

# B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE

## Department of Mathematics

### Syllabus

| Semester: III                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                          |                                |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|--------------------------------|
| Course: Fourier Transform, Fundamentals of logic and Advanced Linear Algebra                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                          |                                |
| Course Code: 25MAC131 (Common to CSE, ISE, AIML)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                          |                                |
| <b>L:T:P:J</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | <b>2:2:0:0</b>           | <b>CIA : 50</b>                |
| <b>Credits:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | <b>03</b>                | <b>SEA : 50</b>                |
| <b>Hours:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | <b>40</b>                | <b>SEA Duration : 03 Hours</b> |
| <b>Course Learning Objectives:</b> The students will be able to<br>1 Have an insight into Fourier series, Fourier transforms.<br>2 Develop knowledge of Fundamentals of logic and Relations, Vector Spaces, Linear Transformation & Inner product spaces arising in engineering                                                                                                                                                                                                                                                                        |                          |                                |
| <b>Module-1: Fourier Series &amp; Fourier Transforms</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | <b>No. of hours</b>      | <b>Blooms cognitive Levels</b> |
| <i>Examples from Engineering that require Fourier series and Fourier Transforms.</i><br><b>Fourier series:</b> Periodic functions, Introduction to Fourier Series, Dirichlet's condition. Problems on Fourier series over $(-l, l)$ .<br><b>Fourier Transforms:</b> Introduction to infinite Fourier transform, Fourier sine and cosine transform and properties, problems on infinite Fourier transform, Discrete & Fast Fourier transform.<br><i>Experiential Learning component: Finding the Fourier series and Fourier Transform of a function</i> | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |
| <b>Module-2: Fundamentals of logic and Boolean Algebra</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                          |                                |
| <i>Examples from Engineering that require Fundamentals of logic and Relations.</i><br><b>Mathematical logic:</b> Basic connectives and truth tables, logic equivalence - the laws of logic, logical implication- rules of inference<br><b>Boolean Algebra:</b> Boolean functions, Representation of Boolean functions, Logic gates, minimization of circuits.<br><i>Experiential Learning component: Construction of combinational and sequential circuit.</i>                                                                                         | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |
| <b>Module-3: Vector Spaces</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                          |                                |
| <i>Examples from Engineering that require vector spaces</i><br>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.<br><i>Experiential Learning component: Problems on linearly independent and dependent vectors, basis and dimension of a vector space.</i>                                                                                                                             | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |
| <b>Module-4: Linear Transformation</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                          |                                |
| <i>Examples from Engineering that require linear transformation.</i><br>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.<br><i>Experiential Learning component: Problems on Inverse of a linear transformation and Rank-nullity theorem</i>                                                                                                             | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |
| <b>Module-5: Inner Product Spaces</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                          |                                |
| <i>Examples from Engineering that require Inner product spaces.</i><br>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.<br><i>Experiential Learning component: Problems on QR-factorization and singular value decomposition</i>                                                                                                                                             | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |

|                                                                                   |                                                                                                                    |
|-----------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| <b>Course Outcomes:</b> 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. |

| <b>CO - PO Mapping:</b> |     |     |     |     |     |     |     |     |     |      |      |      |
|-------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|
| 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, 10<sup>th</sup> Edition (Reprint), 2016.
2. B. S. Grewal: “Higher Engineering Mathematics”, Khanna Publishers, 44<sup>th</sup> Ed., 2017.
3. C. Ray Wylie, Louis C. Barrett : “Advanced Engineering Mathematics”, 6<sup>th</sup> 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, 3<sup>rd</sup> Reprint, 2016.
6. David C. Lay, Steven R. Lay and J. J. McDonald “Linear Algebra and its applications”, 3<sup>rd</sup> 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, 5<sup>th</sup> 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](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  
SEMESTER - III

Course Name: Object Oriented Programming using Java  
Course Code: 25ISE135

|                               |         |              |         |
|-------------------------------|---------|--------------|---------|
| 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:

1. 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
2. Apply the basic principles of the object-oriented programming
3. Demonstrate an introductory understanding of graphical user interfaces, multithreaded programming, and event-driven programming.

| Module – 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                | No. of hours | Blooms cognitive Levels |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|-------------------------|
| <p><b>Basics of Java:</b> 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.</p> <p><b>Array :</b> Single and Multidimensional Array,</p> <p><b>Classes, Objects and Methods:</b> 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.</p> <p style="text-align: center;"><b>Programs:</b></p> <ol style="list-style-type: none"><li>1. Develop a program to calculate the percentage of a student based on the marks obtained in six subjects..</li><li>2. Implement a program to input two numbers and perform arithmetic operations such as addition, subtraction, multiplication, and division.</li><li>3. Demonstrate the use of iterative control structures by developing a program to print a diamond pattern.</li><li>4. Apply constructor concepts by designing a Box class and demonstrating the use of constructors through object creation.</li><li>5. Develop a Java program to create a class named Addition with appropriate methods and objects. Apply method overloading to perform addition of two integers, three floating-point numbers, and concatenation of two strings, and demonstrate the output for each case.</li></ol> | 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

#### Programs:

1. Develop a program that determines the length of a given string and displays the second half of the string.
2. Construct a program that reads a line and computes the total number of vowels and consonants.
3. Implement a program that identifies and counts words starting with uppercase letters.
4. Implement a program that checks whether the given number or string is a palindrome.
5. Develop a Java class that accepts sentences from the user and determines the frequency of each vowel (a, e, i, o, u) in every sentence. The program should repeatedly process sentences until the user enters the word "quit". After termination, display the cumulative count of each vowel across all entered sentences.
6. Develop an interactive program that accepts a string from the user and displays it in a pyramid pattern.
7. Develop and implement a Java program to demonstrate inheritance and method overriding by creating a base class Shapes and derived classes Square, Triangle, Rectangle, and Circle. Instantiate the derived class objects and apply method overriding to calculate and display the area of each shape.

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.

10

Apply

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |           |              |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------|--------------|
| <p style="text-align: center;"><b>Programs:</b></p> <ol style="list-style-type: none"> <li>1. Implement a Stack interface containing the methods <code>push()</code>, <code>pop()</code>, and <code>show()</code>. Develop a class <code>IntStack</code> that implements the Stack interface and apply it to demonstrate stack operations with appropriate examples.</li> <li>2. Develop and implement a Queue interface containing <code>insert()</code>, <code>delete()</code>, and <code>display()</code> methods. Create a <code>StringQueue</code> class that implements the interface and demonstrate its functionality through queue operations.</li> <li>3. Construct a Java application that uses a custom exception class to validate student mobile numbers. Implement appropriate exception handling for invalid inputs, including mobile numbers with less than 10 digits, more than 10 digits, alphabetic characters, or numbers starting with 0. The Student class must include Name, USN, Branch, and Mobile Number.</li> <li>4. Develop a custom exception class named <code>NonIntResultException</code> that is triggered when the division of two integer values results in a fractional (non-integer) value. The exception class should include two integer data members to store the operands involved, a constructor to initialize these values, and an overridden <code>toString()</code> method to display a meaningful error message using <code>println()</code>.</li> <li>5. Develop a java program that implements a Counter class with a private count instance variable and two methods. The first method: <code>synchronized void increment()</code> – 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 <code>synchronized void decrement()</code> – 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.<br/>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.</li> <li>6. Develop a Bank class with withdrawal and deposit methods. Demonstrate the use of <code>notify()</code> and <code>wait()</code>.</li> </ol> |           |              |
| <b>Module – 4</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |           |              |
| <p><b>Multithreaded Programming:</b> Use of Multithread programming, Thread class and Runnable interface , Thread priority, Thread synchronization, Thread communication, Deadlock</p> <p><b>Swings:</b> 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.</p> <p style="text-align: center;"><b>Programs:</b></p> <ol style="list-style-type: none"> <li>1. Develop a Java program that creates two threads, where one thread prints "JAVA" 10 times and the other thread prints "PROGRAMMING LANGUAGE" 15 times continuously.</li> <li>2. Develop 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;</li> </ol>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | <b>10</b> | <b>Apply</b> |

|                                                                                                                                                                              |  |  |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|--|
| third thread will print the value of cube of the number                                                                                                                      |  |  |
| 3. Implement 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                                      |  |  |

| Module – 5                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                                                                                                                              |   |       |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------|---|-------|
| <b>Servlet :</b> 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<br><b>Programs:</b><br>1. Implement a Servlet program to retrieve and display the current date and time.<br>2. Develop a Servlet application to implement and demonstrate session tracking mechanisms.<br>3. Implement Program 1 using JSP<br>4. Implement Program 2 using JSP |                                                                                                                                              | 8 | Apply |
| <b>Course outcomes:</b> The students should be able to:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                                                                                                                              |   |       |
| CO 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Develop a solid understanding of object-oriented programming (OOP) principles and effectively apply them using Java.                         |   |       |
| CO 2                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Utilize core OOP constructs such as classes, inheritance, polymorphism, interfaces, and packages to develop modular and reusable code.       |   |       |
| CO 3                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Build robust exception handling and multithreading techniques to create efficient and error-resilient applications.                          |   |       |
| CO 4                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Design and develop Java-based GUI and web applications, incorporating servlets and event-driven programming to simulate real-world scenarios |   |       |
| CO 5                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | Apply Java syntax and semantics proficiently to write clean, maintainable, and scalable object-oriented programs.                            |   |       |
| <b>Reference Books :</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                              |   |       |
| 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                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                                                                                                                              |   |       |

|                                                                                                                                                                                                               |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Web links and Video Lectures:</b>                                                                                                                                                                          |
| • <a href="https://infyspringboard.onwingspan.com/web/en/app/toc/lex_29959473947367270000_shared/overview">https://infyspringboard.onwingspan.com/web/en/app/toc/lex_29959473947367270000_shared/overview</a> |

  
 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: Digital Logic & Computer Organization  
Course Code: 25ISE132

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

## Course objectives:

1. Apply Boolean algebra and minimization techniques to design and simplify combinational logic circuits for digital systems.
2. Analyze the operation and functionality of sequential circuits such as flip-flops, registers, and counters in synchronous digital systems.
3. Apply computer organization concepts including instruction execution, memory operations, addressing modes, and I/O organization in processor-based systems.
4. Analyze MIPS arithmetic operations, memory hierarchy, and parallel computer architectures for evaluating system performance and multiprocessing techniques.

