



**Vivekanand Education Society's
College of Arts, Science and Commerce
(Autonomous)**

Sindhi Society, Chembur, Mumbai, Maharashtra - 400 071.

Accredited by NAAC "A Grade" in 3rd Cycle-2017

Best College Award- Urban Area, University of Mumbai (2012-13)

Recipient of FIST Grant (DST) and STAR College Grant (DBT)

**Affiliated to the
University of Mumbai**

Credit Based System for Holistic Development

Syllabus for **SYBSc Computer Science**

Program: B.Sc. Computer Science(Major)
(Program code: VESUSCS)

As per NEP 2020

with effect from Academic Year 2026-2027

PREFACE

The NEP-2020 is aimed to make India a global knowledge superpower with equitable access to the highest-quality education for all learners. The holistic and multidisciplinary education proposed under NEP-2020 is expected to facilitate the all-round development of learners with enhancement in employability potential and skills for new jobs in emerging fields. NEP envisages the development of guidelines for curriculum and credit framework for four year undergraduate programmes, regulations prescribing a minimum standard and the procedure for the award of the one year Certificate, two year Diploma, three years Bachelor Degree, four years Bachelor Degree (Research/ Honors) and Master's degree with multiple entry and exit options and transformation of HEIS into multidisciplinary institutions.

From this academic year, VESASC as an autonomous college, is going to implement NEP for the first year students as per the guidelines given by Government of Maharashtra through the 20th April 2023 GR. This model can be implemented in various ways out of which the **Department of Computer Science would be following the Department/ Subject Specific(DSC) model for Computer Science under which computer science should be pursued by the students as a mandatory requirement. DSCs shall be the core credit** courses which will be appropriately arranged and graded across all the semesters of the study.

Computer Science (CS) has been evolving as an important branch of science throughout the world in the last couple of decades. Nowadays, practically everyone is a computer user, and many people are even computer programmers. Computer Science can be seen on a higher level, as a science of problem solving and problem solving requires precision, creativity, and careful reasoning.

Computer Science education at undergraduate level (+3) will result in earning a Bachelor of Science (BSc) degree in CS. BSc with CS and BSc(Hons) in CS are aimed at undergraduate level training, laying a strong foundation of CS at an early stage of the career and facilitating multiple career paths. Students so graduated, can take up postgraduate programmes in CS leading to research as well as R&D, can be employable at IT industries(as Web Developer, Software Engineer, Network Engineer, Data Scientist, or AI/ML personnel), or can pursue a teachers' training programme such BEd in Computer Education, or can adopt a business management career.

Program Learning Outcomes

- To develop an understanding and knowledge of the basic theory of Computer Science with a good foundation on theory, systems and applications such as algorithms, data structures, data handling, data communication and computation.
- To develop the ability to use this knowledge to analyse new situations
- To acquire necessary and state-of-the-art skills to take up industry challenges. The objectives and outcomes are carefully designed to suit the above-mentioned purpose.
- The ability to synthesize the acquired knowledge, understanding and experience for a better and improved comprehension of the real-life problems
- To learn skills and tools like mathematics, statistics, IoT to find the solution, interpret the results and make predictions for the future developments.

SYBSC COMPUTER SCIENCE

MAJOR COURSES

Sem.	Theory / Practical	Course Title	No of Credits	No of Hours	Total Credits
III	Theory	Object Oriented Programming Using Java	2	30	8
	Theory	Data Structures	2	30	
	Theory	Software Engineering	2	30	
	Practical	Computer Science Practical 3	2	60	
IV	Theory	Advanced Application Development	2	30	8
	Theory	Computer Networks	2	30	
	Practical	IoT	2	60	
	Practical	Computer Science Practical 4	2	60	

MINOR COURSES

Sem.	Theory / Practical	Course Title	No of Credits	No of Hours	Total Credits
III	Theory	Discrete Mathematics	3	45	4
	Practical	Discrete Mathematics Practical	1	30	
IV	Theory	Calculus	3	45	4
	Tutorial	Calculus Practical	1	30	

OPEN ELECTIVE COURSES (offered by Computer Science Department)

Sem.	Theory/ Practical	Course Title	No of Credits	No of Hours	Total Credits
III & IV	Theory	Introduction to Cyber Security	2	30	2

Vocational Skill Courses (VSC)

Sem.	Theory/ Practical	Course Title	No of Credits	No of Hours	Total Credits
III	Practical	Advanced Database Concepts	2	60	2

Skill Enhancement Courses (SEC)

Sem.	Theory/Practical	Course Title	No of Credits	No of Hours	Total Credits
IV	Practical	Mobile Application Development	2	60	2

Ability Enhancement Courses (AEC)

Sem.	Theory / Practical	Course Title	No of Credits	No of Hours	Total Credits
III	Theory	AEC : Modern Indian Language *	2	30	2
IV	Theory	AEC : Modern Indian Language *	2	30	2

*As offered by the College

Co-Curricular Courses (CC)

Sem.	Theory / Practical	Course Title	No of Credits	No of Hours	Total Credits
III	Activity Based	CC : NSS / Health & Wellness / Sports / Music / Cultural Activities / Sunday School / Department Activities etc	2	60	4
IV	Activity Based	CC : NSS / Health & Wellness/ Sports / Music / Cultural Activities / Sunday School / Department Activities etc	2	60	

Field Project(FP) + Community Engagement Program(CEP)

Sem.	Theory / Practical	Course Title	No of Credits	No of Hours	Total Credits
III	Practical	Field Project(FP)*	2		2
IV	Practical	Community Engagement Program(CEP)*	2		2

*As decided by the College

Semester III

Course Code	Course Title	Credits	Lectures/ Week
UMMCSS 3-201	Object Oriented Programming Using Java	2	2
About the course: The objective of this course is to teach the learner how to use the Object Oriented paradigm to develop code and understand the concepts of Core Java and explore advanced topic of Java programming for solving problems.			
Course Objectives: • To provide insight into OOP concepts. • code and understand the concepts of Core Java and to cover its prerequisites • To provide understanding of how java can be used to design different types of applications			
Learning Outcomes: After successful completion of this course, students would be able to • Object oriented programming concepts using Java. • Knowledge of input, its processing and getting suitable output. • Design basic applications in java using Graphical User Interface. • The learner will be able to develop applications using swings • The learner will be able to connect databases with java			
Unit	Topic	No. of lectures	
I	Introduction: History, Features of Java, Java Development Kit, Java Application Programming Interface, Java Virtual Machine Java Program Structure, Java Tokens, Typecasting, Arrays OOPS: Introduction, Class, Object, Static Keywords, Constructors, this Keyword, Inheritance, super Keyword, Polymorphism (overloading and overriding), Abstraction, Encapsulation, Abstract Classes, Interfaces Introduction to UML, UML Class diagrams, Representing relationships using association, generalization-specialization, composition-aggregation String Manipulations: String, String Buffer, String Tokenizer	10	
II	Packages: Introduction to predefined packages (java.lang, java.util, java.io, java.sql, java.swing), User Defined Packages, Access specifiers Exception Handling: Introduction, Pre-Defined Exceptions, Try-Catch-Finally, Throws, throw, User Defined Exception examples Multithreading: Thread Creations, Thread Life Cycle, Life Cycle Methods, Synchronization, Wait() notify() notify all() methods.	10	

