Nirmala Memorial Foundation College of Commerce and Science

(Autonomous)

Re-accredited by NAAC with B++, ISO 9001-2015 Certified, Recognised under section 2(f) & 12(B) of the UGC Act 1956

Permanently Affiliated to the University of Mumbai,

Syllabus

Bachelor of Science in Information Technology

(B.Sc. Information Technology)

(Semester III and IV)

(Programme Code: UGIT01)

As Per New Education Policy (NEP) 2020

(To be implemented from the Academic Year 2025-2026)

Approved in the Academic Council Meeting held on **5th July 2025**

INTRODUCTION OF THE PROGRAMME

Information technology (IT) continues to be a dynamic and rapidly evolving field with high demand for skilled professionals. The demand for IT workers is driven by various factors, and the landscape may have evolved over a period of time. NEP envisages the multidisciplinary approach thus making IT much more applicable in all fields of life. This facilitates multi-institutional mobility of the students within India as well as abroad thus making the students attain different proficiency levels right from certificate to B.Sc Honours with Research. This new syllabus under NEP will thus enables the students for higher education, research and career in the field of IT

Aims and Objectives

The aims and objectives of a Bachelor of Science (B.Sc) program in Information Technology (IT) generally revolve around providing students with a comprehensive understanding of the principles, technologies, and applications within the field of information technology. The entire program collectively aim to produce graduates who are well-rounded IT professionals, capable of contributing to the design, development, and management of information technology systems in various industries. The specific details of the curriculum may vary among institutions offering B.Sc in Information Technology programs.

PROGRAM OUTCOME

- **Knowledge of Business Disciplines**

Develop a strong foundation in core areas of commerce such as accounting, economics, business law, management, and statistics, enabling students to understand and respond to contemporary business challenges.

- **Problem-Solving and Decision-Making Skills**

Apply logical reasoning and quantitative techniques to analyze business problems and make informed decisions in diverse organizational settings.

- **Communication and Interpersonal Skills**

Demonstrate proficiency in written, verbal, and digital communication, essential for effective collaboration and negotiation in business environments.

- **Ethical, Social, and Environmental Awareness**

Recognize ethical issues, social responsibilities, and environmental sustainability in business operations and decision-making.

- **Lifelong Learning and Adaptability**

Develop the ability to adapt to a dynamic global business environment by engaging in continuous learning, embracing technology, and cultivating personal and professional growth.

- **Technical Proficiency:**

Demonstrate a comprehensive understanding of fundamental concepts, principles, and technologies in information technology. Apply programming and software development skills to design and implement IT solutions.

- **System Thinking and Analysis:**

Apply system analysis and design methodologies to analyze and address complex problems. Design and develop IT systems that meet user requirements and organizational needs.

- **Database Management:**

Design, implement, and manage relational databases to store and retrieve information effectively. Demonstrate proficiency in using database management systems and querying languages.

- **Networking and Security:**

Understand and implement computer networks, protocols, and security measures.
Evaluate and implement security solutions to protect information systems.

- **Web Technologies:**

Develop web applications using a variety of technologies and programming languages.
Design and create user interfaces that adhere to web design principles.

- **Project Management:**

Apply project management principles to plan, execute, and deliver IT projects.
Demonstrate the ability to work effectively within project teams.

- **Emerging Technologies:**

Stay informed about and adapt to emerging technologies in the IT field.
Apply concepts of artificial intelligence, machine learning, cloud computing, and IoT to solve real-world problems.

- **Critical Thinking and Problem-Solving:**

Analyze and solve complex IT problems using critical thinking skills.
Apply problem-solving strategies to troubleshoot and resolve technical issues.

- **Communication Skills:**

Effectively communicate technical information to diverse audiences, both orally and in writing. Collaborate with team members and stakeholders to achieve common goals.

- **Ethics and Professionalism:**

Demonstrate ethical behavior and professionalism in all aspects of the IT profession.
Adhere to ethical standards and legal considerations related to information technology.
Demonstrate the ability to work effectively within project teams.

SEMESTER III
COURSE STRUCTURE

VERTICAL		COURSE CODE	SUBJECT	CREDIT
Vertical 1	Major	USIT.3MAJ01	Python for AI	2
	Major	USIT.3MAJ02	Data Structures for Problem Solving	2
	Major	USIT.3MAJ03	Operating System	2
	Major	USIT.3MAJ04	Python for AI + Data Structures for Problem Solving Practical	2
Vertical 2	Minor	USDS.3MIN01	Programming with R	2
	Minor	USDS.3MIN02	DS_Scala for Data Science	2
Vertical 3	OE	UC.3OE01	Financial Literacy	2
Vertical 4	VSC	USIT.3VSC01	Applied Mathematics	2
	SEC	***		
Vertical 5	AEC	UG.3AEC01	Hindi Bhasha: Vyavharik Prayog	2
	VEC	***		
	IKS	***		
Vertical 6	CC	UG.3CCS	Sports -I	2
	FP	UG.3FP	Field Project	2
			Total Credits	**22**

SEMESTER IV
COURSE STRUCTURE

VERTICAL		COURSE CODE	SUBJECT	CREDIT
Vertical 1	Major	USIT.4MAJ01	Core Java Techniques	2
	Major	USIT.4MAJ02	Software Engineering	2
	Major	USIT.4MAJ03	Computer Networks	2
	Major	USIT.4MAJ04	Core Java Techniques + Computer Networks Practical	2
Vertical 2	Minor	USDS.4MIN03	Advanced Python for Data Science	2
	Minor	USDS.4MIN04	Data Analysis with SAS/SPAA/R	2
Vertical 3	OE	UC.4OE01	Management Skills	2
Vertical 4	VSC	***		
	SEC	USIT.4SEC01	Arduaino Programming Practical	2
Vertical 5	AEC	UG.4AEC03	Technical writing in English	2
	VEC	***		
Vertical 6	CC	UG.4CCS	Sports-II	2
	CEP	UG.4CEP01	Community Engagement Project	2
			Total Credits	**22**

Vertical – I

Semester- III

Mandatory Courses

Name of the Course: Python Programming for AI

Sr. No.	Heading	Particulars
1	**Description the course:**	This course introduces Python, a simple yet powerful language widely used in web development, data analysis, and AI. Students will learn core programming concepts like syntax, control flow, and data structures through hands-on practice. Python's ease of use and high demand across industries make this course a valuable starting point for careers in tech, data, and software development.
2	**Vertical:**	**Major**
3	**Type:**	**Theory**
4	**Credits:**	**2 credits**
5	**Hours Allotted:**	**30 Hours**
6	**Marks Allotted:**	**50 Marks**
7	**Course Objectives:** **1.** Gain a solid understanding of Python's core features, execution model, and built-in data types. **2.** Strengthen problem-solving skills through effective use of control structures like conditionals and loops. **3.** Handle arrays, strings, and structured data efficiently using Python's rich data manipulation features. **4.** Utilize functions, modules, and string processing techniques to design flexible and reusable code. **5.** Perform file handling, apply regular expressions, and work with date and time modules to build complete Python solutions.	
8	**Course Outcomes:** **CO1.** Exhibit proficiency in Python's core features to solve diverse programming problems effectively. **CO2**. Apply control flow constructs to develop logically correct and efficient programs. **CO3**. Skilfully manipulate arrays, strings, and complex data structures for enhanced data processing. **CO4**. Create modular and optimized programs using functions, modules, and advanced string handling. **CO5.** Perform file management, utilize regular expressions, and handle date/time operations to build robust applications.	
9	**Modules:-** **Module 1 (15 hours):**	
	Basic Elements of Python Programming: Features of Python, Execution of a Python Program, Python Interpreter, Comments,	