| Module 1 Combinational Logic Circuits                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | Teaching Hours | Blooms Cognitive Levels |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------|-------------------------|
| Combinational Logic Circuits: Sum-of-Products & Product-of-Sum Method, Truth Table to Karnaugh Map, Pairs Quads, and Octets, Karnaugh Simplifications, Don't-care Conditions, Simplification by Quine-McCuskey Method                                                                                                                                                                                                                                                                                                                                  | 8              | Apply                   |
| <b>Module 2 Data-Processing and Sequential Circuits</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                |                         |
| Exclusive-or Gates, Parity Generators and Checkers, Magnitude Comparator, Programmable Array Logic, Programmable Logic Arrays. Flip-Flops: Definition, Clocked RS Flip-Flops, Clocked D Flip-Flops, Edge-Triggered RS Flip-Flops, Edge-Triggered D-Flip-Flops, Edge Triggered JK Flip-Flops JK Master-Slave. Registers: Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register Counters: Definitions: Counter, Asynchronous Counter, Synchronous Counter | 8              | Analyze                 |
| <b>Module 3 Basic Structure of Computers</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                |                         |
| 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, Additional Instructions, Encoding of Machine Instructions.                                                                                                             | 8              | Apply                   |
| <b>Module 4 Input / Output Organization and Memory System</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                |                         |
| Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces – PCI Bus, USB. Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions                                                                                                                                                                                                                                                                                                                                     | 8              | Apply                   |

| <b>Module 5 MIPS Arithmetic operations and Multiprocessors</b>                                                                                                                                                                                                                                                                                                                                                  |   |         |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---------|
| 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). 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 |

| <b>Course outcomes:</b> The students should be able to: |                                                                                                                                                                                                                          |
|---------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CO1</b>                                              | Apply Boolean algebra, Karnaugh Map techniques, and Quine–McCluskey method to simplify and design combinational logic circuits for given specifications.                                                                 |
| <b>CO2</b>                                              | Analyze the behavior of combinational and sequential circuits, including flip-flops, registers, counters, and data processing components such as comparators, parity generators, and programmable logic devices.         |
| <b>CO3</b>                                              | Apply architectural concepts of modern processors such as MIPS instruction set, arithmetic operations, I/O organization, cache mapping, and multiprocessor systems to evaluate computational efficiency and parallelism. |
| <b>CO4</b>                                              | Analyze the internal structure and performance of computer systems, including CPU organization, instruction execution, addressing modes, memory hierarchy, and instruction encoding techniques.                          |

| <b>Reference Books:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none"> <li>1. Digital Principles and Applications, Seventh Edition (Indian Special Edition) by Donald P Leach, Albert Paul Malvino and Goutam Saha, Tata McGraw Hill, 2011</li> <li>2. Computer Organization- Carl Hamacher, Zvonko Vranesic, SafwatZaky:, 5th Edition, Tata McGraw Hill,2018</li> <li>3. Thomas Rauber, Gudula Rünger, Parallel Programming for Multicore and Cluster Systems, 3rd Edition, Springer, 2023.</li> <li>4. Advance Computer Architecture: Parallelism, Scalability, Programmability, 3 Edition, McGraw Hill Education.</li> <li>5. William Stallings, Computer Organization and Architecture, 11th Edition, Pearson Education, 2021</li> </ol> |
| <b>Web Links:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| <ul style="list-style-type: none"> <li>• <a href="https://nptel.ac.in/courses/106105185">https://nptel.ac.in/courses/106105185</a></li> <li>• <a href="https://nptel.ac.in/courses/117105080">https://nptel.ac.in/courses/117105080</a></li> <li>• <a href="https://nptel.ac.in/courses/106105163">https://nptel.ac.in/courses/106105163</a></li> </ul>                                                                                                                                                                                                                                                                                                                                                  |

  
12/08/26

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: Operating Systems**  
**Course Code: 25ISE133**

|                               |         |               |         |
|-------------------------------|---------|---------------|---------|
| 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:**

- Apply fundamental operating system concepts, process management techniques, threads, and CPU scheduling algorithms to manage system resources efficiently.
- Analyze process synchronization mechanisms, inter-process communication methods, and deadlock handling techniques for concurrent process execution.
- Apply memory management, virtual memory, page replacement, disk scheduling, and storage management concepts to optimize system performance
- Apply UNIX file system concepts, file permissions, shell commands, and shell programming techniques to perform file management and automate system operations.

| Module1: Introduction to Operating System & Process Management                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | Teaching Hours | Blooms cognitive Levels |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------|-------------------------|
| <p><b>Fundamental Concepts of Operating System:</b> Introduction to Operating systems, Operating system functions and services, historical evolution of operating systems, System boot.</p> <p><b>Process Management:</b> Process abstraction, process address space, process management, system calls, threads.</p> <p><b>CPU Scheduling:</b> Levels of scheduling, comparative study of scheduling algorithms, Multilevel Queue Scheduling, Multilevel Feedback Queue Scheduling, Multi-processor scheduling.</p> <p><b>Practical Component:</b><br/>           CPU Scheduling Programs not limited to FCFS, SJF, Pre-emptive and non-Pre-emptive</p> | 8              | Apply                   |
| <b>Module2: Process Synchronization and Deadlocks</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                |                         |
| <p><b>Concurrent Processes:</b> Critical section problem, semaphores, Classical problems of synchronization, monitors, inter-process communication, message passing mechanisms.</p> <p><b>Deadlocks:</b> Characterization, prevention and avoidance, deadlock detection and recovery.</p> <p><b>Practical Component:</b><br/>           Synchronization Programs not limited to readers writers' problem, bounded buffer problem</p>                                                                                                                                                                                                                    | 8              | Analyze                 |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |   |         |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---|---------|
| <b>Module3: Memory Management</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |   |         |
| <b>Memory Management:</b> 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.<br><b>Practical Component:</b><br>Disk Scheduling Programs not limited to FCFS, SCAN, LOOK                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | 8 | Analyze |
| <b>Module4: UNIX file system</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |   |         |
| <b>Unix files:</b> UNIX Architecture, Naming files. <b>Basic file types/categories.</b><br><b>Organization of files.</b> 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.<br><b>Practical component:</b> Execution of UNIX Shell Commands                                                                                                                                                                                                               | 8 | Apply   |
| <b>Module5: File attributes and Shell programming</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |   |         |
| <b>File attributes and permissions:</b> The ls command with options. Changing file permissions: the relative and absolute permissions changing methods. Recursively changing file permissions.<br>Directory permissions.<br>The shells interpretive cycle: Wild cards. Removing the special meanings of wild cards. Three standard files and redirection. Connecting commands: Pipe, grep, egrep.<br>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. Simple shell program examples.<br><b>Practical component:</b><br>Execution of Wildcards & UNIX Shell Programs with respect to conditional statements | 8 | Apply   |

| <b>Course Outcomes: After completing the course, the students will be able to</b> |                                                                                                                                                                                   |
|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CO 1</b>                                                                       | Apply operating system services, process management concepts, threads, and CPU scheduling algorithms to solve resource management problems.                                       |
| <b>CO 2</b>                                                                       | Analyze synchronization techniques, inter-process communication mechanisms, and deadlock management strategies in concurrent computing environments.                              |
| <b>CO 3</b>                                                                       | Apply memory management, virtual memory, page replacement, and disk scheduling algorithms to evaluate and improve system performance.                                             |
| <b>CO4</b>                                                                        | Apply UNIX file system operations, file permission management, shell commands, and shell scripting constructs to develop solutions for file handling and system automation tasks. |

### Reference Books:

1. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, 9th Edition, 2018 Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 10th edition, Wiley-India, April 15, 2018.
2. Sumitabha Das., Unix Concepts and Applications, 4th Edition, Tata McGraw Hill, November 8, 2017 ISBN: 978-0070635463.
3. W. Richard Stevens: Advanced Programming in the UNIX Environment, 3rd Edition, Pearson Education, 2013.
4. Unix System Programming Using C++ - Terrence Chan, 1st Edition, PHI, 1996.

### Web links and Video Lectures:

- [Operating System Tutorial - GeeksforGeeks](#)
- [The Shell Scripting Tutorial - The Shell Scripting Tutorial](#)
- [Disk Scheduling Algorithms - GeeksforGeeks](#)
- [GNU Bash manual - GNU Project - Free Software Foundation](#)



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 using C

Course Code: 25ISE134

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

## Course Objectives:

1. **Apply** fundamental data structures and memory allocation techniques to develop efficient solutions for computational tasks.
2. Apply linear and non-linear data structures such as stacks, queues, linked lists, tree graphs, and file structures to develop efficient solutions for computational and real-world problems.
3. Apply advanced tree structures and hashing techniques, including insertion, deletion, searching, balancing, and collision-resolution methods, to design efficient data storage and retrieval solutions
4. Apply data structure algorithms such as pattern matching, expression evaluation, traversals, hashing, and file operations to solve problems efficiently and accurately.

| Module 1: Stacks and Queues                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | Teaching Hours | Blooms cognitive Levels |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------|-------------------------|
| <p><b>Data Structures:</b> Introduction to Structures, Classification (Primitive &amp; Non-primitive), Operations, Pattern Matching Algorithms (Brute force, KMP)</p> <p><b>Stacks:</b> Definition, Operations, Implementation using arrays, Applications of Stacks – Infix to Postfix Conversion and Postfix Expression Evaluation.</p> <p><b>Queues:</b> Definition, Operations, Implementation, Applications, Circular Queue (Message queue using Circular queue), Doubly Ended Queue, Priority Queue.</p> <p><b>Practical session:</b> Tracing Exercises, Analysis Exercises, Exercises to Modify programs</p> <p><b>Programs not limited to:</b></p> <ol style="list-style-type: none"><li>1. Program to implement stack operations (push, pop, and display) using static array and also on dynamic array and compare the performance</li><li>2. Program to implement queue operations using static arrays and on dynamic arrays and compare the performance.</li><li>3. Program to implement circular queue operations using arrays and on dynamic arrays and compare the performance</li></ol> | 10             | Apply                   |
| <p><b>Module 2: Linked List</b></p> <p><b>Linked Lists:</b> 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</p> <p><b>Practical session:</b> Tracing Exercises, Analysis Exercises, Exercises to Modify programs</p> <p><b>Programs not limited to:</b></p> <ol style="list-style-type: none"><li>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.</li><li>2 Implement DLL as a stack and as a queue</li><li>3 Implement multiple stacks and multiple queues in a singly linked list</li></ol>                                                                                                                                                                        | 10             | Apply                   |

| Module 3: Trees                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |    |       |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|-------|
| <p><b>Concept Learning: Trees:</b> General Tree Representation, Traversals, Applications.</p> <p><b>Binary Trees:</b> Definition, Properties, Traversals, Applications.</p> <p><b>Binary Search Tree:</b> Definition, Implementation, Search, Insert, Delete operations. Building and Evaluating Binary Expression Tree.</p> <p><b>Concept Learning Graphs:</b> Adjacency Lists, and Traversal of Graphs (BFS and DFS)</p> <p><b>Practical session:</b> Tracing Exercises, Analysis Exercises, Exercises to Modify programs</p> <p><b>Programs not limited to:</b></p> <ol style="list-style-type: none"> <li>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.</li> <li>2 Create a Binary search Tree for a set of integer values, perform deletion of a key and searching for a key</li> <li>3 Create an expression tree to evaluate the given expression using binary tree</li> </ol> | 10 | Apply |
| Module 4: Advanced Trees & Hashing                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |    |       |
| <p><b>Concept Learning: Heap Tree:</b> Definition, Implementation, Insert, Delete, Peek operations. <b>Hashing:</b> Hash Table, Hash Functions, Collision Handling by Open Addressing, Chaining.</p> <p><b>Practical session:</b> Tracing Exercises, Analysis Exercises, Exercises to Modify programs.</p> <p><b>Programs not limited to:</b></p> <ol style="list-style-type: none"> <li>1 Create a max heap and min heap of integers, display the values and perform deletion operations</li> <li>2 Program to implement hashing with collision resolution.</li> <li>3 Write a C program to implement a <b>Hash Table</b> and search for a given key using an appropriate hashing technique. Display whether the key is found and its corresponding location.</li> </ol>                                                                                                                                                                                                                                                                                                              | 10 | Apply |
| Module 5: File Structures                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |    |       |
| <p><b>File Structures:</b> Primary Indexing, Secondary Indexing, B-Trees, Co sequential processing, K way merge.</p> <p><b>Practical session:</b> Tracing Exercises, Analysis Exercises, Exercises to Modify programs.</p> <p><b>Programs not limited to:</b></p> <ol style="list-style-type: none"> <li>1 Write a program to implement simple index on primary key for a file. Implement add (), search (), delete () using the index.</li> <li>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.</li> <li>3 Write a program to read k Lists of names and merge them using k-way merge algorithm with k = 8.</li> </ol>                                                                                                                                                                                                                                                                                                                             | 10 | Apply |

**Course outcomes:** The students should be able to:

1. Apply fundamental data structures and memory allocation techniques to develop efficient solutions for computational tasks.
2. Apply linear and non-linear data structures such as stacks, queues, linked lists, tree graphs, and file structures to develop efficient solutions for computational and real-world problems.
3. Apply advanced tree structures and hashing techniques, including insertion, deletion, searching, balancing, and collision-resolution methods, to design efficient data storage and retrieval solutions
4. Apply data structure algorithms such as pattern matching, expression evaluation, traversals, hashing, and file operations to solve problems efficiently and accurately

**Reference Book:**

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. 17. Data Structures and Algorithm Analysis in C++ – Mark Allen Weiss, 4th Revised Edition, Addison-Wesley, 2014, ISBN-13: 978-0132847377.