III	JDBC: Introduction, JDBC Architecture, JDBC Drivers, JDBC Connectivity Model, java.sql package, Using Statement, PreparedStatement, CallableStatement, ResultSet, Scrollable and Updatable ResultSet, Navigating and manipulating data, ResultSetMetaData, Managing Transactions in JDBC Introduction to JFC and Swing- Features of the Java Foundation Classes, Swing API Components, JComponent Class, Windows, Dialog Boxes, and Panels, Labels, Buttons, Check Boxes, Menus, Toolbars, Implementing Action interface, Pane, JScrollPane, Desktop pane, Scrollbars, Lists and Combo Boxes, Text-Entry Components, Colors and File Choosers, Tables Event Handling: Delegation Event Model, Events, Event classes, Event listener interfaces, Using delegation event model, adapter classes.	10
Text Books : 1) Herbert Schildt, Java The Complete Reference, Ninth Edition, McGraw-Hill Education, 2014 Additional References: 1) E. Balagurusamy, Programming with Java, Tata McGraw-Hill Education India, 2014 2) Programming in JAVA, 2nd Ed, Sachin Malhotra & Saurabh Choudhary, Oxford Press 3) The Java Tutorials: http://docs.oracle.com/javase/tutorial/		

Course Code	Course Title	Credits	Lectures/ Week
UM MCS S3-20 2	Data Structures	2	2
About the course: The course focuses to give an understanding of different types of data structures that can be used to store data in memory, how to create-manipulate them and to use them in the best possible manner as per the requirements of the application.			
Course Objectives: ● To introduce data abstraction and data representation in memory ● To describe, design and use of elementary data structures such as stack, queue, linked list, tree and graph ● How and why different data structures are used for different types of problems.			
Learning Outcomes: After successful completion of this course, students would be able to- ● Create different types of data structures. ● Understand which data structure to be used based on the type of the problem. ● Apply combined knowledge of algorithms and data structures to write highly effective programs in various domains.			

Unit	Topic	No. of lectures
I	Abstract Data Type: Different Data Types, different types of data structures & their classifications, Introduction to ADT, Creating user-specific ADT. Linked Structures: ADT for linked list, Advantages & Disadvantages, Singly Linked List-Traversing, Searching, Prepending and Removing Nodes, applications of linked lists like polynomial equations. Stacks: Stack ADT for Stack, Advantages & Disadvantages, Applications of stack like balanced delimiter, prefix to postfix notation. Queues: Queue ADT, Advantages & Disadvantages, linked representations. Circular Queue operations, Dequeues, applications of queue like job scheduling queues.	10
II	Doubly Linked list: ADT of doubly linked list, Advantages & Disadvantages, Insertion and deletion of nodes at various positions. Trees: ADT for Tree Structure. Advantages & disadvantages, Binary Tree-Properties, Implementation and Traversals, Binary Search Tree, Balanced BST, Threaded Binary Trees, AVL Trees, Applications of Tree like Huffman Coding. Priority Queues & Heaps: Priority Queue, Priority Queue ADT.	10
III	Graph: Introduction, Graph ADT, Advantages and Disadvantages, Graph Representation using adjacency matrix and adjacency list, Graph operations like insertion and deletion of nodes, Graph Traversals using BFS & DFS, Applications of Graphs like shortest path algorithms. Hashing: Hash Table ADT, Advantages & Disadvantages, Concept of hashing, hash table, hash functions, collision, collision avoidance techniques, Applications of hashing.	10
TextBooks : 1. Data Structures And Algorithms Made Easy, Narasimha Karumanchi, 2021 2. Data Structures and Algorithms in Python, Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, Wiley, 2016 Additional References: 1. Introduction to Algorithm, Thomas H Cormen, PHI 2. Fundamentals of Computer Algorithms, Sartaj Sahni and Sanguthevar Rajasekaran Ellis Horowitz, Universities Press, 2018		

Course Code	Course Title	Credits	Lectures/ Week
UMMC SS3-203	Software Engineering	2	2
About the course: The course explores software engineering concepts including but not limited to Software Process, Agile development, Reliable programming , Software Testing and Management to build secure and resilient software products.			
Course Objectives: - Comprehend the processes involved in the development of a software product. - Elaborate on Agile Development Techniques and associated artefacts. - Compare various testing techniques and analyse them based on their purpose. - Develop the skills for project planning and management.			
Learning Outcomes: Students will be able to - Create prototypes for software products with the aid of Software Requirements Document . - Apply agile methods and use the scrum cycle for efficient software development. - Use testing techniques like test cases and code reviews for software testing. - Develop project plans and ensure quality through inspections and reviews.			
Unit	Topic	No. of lectures	
I	Software Process: Software Process models, Process activities , Coping with change, process improvement. Requirements Engineering: Functional and Non-Functional Requirements, Requirements Engineering processes, Requirements Elicitation, Requirements specification, Requirements validation and change, System Modelling using UML Software Products: Product Vision, Software Product Management, Product Prototyping.	10	
II	Software Architecture: Importance, Architectural Design, System decomposition. Agile Software Engineering: Agile Methods, Extreme Programming and Scrum. Features, Scenarios and User Stories: Personas, Scenarios, User Stories, Feature Identification Reliable Programming: Fault Avoidance, Input Validation, Failure Management	10	
III	Software Testing: Functional Testing, Test -automation, Test - driven Development, Security Testing, Code Reviews. Project Planning : Software pricing, Plan-Driven development, Project Scheduling, Agile planning, Estimation Techniques, COCOMO Cost Modelling. Quality management: Software quality, Software Standards, Reviews and Inspections, Quality management and Agile Development, Software Measurement.	10	

TextBooks :

1. Software Engineering, 10th, edition, Ian Sommerville ,2015
2. Engineering Software Products An Introduction to Modern Software Engineering Ian Sommerville,Global Edition, 2019

Additional References:

1. Modern Software Engineering: Doing What Works to Build Better Software Faster David Farley , 1st Edition, 2021.
2. Fundamentals Of Software Architecture An Engineering Approach, 2020, Mark Richards , Neal Ford , GrayScale Edition.