	IDLE, Data types, Dictionary, Variables, Input Function, Output Statements, Operators, Precedence of Operators. **Control Statements:** Conditional statements (if, if-else, if-elif-else, nested if), Loops in python (while, for, nested loops), Loop manipulation statements (continue, pass, break, else) **Arrays:** Creating Arrays, Indexing and Slicing of Arrays, Basic Array Operations, Array Indexing in NumPy Arrays **Functions:** Function definition and call, Returning Results, Returning Multiple Values from a Function, Built-in Functions Pass Value by Object Reference, Parameters and Arguments, Recursive Functions, Anonymous or Lambda Functions, Modules in Python. **Strings:** Creating Strings, Working with String functions, Formatting Strings
	Module 2 (15 hours):
	List: Exploring List, Tuples and Dictionaries: List Functions and Methods, List Operations, List Slices, Nested Lists, Tuples Functions, creating a Dictionary, Operators in Dictionary, Dictionary Methods, Using for Loop with Dictionaries, Operations on Dictionaries **Files in Python:** Opening and Closing a File, Working with Text Files and Binary Files, The 'with' statement, The seek() and tell() Methods, Random Accessing of Binary Files, Zipping and Unzipping Files. **Regular Expressions:** Introduction, Sequence Characters in Regular Expressions, Special Characters in Regular Expressions, Using Regular Expression on Files, Retrieving Information from HTML File **Date And Time in Python:** Time, Date, Date and Time Now, combining date and times, formatting date and time, Working with Calendar Module Introduction: What is Artificial Intelligence? Foundations of AI, History, the state of art AI today.
10	**Text Books** 1. Learning Python, Fourth Edition by Mark Lutz Copyright © 2009 Mark Lutz. Published by O'Reilly Media, Inc. 2. Python Basics: A Practical Introduction to Python 3. Revised and Updated 4th Edition David Amos, Dan Bader, Joanna Jablonski, Fletcher Heisler

11	**Reference Books** 1. Let Us Python, Yashwant. B. Kanetkar, BPB Publication, 2019 2. Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018 3. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017	
12	**Internal Continuous Assessment: 40%**	**Semester End Examination: 60%**
13	**Continuous Evaluation through:** Class test of 15 marks Quizzes/ Presentations/ Assignments: 5 marks Total: 20 marks	**External Examination (30 Marks)– 1 hr duration**
14	**Format of Question Paper: (Semester End Examination: 30 Marks. Duration:1 hour)** Q1: Attempt any three (out of five) from Module 1 (15 marks) Q2: Attempt any three (out of five) from Module 2 (15 marks)	

Name of the Course: Data Structures for Problem Solving

Sr. No.	Heading	Particulars
1	**Description the course:**	Data Structures is a core subject that deals with the way data is organized, stored, and manipulated. It equips learners with methods and strategies to efficiently handle and process data, serving as a foundation for designing algorithms and building software applications.
2	**Vertical:**	**Major**
3	**Type:**	**Theory**
4	**Credits:**	**2 credits**
5	**Hours Allotted:**	**30 Hours**
6	**Marks Allotted:**	**50 Marks**
7	**Course Objectives:** 1. To develop the ability to evaluate the performance and efficiency of data structure operations and algorithms. 2. To offer hands-on experience in implementing common data structures through programming. 3. To explore the characteristics, use cases, and implementation of arrays, linked lists, stacks, and queues. 4. To enable students to convert theoretical data structure concepts into executable cod. 5. To apply suitable data structures for solving standard computational problems such as searching and sorting. 6. To understand the organization, properties, and traversal techniques of binary trees and binary search trees.	
8	**Course Outcomes:** **CO1.** Upon successful completion of this course, students will be able to: **CO2**. Understand and apply fundamental data structures and their associated operations in problem-solving. **CO3**. Analyze algorithms to assess their efficiency in terms of time and space complexity, and select appropriate solutions. **CO4**. Implement algorithmic solutions effectively using a programming language of their choice. **CO5**. Construct and traverse binary trees and binary search trees, demonstrating a clear conceptual and practical grasp of tree structures.	
9	**Modules:-** **Module 1 (15 hours):**	
	Introduction and Fundamentals Basic Terminology: Concepts of data, information, data structures, and Abstract Data Types (ADT) Classification of Data Structures: Linear and Non-linear structures **Algorithm Analysis**: Introduction to algorithm efficiency, time complexity, and Big O notation Arrays and Linked Lists Arrays: Representation, traversal, insertion, deletion, and searching techniques	

	Linked Lists: Singly linked list – structure, creation, insertion, deletion, and traversal **Comparative Analysis**: Arrays vs. Linked Lists – use cases, advantages, and limitations Stacks, Queues, and Recursion Stacks: Stack ADT and operations: Push, Pop, Peek Array-based implementation Applications: Expression evaluation (postfix/prefix) Queues: Queue ADT and operations: Enqueue, Dequeue, Peek Array-based implementation Applications of queues in real-life scenarios Recursion: Introduction to recursion with basic examples (e.g., factorial, Fibonacci series) Understanding recursive logic and its relation to stack behaviour	
	Module 2 (15 hours):	
	Trees,Binary Trees: Representation of binary trees, Tree traversal techniques: Inorder, Preorder, and Postorder **Binary Search Trees (BST)**: Operations: Insertion, Deletion, and Search, Applications of trees in computing (e.g., hierarchical data representation, searching) **Hashing**: Introduction to Hashing and Hash Functions, Implementation of Hash Tables, Collision Resolution Techniques: Separate Chaining, Applications of Hashing **Sorting and Searching Algorithms :- Sorting Techniques**: Bubble Sort, Insertion Sort, Selection Sort **Searching Techniques**:Linear Search, Binary Search	
10	**Text Books** 1. Data Structures and Algorithms made Easy: Data Structures and Algorithmic Puzzles, Narasimha Karumanchi ,5th Edition 2017 2. A Simplified Approach to Data Structures, Lalit Goyal, Vishal Goyal, Pawan Kumar SPD,1st 2014	
11	**Reference Books** 1. Problem Solving in Data Structures & Algorithms Using C by Hemant Jain ,1st Edition, BPB Publications, 2018 2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, 4th Edition, MIT Press,2022	
12	**Internal Continuous Assessment: 40%**	**Semester End Examination: 60%**
13	**Continuous Evaluation through:** Class test of 15 marks Quizzes/ Presentations/ Assignments: 5 marks Total: 20 marks	**External Examination (30 Marks)– 1 hr duration**
14	**Format of Question Paper: (Semester End Examination: 30 Marks. Duration:1 hour)** Q1: Attempt any three (out of five) from Module 1 (15 marks) Q2: Attempt any three (out of five) from Module 2 (15 marks)	

