such as search strategies, decision trees, Bayesian classifiers, and neural networks to solve real-world problems.
CO3	Apply agent-based and machine learning techniques to develop intelligent systems capable of learning, reasoning, and decision-making.
CO4	Analyze and compare the effectiveness of AI models and learning algorithms in different problem-solving contexts.
CO5	Analyze AI approaches to handle uncertainty and learning in dynamic environments using probabilistic and reinforcement learning models.
Reference Books:	
1. Artificial Intelligence: A Modern Approach, Stuart Russell, Peter Norvig, 4th Edition, Pearson Education, 2020, ISBN-13: 978-0134610993. 2. Machine Learning, Tom M. Mitchell, 1st Edition, McGraw Hill Education, 2017, ISBN-13: 978-12590969521. 3. Artificial Intelligence: A Modern Approach Stuart Russel Peter Norvig Pearson Education 4th edition 2020. 4. Machine Learning Tom M Mitchell McGraw Hill Education, 1st Edition, 2017. 5. Hands-On Machine Learning with Scikit-Learn and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems Aurelien Geron Shroff/O'Reilly Media 3rd Edition, 2022. 6. Introduction to Machine Learning Ethem Alpaydin PHI Learning Pvt. Ltd 4th Edition, 2020. 7. Structures and Strategies for Complex Problem-Solving George FLuger Pearson Education 5th Edition, 2011.	
WEB links and Video Lectures	
1. https://cloud.google.com/learn/artificial-intelligence-vs-machine-learning 2. https://onlinecourses.nptel.ac.in/noc19_me71/preview 3. https://youtube.com/playlist?list=PL1xHD4vteKYVpaliy295pg6_SY5qznc77&feature=shared	



Head of the Department
 Dept. of Information Science & Engineering
B.N.M Institute of Technology
 Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE

Department of Information Science and Engineering

SEMESTER - VI

Course Name: Virtual and Augmented Reality

Course Code: 24ISE164

L:T:P:J	0:0:2:2	CIA Marks	50
Credits	2	SEA Marks	50
Total Number of Lecture Hours	25	Exam Hours	03

Course Learning Objectives:

- Experience the fundamental Computer Vision, Computer Graphics and Human-Computer interaction Techniques related to VR/AR
- Demonstrate the Geometric Modelling Techniques Review the Virtual Environment
- Develop VR/AR Technologies Simulate and Apply Virtual/Augmented Reality to varieties of Applications.

Module 1	Number of Hours	Bloom's Level
Blender Introduction • Demonstrate Blender Interface, Overlay Reference, Position on Overlay, Managing Layouts, Camera Overlay • Demonstrate Scene view navigation, Scene view camera, Game View. • Project Creation, Plane creation, cube creation, Explanation of cube camera and position, Material Selection, and class file same, adding motion to the script, Run. • Understanding different 3D projectiles	4+ 2(Project)	Apply
Module 2		
Blender Programming • Show the use of blender for Creating, loading and saving the scenes. • Demonstrate the Transforms, Components, 3D objects, creating components, Saving the work. • Demonstrate the Layers, Constraints, Editor Feature 3d mode, Preferences, Build settings, Sculpting, preference add Ons	4+ 2(Project)	Apply
Module 3		
Graphics • Demonstrate the use of Render Pipelines, Cameras, lighting, models, mirroring objects • Demonstrate the use of Meshes, Textures, shaders, materials • Demonstrate the concept Visual effects, sky, colour for suitable example. • Demonstrate the concept of World Building, Terrain, tree Editor for suitable example. • Demonstrate Creation of left-hand, left-hand Controller and right-hand controller.	4+ 2(Project)	Apply