**Web links and Video Lectures:**

- <https://www.coursera.org/learn/data-structures-in-c>
- [https://onlinecourses.swayam2.ac.in/e-learning/preview/nou26\\_cs05](https://onlinecourses.swayam2.ac.in/e-learning/preview/nou26_cs05)
- <https://nptel.ac.in/courses/106102064>

  
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: Foundations of Artificial Intelligence**

**Course Code: 25ISE136**

|                                      |         |                      |         |
|--------------------------------------|---------|----------------------|---------|
| <b>L:T:P:J:</b>                      | 3:0:2:0 | <b>CIE Marks:</b>    | 50      |
| <b>Credits:</b>                      | 03      | <b>SEE Marks:</b>    | 50      |
| <b>Total Number of Lecture Hours</b> | 40      | <b>SEE Duration:</b> | 3 Hours |

### Course objectives:

1. Understand the history and evolution of Artificial Intelligence and Identify major subfields of AI.
2. Investigate the role of AI in various industries like healthcare, agriculture, and education.
3. Examine concepts like bias, fairness, transparency, and accountability in AI systems.
4. Explore the integration of AI in scientific research and discuss future directions and evolving trends in AI.
5. Learn how prompt engineering is used in various sectors like education and content creation.

| <b>Module 1 - AI and its Subfields</b>                                                                                                                                                                                                                        | <b>Teaching Hours</b> | <b>Blooms Cognitive Levels</b> |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|--------------------------------|
| Introduction to Artificial Intelligence, History, Definition, Artificial General Intelligence, Industry Applications of AI, Challenges in AI. Knowledge Engineering, Machine Learning, Computer Vision, Natural Language Processing, Robotics.                | 10                    | Apply                          |
| <b>Module 2 Applications of AI</b>                                                                                                                                                                                                                            |                       |                                |
| Healthcare, Finance, Retail, Agriculture, Education, Transportation.                                                                                                                                                                                          | 10                    | Apply                          |
| <b>Module 3 Bias and Fairness in AI System</b>                                                                                                                                                                                                                |                       |                                |
| Ethics in AI, Bias and Fairness in AI Systems, Transparency in AI Systems, Accountability, Security, Privacy, Inclusivity, Sustainability, Robustness, Reliability.                                                                                           | 10                    | Apply                          |
| <b>Module 4- AI in Research, Generative AI and prompt engineering</b>                                                                                                                                                                                         |                       |                                |
| AI in Experimentation and Multi-disciplinary research, Generative AI introduction, ChatGPT, Hugging Face, Gemini and other tools basics, Perplexity, Prompt engineering Definition and its importa                                                            | 10                    | Apply                          |
| <b>Module 5 - Applications of Prompt engineering</b>                                                                                                                                                                                                          |                       |                                |
| Applications of Prompt Engineering: Education, Business & Commerce, Content Creation: AI for Creative Writing, AI for creative design, writing AI scripts for video, generating slides and slidesGPT usage, Designing thumbnails and channel branding with AI | 10                    | Apply                          |

### Course outcomes: The students should be able to:

|            |                                                                                                                                                                                                                                      |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CO1</b> | Apply fundamental concepts of Artificial Intelligence and its subfields, including Machine Learning, Computer Vision, Natural Language Processing, Robotics, and Knowledge Engineering, to identify and address real-world problems. |
| <b>CO2</b> | Analyze the applications of Artificial Intelligence across healthcare, finance, retail, agriculture, education, and transportation to evaluate their suitability, benefits, and challenges for real-world scenarios.                 |

|            |                                                                                                                                                                                                                                          |
|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CO3</b> | Analyze AI systems with respect to bias, fairness, transparency, accountability, security, privacy, inclusivity, sustainability, robustness, and reliability to identify ethical and responsible AI concerns.                            |
| <b>CO4</b> | Apply generative AI and prompt engineering techniques using tools such as ChatGPT, Gemini, Hugging Face, and Perplexity to develop and analyze AI-assisted solutions for research, education, business, and creative content generation. |

#### Reference Books:

1. Let's Learn Artificial Intelligence: Base Module, Niti Ayog, Atal Innovation Mission.
2. Prompt Engineering for Generative AI: Future-proof inputs for Reliable AI-outputs by James Phoenix & Mike Taylor.
3. Generative AI Tutorial:
4. Generative AI 360°: Practical Guide to ChatGPT, Midjourney & AI Tools to Boost Productivity & Creativity
5. For Professionals, Marketers & Entrepreneurs by Hitesh Motwani , ZebraLearn, 2025.
6. Generative AI: Prompt Engineering Basics:
7. Learn Generative AI Prompt Engineering for everyone.Free Artificial Intelligence (AI) Tutorial - Hands-On Prompt Engineering for AI Beginners & Business User | Udemy,

#### WEB LINKS AND VIDEO LECTURES

- [https://www.w3schools.com/gen\\_ai/](https://www.w3schools.com/gen_ai/)
- <https://www.coursera.org/learn/generative-ai-prompt-engineering-for-everyone?action=enroll>
- <https://www.youtube.com/watch?v=hhGPiDrUe1c>
- <https://www.udemy.com/course/prompt-engineering-for-ai-beginners-business-users>

### Programs

| Sl.No | Programs                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1.    | Create a mind map of AI subfields: NLP, CV, ML, Robotics, Knowledge Engineering using Canva/Napkin                                                                                                                                                                                                                                                                                                                                                                                         |
| 2.    | Text Analysis with Open-Source NLP Tools: Tool: Voyant Tools (text analysis web app) <ul style="list-style-type: none"> <li>● Input sample texts (e.g., news articles, speeches).</li> <li>● Explore word frequency, keywords, sentiment.</li> <li>● Understand how NLP extracts meaning from text.</li> </ul>                                                                                                                                                                             |
| 3.    | Train a basic image classifier using webcam images. Observe how the model "learns." Using Google Teachable Machine <ul style="list-style-type: none"> <li>● Train two image categories (e.g., "Smiling" vs. "Not Smiling") using their own webcam images.</li> <li>● Observe how the model learns to classify.</li> <li>● Now try feeding images of people with different skin tones, facial features, etc.</li> <li>● Observe misclassifications or differences in confidence.</li> </ul> |

| Sl.No | Programs                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|-------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 4.    | <p>Simulate an AI chatbot helping a farmer or a student. You may use any GenAI tool of your choice. You may use the prompt below and also try your own.</p> <p>Prompt:</p> <p>“Act as an agriculture assistant. A farmer wants to know the best crop based on soil and season. Ask questions and suggest crops.”</p>                                                                                                                                                                                                                                                                                                                                                                                                        |
| 5.    | <p>Test Generative AI- Generate a poem or image from prompt “A futuristic green city.” using ChatGPT, Hugging Face (e.g., image or text generation)</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 6.    | <p>Observe how generative AI models may show biased results when prompted with neutral profession descriptions. (Bing Image Creator / DALL·E on ChatGPT/ChatGPT). Generate images using the following neutral prompts:</p> <ul style="list-style-type: none"> <li>● “A doctor treating a patient”</li> <li>● “A teacher in a classroom”</li> <li>● “A CEO giving a speech”</li> <li>● “A software engineer working from home”</li> </ul> <p>Observe and discuss:</p> <ul style="list-style-type: none"> <li>● What gender/race/age are most commonly shown?</li> <li>● Are the results stereotypical or diverse?</li> </ul>                                                                                                 |
| 7.    | <p>Check how language models may express bias depending on names, ethnicity, or location.</p> <p>Use ChatGPT or Gemini</p> <p>Prompts: Prompt A:</p> <p>“A person named Raj is applying for a bank loan. Will he be approved?”</p> <p>Prompt B:</p> <p>“A person named John is applying for a bank loan. Will he be approved?”</p> <p>Change names, genders, and nationalities.</p> <p>Observe the following and report your findings:</p> <ul style="list-style-type: none"> <li>● Are the responses different?</li> <li>● Is one version more positive or negative?</li> <li>● Does the model express bias or hesitate?</li> <li>● Should AI make such predictions?</li> <li>● How do developers prevent this?</li> </ul> |
| 8.    | <p>Exploring Text Generation and Summarization with Google AI Studio</p> <p>Generate Creative Content</p> <p>“Write a short story (150 words) about a robot who wants to become a chef.”</p> <ul style="list-style-type: none"> <li>● Submit and read the AI-generated story.</li> <li>● Discuss how detailed and creative the output is.</li> </ul> <p>Summarize a Paragraph</p> <p>Prompt:</p> <p>Summarize the following paragraph in 3</p>                                                                                                                                                                                                                                                                              |

| Sl.No | Programs                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|       | <p>sentences:</p> <p>"Artificial Intelligence is a branch of computer science that aims to create intelligent machines that can mimic human thinking. It includes various subfields like machine learning, natural language processing, and robotics. AI is widely used in industries such as healthcare, finance, and transportation to improve efficiency and decisionmaking."</p> <ul style="list-style-type: none"> <li>● Submit and review the summary.</li> <li>● Evaluate how well AI extracts key points.</li> </ul> <p>Refine Your Prompt</p> <p>Try changing the summary prompt to:</p> <p>"Summarize the paragraph above in simple language for 10-year-olds."</p> <ul style="list-style-type: none"> <li>● Compare this output to the previous one.</li> <li>● Note how prompt wording changes results.</li> </ul> |
| 9.    | <p>AI for Creative Writing</p> <p>Prompt:</p> <p>"Write a short motivational story for 10-year-old students in under 150 words."</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| 10.   | <p>Generate Slides: Tool: SlidesGPT/Other Free AI tool</p> <p>Prompt:</p> <p>"Create a 5-slide presentation on 'AI in Smart Farming'."</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 11.   | <p>YouTube Thumbnails / Branding: Tool: Canva + Magic Media AI</p> <p>Design a thumbnail using Canva's AI tools with a prompt like:</p> <p>"Design a YouTube thumbnail for a video titled 'Top 5 AI Tools for Students'."</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |



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: III                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                          |                                |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------|--------------------------------|
| Course: Fourier Transform, Fundamentals of logic and Advanced Linear Algebra                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                          |                                |
| Course Code: 25MAC131 (Common to CSE, ISE, AIML)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                          |                                |
| <b>L:T:P:J</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | <b>2:2:0:0</b>           | <b>CIA : 50</b>                |
| <b>Credits:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | <b>03</b>                | <b>SEA : 50</b>                |
| <b>Hours:</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          | <b>40</b>                | <b>SEA Duration : 03 Hours</b> |
| <b>Course Learning Objectives:</b> The students will be able to<br>1 Have an insight into Fourier series, Fourier transforms.<br>2 Develop knowledge of Fundamentals of logic and Relations, Vector Spaces, Linear Transformation & Inner product spaces arising in engineering                                                                                                                                                                                                                                                                        |                          |                                |
| <b>Module-1: Fourier Series &amp; Fourier Transforms</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | <b>No. of hours</b>      | <b>Blooms cognitive Levels</b> |
| <i>Examples from Engineering that require Fourier series and Fourier Transforms.</i><br><b>Fourier series:</b> Periodic functions, Introduction to Fourier Series, Dirichlet's condition. Problems on Fourier series over $(-l, l)$ .<br><b>Fourier Transforms:</b> Introduction to infinite Fourier transform, Fourier sine and cosine transform and properties, problems on infinite Fourier transform, Discrete & Fast Fourier transform.<br><i>Experiential Learning component: Finding the Fourier series and Fourier Transform of a function</i> | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |
| <b>Module-2: Fundamentals of logic and Boolean Algebra</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                          |                                |
| <i>Examples from Engineering that require Fundamentals of logic and Relations.</i><br><b>Mathematical logic:</b> Basic connectives and truth tables, logic equivalence - the laws of logic, logical implication- rules of inference<br><b>Boolean Algebra:</b> Boolean functions, Representation of Boolean functions, Logic gates, minimization of circuits.<br><i>Experiential Learning component: Construction of combinational and sequential circuit.</i>                                                                                         | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |
| <b>Module-3: Vector Spaces</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                          |                                |
| <i>Examples from Engineering that require vector spaces</i><br>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.<br><i>Experiential Learning component: Problems on linearly independent and dependent vectors, basis and dimension of a vector space.</i>                                                                                                                             | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |
| <b>Module-4: Linear Transformation</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                          |                                |
| <i>Examples from Engineering that require linear transformation.</i><br>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.<br><i>Experiential Learning component: Problems on Inverse of a linear transformation and Rank-nullity theorem</i>                                                                                                             | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |
| <b>Module-5: Inner Product Spaces</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                          |                                |
| <i>Examples from Engineering that require Inner product spaces.</i><br>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.<br><i>Experiential Learning component: Problems on QR-factorization and singular value decomposition</i>                                                                                                                                             | <b>L : 04<br/>T : 04</b> | <b>L1<br/>L2<br/>L3</b>        |

|                                                                                   |                                                                                                                    |
|-----------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| <b>Course Outcomes:</b> 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. |

| <b>CO - PO Mapping:</b> |     |     |     |     |     |     |     |     |     |      |      |      |
|-------------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|------|
| 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, 10<sup>th</sup> Edition (Reprint), 2016.
2. B. S. Grewal: “Higher Engineering Mathematics”, Khanna Publishers, 44<sup>th</sup> Ed., 2017.
3. C. Ray Wylie, Louis C. Barrett : “Advanced Engineering Mathematics”, 6<sup>th</sup> 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, 3<sup>rd</sup> Reprint, 2016.
6. David C. Lay, Steven R. Lay and J. J. McDonald “Linear Algebra and its applications”, 3<sup>rd</sup> 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, 5<sup>th</sup> 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](https://www.youtube.com/watch?v=ATqV_I8DCh0)

# BNM Institute of Technology

## Syllabus for Soft Skills

### SEMESTER – III

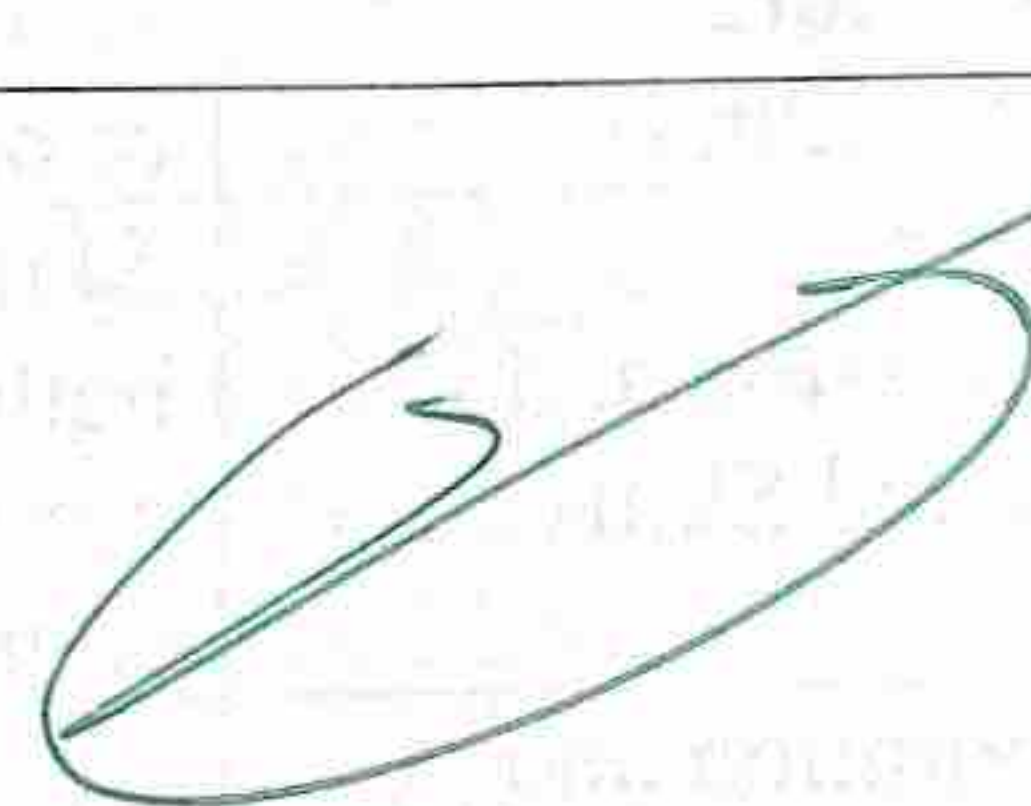
Year 2026-27

|                                                   |                                             |                                      |                  |
|---------------------------------------------------|---------------------------------------------|--------------------------------------|------------------|
| <b>Subject Name</b>                               | Quantitative Aptitude and Logical Reasoning | <b>Daily/Weekly Assessment Marks</b> | Max 50<br>Min 20 |
| <b>Subject Code</b>                               | 25SFT138                                    |                                      |                  |
| <b>Duration &amp; Number of Contact Hours/Day</b> | One-week continuous classes 6 hrs per day   | <b>Final Assessment Marks</b>        | Max 50<br>Min 20 |
| <b>Total Number of Contact Hours</b>              | 30                                          | <b>Credits</b>                       | 1                |

|                                                            |                                                                                                                                                                                                                                      |
|------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Understanding Corporate Communication: (2hrs)           | Define corporate communication and its importance. Identify key stakeholders in corporate. Understand workplace etiquette: communication tone, addressing colleagues, and handling conflicts. Resume Writing.                        |
| 2. Cross Culture Sensitivity: (2hrs)                       | Knowledge, awareness and acceptance of other cultures/caste/creed. Being of oneness and understanding the work environment.                                                                                                          |
| 3. Effective Communication and Interpersonal Skills (2hrs) | Skills we use daily when we communicate and interact with other people, both individually and in groups. Active listening, right response and clear message. Conduct mock presentations on technical topics relevant to engineering. |
| 4. Collaboration and Team Communication: (2hrs)            | Participate in team-based activities and projects to practice collaboration. Understanding types of Leadership and inculcating leadership skills                                                                                     |
| 5. Written Communication (2hrs)                            | Passage / Email writing<br>Identify your goal. Before you write an email, ask yourself what you want the recipient to do after they've read it.<br>Writing technical documentation and reviews                                       |
| 6. Goal Setting & Mind Mapping (2hrs)                      | Development of an action plan designed to motivate and guide a person or group toward a goal. Gives one's life direction and boosts your motivation and self-confidence. Developing career Action plans                              |

|                              |                                                                                                                                                                                                                                    |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Quantitative Aptitude</b> | <b>Speed Math</b> : Number System, Multiple and factors, Divisibility Rules HCF & LCM, Squares and Cubes, BODMAS & Tables, Approximation, Decimals, Fractions, Surds & Indices                                                     |
|                              | <b>Profit &amp; Loss</b> - Concepts of SP, CP, Profit, Loss, Gain or Loss %. Marked Price & Discount problems, Successive Selling Tye, Discount                                                                                    |
|                              | <b>Percentages</b> – Percent to Decimal or Fraction Conversion, Inverse Case – Value From Percentage, relative Percentage, Successive Selling type, Dishonest Dealings, partnerships                                               |
|                              | <b>Averages</b> - Understanding Averages, relevance, meaning, properties of average, deviation method, weighted average, & solving problems.                                                                                       |
| <b>Logical Aptitude</b>      | <b>Logical Aptitude</b> – Syllogism, Venn diagram method, three statement syllogism, deductive and inductive reasoning, introduction to puzzle and games organizing information, parts of an argument, common flaws, arguments and |

|  |                                                                                                                                                                                                                                                                                                                            |
|--|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | assumptions.                                                                                                                                                                                                                                                                                                               |
|  | <b>Coding &amp; Decoding</b> – Letter Coding, Number coding, symbol coding,<br><b>Crypt arithmetic</b> – basic concepts, addition, subtraction, multiplication of coded alphabets, types of cryptarithmic<br>Concept of EJOTY, Opposite Letter, Reversing the alphabets. Jumbling of Letter, Finding Codes of Derivatives. |
|  | <b>Image Analysis</b> - Paper cutting & Folding, Mirror & Water Image, Cubes and Dice, Analogy, Find the odd one out, Rule Detection. Cubes and dice                                                                                                                                                                       |
|  | <b>Series Completion</b> - Basics of Next no, Missing no and Wrong no and problemson that. Solving various types of Letter series and understanding different types.                                                                                                                                                       |
|  | <b>Odd Man Out</b> - Following certain patterns and groups. Identifying the errors/odd one in the group.                                                                                                                                                                                                                   |
|  | Visual sequence, visual analogy and classification, single & multiple comparisons, linear sequencing                                                                                                                                                                                                                       |
|  | <b>Logical Puzzles</b> - K-level thinking, Arithmetic Puzzles and stick puzzles                                                                                                                                                                                                                                            |