Name of the Course: Operating System

Sr. No.	Heading	Particulars
1	**Description the course:**	Introduce operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file systems and protection) Introduce the issues to be considered in the design and development of operating system (memory, file and disk).
2	**Vertical:**	**Major**
3	**Type:**	**Theory**
4	**Credits:**	**2 credits**
5	**Hours Allotted:**	**30 Hours**
6	**Marks Allotted:**	**50 Marks**
7	**Course Objectives:** 1. Understand the fundamental concepts of operating systems, including their types, structure, and core functionalities. 2. Comprehend process management techniques including process scheduling, threading, and synchronization mechanisms. 3. Analyse memory management strategies, including paging, segmentation, and virtual memory concepts. 4. Understand the principles of deadlock, including its characterization, prevention, and recovery techniques. 5. Explore file system interfaces, including file organization, directory structures, and access methods. 6. Recognize the need for protection and security in operating systems.	
8	**Course Outcomes:** **CO1.** Explain the structure and functions of an operating system and how it acts as an interface between hardware and users. **CO2.** Describe and analyse the different states of a process and demonstrate knowledge of process scheduling and multithreading. **CO3.** Apply synchronization techniques to solve critical-section problems and avoid race conditions using semaphores and other methods. **CO4.** Illustrate various memory management techniques, including paging and segmentation, and analyse page replacement algorithms. **CO5.** Identify and handle deadlocks using appropriate algorithms for detection, prevention, and recovery. **CO6.** Describe file system architecture, access methods, and implement directory structures. **CO7.** Demonstrate an understanding of system protection and security mechanisms to ensure OS stability and integrity.	
9	**Modules:-** **Module 1 (15 hours):**	
	Operating System Overview, Basics of operating systems, Generations, Types and Functions, Structure, Services, System Calls, System Boot, System Programs, Protection and Security, Process Management: Process Concepts, Process Creation and Deletion,	

	Process Control Block, Scheduling-Criteria, Scheduling Algorithms and their Evaluation, Threads, Threading Issues. Introduction to process Synchronization, Inter process Communication, Critical-Section Problem, Peterson's Algorithm, Synchronization Hardware, Semaphores, Classic Problems of Synchronization.	
	Module 2 (15 hours):	
	Main Memory, Swapping, Contiguous Memory Allocation, Paging, Structure of Page Table, Segmentation, Virtual Memory, Demand Paging, Page Replacement Algorithms, Allocation of Frames, Thrashing. Deadlock: System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery from Deadlock, Banker's Algorithm, File System Interface, File Concept, Access Methods, Directory Structure, and File System Structure.	
10	**Text Books** 1. "Operating Systems: Design and Implementation" by Andrew S. Tanenbaum & Albert S. Woodhull. 2. "Operating Systems: Three Easy Pieces (OSTEP)" by Remzi H. Arpaci-Dusseau & Andrea C. Arpaci-Dusseau	
11	**Reference Books** 1. Operating Systems – Internals and Design Principles William Stallings, Pearson 9th , 2009 2. Operating System Concepts, Abraham Silberschatz, Wiley, 8th Edition 3. Operating Systems, Godbole and Kahate, Godbole and Kahate, 3rd Edition.	
12	**Internal Continuous Assessment: 40%**	**Semester End Examination: 60%**
13	**Continuous Evaluation through:** Class test of 15 marks Quizzes/ Presentations/ Assignments: 5 marks Total: 20 marks	**External Examination (30 Marks)– 1 hr duration**
14	**Format of Question Paper: (Semester End Examination: 30 Marks. Duration:1 hour)** Q1: Attempt any three (out of five) from Module 1 (15 marks) Q2: Attempt any three (out of five) from Module 2 (15 marks)	

Name of the Course: Python for AI + Data Structures for Problem Solving Practical

Sr. No.	Heading	Particulars
1	**Description the course:**	**Python for AI** This course offers a comprehensive exploration of advanced Python programming concepts, designed to equip students with the tools to tackle real-world problems efficiently. Key topics include regular expressions, Python's built-in date and time modules, calendar-based operations, and file handling techniques for both text and binary files. Students will also learn methods for extracting and processing structured data. **Data Structures for Problem Solving** In a separate module, the course covers essentials of Data Structures. This includes the study and application of structures such as arrays, linked lists, stacks, queues, trees, and hash tables. Emphasis is placed on understanding their underlying logic, practical implementation, and performance analysis. This dual-language approach enhances students' programming versatility and problem-solving skills.
2	**Vertical:**	**Major**
3	**Type:**	**Practical**
4	**Credits:**	**2 credits (60 hours practical work in a semester)**
5	**Hours Allotted:**	**30 Hours (Python for AI Practical)+ 30 Hours (Data Structures for Problem Solving practical)**
6	**Marks Allotted:**	**50 Marks**
7	**Course Objectives:** 4. Develop a solid understanding of core programming concepts in Python, including input/output operations, conditional statements, and loop constructs. 5. Apply array operations, indexing, slicing, and mathematical functions using the NumPy library for efficient numerical computing. 6. Strengthen problem-solving abilities through the use of functions, recursion, lambda expressions, and modular programming in Python. 7. Utilize built-in Python data structures such as lists and dictionaries, along with performing file operations for both text and binary files. 8. Work with text processing, date and time manipulation, and calendar-based operations in Python. 9. Gain hands-on experience in implementing fundamental data structures—such as arrays, linked lists, stacks, and queues—using C++. 10. Develop skills in designing and analyzing algorithms to solve computational problems using appropriate data structures. 11. Learn to select and justify the use of suitable data structures for various real-world applications. 12. Enhance understanding of dynamic memory allocation and efficient data management techniques in C++. 13. Build the ability to debug, test, and optimize code related to data structure operations for improved performance and reliability.	

8	**Course Outcomes:** **CO1.** Apply Python programming concepts such as input/output operations, conditional statements, and loops to solve fundamental problems. **CO2.** Demonstrate proficiency in performing array operations, indexing, slicing, and analyzing attributes of arrays using NumPy. **CO3.** Solve computational problems using functions, recursion, and lambda expressions, and implement modular programming for efficient and reusable code. **CO4.** Implement real-world solutions using Python data structures like lists and dictionaries, along with file operations. **CO5.** Process and analyze text data, extract relevant information, handle date and time operations, and measure program execution time. **CO6.** Gain hands-on experience in implementing fundamental data structures including arrays, linked lists, stacks, queues, trees, and graphs using C++. **CO7.** Design and analyze algorithms to solve computational problems using appropriate data structures. **CO8.** Choose suitable data structures for different applications and justify their use based on problem requirements. **CO9.** Understand and apply dynamic memory allocation and effective data management techniques in C++. **CO10.** Debug and optimize data structure implementations for correctness and performance.
9	**Modules:-** **Module 1 :**

Practical 1:-Write programs for the following:

a. Write a program that asks the user for their name and the age they will turn this year. Then, calculate and print the year they were born, addressing them by name.
b. Write a program to accept a number from the user and depending on whether the number is even or odd, print out an appropriate message to the user.
c. Write a program that accepts a signal strength value (in dBm) from the user and prints the corresponding signal quality based on the following scale:

Signal Strength (dBm)	Signal Quality
-50 to 0	Excellent
-70 to -51	Good
-85 to -71	Fair
-100 to -86	Poor
Below -100	No Signal

Practical 2:-Write programs for the following:

a. Write a program to print factorial of the number.
b. Write a program to generate the Fibonacci series.
c. Write a program to accept a number from the user display sum of its digits.