Course Code	Course Title	Credits	Practicals/ Week
UMMCSS3-2 04	Practicals Based on (Computer Science Practical 1)	2	4

Practical of Data Structures -

1. Write a program to implement Abstract Data Types (ADT)
2. Write a program to implement Singly Linked list with insertion, deletion, traversal operations
3. Write a program to implement Doubly Linked list with insertion, deletion, traversal operations
4. Write a program to implement Stack with insertion, deletion, traversal operations
5. Write a program to implement Queue with insertion, deletion, traversal operations
6. Write a program to implement Priority Queue with insertion, deletion, traversal operations
7. Write a program to implement Binary Tree with insertion, deletion, traversal operations
8. Write a program to implement Huffman Coding
9. Write a program to implement Graph with insertion, deletion, traversal operations
10. Write a program to implement Travelling Salesman Problem
11. Write a program to create basic Hash Table for insertion, deletion, traversal operations(assume that there are no collisions)
12. Write a program to create hash table to handle collisions using overflow chaining

Practical of Object Oriented Programming Using Java -

1.
 - a. Accept integer values for a, b and c which are coefficients of quadratic equation. Find the solution of quadratic equation.
2. Accept two n x m matrices. Write a Java program to find addition of these matrices.
3. Accept n strings. Sort names in ascending order
- 4
 - a. Write a program to create a class and implement the concepts of Constructor Overloading, Method Overloading, Static methods
 - b. Write a program to implement the concept of Inheritance and Method Overriding
5.
 - a. Write a program to implement the concepts of Abstract classes and methods
 - b. Write a program to implement the concept of interfaces
6.

Write a program to define user defined exceptions and raise them as per the requirements
7.
 - a. Write a JDBC program that displays the data of a given table
 - b. Write a JDBC program to return the data of a specified record from a given table
 - c. Write a JDBC program to insert / update / delete records into a given table

8.

- a. Construct a simple calculator using the JAVA Swings with minimum functionality.
- b. Construct a GUI using JAVA Swings to accept details of a record of a given table
- c. Write a program using various swing components design Java application to accept a student's resume. (Design form)

Course Code	Course Title	Credits	Lectures/ Week
UMN CSS3-216	Discrete Mathematics	3	3
About the course: Discrete Mathematics provides an essential foundation for virtually every area of Computer Science. The problem-solving techniques honed in Discrete Mathematics are necessary for writing complicated software. Discrete mathematics also builds the gateway to advanced courses in Mathematical Sciences, Data Science, Machine Learning, Software Engineering, etc.			
Course Objectives: • The purpose of the course is to familiarize the prospective learners with mathematical structures that are fundamentally discrete. • This course will enhance prospective learners to reason and ability to articulate mathematical problems. • This course will introduce functions, forming and solving recurrence relations and different counting principles. These concepts will be useful to study or describe objects or problems in computer algorithms and programming languages and these concepts can be used effectively in other courses.			
Learning Outcomes: • After successful completion of this course, learners would be able to: • Define mathematical structures (relations, functions, graphs) and use them to model real life situations. • Understand, construct and solve simple mathematical problems. • Solve puzzles based on counting principles. • Provide basic knowledge about models of automata theory and the corresponding formal languages. • Develop an attitude to solve problems based on graphs and trees, which are widely used in software.			
Unit	Topic	No. of lectures	
I	Functions: Definition of function; Domain, co-domain, range of a function; Examples of standard functions such as identity and constant functions, absolute value function, logarithmic and exponential functions, flooring and ceiling functions; Injective, surjective and bijective functions; Composite and inverse functions. Relations:	15	

	Definition and examples of relation; Properties of relations, Representation of relations using digraphs and matrices; Equivalence relation; Partial Order relation, Hasse Diagrams, maximal, minimal, greatest, least element, Lattices.	
II	Recurrence Relations: Definition and Formulation of recurrence relations; Solution of a recurrence relation; Solving recurrence relations- Backtracking method, Linear homogeneous recurrence relations with constant coefficients; Homogeneous solution of linear homogeneous recurrence relation with constant coefficients; Particular solution of non-linear homogeneous recurrence relation with constant coefficients; General solution of nonlinear homogeneous recurrence relation with constant coefficients; Applications- Formulate and solve recurrence relation for Fibonacci numbers, Tower of Hanoi, Intersection of lines in a plane, Sorting Algorithms. Counting Principles: Basic Counting Principles (Sum and Product Rule); Pigeonhole Principle (without proof) - Simple examples; Inclusion Exclusion Principle (Sieve formula) (without proof); Counting using Tree diagrams.	15
III	Permutations and Combinations: Permutation without and with repetition; Combination without and with repetition; Binomial numbers and identities: Pascal Identity, Vandermonde's Identity, Pascal triangle, Binomial theorem (without proof) and applications; Multinomial numbers, Multinomial theorem (without proof) and applications. Languages, Grammars and Machines: Languages and Grammars – Introduction, Phase structure grammar, Types of grammar, derivation trees; Finite-State Machines with Output; FiniteState Machines with No Output; Regular Expression and Regular Language.	15

Textbooks:

1. Discrete Mathematics and Its Applications, Seventh Edition by Kenneth H. Rosen, McGraw Hill Education (India) Private Limited. (2011)
2. Discrete Mathematics: Semyour Lipschutz, Marc Lipson, Schaum's out lines, McGraw-Hill Inc. 3rd Edition

Additional References:

1. Elements of Discrete Mathematics: C.L. Liu, Tata McGraw- HillEdition.
2. Concrete Mathematics (Foundation for Computer Science): Graham, Knuth, Patashnik Second Edition, Pearson Education.
3. Foundations in Discrete Mathematics: K.D. Joshi, New Age Publication, NewDelhi.

Course Code	Course Title	Credits	Practicals/ Week
-------------	--------------	---------	---------------------

UMNCSS3-2 17	Practicals Based on Discrete Mathematics	1	2
1 Functions – • Identify if the given mapping is a function • Finding domain and range of a given function • Check if the given function is injective/surjective/bijective • Find the inverse of a given function • Operations on functions • Graphs of functions using any online tool 2 Relations – • Representation of relations • Determine if the given relation satisfies equivalence relation/partial order relation • Draw Hasse diagrams • Find maximal, minimal, greatest, least element in a poset • Determine if a given poset is a lattice 3 Recurrence Relation – • Solve recurrence relation using backtracking method • Solve linear homogeneous recurrence relations with constant coefficients • Find homogeneous, particular, general solution of a recurrence relation • Formulate and solving recurrence relation 4 Counting Principles – • Sum and product rule • Pigeonhole Principle • Inclusion Exclusion Principle • Counting using Tree diagrams 5 Permutations and Combinations – • Permutations • Permutations with repetitions • Combinations • Combinations with repetitions • Binomial numbers and Identities • Applications on Binomial theorem			

- Applications on Multinomial theorem

6 Languages and Grammars –

- Find the language generated by given grammar
- Check if a given string belongs or not to a given language/grammar
- Operations on languages
- Identify the type of grammar