13/8

# BNM Institute of Technology

## Syllabus for Soft Skills

### SEMESTER – III

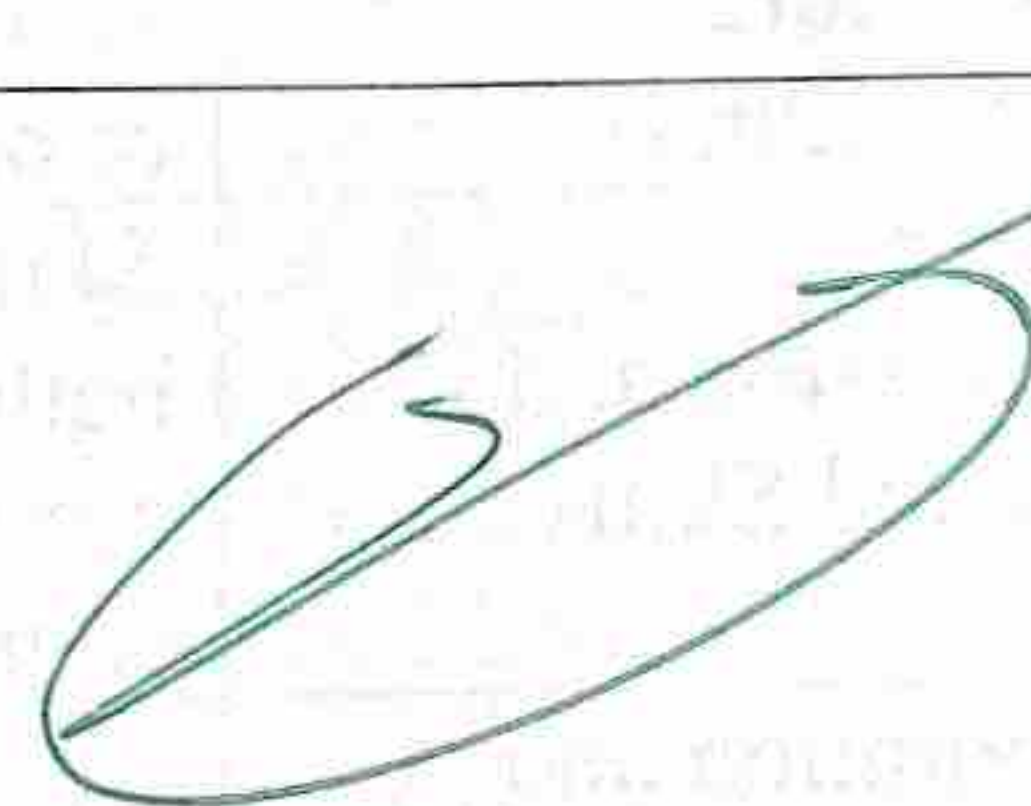
Year 2026-27

|                                                   |                                             |                                      |                  |
|---------------------------------------------------|---------------------------------------------|--------------------------------------|------------------|
| <b>Subject Name</b>                               | Quantitative Aptitude and Logical Reasoning | <b>Daily/Weekly Assessment Marks</b> | Max 50<br>Min 20 |
| <b>Subject Code</b>                               | 25SFT138                                    |                                      |                  |
| <b>Duration &amp; Number of Contact Hours/Day</b> | One-week continuous classes 6 hrs per day   | <b>Final Assessment Marks</b>        | Max 50<br>Min 20 |
| <b>Total Number of Contact Hours</b>              | 30                                          | <b>Credits</b>                       | 1                |

|                                                            |                                                                                                                                                                                                                                      |
|------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Understanding Corporate Communication: (2hrs)           | Define corporate communication and its importance. Identify key stakeholders in corporate. Understand workplace etiquette: communication tone, addressing colleagues, and handling conflicts. Resume Writing.                        |
| 2. Cross Culture Sensitivity: (2hrs)                       | Knowledge, awareness and acceptance of other cultures/caste/creed. Being of oneness and understanding the work environment.                                                                                                          |
| 3. Effective Communication and Interpersonal Skills (2hrs) | Skills we use daily when we communicate and interact with other people, both individually and in groups. Active listening, right response and clear message. Conduct mock presentations on technical topics relevant to engineering. |
| 4. Collaboration and Team Communication: (2hrs)            | Participate in team-based activities and projects to practice collaboration. Understanding types of Leadership and inculcating leadership skills                                                                                     |
| 5. Written Communication (2hrs)                            | Passage / Email writing<br>Identify your goal. Before you write an email, ask yourself what you want the recipient to do after they've read it.<br>Writing technical documentation and reviews                                       |
| 6. Goal Setting & Mind Mapping (2hrs)                      | Development of an action plan designed to motivate and guide a person or group toward a goal. Gives one's life direction and boosts your motivation and self-confidence. Developing career Action plans                              |

|                              |                                                                                                                                                                                                                                    |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Quantitative Aptitude</b> | <b>Speed Math</b> : Number System, Multiple and factors, Divisibility Rules HCF & LCM, Squares and Cubes, BODMAS & Tables, Approximation, Decimals, Fractions, Surds & Indices                                                     |
|                              | <b>Profit &amp; Loss</b> - Concepts of SP, CP, Profit, Loss, Gain or Loss %. Marked Price & Discount problems, Successive Selling Tye, Discount                                                                                    |
|                              | <b>Percentages</b> – Percent to Decimal or Fraction Conversion, Inverse Case – Value From Percentage, relative Percentage, Successive Selling type, Dishonest Dealings, partnerships                                               |
|                              | <b>Averages</b> - Understanding Averages, relevance, meaning, properties of average, deviation method, weighted average, & solving problems.                                                                                       |
| <b>Logical Aptitude</b>      | <b>Logical Aptitude</b> – Syllogism, Venn diagram method, three statement syllogism, deductive and inductive reasoning, introduction to puzzle and games organizing information, parts of an argument, common flaws, arguments and |

|  |                                                                                                                                                                                                                                                                                                                            |
|--|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | assumptions.                                                                                                                                                                                                                                                                                                               |
|  | <b>Coding &amp; Decoding</b> – Letter Coding, Number coding, symbol coding,<br><b>Crypt arithmetic</b> – basic concepts, addition, subtraction, multiplication of coded alphabets, types of cryptarithmic<br>Concept of EJOTY, Opposite Letter, Reversing the alphabets. Jumbling of Letter, Finding Codes of Derivatives. |
|  | <b>Image Analysis</b> - Paper cutting & Folding, Mirror & Water Image, Cubes and Dice, Analogy, Find the odd one out, Rule Detection. Cubes and dice                                                                                                                                                                       |
|  | <b>Series Completion</b> - Basics of Next no, Missing no and Wrong no and problemson that. Solving various types of Letter series and understanding different types.                                                                                                                                                       |
|  | <b>Odd Man Out</b> - Following certain patterns and groups. Identifying the errors/odd one in the group.                                                                                                                                                                                                                   |
|  | Visual sequence, visual analogy and classification, single & multiple comparisons, linear sequencing                                                                                                                                                                                                                       |
|  | <b>Logical Puzzles</b> - K-level thinking, Arithmetic Puzzles and stick puzzles                                                                                                                                                                                                                                            |



13/8

# 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: 25MAC141 (Common to CSE, ISE, AIML)**

|                 |                |                               |
|-----------------|----------------|-------------------------------|
| <b>L:T:P:J</b>  | <b>2:2:0:0</b> | <b>CIA: 50</b>                |
| <b>Credits:</b> | <b>03</b>      | <b>SEA: 50</b>                |
| <b>Hours:</b>   | <b>40</b>      | <b>SEA Duration: 03 Hours</b> |

**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.

| <b>Module-1: Curve fitting &amp; Statistical methods</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |  | <b>No. of hours</b>          | <b>Blooms cognitive Levels</b>      |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|------------------------------|-------------------------------------|
| <p><i>Examples from Engineering that require curve fitting and statistical methods.</i></p> <p><b>Curve Fitting:</b> Curve fitting by the method of least squares-fitting the curves of the form: <math>y = ax+b</math>, <math>y = ax^b</math> and <math>y = ax^2 + bx + c</math>.</p> <p><b>Statistical methods:</b> Introduction to Moments, Skewness, Kurtosis and problems. Karl Pearson's coefficient of correlation and lines of regression.</p> <p><i>Experiential Learning component: Problems on curve fitting and statistical methods</i></p>                                                                                                                                                                                                       |  | <b>L: 04</b><br><b>T: 04</b> | <b>L1</b><br><b>L2</b><br><b>L3</b> |
| <b>Module-2: Probability distributions &amp; Joint probability distribution</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |  |                              |                                     |
| <p><i>Examples from Engineering that require Probability and Joint probability distribution</i></p> <p><b>Probability distributions:</b> Review of basic probability theory. Discrete and continuous Random variables, probability mass/density functions (definitions only). Binomial, Poisson, exponential and normal distributions (without proof).</p> <p><b>Joint probability distribution:</b> Joint Probability distribution for two discrete random variables, expectation, covariance and correlation.</p> <p><i>Experiential Learning component: Problems on Binomial, Poisson, Exponential and Normal distributions</i></p>                                                                                                                        |  | <b>L: 04</b><br><b>T: 04</b> | <b>L1</b><br><b>L2</b><br><b>L3</b> |
| <b>Module-3: Markov chain &amp; Sampling theory</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |  |                              |                                     |
| <p><i>Examples from Engineering that require Markov Chain and Sampling Theory</i></p> <p><b>Markov chain:</b> 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.</p> <p><b>Sampling theory:</b> 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.</p> <p><i>Experiential Learning component: Problems on Markovian processes and, Sampling Theory</i></p> |  | <b>L: 04</b><br><b>T: 04</b> | <b>L1</b><br><b>L2</b><br><b>L3</b> |
| <b>Module-4: Queuing theory</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |  |                              |                                     |
| <p><i>Examples from Engineering that require queueing theory</i></p> <p>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: <math>\infty</math>/FCFS), when <math>\lambda_n = \lambda</math> and <math>\mu_n = \mu (\lambda &lt; \mu)</math>, Multiple server Poisson queueing model with infinite capacity (M/M/S: <math>\infty</math>/ FCFS), when <math>\lambda_n = \lambda</math> for all <math>n, (\lambda &lt; S\mu)</math>,</p> <p><i>Experiential Learning component: Problems on (M/M/1: <math>\infty</math>/FCFS) and (M/M/S: <math>\infty</math>/ FCFS) queueing models</i></p>                                            |  | <b>L: 04</b><br><b>T: 04</b> | <b>L1</b><br><b>L2</b><br><b>L3</b> |
| <b>Module-5: Graph theory</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |  |                              |                                     |
| <p><i>Examples from Engineering that require graph theory</i></p> <p>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.</p> <p><i>Experiential Learning component: Problems on detection of planar and non-planar graphs</i></p>                                                                                                                                                                        |  | <b>L: 04</b><br><b>T: 04</b> | <b>L1</b><br><b>L2</b><br><b>L3</b> |

**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, 10<sup>th</sup> Edition(Reprint), 2016.
2. B. S. Grewal: "Higher Engineering Mathematics", Khanna Publishers, 44<sup>th</sup> 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", 6<sup>th</sup> Edition, McGraw-Hill Book Co., New York, 1995.
6. B. V. Ramana: "Higher Engineering Mathematics" 11<sup>th</sup> Edition, Tata McGraw-Hill, 2010.
7. Srimanta Pal & Subodh C. Bhunia: "Engineering Mathematics", Oxford University Press, 3<sup>rd</sup> 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:25ISE142**

|                 |                |                      |                |
|-----------------|----------------|----------------------|----------------|
| <b>L:T:P:J</b>  | <b>3:1:1:0</b> | <b>CIA:</b>          | <b>50</b>      |
| <b>Credits:</b> | <b>03</b>      | <b>SEA:</b>          | <b>50</b>      |
| <b>Hours:</b>   | <b>50</b>      | <b>SEA Duration:</b> | <b>3 HOURS</b> |