Module 4		
Scripting, Audio Video and Animation		
• Demonstrate the Setting up scripting Environment. • Apply the concepts for Creating frames, creating .mkv files, namespaces, attributes • audio files, tracker Modules, Audio Group inspector. • Demonstrate for suitable example to create Animation, Rotation in animation, animation clips, Humanoid avatars, Animation.		
	4+ 2(Project)	Apply
Module 5		
Augmented reality		
• Program to show augmented reality. • Program to show The Relationship Between Augmented Reality and Other Technologies-Media, Technologies. • Program to show Spectrum Between Real and Virtual Worlds, applications of augmented reality Augmented.		
	4+ 2(Project)	Create
Mini project • Using VR exploring the human body level by level, including cell level. • Using VR describing how medicine and body cures the illness. • Touchless ATM Using Augmented Reality. • Augmented Reality Controlled Hologram. • Augmented Reality House devices. • Augmented Reality Agriculture Field. • Augmented Reality and Medical devices. • VR Game Development. • VR Application Development. • Development of AI controlled VR Device, that accurately work to keep the clarity very good and using technology keep the eye healthy. • Development of Cardboard VR device as activity. • Improvising the quality of the VR Device and determining the accuracy. • Development of Satellite with the capability of VR Camera and more features. • Ability to view the Solar system using the VR Device using compatible camera sent through the satellite.		
Course Outcomes: The students will be able to • Apply navigation skills and utilize the Blender interface to create and manipulate basic 3D objects, integrating materials, textures, and basic animations • Apply programming techniques in Blender for creating and managing 3D scenes using transformations, constraints, layers, and XR integration • Use advanced graphics features such as render pipelines, shaders, lighting, world-building, and XR controllers to develop immersive virtual environments • Develop scripting environments to create animations, integrate audio/video elements, and produce realistic humanoid avatars and dynamic scenes. • Design and construct functional Augmented Reality applications, and explore AR's relationship with other technologies through innovative real-world mini-projects		

References:

1. Gregory C. Burdea & Philippe Coiffet, Virtual Reality Technology, 3rd Edition, Wiley-IEEE Press, 2024, ISBN 978-1394306947
2. Allan Fowler, AR Game Development, 1st Edition, Apress Publications, 2018, ISBN 978-1484236178
3. Allan Fowler, Beginning iOS AR Game Development: Developing Augmented Reality Apps with Unity and C#, 1st Edition, Apress Publications, 2018, ISBN 978-1484236178
4. Donald D. Hearn, M. Pauline Baker & Warren Carithers, Computer Graphics with OpenGL, 4th Edition, Pearson Education, 2011, ISBN 978-0136053583
5. Unity Manual – Scripting Documentation (2023.2)
6. Steven M. LaValle, Virtual Reality, Cambridge University Press, 2023, ISBN 978-1107198937
7. William R. Sherman & Alan B. Craig, Understanding Virtual Reality: Interface, Application, and Design, 2nd Edition, Morgan Kaufmann Publishers, 2018, ISBN 978-0128010389



Head of the Department
Dept. of Information Science & Engineering
B.N.M. Institute of Technology
Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

SEMESTER – VII

Course Name: Big Data Analytics

Course Code: 24ISE171

L:T:P: J	2:2:2:0	CIA:	50
Credits:	4	SEA:	50
Hours:	50	SEE Duration:	3 Hours

Course Learning Objectives: This course will enable students to:

1. Understand the fundamentals and evolution of Big Data along with its structure, types, and significance in modern data-driven industries.
2. Explore real-world applications of Big Data in business sectors like social networking, fraud detection, and retail to gain insights into its practical relevance
3. Introduce and explain technologies and frameworks for handling Big Data such as Hadoop, Spark, Pig, Hive, and Cassandra.
4. Enable learners to develop and execute data processing jobs using MapReduce and optimize their performance.
5. Familiarize learners with Big Data storage and retrieval concepts using tools like HDFS, Hive, and Cassandra with an emphasis on distributed and NoSQL systems.

Module 1	Teaching Hours	Blooms cognitive Levels
Getting an Overview of Big Data: What is Big Data? History of Data, Management – Evolution of Big Data, Structuring Big Data, Types of Data, Elements of Big Data, Big Data Analytics, Careers in Big data, Advantages of Big Data Analytics, Future of Big Data. Exploring the Use of Big Data in Business Context: Use of Big Data in Social Networking, Use of Big Data in Preventing Fraudulent Activities, Use of Big Data in Retail Industry, Introducing Technologies for Handling Big Data: Distributed and Parallel Computing for Big Data, Introducing Hadoop, Cloud Computing and Big Data, In-Memory Computing Technology for Big Data.	10	Understand
Module 2		
Understanding Hadoop Ecosystem: Hadoop Ecosystem, Hadoop Distributed File System: Concepts of Blocks in HDFS Architecture, NameNodes and DataNodes, The command line interface, Using HDFS Files, Hadoop specific File System Files, HDFS commands, The org.apache.hadoop.io.package, HDFS High Availability , Features of HDFS.	10	Understand