7 Finite State Machines –

- Check if a given string is accepted or rejected by FSM without output
- Find the output for a FSM with output
- Describe a machine (diagram/table)

8 Regular Expression and Regular Language –

- Describe the regular expressions represented by given language
- Describe the language represented by given regular expression

Sr. No.	OE Subject	Dept
1	Introduction to Economics	BAF
2	Taxation Literacy	BAF
3	An Introduction to Insurance Sector	BBI
4	Personal Financial Planning	BFM
5	Entrepreneurship and Startup Management	BMS
6	Brand Building	BMS
7	Introduction to Cyber Security	CS
8	AI & us	IT
9	Hormones & you	BT
10	Content Creation & New Media	BAMMC

Course Code	Course Title	Credits	Lectures/ Week
UOEC SS3-25 3	Introduction to Cyber Security	2	2

About the course:

The evolution of Information Communication Technology (ICT) and growing security concerns demands a flexible and generally comprehensive approach to the issue of cyber security. A need has

been felt to address cyber security broadly, as also in sufficient depth so that even students from non-technical streams will develop a more complete picture of the cyber security issues.

Course Objectives:

- Understand the fundamentals of cyber security and cyber crimes.
- Learn the foundations of Cyber security and the threat landscape.
- To equip students with the technical knowledge and skills needed to protect and defend against cyber threats.
- To expose students to responsible use of online social media networks.

Learning Outcomes:

1. Develop a deeper understanding and familiarity with various types of cyberattacks, cyber crimes, vulnerabilities and remedies thereto
2. Analyse and evaluate the importance of personal data its privacy and security
3. Analyse and evaluate the digital payment system security and remedial measures against digital payment frauds.
4. Analyse and evaluate the security aspects of social media platforms and ethical aspects associated with use of social media.
5. Gain awareness of ethical and legal considerations in cybersecurity.

Unit	Topic	No. of lectures
I	Introduction to cyber security ,Definition and importance of cyber security, Evolution of cyber threats Roles and responsibilities in cyber security Types of Cyber Attacks : Web Based, System Based, Passive and Active Attacks Types of Attackers : Cyber Criminal, Hacktivists, State-sponsored, Insiders Threats Web attack: Browser Attacks, Web Attacks Targeting Users, Obtaining User or Website Data, Email Attacks.	10
II	Cyber Security Principals, Data Security Consideration, Cyber Security Technology, Computer Security, Threats, Harm, Vulnerabilities, Controls, Authentication, Authorization, Access Control and Cryptography. Firewall: Introduction, Windows Firewall, Firewall Basics, Packet Filter Vs Firewall, How a Firewall Protects a Network, Packet Characteristic to Filter, IDS/IPS, and other network security devices Security in Mobile Device: Mobile device security principles, Mobile application security, Mobile device management (MDM) and BYOD policies	10

III	Social Media and Ecommerce Security: Types of Social media, Social media platforms, Social media monitoring Social media privacy, Challenges, Security issues related to social media, Flagging and reporting of inappropriate content, Laws regarding posting of inappropriate content, Best practices for the use of Social media E-Commerce and Digital ji jiPayments Definition of E- Commerce, Main components of E-Commerce, Elements of E-Commerce security, E-Commerce threats, Digital payments related common frauds and preventive measures. Information Security and Cyber Law: Need for An Information Security Policy, Information Security Standards – ISO, Introducing Various Security Policies and Their Review Process, Incident Response process and procedure, digital forensic investigation techniques, Introduction to Indian Cyber Law, Objective and Scope of the IT Act.	10
-----	--	----

Text Books :

1. Jennifer L. Bayuk: Cyber Security, Policy Guide Book, Wiley Publisher
2. Nina Godbole: Cyber Security, Wiley Publisher, Latest Edition

Additional References:

1. Cyber Laws: Intellectual Property & E-Commerce Security by Kumar K, Dominant Publishers
2. James Graham, Ryan Olson, Rick Howard, “Cyber Security Essentials”, CRC Press, 15-Dec 2010.

Course Code	Course Title	Credits	Practical/ Week
UVSCS S3-226	Advanced Database Concepts	2	4

About the course:

This course deals with the basic understanding of programming in databases. It touches security, recovery, and transaction aspects of the database. The course will increase the confidence among the learner while dealing with databases.

Course Objectives :

- To develop understanding of concepts and techniques for data management and learn about widely used systems for implementation and usage.
- To develop understanding of Transaction management and crash recovery.
- To develop concepts of programming concepts of databases.

Learning Outcomes

- After successful completion of this course, students would be able to
- Master concepts of stored procedure, functions, cursors and triggers and its use.
- Learn about using PL/SQL for data management.
- Use Collections and records efficiently.

- Understand concepts and implementations of transaction management and crash recovery.

Practicals -

Revision of DDL, DML & DQL Commands

1 Writing PL/SQL Blocks with basic programming constructs by including following:

- Sequential Statements
- unconstrained loop

2 Sequences:

- Creating simple Sequences with clauses like START WITH, INCREMENT BY, MAXVALUE, MINVALUE, CYCLE | NOCYCLE, CACHE | NOCACHE, ORDER | NOORDER.
- Creating and using Sequences for tables.

3 Writing PL/SQL Blocks with basic programming constructs by including following:

- If...then...Else, IF...ELSIF...ELSE... END IF
- Case statement

4 Writing PL/SQL Blocks with basic programming constructs for following Iterative Structure:

- While-loop Statements
- For-loop Statements.

5 Writing PL/SQL Blocks with basic programming constructs by including a GoTO to jump out of a loop and NULL as a statement inside IF.

6 Writing Procedures in PL/SQL Block

- Create an empty procedure, replace a procedure and call procedure
- Create a stored procedure and call it
- Define procedure to insert data
- A forward declaration of procedure

7 Writing Functions in PL/SQL Block.

- Define and call a function
- Define and use function in select clause,
- Call function in dbms_output.put_line
- Recursive function
- Count Employee from a function and return value back
- Call function and store the return value to a variable

8 Creating and working with Insert/Update/Delete Trigger using Before/After clause.

9 Write an Implicit and explicit cursor to complete the task.

10 Create packages and use it in SQL block to complete the task.

11. Write a SQL block to handle exception by writing:

- Predefined Exceptions,
- User-Defined Exceptions,
- Redeclared Predefined Exceptions,

Text Books :

- 1.Mastering PL/SQL Through Illustrations: From Learning Fundamentals to Developing Efficient PL/SQL Blocks, Dr. B. Chandra, BPB Publication, 2020
2. Oracle Pl/Sql Training Guide., Training guide, BPB Publications, 2016
3. Raghu Ramakrishnam, Gehrke, Database Management Systems, McGraw-Hill,3rd Edition, 2014

4. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, 6th Edition 2019

5. RDBMS In-Depth, Dr. Madhavi Vaidya, BPB Publications, Feb 2021

Additional References:

1. Ivan Bayross, "SQL, PL/SQL -The Programming language of Oracle", B.P.B. Publications 2009

2. Ramez Elmasri & Shamkant B. Navathe, Fundamentals of Database Systems, Pearson Education, 2008

Course Code	Course Title	Credits	Lectures/Week
UAECSS3-242 to UAECSS3-245	AEC - Modern Indian Language	2	2
Marathi / Hindi / Sanskrit / Sindhi			

Course Code	Course Title	Credits	Activity Based Hours
UCCCSS3-276	Co-curricular Courses	2	
NSS / Sports / Health & Wellness / Music / Cultural Activities / Sunday School / Department Activities etc			

SEMESTER IV

Course Code	Course Title	Credits	Lectures/ Week
UMM CSS4-2 01	Advanced Application Development	2	2
About the course: The course aims at developing scalable, robust, and maintainable web applications using MEAN stack and developing advanced mobile applications using Flutter			
Course Objectives: - To understand all the necessary and important technologies such as MongoDB, Express.js, AngularJS, and Node.js. - To understand modern app development using Flutter			
Learning Outcomes: After successful completion of this course, students would be able to - Store the data in NoSQL, a document-oriented MongoDB database that brings performance and scalability. - Use Node.js and Express Framework for building fast, scalable network applications - Use AngularJS framework that offers declarative, two-way data binding for web applications. - Integrate the front-end and back-end components of the MEAN stack. - Develop robust mobile applications using Flutter.			
Unit	Topic	No. of lectures	
I	Node.js (N): Introduction to Node.js. Installing Node.js. The package.json File. The Node.js Event Loop. The I/O Cycle. The Anatomy of a Node.js Module. Creating Node Modules. Exploring the Node.js HTTP Module. Creating an HTTP Web Server with Node.js. Responding to HTTP Requests. Routing in Node.js. Creating a Sample Node.js Application. MongoDB(M): Introduction to MongoDB. Installing MongoDB. Using MongoDB Compass. Using Mongo Shell Interface. Connecting to MongoDB. Creating Schemas and Models. CRUD operations, MongoDB Atlas	10	
II	Server-Side Development with Express (E): Introduction to the Express Framework. Installing and Testing Express. Creating a Node.js Express App. Restructuring an Express App. Creating Templates. Using Express Middleware Functions. Creating the List Page. Creating the Details Page. Creating the Edit Page. Creating the Add Page. Deleting Data. REST API Basics. Testing REST APIs. Refactoring APIs. Understanding Angular.JS(A): Getting Started with Angular. Creating an Angular Application. Angular Project File Structure. Anatomy of an Angular Component. One-way Data Binding. Two-way Data Binding. Using NgIf Directive. Using NgForOf Directive. Angular Modules. Configuring Templates. Creating Navigations. Working with Template-driven Forms. Working with Reactive Forms. Validating Form Data. Reading Data from Database. Inserting Data into Database. Updating Data in the Database. Delete Data from Database.	10	

III	Understanding Flutter: Importance of Flutter, Flutter Framework, Android Studio, Flutter SDK, Installing and Configuring Flutter SDK. Dart Programming: main() function, Dart Variables, Dart Data Types, Dart Conditional Operators, Control Flow & Loops. Dart Functions - Functions, Function Structure, creating a Function, Function Returning Expression. Object-Oriented Programming (OOP) - Creating a Class, Adding Methods to Classes, Class — Getters and Setters, Class Inheritance, Abstract Class. Flutter Widgets Fundamentals: Scaffold Widget, Image Widget, Container Widget, Column and Row Widgets, Icon Widget, Layouts in Flutter, Card Widget, Hot Reload and Hot Restart, Stateful and Stateless.	10
-----	---	----

Text Books:

1. Node.js, MongoDB and Angularjs Web Development: The definitive guide to using the MEAN stack to build web applications by Brad Dayley, Brendan Dayley, Caleb Dayley, Pearson, 2018.
2. Beginning Flutter: A Hands On Guide to App Development by Marco L. Napoli, Wrox, 2019.

Additional References:

- 1.Full Stack Javascript Development with Mean - MongoDB, Express, AngularJS, and Node.JS by Adam Bretz, Colin J Ihrig, Shroff/SitePoint, 2015.
2. Practical Flutter by Zammetti Frank, Apress, 2019.

Course Code	Course Title	Credits	Lectures/ Week
UMM CSS4-2 02	Computer Networks	2	2
About the course: This course introduces computer networks, with a special focus on the Internet architecture and protocols. The course includes topics such as network architectures, addressing, naming, forwarding, routing, communication reliability, the client-server model, web, email and other application layer protocols.			
Course Objectives: ● To Understand Basic Concepts of Networking. ● To Understand Working of Network Layer Architecture. ● To Learn Practical Implementation of Basic Routing Algorithms. ● To Learn Different Networking Protocols.			
Learning Outcomes: After successful completion of this course, students would be able to ● Learn basic networking concepts and layered architecture. ● Understand the concepts of networking, which are important for them to be known as a networking professionals			
Unit	Topic	No. of lectures	

I	Introduction: Networking standards and Administrations, networks, network types – LAN, MAN, WAN. Network Models: The OSI model, TCP/IP protocol suite Digital transmissions: Digital-to-digital conversion, analog-to-digital conversion, transmission modes Analog transmissions: digital-to-analog conversion, analog-to-analog conversion. Transmission media: Guided Media, Unguided Media Switching: Introduction, Circuit Switched Network, Packet Switching.	10
II	Introduction to Data Link Layer: Concepts, Features, Address resolution protocol. Error detection and correction: -Block coding, cyclic codes, checksum, forward error correction, error correcting codes, error detecting codes. Wired LANs Ethernet: Ethernet Protocol Wireless LANs: Introduction, LiFi Technology, Bluetooth Introduction to Network Layer: IPv4 addressing, forwarding of IP packets, Network Layer Protocols : Internet Protocol, ICMPv4.	10
III	Unicast Routing: Introduction, routing algorithms, unicast routing protocols. Next generation IP: IPv6 addressing, IPv6 protocol, transition from IPv4 to IPv6. Introduction to the Transport Layer: Transport Layer Protocol, User Datagram Protocol, Transmission Control Protocol. Introduction to Application Layer: Protocols: WWW, HTTP, FTP, Electronic Mail, TELNET, Secure Shell, DNS, SNMP Quality of Service: Flow Characteristics, QoS Parameters, Data Flow to improve QoS	10
Textbooks: 1. Data Communications and Networking, Behrouz A. Forouzan, Fifth Edition, TMH, 2018. 2. Computer Network, Andrew S. Tanenbaum, David J. Wetherall, Fifth Edition, Pearson Education, 2018. Additional References: 1. Computer Network, Bhushan Trivedi, Oxford University Press, 2016 2. Data and Computer Communication, William Stallings, PHI, 2017		

Course Code	Course Title	Credits	Practicals/ Week
USECSS 4-231	IoT	2	4

About the course: The course aims to provide basic understanding of IoT, different types of IoT platforms and different types of applications that can be built.