Practical 3:-Write programs for the following:

a. Write a program that accepts a list of daily temperatures (in °C) for a week from the user. Perform the following:
   i. Print the full list of temperatures.
   ii. Print the temperature of the first and last day.
   iii. Print temperatures from day 3 to day 5.
   iv. Calculate and print the average temperature for the week.
b. Write a program to implement mathematical functions on arrays.
c. Write a program to perform array aliasing and copying. **Practical 4:-**Write

	programs for the following: a. You are analyzing the daily temperatures (in °C) of a city for a week. Write a Python program using NumPy to do the following: i. Store the temperatures in a NumPy array: [30, 32, 31, 35, 36, 34, 33]. ii. Print the temperature on the 4th day (using basic indexing). iii. Print the temperatures from day 2 to day 5 (using slicing). iv. Display the temperatures on alternate days (every second element). v. Display all temperatures above 33°C (using boolean indexing). b. Write a program to analyze dimensions and attributes of arrays **Practical 5:-**Write programs for the following: a. Write a function to check the input value is Armstrong and also write the function for Palindrome. b. Write a recursive function to print the factorial for a given number. c. Write a lambda function that checks whether a given string starts with a specific character. **Practical 6:-**Write programs for the following: a. Write a program to compute number of characters and words in a string. b. Write a program to create a custom Python module containing at least two functions. Then write a separate script to import and test these functions. **Practical 7:-**Write programs for the following: a. Write a program that takes two lists and returns True if they have at least one common member. b. Write a Python script to sort (ascending and descending) a dictionary by value. **Practical 8:-**Write programs for the following: a. Write a program to accept and pass radius to a function that returns area and circumference (using tuple). b. Write a program to perform basic file operations on text files and binary files. c. Write a Python program to read last n lines of a file. **Practical 9:-**Write programs for the following: a. Write a program to count the occurrences of a specific word in a file using regular expressions. b. Write a program to extract all hyperlinks (<a href="…">) from HTML File. **Practical 10:-**Write programs for the following: a. Write a program that compares two dates (in DD/MM/YYYY format) and prints which one is earlier. b. Write a program to measure program execution time. c. Write a program using the calendar module to print the weekday of the first day of a given month and year.
	Module 2 :
	Practical 1:-Write a program that performs basic operations on an array, including: a. Adding an element at a given index within the array. b. Removing an element from a specified index in the array. **Practical 2:-**Implement a program to manage a singly linked list with the following functionalities: a. Initialize an empty singly linked list. b. Add a new node at the start, at the end, and at a specified position within the list. c. Delete a node from a given position in a linked list. **Practical 3:-**Develop a program that performs the following tasks: a. Implement a stack data structure using an array.

	b. Evaluate a postfix expression using the stack implementation. **Practical 4:-**Write a program to: a. Implement a queue data structure using an array. b. Simulate a basic queue system, such as managing customers waiting in line. **Practical 5:-**Design a program to manage a Binary Search Tree (BST) with the following capabilities: a. Construct an empty Binary Search Tree. b. Insert new nodes into the BST following BST properties. c. Search for a specific node within the BST. **Practical 6:-**Write a program to perform different types of tree traversals on a binary tree, including: a. Inorder traversal b. Preorder traversal c. Postorder traversal **Practical 7:-**Develop a program to a. Store and retrieve data from the hash table. b. Write a program to implement the collision technique **Practical 8:-**Write programs to implement and compare the following sorting algorithms: a. Bubble sort b. Insertion sort c. Selection sort **Practical 9:-**Write programs to implement and compare: a. Linear search b. Binary search (on a sorted array) **Practical 10:-**Design a simple program that integrates and manipulates more than one data structure.	
10	**Text Books** 1. Learning Python, Fourth Edition by Mark Lutz Copyright © 2009 Mark Lutz. Published by O'Reilly Media, Inc. 2. Python Basics: A Practical Introduction to Python 3 Revised and Updated 4th Edition David Amos, Dan Bader, Joanna Jablonski, Fletcher Heisler 3. Data Structures and Algorithms made Easy: Data Structures and Algorithmic Puzzles, Narasimha Karumanchi ,5th Edition 2017	
11	**Reference Books** 1. Let Us Python, Yashwant. B. Kanetkar, BPB Publication, 2019 2. Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018 3. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017 4. A Simplified Approach to Data Structures, Lalit Goyal, Vishal Goyal, Pawan Kumar SPD,1st 2014 5. Problem Solving in Data Structures & Algorithms Using C by Hemant Jain ,1st Edition, BPB Publications, 2018 6. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, 4th Edition, MIT Press,2022	
12	**Internal Continuous Assessment: 40%**	**Semester End Examination: 60%**
13	**Continuous Evaluation through:** Performance and write-up submission of 20 marks.	**30 marks practical exam of 2 hours duration**

14	**Format of Question Paper: (Semester End Examination: 30 Marks. Duration:2 hour)** **Practical Slip:** Q1. From Module 1 13 marks Q2. From Module 2 12marks Q3. Journal and Viva 05 marks

Nirmala Memorial Foundation College of Commerce and Science

(Autonomous)

Re-accredited by NAAC with B++, ISO 9001-2015 Certified, Recognised under section 2(f) & 12(B) of the UGC Act 1956

Permanently Affiliated to the University of Mumbai,

Syllabus

Bachelor of Science in Information Technology

(B.Sc. Information Technology)

(Semester III)

Minor

(Programme Code: UGIT01)

As Per New Education Policy (NEP) 2020

(To be implemented from the Academic Year 2025-2026)