Understanding MapReduce Fundamentals: The MapReduce Framework. Exploring the Features of MapReduce. Working of MapReduce. Exploring Map and Reduce Functions. Techniques to Optimize MapReduce Jobs. Hardware/Network Topology, Synchronization, File System, Uses of MapReduce. Practical Component: Installation of Apache Hadoop (Not limited to this program)		
Module 3		
Processing Your Data with MapReduce: Recollecting the Concept of MapReduce Framework, Developing Simple MapReduce Application, Building the Application, Executing the Application, Points to Consider while Designing MapReduce. Customizing MapReduce Execution: Controlling MapReduce Execution with InputFormat, Reading Data with Custom Record Reader, Organizing Output Data with OutputFormats, Customizing Data with Record Writer, Optimizing MapReduce Execution with Combiner, Controlling Reducer Execution with Partitioners. Practical Component: Develop a MapReduce program to calculate the frequency of a given word in each file. Develop a MapReduce program to find the maximum temperature in each year. Develop a MapReduce program to implement Matrix Multiplication	10	Apply
Module-4		
Pig: Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators. Hive: Hive Shell, Hive Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User Defined Functions. Practical Component: • Develop a MapReduce to analyze weather data set and print whether the day is shinny or cool day • Write queries to sort and aggregate the data in a table using HiveQL • Demonstrate Hive UDF to convert table data into uppercase letter. • Compute the average of a given numeric values using pig.	10	Apply

Module 5

Spark: Introduction, Spark Applications, Jobs, Stages, and Tasks, Resilient Distributed Datasets, Creation, Transformations and Actions, Persistence, Serialization, Shared Variables, Broadcast Variables, Accumulators, Anatomy of a Spark Job Run, Job Submission, DAG Construction, Task Scheduling, Task Execution, Executors and Cluster Managers, Spark on YARN. Introduction to Apache Cassandra: Introduction and its history, Key features and advantages of Cassandra, Comparison with other database systems, understanding data modeling concepts in a NoSQL context, Designing a Cassandra data model, Primary keys and clustering keys, Data types in Cassandra. Practical Component: • Create a spark data frame and perform the following operations. ○ Add a column to a Spark DataFrame ○ Filter a Spark DataFrame ○ Group by aggregation on Spark DataFrame ○ Query the DataFrame with SQL • Develop an application to find the maximum temperature using Spark	10	Analyze
---	-----------	----------------

Course outcomes: The students will be able to

CO1	Understand the core concepts, architecture, and technologies involved in Big Data processing and storage.
CO2	Apply Big Data tools and frameworks such as Hadoop, MapReduce, Pig, Hive, Spark, and Cassandra to process and analyze large datasets.
CO3	Apply Big Data tools and techniques to build end-to-end data pipelines for efficient data processing, storage, and analysis.
CO4	Analyze the suitability and efficiency of Big Data technologies for different application scenarios.

Reference Books

1. **Big Data Black Book**, DT Editorial Services, 2016 Edition, Dreamtech Press, ISBN-13: 978-9351199311.
2. **Hadoop: The Definitive Guide**, Tom White, 4th Edition, O'Reilly Media, 2015, ISBN-13: 978-1491901632.
3. **Big Data Glossary**, Pete Warden, 1st Edition, O'Reilly Media, 2011, ISBN-13: 978-1449314590.
4. **Big Data and Analytics**, Seema Acharya, Subhashini Chellappan, 2nd Edition, Wiley India Publications, 2019, ISBN-13: 978-8126579518.

[Signature]
Head of Department
Dept. of Information & Engineering
B.N.M. Institute of Technology
Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering
SEMESTER-VII

Course Name: Prompt Engineering
Course Code: 24ISE1723

L:T:P:J:	3:0:0:0	CIA Marks:	50
Credits:	3	SEA Marks:	50
Total Number of Lecture Hours	40	SEA Duration:	3 Hours

Course Objectives:

- To provide a foundational understanding of the historical development and evolution of language modeling techniques, particularly large language models.
- To explore advanced prompt engineering strategies and evaluation methods to enhance model performance and reasoning capabilities.
- To examine ethical considerations, alignment challenges, and real-world domain-specific applications of prompt engineering in diverse professional fields.