Course Objectives:

- Introduce various types of IoT platforms
- Interfacing various types of devices using different protocols with IoT
- Understand practical applications of IoT in real life world

Learning Outcomes: After successful completion of this course, students would be able to

- understand IoT
- use different types of IoT Platforms and interfaces
- understand and implement an idea of various types of applications built using IoT

IoT Platform - Raspberry PI or any other platform

1. Preparing IoT Platform: Hardware preparation and Installation
2. Determining Voltage, Current and Resistor requirements for connecting components
3. Demonstrate interfacing with Raspberry Pi with basic LED & buttons
4. Dimming LED : PWM to control LED intensity using IoT Platform
5. Stepper Motor Control: PWM to control stepper motor speed using IoT Platform
6. GPIO LED Grid Module: Program the 8X8 Grid with Different Formulas
7. SPI : Camera Connection and capturing Images/Videos using SPI.
8. SPI : Capture Image using SPI Camera and perform face detection using opencv library.
9. Use basic sensors with IoT Platform(from 32 sensor kit)
 - a. Door open/close sensor using hall magnet sensor
 - b. Turn on LED when ambient lighting is dark using photo-resistor sensor
10. Use advanced sensors with IoT Platform
 - a. Fetch GPS location using GPS Sensor and display on 8X8 Grid
 - b. Record Fingerprint, turn on green LED if fingerprint matches, turn on Red LED if fingerprint does not match.
11. Node RED(or any other IoT Cloud platform): Connect LED to Internet of Things (or any suitable platform) and perform various experiments
12. Communicating with Web Server to send/receive values from/to the IoT platform using specific protocols.

Text Books :

- 1.Introduction to IoT Paperback by Sudip Misra , Anandarup Mukherjee , Arijit Roy Cambridge Press, 2022
- 2.Jain, Prof. Satish, Singh, Shashi, “Internet of Things and its Applications”, 1st Edition, BPB,2020.
- 3.Shriram K Vasudevan, Abhishek S Nagarajan, RMD Sundaram, Internet of Things, Wiley, India,2019
4. IoT and Edge Computing for Architects - Second Edition, by Perry Lea, Publisher: Packt Publishing, 2020
5. 21 IOT Experiments Paperback – 1 January 2018,Yashavant Kanetkar, Shrirang Korde, BPB Publications

Additional References:

- 1.Internet of Things by Vinayak Shinde, SYBGEN Learning India Pvt. Ltd, 2020
2. Internet of things, Dr. Kamlesh Lakhwani, Dr. Hemant kumar Gianey, Josef Kofi Wireko, Kamalkant Hiran, BPB Publication, 2020

Course Code	Course Title	Credits	Practicals/ Week
UMMCSS4-204	Practicals Based on (Computer Science Practical 4)	2	4

Practical of Advanced Application Development

1. Write a program to implement MongoDB data models
2. Write a program to implement CRUD operations on MongoDB
3. Write a program to perform validation of a form using AngularJS
4. Write a program to create and implement modules and controllers in AngularJS
5. Write a program to implement Error Handling in AngularJS
6. Create an application for Customer / Students records using AngularJS
7. Write a program to create a simple web application using Express, Node JS and Angular JS
8. Create a simple HTML “Hello World” Project using AngularJS Framework and apply ng-controller, ng-model and expressions
9. Write a Dart program to implement a conditional statement and loop.
10. Write a Dart program to implement a different type of function.
11. Write a Dart program to implement OOPs concept.

Practical of Computer Networks

- 1 Using Windows-cmd, execute following networking commands and note the output: ping, traceroute, netstat, arp, ipconfig, Getmac, hostname, NSLookUp,pathping, SystemInfo
- 2 Using Packet Tracer, create a basic network of two computers using appropriate network wire. Use Static IP address allocation and show Connectivity
- 3 Using Packet Tracer, create a basic network of One server and two computers using appropriate network wire. Use Dynamic IP address allocation and show connectivity

- 4 Using Packet Tracer, create a basic network of One server and two computers and two mobile / movable devices using appropriate network wire. Show connectivity
- 5 Using Packet Tracer, create a network with three routers with RIPv1 and each router associated network will have minimum three PC. Show Connectivity
- 6 Using Packet Tracer, create a network with three routers with RIPv2 and each router associated network will have minimum three PC. Show Connectivity
- 7 Using Packet Tracer, create a network with three routers with OSPF and each router associated network will have minimum three PC. Show Connectivity
- 8 Using Packet Tracer, create a network with three routers with BGP and each router associated network will have minimum three PC. Show Connectivity
- 9 Using Packet Tracer, create a wireless network of multiple PCs using appropriate access point.
- 10 Using Wireshark, network analyzer, set the filter for ICMP, TCP, HTTP, UDP, FTP and perform respective protocol transactions to show/prove that the network analyzer is working

Course Code	Course Title	Credits	Lectures/ Week
UMNCS S4-216	CALCULUS	3	3
About the course: Calculus is a branch of mathematics that involves the study of rates of change. In Computer Science, Calculus is used in Machine Learning, Data Mining, Scientific Computing, Image Processing, and creating the graphics and physics engines for video games, including the 3D visuals for simulations.			
Course Objectives: • The primary objective of this course is to introduce the basic tools of Calculus which are helpful in understanding their applications to real world problems. • The course is designed to have a grasp of important concepts of Calculus in a scientific way. • It covers topics from as basic as definition of functions to partial derivatives of functions in a gradual and logical way. • The learner is expected to solve as many examples as possible to get complete clarity and understanding of the topics covered.			
Learning Outcomes: After successful completion of this course, learners would be able to: • Develop mathematical skills and enhance the thinking power of learners. • Understand mathematical concepts like limit, continuity, derivative, integration of functions, partial derivatives. • Appreciate real world applications which use the learned concepts. • Skill to formulate a problem through Mathematical modelling and simulation.			