Approved in the Academic Council Meeting held on 5th July 2025

Vertical – II

Semester – III

Minor Courses

Name of the Course: Programming with R

Sr. No.	Heading	Particulars
1	**Description the course:**	This course introduces learners to the fundamentals of R, a powerful programming language widely used for statistical computing, data analysis, and visualization. Students will gain hands-on experience with core R concepts, including creating and manipulating variables, performing arithmetic and logical operations, working with vectors, matrices, lists, and data frames, and handling categorical data using factors. The course also covers the use of built-in summary functions for statistical analysis, as well as writing simple custom functions. By the end of the course, learners will be equipped with the foundational skills necessary to explore, analyze, and manage data efficiently using R.
2	**Vertical:**	**Minor**
3	**Type:**	**Practical**
4	**Credits:**	**2 credits (1 credit = 15 Hours for Theory or 30 Hours of Practical work in a semester)**
5	**Hours Allotted:**	**30 Hours**
6	**Marks Allotted:**	**50 Marks**
7	**Course Objectives:** 1. Introduce the fundamentals of R programming, including syntax, variables, and data structures. 2. Develop the ability to perform operations, manipulate data, and use built-in functions for analysis. 3. Build foundational skills to create custom functions and prepare for advanced R applications in data science.	
8	**Course Outcomes:** After successful completion of this course, students would be able to - CO1. Apply basic R syntax to create and manipulate variables, vectors, matrices, lists, and data frames. CO2. Perform arithmetic, logical, and statistical operations using R's built-in functions. CO3 Manage categorical data effectively using factors. CO4. Write and execute simple custom functions to solve programming tasks in R.	
9	**Modules:-** **Module 1 (15 hours):**	
	This set of practicals introduces fundamental R programming concepts, including basic commands, data structures (arrays, lists, and data frames), and matrix operations such as addition, transpose, inverse, and multiplication. It also covers essential statistical functions like mean, median, mode, quartiles, and histogram plotting. Finally, it demonstrates how to import data from Excel or CSV files and perform statistical analysis on real-world datasets.	

	Practical 1: Using R execute the basic commands, array, list and frames. **Practical 2:** Create a Matrix using R and Perform the operations addition, inverse, transpose and multiplication operations. **Practical 3:** Using R Execute the statistical functions: mean, median, mode, quartiles, range, inter quartile range histogram. **Practical 4:** Using R import the data from Excel / .CSV file and Perform the above functions.	
	Module 2 (15 hours): These practicals focus on advanced statistical analysis using R. They include calculating standard deviation, variance, covariance, and analyzing skewness from imported Excel/CSV data. Learners also perform hypothesis testing, Chi-squared tests, and apply probability distributions (binomial and normal). Finally, the concept of linear regression is introduced for modeling and predicting data relationships. **Practical 5:** Using R import the data from Excel / .CSV file and Calculate the standard deviation, variance, co-variance. **Practical 6:** Using R import the data from Excel / .CSV file and draw the skewness. **Practical 7:** Import the data from Excel / .CSV and perform the hypothetical testing. **Practical 8:** Import the data from Excel / .CSV and perform the Chi-squared Test. **Practical 9:** Using R perform the binomial and normal distribution on the data **Practical 10:** Perform the Linear Regression using R.	
10	**Text Books** 1. A Practical Approach to R Tool R.B. Patil, H.J. Dand and R. Dahake SPD First 2011 2. R for Data Science- Hadley Wickham & Garrett Grolemund- https://r4ds.hadley.nz/	
11	**Reference Books** 1. Hands-On Programming with R – Garrett Grolemund (2014) 2. Advanced R – Hadley Wickham (2nd Edition, 2019) 3. The Art of R Programming – Norman Matloff (2011) 4. R Programming – Anil Kumar Verma, Cengage India (2017) 5. R Programming By Example – Omar Trejo Navarro (2018)	
12	**Internal Continuous Assessment: 40%**	**Semester End Examination: 60%**
13	**Continuous Evaluation through:** Practical journal submission • Journal Submission : 20 Marks Total: 20 marks	**A semester end practical examination of 2 hours duration for 30 marks as the paper pattern given below.** *Its compulsory to carry certified journal at the time of practical exam*

14	Format of Question Paper: (Semester End Practical Examination : 30 Marks. Duration:2 hours) Q1: Module 1 (12 marks) Q2: Module 2 (12 marks) Q.3 Viva (06 Marks)

Name of the Course: DS_Scala for Data Science

Sr. No.	Heading	Particulars
1	**Description the course:**	This course provides hands-on experience with Scala and its ecosystem for data analysis and machine learning. Students will learn statistical methods, machine learning algorithms, and data processing techniques using Breeze and Apache Spark. The course also covers time-series analysis, feature engineering, and building scalable data pipelines. Through practical exercises, students will gain proficiency in implementing regression models and clustering while handling real-world datasets effectively.
2	**Vertical:**	**Minor**
3	**Type:**	**Practical**
4	**Credits:**	**2 credits (1 credit = 15 Hours for Theory or 30 Hours of Practical work in a semester)**
5	**Hours Allotted:**	**30 Hours**
6	**Marks Allotted:**	**50 Marks**
7	**Course Objectives:** 1. To set up and configure Scala, SBT, and Apache Spark for programming, data analysis, and large-scale data processing. 2. To perform statistical calculations, including correlation, frequency distribution, and moving averages using Scala and Breeze, and visualize data insights with Breeze-viz. 3. To implement machine learning models such as linear regression, logistic regression, and k-means clustering, along with feature engineering for predictive modeling. 4. To utilize the Breeze library for numerical computations, matrix operations, and time-series data analysis to extract meaningful insights.	
8	**Course Outcomes:** After successful completion of this course, students would be able to - CO1. Students will set up a functional Scala development environment with SBT and execute basic programs for data analysis. CO2. Students will utilize Breeze for numerical operations, matrix manipulations, and statistical computations such as correlation and frequency distribution. CO3. Students will create data visualizations using Breeze-viz and implement machine learning models, including regression and clustering, using Breeze. CO4. Students will work with Apache Spark for large-scale data processing, machine learning pipelines, and time-series analysis to extract meaningful insights.	
9	**Modules:-** **Module 1 (15 hours):**	
	Set up Scala and SBT on your system. 1. Write a simple Scala program that prints a welcome message for data scientists. 2. Calculate mean, median, and mode of a list of numbers. Implement basic statistical	

	calculations using Scala collections. 3. Generate a random dataset of 10 numbers and calculate its variance and standard deviation. 4. Create a dense vector using Breeze and calculate its sum, mean, and dot product with another vector. 5. Generate a random matrix using Breeze and compute its transpose and determinant. 6. Slice a Breeze matrix to extract a sub-matrix and calculate its row and column sums. 7. Write a program to perform element-wise addition, subtraction, multiplication, and division of two Breeze matrices. 8. Read a CSV file and calculate basic statistics for each numeric column. Use the scala-csv library or similar tools. 9. Handle missing values in a dataset. Replace missing values with the column mean. 10. Filter rows in a dataset where a specific column value exceeds a threshold. 11. Write a program to tokenize and count the frequency of words in a text file. 12. Implement one-hot encoding for a categorical column in a dataset. 13. Create a scatter plot of random data using Breeze-viz. Label the axes and customize the color of points. 14. Generate a histogram of a dataset using Breeze-viz. Experiment with different bin sizes. 15. Plot a line graph for a dataset showing a trend over time. 16. Combine two plots (e.g., scatter and line plot) in a single visualization using Breeze-viz.
	Module 2 (15 hours):
	1. Find the correlation between two lists of numbers. Implement the formula for Pearson correlation coefficient. 2. Calculate the moving average of a time series data using Scala collections. 3. Write a program to compute frequency distribution and cumulative frequency of a dataset. 4. Sort a dataset by a specific column and extract the top 5 rows. 5. Implement linear regression using Breeze. Fit a model to a small dataset and predict a value. 6. Perform logistic regression using Breeze. Classify a dataset with binary labels. 7. Compute the Euclidean distance between two Breeze vectors. Use it for nearest neighbor classification. 8. Cluster a dataset into two groups using k-means clustering in Breeze. 9. Set up Apache Spark locally and count the frequency of words in a text file. 10. Filter rows in a CSV file using Spark DataFrames where a numeric column exceeds a threshold. 11. Perform a group-by operation in Spark DataFrames to compute the average of each group. 12. Join two CSV files in Spark DataFrames based on a common column and write the