Module1: Introduction and History of Language Modeling	Teaching Hours	Blooms cognitive Levels
History of language modeling , providing a foundational understanding of how large language models have evolved over time.	8	Understand
Module2: Prompt Engineering Strategies and Techniques		
various prompt engineering strategies and techniques, including prompt-tuning, chain-of-thought prompting, multi-hop reasoning, and multi-stage prompting	8	Apply
Module3: Benchmarking and Evaluation		
Prompt engineering benchmark datasets , task goals, and evaluation metrics. self-consistency and calibration methods to evaluate the efficacy of prompt designs.	8	Apply
Module4: Alignment and Ethics of Large Language Models		
Ethical considerations and alignment challenges associated with large language models. The importance of responsible AI usage and the potential societal impacts.	8	Analyze
Module5: Domain-Specific Applications		
Domain-specific applications of prompt engineering across fields such as law, medicine, and computer science. Practical implementations and challenges in various professional contexts	8	Apply

Course Outcomes: After completing the course, the students will be able to	
CO 1	Understand the evolution of language modeling and demonstrate an understanding of prompt engineering strategies, including chain-of-thought prompting and multi-hop reasoning.
CO 2	Apply appropriate prompt engineering techniques to solve domain-specific problems and utilize benchmarking tools to evaluate model performance using relevant metrics.
CO 3	Analyze the ethical and alignment challenges in large language models and assess the impact of prompt design on model behavior
Reference Books:	
1. Speech and Language Processing, Daniel Jurafsky and James H. Martin, 3rd Edition (Draft), 2025, Online Manuscript, ISBN: N/A. This is an updated draft released on January 12, 2025, available online. Stanford University 2. Natural Language Processing with Transformers, Lewis Tunstall, Leandro von Werra, and Thomas Wolf, Revised Edition, 2022, O'Reilly Media, ISBN-13: 978-1098136796. 3. Evaluating Natural Language Processing Systems: An Analysis and Review, Karen Spärck Jones and Julia R. Galliers, 1st Edition, 1996, Springer, ISBN-13: 978-3540613091. 4. Ethics of Artificial Intelligence and Robotics, edited by Vincent C. Müller, 2020, Stanford Encyclopedia of Philosophy (Summer 2020 Edition), ISBN: N/A. 5. Artificial Intelligence in Healthcare, Adam Bohr and Kaveh Memarzadeh, 1st Edition, 2020, Academic Press, ISBN-13: 978-0128184387.	


Head of the Department
 Dept. of Information Science & Engineering
B.N.M Institute of Technology
 Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science and Engineering

SEMESTER – VII

Course Code: Research Methodology & Intellectual Property Rights

Course Code: 24ISE174

L:T:P:J	1:2:0:0	CIA:	50
Credits:	02	SEA:	50
Hours:	25	SEA Duration:	3 hours

Course Learning Objectives: The students will be able to

1. To equip learners with a strong foundation in engineering research methods, ethical practices, and problem-solving techniques, including literature review and technical reading.
2. To develop learners' ability to analyze, synthesize, and manage knowledge through bibliographic tools and proper citation practices in scholarly work.
3. To provide comprehensive knowledge of Intellectual Property (IP) rights—including patents, copyrights, trademarks, industrial designs, and geographical indications—with a focus on their legal frameworks, registration processes, and real-world applications.

Module 1	Teaching Hours	Blooms cognitive Levels
Introduction: Meaning of Research, Objectives of Engineering Research, and Motivation in Engineering Research, Types of Engineering Research, Finding and Solving a Worthwhile Problem. Ethics in Engineering Research, Ethics in Engineering Research Practice, Types of Research Misconduct, Ethical Issues Related to Authorship.	5	Understand
Module 2		
Literature Review and Technical Reading: New and Existing Knowledge, Analysis and Synthesis of Prior Art Bibliographic Databases, Web of Science, Google and Google Scholar, Effective Search: The Way Forward Introduction to Technical Reading Conceptualizing Research, Critical and Creative Reading, Taking Notes While Reading, Reading Mathematics and Algorithms, Reading a Datasheet. Attributions and Citations: Giving Credit Wherever Due, Citations: Functions and Attributes, Impact of Title and Keywords on Citations, Knowledge Flow through Citation, Citing Datasets, Styles for Citations, Acknowledgments and Attributions, What Should Be Acknowledged, Acknowledgments in, Books Dissertations, Dedication or Acknowledgments.	5	Understand
Module 3		
Introduction To Intellectual Property: Role of IP in the Economic and Cultural Development of the Society, IP Governance, IP as a Global Indicator of Innovation, Origin of IP History of IP in India. Major Amendments in IP Laws and Acts in India.	5	Understand