**Course Learning Objectives: The students will be able to**

1. Apply the fundamentals of ARM-based embedded systems, including ARM architecture, instruction sets, hardware components, selection criteria, and embedded system design principles, to solve embedded system problems.
2. Develop proficiency in programming the ARM7TDMI/LPC2148 microcontroller, using Embedded C and Keil  $\mu$ Vision-4 tools, and to implement and test embedded applications using appropriate instructions and interfacing techniques.
3. Apply real-time operating system (RTOS) concepts and system integration techniques to design, test, and evaluate ARM-based embedded systems on development boards.

| <b>Module-1</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             | <b>No. of Hours</b> | <b>Blooms cognitive Levels</b> |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------|--------------------------------|
| 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                          |
| <b>Module-2</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                     |                                |
| 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.<br>List of programs:<br>1. Write a program to add an array of 16 bit numbers & store the 32 bit result in internal RAM<br>2. Write a program to find the square of a number (1to10) using look-up table.<br>3. Write a program to find the largest number in an array of 32 numbers.<br>4. Write a program to arrange a series of 32 bit numbers in ascending order.<br>5. Write a program to count the number of ones and zeros in two consecutivememory locations. | 10                  | Apply                          |
| <b>Module-3</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                     |                                |
| Embedded System Components: Embedded Vs General computing system, History of embedded systems, Classification of Embedded systems, Major applications areas of embedded systems.<br>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.                                                                                                                                                                                                                                                                                                                                                                                                                    | 10                  | Apply                          |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |    |       |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|-------|
| <p>List of programs:</p> <p>6. Display “Hello World” message using Internal UART.</p> <p>7. Interface a Stepper motor and rotate it in clock wise and anti-clock wise direction.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |    |       |
| <b>Module-4</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |    |       |
| <p>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).</p> <p>List of programs:</p> <p>8. Demonstrate the use of an external interrupt to toggle an LED On/Off.</p> <p>9. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.</p> | 10 | Apply |
| <b>Module-5</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |    |       |
| <p>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, <b>target hardware debugging</b>, Magnifying Glass, Multimeter, Digital CRO, Logic Analyzer, Function Generator, <b>boundary scan</b>.</p> <p>List of programs:</p> <p>10. Demonstrate the use of an external interrupt to toggle an LED On/Off.</p> <p>11. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.</p>                                         | 10 | Apply |

| <b>Course Outcomes: After completing the course, the students will be able to</b> |                                                                                                                                                                             |
|-----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CO 1</b>                                                                       | Apply ARM7 processor architecture concepts and ARM instruction sets to develop and execute assembly language programs for embedded applications.                            |
| <b>CO 2</b>                                                                       | Apply assembly language programming techniques to implement arithmetic, data manipulation, sorting, and memory management operations in ARM-based embedded systems.         |
| <b>CO 3</b>                                                                       | Apply interfacing techniques to integrate and control embedded system peripherals such as UART, LEDs, seven-segment displays, switches, and stepper motors.                 |
| <b>CO4</b>                                                                        | Apply RTOS concepts, debugging tools, hardware-software co-design principles, and testing methodologies for the development and validation of 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. 6. Embedded Systems: Architecture, Programming & Design", published in 2014 by McGraw-Hill Education, ISBN: 9789332901506.

**Web links and Video Lectures:**

- <https://academicearth.org/>
- [https://onlinecourses.nptel.ac.in/noc20\\_ee98/preview](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 & Engineering

SEMESTER: IV

Course Name: Database Management System

Course Code: 25ISE143

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

**Course Learning Objectives: The students will be able to**

- 1 Apply fundamental database concepts, SQL, and NoSQL techniques to create, manage, and manipulate databases for real-world applications.
- 2 Develop relational database schemas and implement SQL queries using relational algebra concepts to solve database problems.
- 3 Analyze database designs using normalization techniques to identify and eliminate redundancy and maintain data integrity.
- 4 Analyze transaction processing and concurrency control mechanisms to evaluate their impact on database consistency, reliability, and performance.

| Module-1: Database System Concepts, Data Modeling                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | No. of Hours | Blooms cognitive Levels |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|-------------------------|
| Databases and Database Users: Characteristics of database Approach, Advantages of using the DBMS Approach.<br>Database System Concepts and Architecture: Data Models-Schemas and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces.<br>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.<br><b>Practical component:</b><br>Draw ER Diagram for the following Databases using GitMind software.<br>Order Database<br>Library Database<br>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<br><b>Practical component:</b><br>Create Schema, insert at least 5 records in each table and add appropriate constraints for the following Employee Database using ORACLE or MySQL DBMS under LINUX/Windows environment<br>STUDENT (Student_ID, Student_Name, Gender, Department, Phone)<br>COURSE (Course_ID, Course_Name, Credits)<br>FACULTY (Faculty_ID, Faculty_Name, Designation)<br>ENROLLMENT (Enroll_ID, Student_ID, Course_ID, Semester, Grade)<br>Write SQL queries to<br>1. Retrieve Student Enrollment Details<br>2. Find Students Enrolled in More Than One Course | 10 | Apply |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|-------|

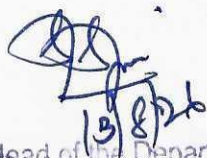
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |    |         |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|---------|
| 3. List Faculty and Courses They Teach<br>4. Update Student Grade<br>5. Delete a Course and Update Related Tables                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |    |         |
| <b>Module-3: SQL</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |    |         |
| <p><b>Basic SQL:</b> 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</p> <p><b>More SQL: Complex Queries, Triggers, Views and Schema Modification:</b> Complex SQL Retrieval Queries, Specifying Constraints as Assertions and actions as Triggers, Views (Virtual Tables) in SQL.</p> <p><b>Practical component:</b><br/>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.</p> <p><b>EMPLOYEE</b> (Emp_ID, Emp_Name, City, Salary, Dept_ID)<br/> <b>DEPARTMENT</b> (Dept_ID, Dept_Name, Location)<br/> <b>PROJECT</b> (Proj_ID, Proj_Name, Budget, Emp_ID)</p> <p>Write SQL queries to</p> <ol style="list-style-type: none"> <li>Count the employees whose salary is above the average salary of employees working in Bangalore.</li> <li>Find the names and IDs of all departments having more than one employee.</li> <li>List all departments and indicate those that have and do not have employees in their locations (Using UNION).</li> <li>Create a View that finds the employee managing the project with the highest budget.</li> </ol>                                                       | 10 | Apply   |
| <b>Module-4: Functional Dependencies and Normalization</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |    |         |
| <p>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]</p> <p><b>Practical component:</b><br/>Create Schema, insert at least 5 records for each table and add appropriate constraints for the following Movie Database using ORACLE or MySQL DBMS under LINUX/Windows environment.</p> <p><b>MOVIE</b> (Movie_ID, Title, Release_Year, Language, Director_ID)<br/> <b>DIRECTOR</b> (Director_ID, Director_Name, Country)<br/> <b>ACTOR</b> (Actor_ID, Actor_Name, Gender)<br/> <b>CASTS</b> (Movie_ID, Actor_ID, Role)<br/> <b>RATINGS</b> (Movie_ID, User_ID, Rating, Review_Date)</p> <p>Write SQL queries to</p> <ol style="list-style-type: none"> <li>List all movie IDs for movies involving an actor named 'Khan', either as cast member or directed by a director from India.</li> <li>Show the resulting ratings if all movies of genre/language 'English' are given a 1-point rating boost.</li> <li>Find total ratings, maximum rating, and average rating of movies released in the '2026' year.</li> <li>Find movies that have and do not have ratings (using UNION).</li> </ol> | 10 | Analyze |

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

| <b>Course Outcomes: After completing the course, the students will be able to</b> |                                                                                                                                           |
|-----------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| CO 1                                                                              | Apply ER modeling techniques to design simple database schemas based on real-world scenarios.                                             |
| CO 2                                                                              | Apply the concepts of relations on RDBMS, constraints, joints using relational algebra operators.                                         |
| CO 3                                                                              | Apply Structured Query Language for database manipulation.                                                                                |
| CO 4                                                                              | Analyze functional dependencies to normalize relations of relational database                                                             |
| CO.5                                                                              | Analyze transactions processing, schedules protocols, serializability issues, deadlocks in DBMS and concepts of NoSQL with its advantages |

| <b>Reference Books</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. Fundamentals of Database Systems, Ramez Elmasri, Shamkant B. Navathe, 8th Edition, 2023, Pearson Education, ISBN-13: 978-0137340530.<br>2. Database System Concepts, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 8th Edition, 2020, McGraw-Hill Education, ISBN-13: 978-9390319108.<br>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. |

| <b>Web links and Video Lectures:</b>                                                                                                                                                                                                                                     |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ul style="list-style-type: none"> <li>• <a href="https://nptel.ac.in/courses/106105175">https://nptel.ac.in/courses/106105175</a></li> <li>• <a href="https://aws.amazon.com/what-is/website-database/">https://aws.amazon.com/what-is/website-database/</a></li> </ul> |

  
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: 25ISE144**

|                                      |                |                      |                |
|--------------------------------------|----------------|----------------------|----------------|
| <b>L:T:P:J</b>                       | <b>3:0:2:0</b> | <b>CIE Marks:</b>    | <b>50</b>      |
| <b>Credits:</b>                      | <b>04</b>      | <b>SEE Marks:</b>    | <b>50</b>      |
| <b>Total Number of Lecture Hours</b> | <b>50</b>      | <b>SEE Duration:</b> | <b>3 Hours</b> |

## **Course objectives:**

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. Apply 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 multiple algorithmic solutions for the same problem to determine their efficiency, scalability, and suitability under practical constraints.
4. Analyze algorithmic approaches for computationally hard problems by their complexity classes such as P, NP, NP-Complete, and NP-Hard, and by selecting feasible solution strategies.