	output to a file. 13. Create a simple Spark MLlib pipeline to classify data. Use logistic regression or decision trees. 14. Perform basic time series analysis in Scala. Generate synthetic time series data (e.g., daily sales over a month). 15. Create polynomial features from a dataset. Given a list of numbers (e.g., [1, 2, 3]), generate polynomial features up to degree 3 (e.g., [1, 1^2, 1^3, 2, 2^2, 2^3, 3, 3^2, 3^3]).	
10	**Text Books** 1. Scala for Data Science, by Pascal Bugnion, Packt Publishing, 1st edition (28 January 2016) 2. Mastering Scala by Dennis Alexander, Packt Publishing, 1st edition (2023) 3. Scala 3 Mastery by John Hunt, Apress, 1st edition (2023) 4. Mastering Scala 3 by John Hunt, Apress, 1st edition (2023)	
11	**Reference Books** 1. Programming Scala by Dean Wampler and Alex Payne, O'Reilly Media, 3rd edition (2021) 2. Scala Cookbook by Alvin Alexander, O'Reilly Media, 2nd edition (2021) 3. Functional Programming in Scala by Paul Chiusano and Rúnar Bjarnason, Manning Publications, 2nd edition (2023)	
12	**Internal Continuous Assessment: 40%**	**Semester End Examination: 60%**
13	**Continuous Evaluation through:** Practical journal submission, viva, assignments • Journal Submission : 10 Marks • Assignments: 10 Marks Total: 20 marks	**A semester end practical examination of 2 hours duration for 30 marks as the paper pattern given below.** *Its compulsory to carry certified journal at the time of practical exam*
14	Format of Question Paper: (Semester End Practical Examination : 30 Marks. Duration:2 hours) Q1: Module 1 (12 marks) Q2: Module 2 (12 marks) Q.3 Viva (06 Marks)	

Nirmala Memorial Foundation College of Commerce and Science

(Autonomous)

Re-accredited by NAAC with B++, ISO 9001-2015 Certified, Recognised under section 2(f) & 12(B) of the UGC Act 1956

Permanently Affiliated to the University of Mumbai,

Syllabus

Open Electives (OE)

As Per New Education Policy (NEP) 2020

Vertical – III - Semester III

(To be implemented from the Academic Year 2025-2026)

Approved in the Academic Council Meeting held on ***5th July 2025***

Name of the Course: Financial Literacy

Sr. No.	Heading	Particulars
1	**Description the course:**	This course provides a comprehensive introduction to the Indian financial system, offering foundational knowledge of its structure, components, and functioning. It delves into the role of banks and other financial institutions in economic development and explores various banking services available to individuals and businesses. The course also introduces students to the basics of insurance and personal financial planning, enabling them to make informed financial decisions and develop sound saving and investment strategies. Through real-world examples, case studies, and practical insights, students will gain a deeper understanding of how financial instruments, services, and institutions interact to support the financial stability of individuals and the economy.
2	**Vertical:**	**Open Electives**
3	**Type:**	**Theory**
4	**Credits:**	**2 credits**
5	**Hours Allotted:**	**30 Hours**
6	**Marks Allotted:**	**50 Marks**
7	**Course Objectives:** 1. Familiarity with different aspects of financial literacy such as savings, investment, taxation, and insurance. 2. Understand the relevance and process of financial planning. **3.** To promote financial well-being.	
8	**Course Outcomes (CO):** **After completing this course, students will have the knowledge and skills to** • **CO1:** Learners will develop proficiency for personal and family financial planning. • **CO2:** Learners will develop the ability to analyse banking and insurance products. • **CO3:** Learners will Apply the concept of investment planning.	
•	**Modules:-** **Module 1 (10 hours): Financial Planning and Financial Products Introduction** A) Introduction to Saving, Management of spending and financial discipline: Meaning and importance of saving, theoretical principles of financial discipline, Impact of inflation and consumption patterns on personal finance. **B)** Banking products and services, Digitization of financial transactions: Types of bank accounts (Savings, Current, and Fixed Deposits), Services offered by banks: loans,	

overdrafts, etc. Concept and working of debit cards, credit cards, Introduction to digital banking: Features of Net Banking and UPI and theoretical benefits and limitations of digitization in finance.

Module 2 (10 hours): Investment Planning and Management

A) Investment opportunity and financial products: Overview of asset classes: equity, debt, mutual funds, real estate, and gold. Risk and return theory. Time value of money (conceptual level). Role of SEBI and regulatory environment.

B) Insurance Planning: Basic principles of insurance: risk pooling, premium, and insurable interest. Types of insurance: life vs non-life (health, motor, property).

C) Introduction to basic Tax Structure in India for personal taxation, Aspects of Personal tax planning: Importance of tax planning in personal finance.

10 Text Books

- ✓ Halan, Monika. *Let's Talk Money*, Harper Business
- ✓ Pathak, Bharati V. *The Indian Financial System* (selected basic chapters)
- ✓ RBI's "Financial Literacy" booklets and SEBI's student guides

11 Reference Books

1. Introduction to Financial Planning (4th Edition 2017) - Indian Institute of Banking & Finance.
2. Bhasham, A.L, Wonder that was India: A Survey of the Culture of the Indian Subcontinent Before the Coming of the Muslims. London, Sidgwick and Jackson, 1954'
3. Sinha, Madhu. Financial Planning: A Ready Reckoner July 2017, McGraw Hill.
4. Halan, Monika, Let's Talk Money: You've Worked Hard for It, Now Make It Work for You, July 2018 Harper Business.
5. Pandit, Amar The Only Financial Planning Book that You Will Ever Need, Network 18 Publications Ltd.