Patents: Conditions for Obtaining a Patent Protection, To Patent or Not to Patent an Invention. Rights Associated with Patents. Enforcement of Patent Rights. Inventions Eligible for Patenting. Non-Patentable Matters. Patent Infringements. Avoid Public Disclosure of an Invention before Patenting. Process of Patenting. Prior Art Search. Choice of Application to be Filed. Patent Application Forms. Jurisdiction of Filing Patent Application. Publication. Pre- grant Opposition. Examination. Grant of a Patent. Validity of Patent Protection. Post-grant Opposition. Commercialization of a Patent. Need for a Patent Attorney/Agent. Can a Worldwide Patent be Obtained. Do I Need First to File a Patent in India. Patent Related Forms. Fee Structure. Types of Patent Applications. Commonly Used Terms in Patenting. National Bodies Dealing with Patent Affairs. Utility Models.		
Module 4		
Copyrights and Related Rights: Classes of Copyrights. Criteria for Copyright. Ownership of Copyright. Copyrights of the Author. Copyright Infringements. Copyright Infringement is a Criminal Offence. Copyright Infringement is a Cognizable Offence. Fair Use Doctrine. Copyrights and Internet. Non- Copyright Work. Copyright Registration. Judicial Powers of the Registrar of Copyrights. Fee Structure. Copyright Symbol. Validity of Copyright. Copyright Profile of India. Copyright and the word 'Publish'. Transfer of Copyrights to a Publisher. Copyrights and the Word 'Adaptation'. Copyrights and the Word 'Indian Work'. Joint Authorship. Copyright Society. Copyright Board. Copyright Enforcement Advisory Council (CEAC). International Copyright Agreements, Conventions and Treaties. Interesting Copyrights Cases. Trademarks: Eligibility Criteria. Who Can Apply for a Trademark. Acts and Laws. Designation of Trademark Symbols. Classification of Trademarks. Registration of a Trademark is Not Compulsory. Validity of Trademark. Types of Trademark Registered in India. Trademark Registry. Process for Trademarks Registration. Prior Art Search. Famous Case Law: Coca-Cola Company vs. Bisleri International Pvt. Ltd.	5	Understand
Module 5		
Industrial Designs: Eligibility Criteria. Acts and Laws to Govern Industrial Designs. Design Rights. Enforcement of Design Rights. Non-Protectable Industrial Designs India. Protection Term. Procedure for Registration of Industrial Designs. Prior Art Search. Application for Registration. Duration of the Registration of a Design. Importance of Design Registration. Cancellation of the Registered Design. Application Forms. Classification of Industrial Designs. Designs Registration Trend	5	Understand

in India. International Treaties. Famous Case Law: Apple Inc. vs. Samsung Electronics Co. Geographical Indications: Acts, Laws and Rules Pertaining to GI. Ownership of GI. Rights Granted to the Holders. Registered GI in India. Identification of Registered GI. Classes of GI. Non-Registerable GI. Protection of GI. Collective or Certification Marks. Enforcement of GI Rights. Procedure for GI Registration Documents Required for GI Registration. GI Ecosystem in India. Case Studies on Patents. Case study of Curcuma (Turmeric) Patent, Case study of Neem Patent, Case study of Basmati patent. IP Organizations In India. Schemes and Programmes		
---	--	--

Course Outcomes : After completing the course, the students will be able to

1	To identify research problems, apply ethical standards in research, and effectively utilize teaching-learning and technical reading methods.
2	To navigate intellectual property systems, including patents, copyrights, trademarks, industrial designs, and geographical indications, understanding their legal frameworks, registration processes, and enforcement mechanisms.
3	To critically analyze scholarly work, apply proper citation and attribution practices, and evaluate real-world case studies to understand the practical application of intellectual property laws.