| <b>Module 1</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  | <b>Teaching Hours</b> | <b>Blooms Cognitive Levels</b> |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|--------------------------------|
| <p>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.</p> <p>Practical Programs not limited to :</p> <ol style="list-style-type: none"><li>1. Implement the following concepts:<ol style="list-style-type: none"><li>a) A library maintains a list of book accession numbers. Write a program to search for a given accession number using the Linear Search technique and display whether the book is available along with its position in the list.</li><li>b) A company records the monthly sales of <math>n</math> products. Write a program to determine the product with the highest sales and the product with the lowest sales.</li></ol></li><li>2. Implement the following:<ol style="list-style-type: none"><li>a) 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.</li><li>b) Find the next three terms of the sequence 15, 23, 38, 61, ... Fibonacci series of the given number using recursion.</li></ol></li></ol> | 10                    | Apply                          |
| <b>Module 2</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                       |                                |
| <p>Brute Force: Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching.</p> <p>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.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | 10                    | Apply                          |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |    |         |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|---------|
| <p><b>Practical Programs:</b></p> <ol style="list-style-type: none"> <li>3. A university needs to prepare a merit list based on student scores. Implement Selection Sort to arrange the scores in ascending order.</li> <li>4. A retail store wants to organize product prices for easy comparison. Implement Bubble Sort to sort the prices in ascending order.</li> <li>5. A hospital maintains a list of patient IDs. Implement Sequential Search to determine whether a given patient ID exists in the records.</li> <li>6. A journal stores book ISBN numbers in sorted order. Implement Binary Search to locate a given ISBN number and measure the time required for the search.</li> <li>7. A company records monthly sales figures for multiple products. Implement the Divide and Conquer technique to identify the highest and lowest sales values and measure the execution time.</li> <li>8. An examination portal needs to rank students based on their scores. Implement Merge Sort to sort the scores and analyze the time taken for different input sizes. Specify the time efficiency class of this algorithm.</li> <li>9. A bank needs to organize transaction amounts for auditing purposes. Implement Quick Sort to sort the transaction values and analyze the time taken for different input sizes. Specify the time efficiency class of this algorithm</li> <li>10. A software development project consists of tasks with dependency constraints. Implement Topological Sort using the Source Removal Method to determine a valid sequence for task execution.</li> </ol> |    |         |
| <p><b>Module 3</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |    |         |
| <p>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.</p> <p><b>Practical Programs:</b></p> <ol style="list-style-type: none"> <li>11. An ATM must dispense a requested amount using the minimum number of currency notes. Write a program using the Greedy method to determine the notes required and measure the execution time.</li> <li>12. A company has multiple jobs with deadlines and profits. Write a program using the Greedy method to schedule jobs and maximize total profit.</li> <li>13. A telecom company plans to connect all its offices with minimum cable cost. Write a program using Prim's algorithm to find the Minimum Cost Spanning Tree.</li> <li>14. A city wants to connect all its locations with roads at minimum construction cost. Write a program using Kruskal's algorithm to find the Minimum Cost Spanning Tree.</li> <li>15. A navigation system must determine the shortest travel distance from a source location to all other locations. Write a program using Dijkstra's algorithm to find the shortest paths.</li> </ol>                                                                                                                                                                                                                                                                                                       | 10 | Apply   |
| <p><b>Module 4</b></p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |    |         |
| <p>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.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 10 | Analyze |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |    |         |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|---------|
| <p>Practical Programs:</p> <ol style="list-style-type: none"> <li>16. A logistics company wants to determine the shortest travel distance between every pair of warehouses. Write a program to find the shortest paths between all warehouses. (All-Pairs Shortest Paths)</li> <li>17. A social networking platform wants to identify whether any user can reach another user through a chain of connections. Write a program to determine connectivity among all users. (Transitive Closure)</li> <li>18. A delivery vehicle has limited carrying capacity. Write a program to select packages that maximize profit without exceeding the vehicle's capacity using dynamic programming.</li> <li>19. An airline booking system needs to find the minimum travel cost from a source airport to all other airports, where promotional discounts may result in negative fare values. Write a program to determine the cheapest routes using the Bellman-Ford algorithm.</li> <li>20. A sales representative must visit multiple client locations exactly once and return to the office. Write a program to determine the minimum travel cost for the entire tour using dynamic programming.</li> </ol>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |    |         |
| <b>Module 5</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |    |         |
| <p>Backtracking: General method, 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</p> <p>Practical Programs:</p> <ol style="list-style-type: none"> <li>21. A security agency needs to place N surveillance cameras on an <math>N \times N</math> grid such that no two cameras monitor each other directly. Write a program using Backtracking to find a valid placement using back tracking.</li> <li>22. A customer wants to purchase items whose prices add up exactly to a given budget. Write a program using Backtracking to identify all possible item combinations that match the budget.</li> <li>23. A tourist wants to visit every attraction 'n' in a city 'G' exactly once and return to the starting point. Write a program using Backtracking to find all Hamiltonian Cycles in a connected undirected Graph G of n vertices.</li> <li>24. A teacher needs to assign N students to N tasks such that each task is assigned to exactly one student with minimum total cost. Write a program using Backtracking to determine the optimal assignment.</li> <li>25. A salesperson must visit multiple cities exactly once and return to the starting city while minimizing travel cost. Write a program using the Branch and Bound method to find the optimal tour.</li> <li>26. A logistics company needs to load items into a vehicle with limited capacity to maximize profit. Write a program using the Branch and Bound method to select the most valuable set of items.</li> </ol> | 10 | Analyze |

**Course outcomes:** The students should be able to:

|            |                                                                                                                                                                                                               |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CO1</b> | Apply fundamental algorithmic paradigms such as brute force, divide and conquer, greedy, dynamic programming, backtracking, and branch and bound to implement standard computational problems.                |
| <b>CO2</b> | Apply appropriate data structures and algorithmic techniques to design and implement solutions for sorting, searching, graph-based, and optimization problems.                                                |
| <b>CO3</b> | Analyze the time and space complexity of algorithms across different paradigms using theoretical and empirical approaches, including performance comparison and efficiency classification.                    |
| <b>CO4</b> | Analyze and differentiate algorithmic strategies in terms of their suitability for solving computational problems, including NP-complete and NP-hard problems, and interpret their computational limitations. |

**Reference Books:**

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 4<sup>th</sup> 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](https://www.tutorialspoint.com/design_and_analysis_of_algorithms/design_and_analysis_of_algorithms_tutorial.pdf).



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: Foundations of Machine Learning**  
**Course Code : 25ISE145**

|                               |         |               |         |
|-------------------------------|---------|---------------|---------|
| L:T:P:J:                      | 3:0:2:0 | CIE Marks:    | 50      |
| Credits:                      | 03      | SEE Marks:    | 50      |
| Total Number of Lecture Hours | 50      | SEE Duration: | 3 Hours |

**Course objectives:**

1. Demonstrate the Machine Learning lifecycle, types of machine learning, and perform data visualization using Python libraries such as Matplotlib and Seaborn.
2. Develop supervised machine learning models using algorithms such as Linear Regression, Logistic Regression, and Naive Bayes for prediction and classification tasks.
3. Apply unsupervised learning techniques such as K-Means Clustering, Hierarchical Clustering, PCA, and KNN for data analysis, dimensionality reduction, and classification problems.
4. Apply neural networks, deep learning concepts, explainable AI, ethical and privacy-preserving machine learning principles, AutoML, and MLOps fundamentals to analyze and address real-world problems in healthcare, finance, cybersecurity, smart cities, agriculture, and Industry 4.0

**Module 1**

**Teaching  
Hours**

**Blooms  
Cognitive  
Levels**

Definition of Machine Learning, The ML Lifecycle. Types of learning: Supervised, Unsupervised, Semi-supervised, and Reinforcement Learning. Challenges of Machine Learning. Data Visualization: General matplotlib, Simple line plots, Simple scatter plots, Histograms, Customizing matplotlib, Visualization with Seaborn.

**10**

**Apply**

**Sample Programs:**

1. Develop a code to draw a Bar Plot using Matplot lib.
2. Develop a code to draw a Scatter Plot using Matplot lib.
3. Develop a code to draw a Histogram Plot using Matplot lib.

**Module 2**

Introduction, Examples of Supervised Machine Learning, Classification Model, Classification Learning Steps, Navie Bayesian algorithm, Introduction to Linear Regression, Simple vs. Multiple Linear Regression. Cost Function (Mean Squared Error), Gradient Descent optimization algorithm. Overfitting and Underfitting. Logistic Regression for binary classification, Sigmoid function, Decision Boundaries

**Sample Programs:**

5. Develop a machine learning model to predict a student's marks based on the number of hours studied using Simple Linear Regression.
6. Design a Naive Bayes classifier to classify emails as spam or non-spam based on the words present in the email.
7. Develop a Multiple Linear Regression model to predict house prices based on features such as area, number of bedrooms, and location.

**10**

**Apply**

**Module 3**

**Introduction to Unsupervised Learning:** Clustering vs. Classification. K-Means Clustering algorithm, Elbow Method for finding optimal . Hierarchical Clustering (Agglomerative).

**Sample Programs:**

8. Apply the K-Nearest Neighbors (KNN) algorithm on a suitable dataset for classification and prediction.
9. Perform k-Means clustering on a real-world dataset.

**10**

**Apply**

**Module 4**

**Unsupervised Learning and Model Evaluation:** Dimensionality Reduction: Introduction to Principal Component Analysis (PCA) and its importance in handling high-dimensional engineering data, Association Rule Mining (Introduction), Apriori Algorithm, Model evaluation and selection, Bias-Variance trade-off, Hyperparameter tuning. Introduction to pipelines in machine learning.

10

Apply

**Sample Programs:**

10. Compare multiple machine learning models using evaluation metrics.
11. Apply Principal Component Analysis (PCA) on a suitable dataset to reduce dimensionality and analyze the transformed data.

**Module 5**

**Advanced Topics and Responsible Machine Learning:** Introduction to Neural Networks, Overview of Deep Learning, Transfer Learning (Introduction), Explainable Machine Learning, Ethical issues in Machine Learning, Bias and fairness in ML models, Privacy-preserving machine learning, AutoML (Introduction), MLOps fundamentals, Machine Learning applications in healthcare, finance, cybersecurity, smart cities, agriculture, and Industry 4.0, Future trends in Machine Learning

10

Apply

**Course outcomes:** The students should be able to:

- |            |                                                                                                                                                                                                                                                                                         |
|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>CO1</b> | Demonstrate the Machine Learning lifecycle, types of machine learning, and perform data visualization using Python libraries such as Matplotlib and Seaborn                                                                                                                             |
| <b>CO2</b> | Develop supervised machine learning models using algorithms such as Linear Regression, Logistic Regression, and Naive Bayes for prediction and classification tasks                                                                                                                     |
| <b>CO3</b> | Apply unsupervised learning techniques such as K-Means Clustering, Hierarchical Clustering, PCA, and KNN for data analysis, dimensionality reduction, and classification problems                                                                                                       |
| <b>CO4</b> | Apply neural networks, deep learning concepts, explainable AI, ethical and privacy-preserving machine learning principles, AutoML, and MLOps fundamentals to analyze and address real-world problems in healthcare, finance, cybersecurity, smart cities, agriculture, and Industry 4.0 |

**Reference Books:**

- 1) Introduction to Machine Learning with Python: A Guide for Data Scientists Andreas C. Müller, Sarah Guido O'Reilly Media First Edition, 2016
- 2) Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow Aurélien Géron O'Reilly Media Third Edition, 2022
- 3) Understanding Machine Learning, Shai Shalev-Shwartz and Shai Ben-David. Cambridge University Press, 2017.
- 4) Tom M. Mitchell, -Machine Learning, McGraw-Hill Education (India) Private Limited, 2013.