12	**Internal Continuous Assessment: 40%**	**External Assessment : 60%**
13	**Continuous Evaluation through:**	**Semester End Examination (30 Marks)**

Sr. No	Particular	Marks
1.	Class Test	15
2.	Class participation	05
	Total	**20**

14	**Format of Question Paper: (Semester End Examination: 30 Marks. Duration:1 hour)**

Questions	Type & Module	Marks
Q1.	Objective Question A) Multiple choice questions B) True or False	 08 Marks 07 Marks
Q2.	Theory Question	15 Marks
	OR	
Q2.	Theory Question	15 Marks
	Total	**30**

Nirmala Memorial Foundation College of Commerce and Science

(Autonomous)

Re-accredited by NAAC with B++, ISO 9001-2015 Certified, Recognised under section 2(f) & 12(B) of the UGC Act 1956

Permanently Affiliated to the University of Mumbai,

Syllabus

Bachelor of Science in Information Technology

(B.Sc. Information Technology)

(Programme Code: UGIT01)

As Per New Education Policy (NEP) 2020

(To be implemented from the Academic Year 2025-2026)
Approved in the Academic Council Meeting held on **5th July 2025**

Vertical – IV

Semester- III

Vocational & Skill Enhancement Courses (VSEC)

Name of the Course: Applied Mathematics

Sr. No.	Heading	Particulars
1	**Description the course:**	This course is designed for developing competency of the students in the applications of various mathematical concepts. It is equipped with Complex numbers, Laplace transform, Inverse Laplace transform, Differential equations of first order with first degree and higher degree. This course introduces basic concepts of Algebra and prepares students to study further courses in linear and abstract algebra.
2	**Vertical:**	Vocational Skill Course
3	**Type:**	Theory
4	**Credits:**	2 credits (1 credit = 15 Hours for Theory in a semester, Total 30 hours)
5	**Hours Allotted:**	30 Hours
6	**Marks Allotted:**	50 Marks
7	**Course Objectives:** 1. Ability to interpret the mathematical results in physical or practical terms for complex numbers. 2. Know and to understand various types of methods to solve Laplace transform. 3. Apply the knowledge of Laplace Transforms to solve the problems. 4. Know and to understand various types of methods to solve differential equations. 5. Apply the knowledge of differential equations to solve the problems. 6. Inculcate the habit of Mathematical Thinking through Indeterminate forms.	
8	**Course Outcomes :** **CO1.** Familiar with the various forms and operations of a complex number. **CO2.** Find the Laplace transform of a function using definition. **CO3**. Find the Inverse Laplace transform of a function using definition. **CO4**. Solve Differential equations of first degree and first order. **CO5**. Solve Differential equations of first degree and higher order.	
9	**Modules:-** **Module 1 (15 hours):**	
	1.1 Complex Numbers: Complex number, Equality of complex numbers, Graphical representation of complex number (Argand's Diagram), Polar form of complex numbers. Polar form of x+iy for different signs of x.y, Exponential form of complex numbers, Mathematical operation with complex numbers and their representation on Argand's Diagram, Circular functions of complex angles, Definition of hyperbolic function. Relations between circular and hyperbolic functions, Inverse hyperbolic functions. **1.2 The Laplace Transform:** Introduction. Definition of the Laplace Transform, Table of Elementary Laplace Transforms. Theorems on Important Properties of Laplace	

	Transformation, First Shifting Theorem, Second Shifting Theorem, Convolution Theorem, Laplace Transform of Derivatives. **1.3 Inverse Laplace Transform:** Shifting Theorem, Partial fractions Methods, Use of Convolution Theorem, Solution of Ordinary Linear Differential Equations with Constant Coefficients, Laplace Transformation of Special Function, a) Periodic Functions, Heaviside Unit Step Function, Dirac-delta Function (Unit Impulse Function).	
	Module 1 (15 hours):	
	2.1 Equation of the first order and of the first degree: Separation of variables, Equations homogeneous in x and y, Non-homogeneous linear equations, Exact differential Equation, Integrating Factor, Linear Equation and equation reducible to this form, Method of substitution. **2.2 Differential equation of the first order of a degree higher than the first**: Introduction, Solvable for p (or the method of factors), Solve for y, Solve for x, Clairaut's form of the equation, Method of Substitution. **2.3 Linear Differential Equations with Constant Coefficients:** Introduction, The Differential Operator, Linear Differential Equation f(D) y = 0, Different cases depending on the nature of the root of the equation f(D) = 0, Linear differential equation f(D) y = X, The complimentary Function, The inverse operator 1/f(D) and the symbolic expression for the particular integral	
10	**Text Books** 1) Advanced Engineering Mathematic, Author**:** Erwin Kreyszig**,** Publisher**:** Wiley, Edition: 10th Edition (Latest: 11th, 2020) 2) Applied Mathematics, Author**:** J. David Logan, Publisher: Wiley, Edition**:** 4th Edition (2013)	
11	**Books and References:** 1.A text book of Applied Mathematies Vol I, P. N. Wartikar and J. N.Wartikar, Pune Vidyathi Griha,7* ,1995 2.A text book of Applied Mathematies Vol II , P. N. Wartikar and J. N. Wartikar, Pune Vidyathi Griha,7" .1995 3.Higher Engineering Mathematies, Dr. B. S.Grewal, Khanna Publications.	
12	**Internal Continuous Assessment: 40%**	**Semester End Examination: 60%**
13	**Continuous Evaluation through:** Class test of 1 of 15 marks Class test of 2 of 15 marks Average of the two: 15 marks Quizzes/ Presentations/ Assignments: 5 marks Total: 20 marks	**External Examination (30 Marks)– 1 hr duration**
14	**Format of Question Paper: (Semester End Examination : 30 Marks. Duration:1 hour)** Q1: Attempt any three (out of five) from Module 1 (15 marks) Q2: Attempt any three (out of five) from Module 2 (15 marks)	

Nirmala Memorial Foundation College of Commerce and Science

(Autonomous)

Re-accredited by NAAC with B++, ISO 9001-2015 Certified, Recognised under section 2(f) & 12(B) of the UGC Act 1956

Permanently Affiliated to the University of Mumbai,

Syllabus

Ability Enhancement Courses

To be implemented from the Academic Year 2025-2026)

Approved in the Academic Council Meeting held on ***5th July 2025***

Vertical – V

Semester – III

हिन्दी भाषा : व्यावहारिक प्रयोग

Sr. No.	Heading	Particulars
1	**Description of the course :**	हिन्दी भाषा का जीवन में अत्यंत महत्व है, क्योंकि यह हमारे विचारों और भावनाओं की अभिव्यक्ति का प्रमुख माध्यम है। वर्तमान युग में तकनीक, मीडिया और राजनीति सहित कई क्षेत्रों में हिन्दी का व्यावहारिक प्रयोग बढ़ा है। भाषाई दक्षता सफलता की कुंजी है, जिससे विद्यार्थी विभिन्न क्षेत्रों में लाभ प्राप्त कर सकते हैं। इसी उद्देश्य से यह पाठ्यक्रम हिन्दी भाषा के शुद्ध और प्रभावी उपयोग पर केन्द्रित है।
2	**Vertical :**	**AEC**
3	**Type :**	**Theory**
4	**Credit**:	**2 credits**
5	**Hours Allotted :**	**30 Hours**
6	**Marks Allotted**:	**50 Marks**
7	**Course Objectives**: 1. विद्यार्थियों को राजभाषा हिन्दी का विविधतापूर्ण ज्ञान प्रदान करना। 2. विद्यार्थियों को राजभाषा हिन्दी के व्याकरण से परिचित कराना। 3. विद्यार्थियों को संज्ञा आदि का ज्ञान प्रदान करना।	
8	**Course Outcomes:** After completion of the course, learners would be able to: CO1. विद्यार्थियों को राजभाषा हिन्दी का ज्ञान एवं उसमें दक्षता प्राप्त होगी। CO2. विद्यार्थियों को हिन्दी के व्याकरणिक प्रयोगों की जानकारी प्राप्त होगी। CO3. विद्यार्थियों को हिन्दी संज्ञा आदि के साथ-साथ शुद्ध एवं व्यावहारिक भाषा प्रयोग का ज्ञान प्राप्त होगा।	
9	**Modules**:-	
	एकक – 1 व्याख्यान – 15 | क्रेडिट – 01 1. हिंदी भाषा : सामान्य परिचय 2. राजभाषा हिंदी : घटनात्मक महत्त्व 3. वर्णमाला : स्वर आणि व्यंजन 4. शब्दप्रकार : सामान्य परिचय (संज्ञा इत्यादी) **एकक – 2 व्याख्यान – 15 | क्रेडिट – 01** 1. वाक्य : सामान्य परिचय 2. शब्दलेखन : शुद्धतेचा उपयोग व खबरदारी 3. कारक व विभक्ती चिन्हे 4. पत्र लेखन : ○ शुभेच्छा पत्र ○ निमंत्रण पत्र ○ सूचना पत्र	