Reference Books

1. **Engineering Research Methodology: A Practical Insight for Researchers**, Dipankar Deb, Rajeeb Dey, Valentina E. Balas, 1st Edition, Springer, 2019, ISBN-13: 978-9811329463.
2. **Intellectual Property: A Primer for Academia**, Prof. Rupinder Tewari, Ms. Mamta Bhardwaj, 1st Edition, Publication Bureau, Panjab University, 2021, ISBN: 81-85322-92-0.
3. **Textbook of Intellectual Property Rights**, N.K. Acharya, 9th Edition, Asia Law House, April 2025, ISBN-13: 978-9390912209.

WEB links and Video Lectures

1. https://onlinecourses.nptel.ac.in/noc25_ge66/preview
2. <https://online.vtu.ac.in/course-details/Research-Methodology>


 Head of the Department
 Dept. of Information Science & Engineering
B.N.M Institute of Technology
 Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU

Department of Information Science & Engineering

SEMESTER: VII

Course Name: Ethical Hacking

Course Code: 24ISE1734

L:T:P:J	3:0:0:0	CIA	:	50
Credits:	3	SEA	:	50
Hours:	40	SEA Duration	:	3 Hours

Course Learning Objectives: The students will be able to

1	To understand ethical hacking fundamentals, networking concepts, IP addressing, routing, and basic network setup.
2	To understand the vulnerability assessment using tools like NMAP, Nessus, and Metasploit for system exploitation and social engineering.
3	To study the cryptographic techniques including key management, hash functions, digital signatures, and explore steganography and biometrics.
4	To know the network-based attacks
5	To know the web application vulnerabilities using tools

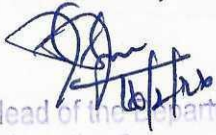
Module-1:	No. of hours	Blooms cognitive Levels
Introduction to Ethical Hacking: Basic Concepts of Networking, TCP/IP Protocol Stack, IP addressing and routing, TCP and UDP, IP subnetting, Routing protocols, IP version 6, Routing examples, Software Installation and Network Setup.	8	Understand
Module-2:		
Information Gathering: Nessus Installation , Port Scanning Using NMAP, Other Features of NMAP, Metasploit Exploiting System Software, Metasploit Exploiting System Software and Privilege, Metasploit Social Eng Attack, MITM (Man in The middle) Attack.	8	Understand
Module-3:		
Basic concepts of cryptography: Private-key cryptography, Public-key cryptography, hash functions, Digital signature and certificate, Applications, Steganography , Biometrics.	8	Understand
Module-4:		
Network Based Attacks: DNS and Email Security, Password cracking, Phishing attack, Malware, Wifi hacking, Dos and DDos attack , Elements of Hardware Security, Side Channel Attacks, Physical Unclonable Function, Hardware Trojan.	8	Apply

Module-5:		
Web Application Vulnerability Scanning, SQL Injection Authentication Bypass, SQL Injection Error Based, SQL Injection Error Based from Web Application, SQLMAP, Cross Site Scripting, File Upload Vulnerability, The NMAP Tool: A Rereook , Network Analysis using Wireshark.	8	Analyze

Course Outcomes: After completing the course, the students will be able to	
CO 1:	Understand the fundamental principles of ethical hacking, cryptography, network-based attacks, and web vulnerabilities, and demonstrate foundational skills in using tools such as NMAP, Nessus, Metasploit, SQLMAP, and Wireshark.
CO 2:	Design and implement a secure test environment to simulate network and application attacks, assess vulnerabilities, and develop strategic countermeasures in line with ethical hacking practices.
CO 3:	Analyze the different types of cyber threats, including cryptographic attacks, social engineering, and web application vulnerabilities, and apply appropriate mitigation techniques using ethical hacking tools.