Unit	Topic	No. of lectures
I	DERIVATIVES AND ITS APPLICATIONS: Review of Basic Concepts: Functions, limit of a function, continuity of a function, derivative function. Derivative In Graphing And Applications: Increase, Decrease, Concavity, Relative Extreme; Graphing Polynomials, Rational Functions, Cusps and Vertical Tangents. Absolute Maxima and Minima, Applied Maximum and Minimum Problems, Newton's Method.	10
II	INTEGRATION AND ITS APPLICATIONS: Integration: An Overview of the Area Problem, Indefinite Integral, Definition of Area as a Limit; Sigma Notation, Definite Integral, Evaluating Definite Integrals by Substitution, Numerical Integration: Simpson's Rule. Applications of Integration: Area between two curves, Length of a plane curve. Mathematical Modeling with Differential Equations: Modeling with Differential Equations, Separation of Variables, Slope Fields, Euler's Method, First-Order Differential Equations and Applications	10
III	PARTIAL DERIVATIVES AND ITS APPLICATIONS: Functions of Several Variables: Functions of two or more variables, Limits and Continuity of functions of two or three variables. Partial Derivatives: Partial Derivatives, Differentiability, Differentials, and Local Linearity, Chain Rule, Implicit Differentiation, Directional Derivatives and Gradients, Applications of Partial Derivatives: Tangent Planes and Normal Vectors, Maxima and Minima of Functions of Two Variables	10

Textbooks:

1. Calculus: Early transcendental (10th Edition): Howard Anton, IrlBivens, Stephen Davis, John Wiley & sons, 2012.

Additional References:

1. Calculus and analytic geometry (9th edition): George B Thomas, Ross L Finney, Addison Wesley, 1995

2. Calculus: Early Transcendentals (8th Edition): James Stewart, Brooks Cole, 2015.

3. Calculus (10th Edition): Ron Larson, Bruce H. Edwards, Cengage Learning, 2013.

4. Thomas' Calculus (13th Edition): George B. Thomas, Maurice D. Weir, Joel R. Hass, Pearson, 2014.

Course Code	Course Title	Credits	Tutorials/ Week
UMNCSS4-2 17	Tutorials/Practicals Based on Calculus	1	1

1. Review of Basic Concepts –

- Functions of one variable, its domain and range, Operations on functions
- Limits of functions of one variable
- Continuity of functions of one variable
- Derivatives of functions of one variable

2. Applications of Derivatives I –

- Increasing and Decreasing functions
- Concavity and inflection points
- Relative Extrema
- Absolute Extrema

3. Applications of Derivatives II –

- Analysis of polynomials
- Graphing rational functions
- Graphs With Vertical Tangents And Cusps
- Newton's method to find approximate solution of an equation

4. Integration –

- Finding area using rectangle method and antiderivative method
- Indefinite and definite integrals
- Properties of integrals
- Numerical integration using Simpson's rule.

5. Applications of Integration –

- Area between two curves
- Length of a plane curve

6. Differential Equations –

- Solution of a first order first degree differential equation using variable separable method
- Solution of a first order linear differential equation using integrating factor
- Numerical solution of first-order equations using Euler's method
- Modeling using differential equation

7. Functions of Several Variables –

- Functions of two or more variables, its domain and range, Operations on functions, level curves
- Limits of functions of two or three variables
- Continuity of functions of two or three variables

8. Partial Derivatives I –

- Partial derivatives of functions, First and Second order partial derivatives, Mixed derivative theorem, Higher order partial derivatives
- Differential for functions of two or three variables
- Local linear approximation for functions of two or three variables

9. Partial Derivatives II –

- Chain rule for functions of two or three variables
- Implicit differentiation
- Directional derivatives and gradient

10. Applications of Partial Derivatives–

- Tangent Planes and Normal Vectors for functions of two or three variables
- Maxima and Minima of Functions of Two Variables

Course Code	Course Title	Credits	Practical/ Week
USEC SS4-23 1	Mobile Application Development	2	4

About the Course: This course introduces students to Flutter, Google's UI toolkit for building cross-platform mobile applications. It covers Dart programming, UI design, navigation, state management, API handling, database integration (SQLite & Firebase), and deployment. By the end of this course, students will be able to develop interactive and feature-rich Android applications using Flutter.

Course Objectives:

- To provide an understanding of Flutter framework and Dart programming language.
- To develop interactive UI designs using Flutter widgets and layouts.
- To enable students to handle user inputs, form validation, and error handling.
- To implement state management techniques for efficient app development.
- To teach students how to fetch and display data from databases (SQLite/Firebase).
- To introduce navigation techniques and gesture handling in Flutter applications.

Learning Outcomes:

After successful completion of this course, students would be able to

- Develop Android applications using Flutter and Dart.
- Design responsive UI layouts using Flutter widgets like Row, Column, and Stack.
- Implement navigation between multiple screens using Named Routes.
- Validate user inputs and handle errors effectively in forms.
- Integrate SQLite and Firebase for data storage and retrieval.
- Implement animations and gestures to enhance user experience.

1. Install Flutter and Dart SDK.

Write a simple Dart program covering variables, functions, and OOP concepts

2. Designing Layouts

Create an Android application using Row, Column, and Stack widgets to design different layouts.

3. Demonstrate Stateless and Stateful widgets.

Implement various widgets (Text, Image, Container, Buttons, Forms, etc)

4. Use Named Routes for better navigation management.

Implement navigation between multiple screens using Navigator.push() & pop().

5. Create an application that implements custom themes and styles.

6. Show error handling techniques for user inputs.

Implement form validation (empty fields, email format, password strength, etc.)

7. Implement Tap, Swipe, Drag & Drop, and Long Press gestures in an app.
8. Create an app using Implicit & Explicit animations (e.g., Animated Container).

9. Connecting to Local Database

Create an app to store and retrieve user data with CRUD operation from SQLite database

10. Connecting to Cloud NoSql Database

Firebase Firestore: Store, retrieve, and update data with authentication.

Text Books :

1. Beginning Flutter: A Hands-On Guide to App Development – By Marco L. Napoli
2. Flutter for Beginners – By Alessandro Biessek

Additional References:

1. <https://docs.flutter.dev/>

Course Code	Course Title	Credits	Lectures/ Week
UOECS S4-253	Introduction to Cyber Security	2	2
About the course: The evolution of Information Communication Technology (ICT) and growing security concerns demands a flexible and generally comprehensive approach to the issue of cyber security. A need has been felt to address cyber security broadly, as also in sufficient depth so that even students from non-technical streams will develop a more complete picture of the cyber security issues.			
Course Objectives: • Understand the fundamentals of cyber security and cyber crimes. • Learn the foundations of Cyber security and the threat landscape. • To equip students with the technical knowledge and skills needed to protect and defend against cyber threats. • To expose students to responsible use of online social media networks.			
Learning Outcomes: 1. Develop a deeper understanding and familiarity with various types of cyberattacks, cyber crimes, vulnerabilities and remedies thereto 2. Analyse and evaluate the importance of personal data its privacy and security 3. Analyse and evaluate the digital payment system security and remedial measures against digital payment frauds. 4. Analyse and evaluate the security aspects of social media platforms and ethical aspects associated with use of social media. 5. Gain awareness of ethical and legal considerations in cybersecurity.			
Unit	Topic	No. of lectures	