	○ तक्रार पत्र ○ आभार पत्र ○ अर्ज ○ माहितीच्या हक्काचा (RTI) अर्ज
10	**संदर्भ ग्रंथ:** 1. बाबूराम सक्सेना – सामान्य भाषा विज्ञान, हिंदी साहित्य सम्मेलन, प्रयाग 2. कामताप्रसाद गुरु – हिंदी व्याकरण, लोकभारती प्रकाशन, इलाहाबाद 3. आचार्य देवेंद्र नाथ शर्मा – भाषा विज्ञान की भूमिका, राधाकृष्ण प्रकाशन, दिल्ली 4. भाषा विज्ञान एवं भाषाशास्त्र – कपिलदेव द्विवेदी, विश्वविद्यालय प्रकाशन, वाराणसी 5. भोलानाथ तिवारी – भाषा विज्ञान, किताब महल, इलाहाबाद
11	**सातत्यपूर्ण मूल्यमापन (Continuous Evaluation) द्वारे:** • होनात्मक कार्य / प्रकल्प इत्यादी – 10 गुण • सादरीकरण / परिसंवाद सहभागिता इत्यादी – 05 गुण • अकादमिक, व्यावसायिक आणि कौशल्यवर्धक उपक्रम – 05 गुण एकूण : 20 गुण
12	अंतिम परीक्षा (Semester End Examination): • एकूण गुण (Marks): 30 • कालावधी (Duration): 01 तास (1 घंटा) Format of Question Paper: for the semester end examination लिखित परीक्षा अंक : 30 समयावधि : 01 घंटा निर्देश: 1. तीन इकाइयों में प्रश्न पूछे जाएं। 2. तीन प्रश्न पूछे जाएं, किन्हीं दो प्रश्नों के उत्तर अपेक्षित हैं। 15×2 = 30 अंक कुलयोग = 30 अंक

Nirmala Memorial Foundation College of Commerce and Science

(Autonomous)

Re-accredited by NAAC with B++, ISO 9001-2015 Certified, Recognised under section 2(f) & 12(B) of the UGC Act 1956

Permanently Affiliated to the University of Mumbai,

Syllabus

Co-Curricular Courses

Sports

As Per New Education Policy (NEP) 2020

(To be implemented from the Academic Year 2025-2026)

Approved in the Academic Council Meeting held on **5th July 2025**

Vertical – VI

Semester III

Yoga and Wellness I

(Course Code: UG.3CCS)

Title of Paper: Yoga and Wellness I

Sr. No.	Heading	Particulars
1	**Description the course :** **Including but Not limited to :**	India is growing rapidly as a global super-power. To face the challenges of the century and to keep up with the pace of the world, maintaining health is of prime importance. Giving thrust to healthy society, Physical Education, Sports, Health & fitness and Yoga are of great significance in today's world. The Government of India insists on Physical Fitness, Mental Health and Overall Development of Personality for every citizen. However, creating efficient and skilled human resource in the field of Physical Education, Sports and Yoga is identified as the need of the hour. Thus, the Governments of India and Government of Maharashtra have included Physical Education, Sports and Yoga as a key area under the NEP 2020.
2	**Vertical :**	Co-Curricular
3	**Type :**	Activity Based
4	**Credit**:	2 credits
5	**Hours Allotted :**	30 Hours
6	**Marks Allotted**:	50 Marks
7	**Course Objectives**: The students will be able to: • **To introduce students to the historical roots and philosophical foundations of Yoga,** with emphasis on Patanjali's Yoga Sutras. • **To develop awareness and practical skills in different forms of Yoga** by learning basic asanas and understanding their physical and mental health benefits.	
8	**Course Outcomes**: **CO1:** Understand the origin, philosophy, and importance of Yoga as described in Patanjali's Yoga Sutras. **CO2:** Identify various types of Yoga and perform basic asanas effectively, demonstrating their benefits on physical and mental well-being.	

9	**Module :** This module introduces the meaning, origin, and significance of Yoga with reference to Patanjali's Yoga Sutras. It explains various types of yoga such as Hatha, Ashtanga, and Vinyasa. Students learn basic asanas like Tadasana, Bhujangasana, Trikonasana, and Vajrasana. The module highlights yoga's benefits in improving both physical fitness and mental well-being.
10	**Reference Books:** • Iyengar, B.K.S. (2001). Light on Yoga. London: Thorsons. • Tiwari, O.P. (1996). Asana: Why and How? Lonavla: Kaivalyadhama Yoga Institute.
11	**Internal Continuous Assessment: 40% (20 Marks)** ❖ Yoga Report/Assignment
12	External Assessment: 60% : **(30 Marks)** • Scanned copy of the Yoga participation certificate. • A short report on Yogasana (handwritten or typed). *Note: Students participating in sports competitions conducted by University at State or National Level, students who have represented Mumbai University or College at Intercollegiate / Inter Zonal / West Zone Inter University / All India Inter University/ International tournament are exempt from submission of report.*

Letter Grades and Grade Points:

Semester GPA/ Program CGPA Semester/ Program	% of Marks	Alpha-Sign / Letter Grade Result		Grade Points
9.00-10.00	90.0-100	O	(Outstanding)	10
8.00-<9.00	80.0-<90.0	A+	(Excellent)	9
7.00-<8.00	70.0-<80.0	A	(Very Good)	8
6.00-<7.00	60.0-<70.0	B+	(Good)	7
5.50-<6.00	55.0-<60.0	B	(Above Average)	6
5.00-<5.50	50.0-<55.0	C	(Average)	5
4.00-<5.00	40.0-<50.0	P	(Pass)	4
Below 4.00	Below 40	F	(Fail)	0
Ab (Absent)	-	AB	(Absent)	0

Signature of BOS Chairman
Ms. Vaishali Mishra
BOS (Information Technology)

Signature of Faculty Member
Dr. Aparna Panigrahy
(Information Technology)

Signature of I/C Principal
Dr. Swiddle D'Cunha