Reference Books:
1. Data and Computer Communications – William Stallings, 10th Edition, 2021 2. TCP/IP Protocol Suite – Behrouz A. Forouzan 5th Edition, McGraw-Hill Education, 2012 3. UNIX Network Programming – W. Richard Stevens et al. Volume 1 (Sockets API), 3rd Edition, 2004 4. Introduction to Computer Networks and Cybersecurity – C-H. Wu & J. D. Irwin CRC Press, 2013 5. Cryptography and Network Security: Principles and Practice – William Stallings, 8th Edition (Global), Print ISBN 978-1292437484, released June 2022

Web links and Video Lectures:
1. https://onlinecourses.nptel.ac.in/noc25_cs142/unit?unit=1&lesson=2


 Head of the Department
 Dept. of Information Science & Engineering
 B.N.M. Institute of Technology
 Bengaluru - 560 070

B.N.M. Institute of Technology

An Autonomous Institution under VTU
Department of Information Science and Engineering
SEMESTER – VIII

Course Name: Business Intelligence & Analytics

Course Code: 24ISE1811

L: T:P: J	3:0:0:0	CIA : 50
Credits:	03	SEA : 50
Hours:	40	SEE Duration: 03

Course Learning Objectives: This course will enable students to:

1. Understand the fundamentals of Business Intelligence and Analytics (BIA)
2. Apply data management principles and relational database concepts.
3. Develop descriptive and predictive analytics solutions using statistical learning, data mining techniques, and visualization to extract insights from structured data.
4. Analyze customer and business problems using classification, clustering, regression, decision trees, ensemble methods, and customer analytics models such as CLV and RFM.
5. Implement advanced analytics techniques using programming tools (Python/R) for real-world business applications including targeted marketing, segmentation, time-series forecasting, and text mining.

Module-1	Teaching Hours	Blooms cognitive Levels
Introduction to Business Intelligence & Analytics (BIA), drivers of BIA, types of analytics: descriptive to prescriptive, vocabulary of business analytics, course plan and resources, Technical architecture of BIA, case analysis of AT&T Long distance, fundamentals of data management, OnLine Transaction Processing (OLTP), design process of databases.	8	Understand
Module-2		
Relational databases, normalisation, SQL queries, ShopSense case of management questions, data warehousing, OnLine Analytical Processing (OLAP), data cube, Descriptive analytics, and visualization, customer analytics, survival analysis, customer lifetime value, case study.	8	Apply
Module-3		
Data mining process, introduction to statistical learning, data pre-processing, data quality, overview of data mining techniques, case study using regression analysis, Introduction to classification, classification techniques, scoring models, classifier performance, ROC and PR curves, Introduction to decision trees, tree induction, measures of purity, tree algorithms, pruning, ensemble methods, Tree implementation in Python: problem of targeted mailing.	8	Apply

Module-4		
Cluster analysis, measures of distance, clustering algorithms, K-means and other techniques, cluster quality, A store segmentation case study using clustering, implementation in Python, profiling clusters, cluster interpretation and actionable insights, RFM sub-segmentation for customer loyalty	8	Apply
Module-5		
Machine learning, Artificial Neural Networks (ANN), topology and training algorithms, back propagation, financial time series modelling using ANN, implementation in Python, Text mining, process, key concepts, sentiment scoring, text mining using R-the case of a movie discussion forum.	8	Apply
Course outcomes: The students will be able to:		
CO1: Explain Business Intelligence and Analytics concepts, types of analytics, BIA architecture, and data-driven decision-making using real-world case studies.		
CO2: Design and manage relational databases and data warehouses by applying normalization, SQL queries, OLTP/OLAP concepts, and data cube operations for analytical processing.		
CO3: Perform data pre-processing, assess data quality, and apply regression and classification techniques to build predictive models and evaluate their performance using metrics such as ROC and PR curves.		
CO4: Apply clustering and customer analytics techniques (K-means, RFM analysis, segmentation) to identify customer groups, profile clusters, and derive actionable business insights.		
CO5: Implement machine learning, neural networks, time-series forecasting, and text mining techniques using Python/R to solve real-world business analytics problems.		
Reference Books:		
1. Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management, Third Edition, Gordon S. Linoff, Michael J. A. Berry 2. Designing a neural network for forecasting financial and economic time series, Neurocomputing 10 (1996) 215-236, ELSEVIER 3. Text Mining with R: A Tidy Approach 1st Edition, by Julia Silge, David Robinson, 2017		
WEB links and Video Lectures		
https://onlinecourses.nptel.ac.in/noc24_cs65/preview		


Head of the Department
Dept. of Information Science & Engineering
B.N.M Institute of Technology
Bengaluru - 560 070