I	Introduction to cyber security ,Definition and importance of cyber security, Evolution of cyber threats Roles and responsibilities in cyber security Types of Cyber Attacks : Web Based, System Based, Passive and Active Attacks Types of Attackers : Cyber Criminal, Hacktivists, State-sponsored, Insiders Threats Web attack: Browser Attacks, Web Attacks Targeting Users, Obtaining User or Website Data, Email Attacks.	10
II	Cyber Security Principals, Data Security Consideration, Cyber Security Technology, Computer Security, Threats, Harm, Vulnerabilities, Controls, Authentication, Authorization, Access Control and Cryptography. Firewall: Introduction, Windows Firewall, Firewall Basics, Packet Filter Vs Firewall, How a Firewall Protects a Network, Packet Characteristic to Filter, IDS/IPS, and other network security devices Security in Mobile Device: Mobile device security principles, Mobile application security, Mobile device management (MDM) and BYOD policies	10
III	Social Media and Ecommerce Security: Types of Social media, Social media platforms, Social media monitoring Social media privacy, Challenges, Security issues related to social media, Flagging and reporting of inappropriate content, Laws regarding posting of inappropriate content, Best practices for the use of Social media E-Commerce and Digital Payments Definition of E- Commerce, Main components of E-Commerce, Elements of E-Commerce security, E-Commerce threats, Digital payments related common frauds and preventive measures. Information Security and Cyber Law: Need for An Information Security Policy, Information Security Standards – ISO, Introducing Various Security Policies and Their Review Process, Incident Response process and procedure, digital forensic investigation techniques, Introduction to Indian Cyber Law, Objective and Scope of the IT Act.	10

Text Books :

1. Jennifer L. Bayuk: Cyber Security, Policy Guide Book, Wiley Publisher
2. Nina Godbole: Cyber Security, Wiley Publisher, Latest Edition

Additional References:

1. Cyber Laws: Intellectual Property & E-Commerce Security by Kumar K, Dominant Publishers
2. James Graham, Ryan Olson, Rick Howard, “Cyber Security Essentials”, CRC Press, 15-Dec 2010.

Course Code	Course Title	Credits	Lectures/Week
UAECSS4-24 2 to UAECSS4-24 5	AEC - Modern Indian Language (Marathi / Hindi / Sanskrit / Sindhi)	2	2
As offered by College from a basket*			

Course Code	Course Title	Credits	Activity Based Hours
UCCCSS4-276	Co-curricular Courses	2	
NSS / Sports / Health & Wellness / Music / Cultural Activities / Sunday School / Department Activities etc			

MODALITY OF ASSESSMENT

The performance of the learners for those exams having Semester End Examinations and Internal Assessment shall be evaluated in two parts as per the following ratio:

Semester End Examination: Internal Assessment [60:40]

The learner's performance shall be assessed by conducting the **Semester-end Examination with 60% marks** and **Continuous Internal Assessment (CIA) with 40% marks**. Practical Examination will consist of Semester-end examination.

Students will have to score 40% of marks INDIVIDUALLY in Internal assessment as well as Semester-end Examination to pass the course.

Internal Assessment: It is defined as the assessment of the learners on the basis of internal evaluation by way of participation of learners in various academic and correlated activities in the given semester of the programme.

Semester End Assessment: It is defined as the assessment of the learners on the basis of Performance in the Semester-end Theory/ Practical examination.

Table-1- Mode of Assessment under NEP 2020

Name of the course	Credits	Nature of Evaluation & Mode of Assessment	Duration	Marks
Major(Theory) OR Minor (Theory) OR Skill Enhancement Course (SEC)(Theory) OR Ability Enhancement Course (AEC)(Theory) OR Value Education Course (VEC)	2 Credit	1. Internal (40%) (Table 2)	90 mins	20
		2. Semester-end Theory Examination(60%) (Table 3)		30
Major(Practical) (2 subjects)	2 Credit	Semester-end Practical Examination (Table-4)	3 hours	50
Open Electives(OE) OR Indian Knowledge system (IKS)	2 Credit	Continuous Internal Evaluation - Assignments/ Presentations/ Group Discussion/ Case Studies etc. Any two modes of assessment with evaluation at Department level (25 M Each)	-	50
Vocation Skill Course (VSC)(Practical)	2 Credit	Practicals (Table-4)	2 horus	50

On Job training (OJT) /NSS/ NCC/ Co-curricular Course(CC)/ Field project (FP)/ Internship	2 Credit	Continuous Internal Evaluation --Tests, Essays, Articles, Group assignments/ Reports/ Journals/ Diaries/ Reviews/ Dissertations/ Observations of Students (As per the nature of the course)	-	
---	---------------------	--	---	--

Note: CC-Co-curricular Courses include involvement/ participation in various areas such as Cultural Activities, Departmental activities, Fine/ Applied/ Visual/ Performing Arts, Sports and fitness, NSS/ NCC, DLLE, The Sunday School (TSS), Health & Wellness, Yoga education etc. Record of involvement /participation by students has to be documented with signatures of staff concerned in students' CC Cards.

Table - 2

Theory - Mode of assessment-Continuous Internal Assessment [40%]

Evaluation type
1. Assignments. 2. Project based learning activities (Group Discussion Research/ Case studies/ Reports / Assignments / Presentations / Skit / Poster / etc.). 3. Class Test (Objective - Multiple Choice Questions/ Subjective). 4. Active participation in class activities. 5. Overall conduct as a responsible student with respect to good behaviour, leadership qualities, interpersonal skills etc. For example - Depending on the subject - 15 Marks MCQ Test + 5 Marks Assignment/Presentation/Case Study OR 10 Marks MCQ Test + 10 Marks Assignment/Presentation/Case Study

Table-3

Paper Pattern for Semester-End Theory Examination
1. Each unit should have equal weightage 2. Minimum one question per unit 3. No MCQ/True-False type of questions should be asked, only descriptive questions are allowed 4. Each question should have extra sub-questions(max 100% option)

Table-4

Semester End Examination for Practical Assessment -50 marks (2-Credit)-Major
[To be conducted at departmental level]

For 1 subject		For 2 subjects	
Any one/ more experiment/s	35	Min two experiments per subject	40
Viva	10	Viva	05
Journal	05	Journal	05
OR			

Practical	30		
Case-Study	20		
Total	50 M	Total	50 M

Evaluation Pattern for Environmental Science

Evaluation Pattern	Credits -2
Continuous Internal Evaluation	Out of 20 marks
Semester End Evaluation	Out of 30 marks
Total Marks	50 marks

Continuous Internal Evaluation

Class Test: 10 marks

Assignment : 10 marks

Question Paper Pattern

Maximum Marks: 30

Questions to be set: 03

Duration: 90 mins

Note: All questions are compulsory carrying 10 marks each

Q.1 Attempt any two - (10)

Q.2 Attempt any two - (10)

Q.3 Attempt any two - (10)

OR

Q.1 Attempt any two- (8)

Q.2 Attempt any two -(8)

Q.3 Attempt any two -(8)

Q.4 Attempt any three- (6)