**Web Links:**

- [https://onlinecourses.nptel.ac.in/e-learning/preview/noc26\\_cs102](https://onlinecourses.nptel.ac.in/e-learning/preview/noc26_cs102)
- <https://www.coursera.org/learn/foundations-of-machine-learning>

  
 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: 25MAC141 (Common to CSE, ISE, AIML)**

|                 |                |                               |
|-----------------|----------------|-------------------------------|
| <b>L:T:P:J</b>  | <b>2:2:0:0</b> | <b>CIA: 50</b>                |
| <b>Credits:</b> | <b>03</b>      | <b>SEA: 50</b>                |
| <b>Hours:</b>   | <b>40</b>      | <b>SEA Duration: 03 Hours</b> |

**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.

| <b>Module-1: Curve fitting &amp; Statistical methods</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |  | <b>No. of hours</b>                     | <b>Blooms cognitive Levels</b> |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--|-----------------------------------------|--------------------------------|
| <p><i>Examples from Engineering that require curve fitting and statistical methods.</i></p> <p><b>Curve Fitting:</b> Curve fitting by the method of least squares-fitting the curves of the form: <math>y = ax+b</math>, <math>y = ax^b</math> and <math>y = ax^2 + bx + c</math>.</p> <p><b>Statistical methods:</b> Introduction to Moments, Skewness, Kurtosis and problems. Karl Pearson's coefficient of correlation and lines of regression.</p> <p><i>Experiential Learning component: Problems on curve fitting and statistical methods</i></p>                                                                                                                                                                                                       |  | <p><b>L: 04</b></p> <p><b>T: 04</b></p> | <p>L1</p> <p>L2</p> <p>L3</p>  |
| <b>Module-2: Probability distributions &amp; Joint probability distribution</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |  |                                         |                                |
| <p><i>Examples from Engineering that require Probability and Joint probability distribution</i></p> <p><b>Probability distributions:</b> Review of basic probability theory. Discrete and continuous Random variables, probability mass/density functions (definitions only). Binomial, Poisson, exponential and normal distributions (without proof).</p> <p><b>Joint probability distribution:</b> Joint Probability distribution for two discrete random variables, expectation, covariance and correlation.</p> <p><i>Experiential Learning component: Problems on Binomial, Poisson, Exponential and Normal distributions</i></p>                                                                                                                        |  | <p><b>L: 04</b></p> <p><b>T: 04</b></p> | <p>L1</p> <p>L2</p> <p>L3</p>  |
| <b>Module-3: Markov chain &amp; Sampling theory</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |  |                                         |                                |
| <p><i>Examples from Engineering that require Markov Chain and Sampling Theory</i></p> <p><b>Markov chain:</b> 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.</p> <p><b>Sampling theory:</b> 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.</p> <p><i>Experiential Learning component: Problems on Markovian processes and, Sampling Theory</i></p> |  | <p><b>L: 04</b></p> <p><b>T: 04</b></p> | <p>L1</p> <p>L2</p> <p>L3</p>  |
| <b>Module-4: Queuing theory</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |  |                                         |                                |
| <p><i>Examples from Engineering that require queueing theory</i></p> <p>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: <math>\infty</math>/FCFS), when <math>\lambda_n = \lambda</math> and <math>\mu_n = \mu (\lambda &lt; \mu)</math>, Multiple server Poisson queueing model with infinite capacity (M/M/S: <math>\infty</math>/ FCFS), when <math>\lambda_n = \lambda</math> for all <math>n, (\lambda &lt; S\mu)</math>,</p> <p><i>Experiential Learning component: Problems on (M/M/1: <math>\infty</math>/FCFS) and (M/M/S: <math>\infty</math>/ FCFS) queueing models</i></p>                                            |  | <p><b>L: 04</b></p> <p><b>T: 04</b></p> | <p>L1</p> <p>L2</p> <p>L3</p>  |
| <b>Module-5: Graph theory</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |  |                                         |                                |
| <p><i>Examples from Engineering that require graph theory</i></p> <p>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.</p> <p><i>Experiential Learning component: Problems on detection of planar and non-planar graphs</i></p>                                                                                                                                                                        |  | <p><b>L: 04</b></p> <p><b>T: 04</b></p> | <p>L1</p> <p>L2</p> <p>L3</p>  |

**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, 10<sup>th</sup> Edition(Reprint), 2016.
2. B. S. Grewal: "Higher Engineering Mathematics", Khanna Publishers, 44<sup>th</sup> 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", 6<sup>th</sup> Edition, McGraw-Hill Book Co., New York, 1995.
6. B. V. Ramana: "Higher Engineering Mathematics" 11<sup>th</sup> Edition, Tata McGraw-Hill, 2010.
7. Srimanta Pal & Subodh C. Bhunia: "Engineering Mathematics", Oxford University Press, 3<sup>rd</sup> 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/>

# BNM Institute of Technology

## Syllabus for Soft Skills

### SEMESTER – IV

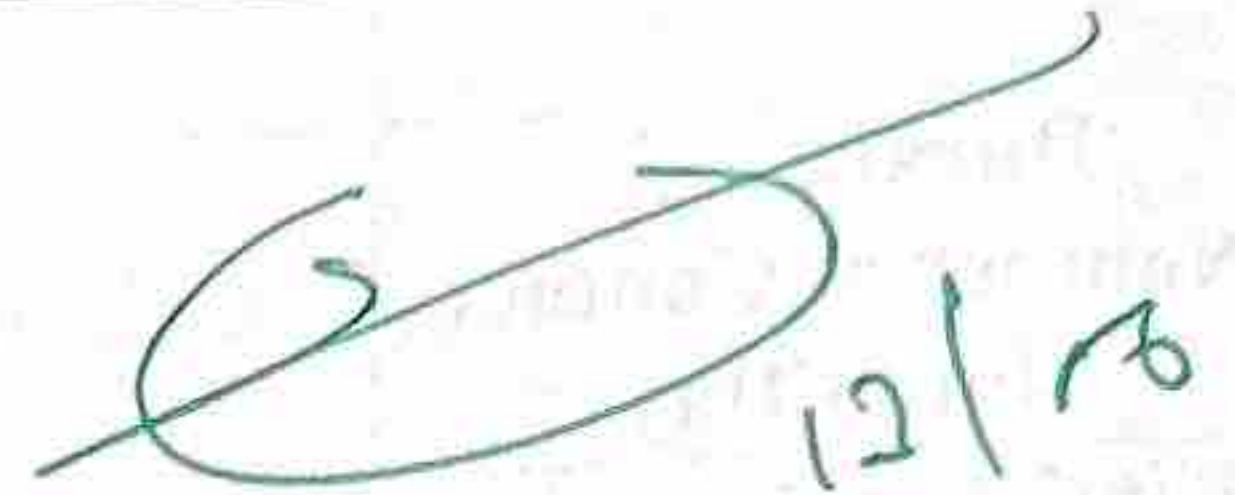
Year 2025-26

|                                                   |                                                                                                                                                                                                                                                                                                                                                               |                                      |                  |
|---------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------|------------------|
| <b>Subject Name</b>                               | Quantitative Aptitude and Logical Reasoning                                                                                                                                                                                                                                                                                                                   | <b>Daily/Weekly Assessment Marks</b> | Max 50<br>Min 20 |
| <b>Subject Code</b>                               | 25SFT148                                                                                                                                                                                                                                                                                                                                                      |                                      |                  |
| <b>Duration &amp; Number of Contact Hours/Day</b> | One-week continuous classes<br>6 hrs per day                                                                                                                                                                                                                                                                                                                  | <b>Final Assessment Marks</b>        | Max 50<br>Min 20 |
| <b>Total Number of Contact Hours</b>              | 30                                                                                                                                                                                                                                                                                                                                                            | <b>Credits</b>                       | 1                |
| <b>Module 1<br/>(Logical Reasoning)</b>           | <b>Ages &amp; Blood Relation</b> - Generation Tree, Family Tree Problems, Statement Based Questions, Coded Blood Relation Question, Blood Relation Based Puzzles.                                                                                                                                                                                             |                                      |                  |
|                                                   | <b>Seating Arrangement</b> - Single or Double rows facing each other or away from each other in the same direction<br>Circular Seating Arrangement<br>· Uni- & Bi-directional problems on<br>· Circular, Square, Rectangular, Hexagonal tables                                                                                                                |                                      |                  |
|                                                   | <b>Distance &amp; Direction</b> - Distance and Displacement between any two points as well as puzzles based on that, Concept of Shadows.                                                                                                                                                                                                                      |                                      |                  |
|                                                   | <b>Clocks &amp; Calendars</b> - Understanding concepts and basic formula along with solving different types of problems.                                                                                                                                                                                                                                      |                                      |                  |
|                                                   | <b>Probability</b> - Single event probability, multi event probability, independent events and dependent events, mutually exclusive events, non-mutually exclusive events, combination method for finding the outcomes.                                                                                                                                       |                                      |                  |
|                                                   | <b>Analogies</b> - Drawing similarity between different but sufficiently similar events, situations, or circumstances using tips and tricks, find the odd one out, rule detection,                                                                                                                                                                            |                                      |                  |
| <b>Module 2<br/>(Quantitative Aptitude)</b>       | <b>Ratio and Proportion</b><br>· Simple Ratios, Compound Ratios<br>· Comprehend and Dividend<br>· Direct & Indirect Proportions<br>· Problems on ages,                                                                                                                                                                                                        |                                      |                  |
|                                                   | <b>Mixtures &amp; Alligation</b> – 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.                                                                                            |                                      |                  |
|                                                   | <b>Data Interpretation</b> – 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. |                                      |                  |
|                                                   | <b>Simple interest and Compound Interest</b> - 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

  
12/10

# BNM Institute of Technology

Syllabus for Soft Skills

SEMESTER – IV

Year 2025-26

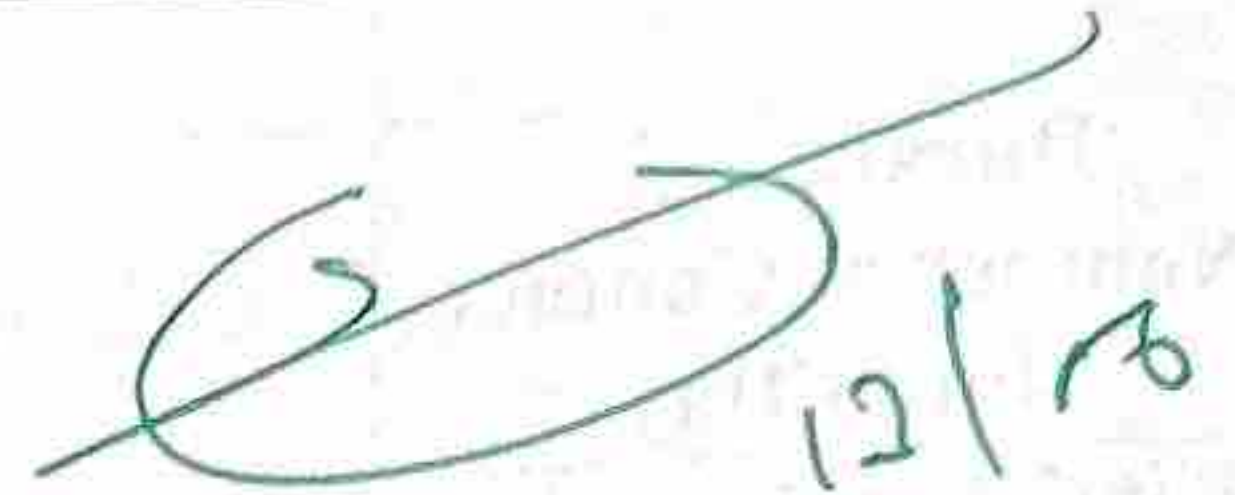
|                                        |                                                                                                                                                                                                                                         |                               |                  |
|----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|------------------|
| Subject Name                           | Quantitative Aptitude and Logical Reasoning                                                                                                                                                                                             | Daily/Weekly Assessment Marks | Max 50<br>Min 20 |
| Subject Code                           | 25SFT148                                                                                                                                                                                                                                |                               |                  |
| Duration & Number of Contact Hours/Day | One-week continuous classes<br>6 hrs per day                                                                                                                                                                                            | Final Assessment Marks        | Max 50<br>Min 20 |
| Total Number of Contact Hours          | 30                                                                                                                                                                                                                                      | Credits                       | 1                |
| Module 1<br>(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<br>Circular Seating Arrangement<br>· Uni- & Bi-directional problems on<br>· 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,                                                             |                               |                  |

|                                     |                                                                                                                                                                                                                                                                                                                                                        |
|-------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Module 2<br>(Quantitative Aptitude) | Ratio and Proportion<br>· Simple Ratios, Compound Ratios<br>· Comprehend and Dividend<br>· Direct & Indirect Proportions<br>· Problems on ages,                                                                                                                                                                                                        |
|                                     | 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

  
12/10