Telephone No. 2419677/2419361
Fax: 0821-2419363/2419301

e-mail : registrar@uni-mysore.ac.in
www.uni-mysore.ac.in

UNIVERSITY OF MYSORE
Estd. 1916

VishwavidyanilayaKaryasoudha
Crawford Hall, Mysuru- 570 005

No.AC2(S)/35/2025-26

Dated: 09-06-2026

Notification

Sub:- Revised Syllabus and Scheme of Examinations of MCA (PG) Computer Applications (PG) programme from the Academic year 2026 27.

Ref:-1. Decision of Board of Studies in Computer Science (PG) meeting held on 23-03-2026 & 24-03-2026.
2. Decision of the Faculty of Science & Technology meeting held on 22-04-2026.
3. Decision of the Academic Council meeting held on 01-06-2026.

*****

The Board of Studies in Computer Science (PG) which met on 23-03-2026 & 24-03-2026 has resolved to recommend and approved the revised Syllabus and Scheme of Examinations of Master of Computer Applications (PG) programme from the Academic year 2026-27.

The Faculty of Science & Technology and Academic Council at their meetings held on 22-04-2026 and 01-06-2026 respectively have also approved the above mentioned Syllabus and Scheme of Examinations, accordingly it is hereby notified.

The Syllabus and Scheme of Examinations contents may be download from the University Website www.uni-mysore.ac.in.

Registrar
Registrar
University of Mysore
Mysore

To;

1. The Registrar (Evaluation), University of Mysore, Mysuru.
2. The Chairman, BOS/DOS in Computer Science, Manasagangothri, Mysuru.
3. The Dean, Faculty of Science & Technology, DOS in Botany, MGM.
4. The Director, PMEB, University of Mysore, Mysuru.
5. The Director, College Development Council, Manasagangothri, Mysuru.
6. The Director, ICD, MGM-Request to publish this Notification in University website.
7. The Deputy Registrar/Assistant Registrar/Superintendent, Administrative Branch and Examination Branch, University of Mysore, Mysuru.
8. The PA to Vice-Chancellor/ Registrar/ Registrar (Evaluation), University of Mysore, Mysuru.
9. Office Copy.

UNIVERSITY OF MYSORE

Department of Studies in Computer Science

Manasagangotri, Mysuru-570006

Regulations and Syllabus

Master of Computer Applications

(MCA)

[Two-year]

Under

Choice Based Credit System (CBCS)

PROGRAM DETAILS

Program Name:

Master of Computer Applications (MCA)

Nature of the Program:

Postgraduate Degree Program

(Choice Based Credit System – CBCS)

Duration of the Program:

Two academic years, consisting of four semesters, or as prescribed by the University from time to time.

Eligibility for Admission:

Candidates for admission to the M.C.A. degree programme must have a B.C.A/B.Sc./B.Com degree with Mathematics as one of the subject at 10+2 level or at graduation, in accordance with the norms prescribed by the University.

Medium of Instruction:

English

Program Structure:

The Program comprises a structured combination of Hard Core (HC) Courses, Soft Core (SC) Courses, Practical Courses, Project works and other components distributed across semesters in accordance with the Choice Based Credit System (CBCS) and University regulations in force from time to time.

Credits:

Credits are assigned to each course based on instructional hours, learning activities, and expected learner effort, in conformity with the Choice Based Credit System (CBCS) norms prescribed by the University.

Academic Regulations

The academic regulations of the M.C.A. Program shall be governed by the rules and ordinances of the University of Mysore and shall be inconformity with the Choice Based Credit System (CBCS) and UGC-LOCF framework in force from time to time.

Attendance Requirements:

- Students shall be required to maintain a minimum percentage of attendance as prescribed by the University of Mysore to become eligible to appear for the end-semester examination in each course.
- Condonation of shortage of attendance, if any, shall be governed by the rules and procedures notified by the University from time to time.

Assessment and Evaluation Scheme:

Evaluation of student performance shall be based on a combination of Continuous Internal Assessment (CIA) and End-Semester Examination (ESE), as prescribed under the CBCS framework of the University of Mysore.

Passing Criteria:

- As per CBCS norms of University of Mysore.

Grading System:

- As per CBCS norms of University of Mysore.

Department Vision

- To be recognized as a Premier Center for Excellence in Computer Science education fostering Research, Innovation and Entrepreneurial attitude in students.

Department Mission

- Empowering students and research scholars with knowledge, advanced research and skills in developing technological based solutions for day-to-day problems.

Department Motto

- Demonstrating abilities in building up Artificial Intelligence (AI) based solutions with cutting edge technology for various day-to-day complex problems.

Program Learning Outcomes (PLOs)

On successful completion of the **Master of Computer Applications (MCA)** programme, the graduates will be able to:

PLO 1: Disciplinary Knowledge

Demonstrate advanced and integrated knowledge of core and emerging areas of Computer Science including algorithms, software systems, databases, networks, operating systems, artificial intelligence, and data analytics to solve complex computing problems.

PLO 2: Problem Solving Skills

Analyze complex computational problems, identify requirements, and design efficient, scalable, and secure software or system-based solutions using appropriate algorithms, tools, and technologies.

PLO 3: Programming Skills

Apply advanced programming concepts, software engineering principles, and modern development frameworks to design, implement, test, and maintain robust software applications and systems.

PLO 4: Research and Scientific Reasoning

Conduct independent and collaborative research by formulating research problems, applying appropriate methodologies, analyzing data, and interpreting results to contribute to knowledge creation or technological innovation.

PLO 5: Use of Modern Tools and Technologies

Select and apply appropriate modern computing tools, programming environments, databases, analytics platforms, and emerging technologies for modeling, simulation, analysis, and solution development.

PLO 6: Teamwork

Function effectively as a team member, or leader in multidisciplinary and multicultural environments to accomplish common objectives through collaboration and mutual respect.

PLO 7: Ethics, Values, and Professional Responsibility

Demonstrate ethical reasoning, professional integrity, and social responsibility in computing practices, including data privacy, cyber security, intellectual property rights, sustainability, and societal impact of technology.

PLO 8: Lifelong Learning and Professional Development

Recognize the need for and engage in independent and lifelong learning to adapt to rapid technological changes, pursue higher studies, research, entrepreneurship, or professional careers.

Graduate Attributes (GAs)

On successful completion of the **Master of Computer Applications (MCA)** programme, graduates will possess the following attributes:

GA 1: Advanced Computing Knowledge

Apply comprehensive knowledge of computer science fundamentals, advanced computing concepts, and emerging technologies to understand and solve real-world problems effectively.

GA 2: Analytical and Problem-Solving Ability

Demonstrate strong analytical thinking and problem-solving skills to design efficient, scalable, and secure solutions for complex computational and real-life challenges.

GA 3: Programming and Software Development Expertise

Exhibit proficiency in programming languages, software engineering practices, and modern development frameworks to build, test, and maintain high-quality software systems.

GA 4: Research, Innovation, and Critical Thinking

Engage in research and innovation by applying scientific reasoning, critical thinking, and appropriate methodologies to address complex problems and contribute to technological advancement.

GA 5: Proficiency in Modern Tools and Technologies

Utilize contemporary tools, platforms, and technologies including data analytics, artificial intelligence, cloud computing, and software development environments for effective solution development.

GA 6: Teamwork, Leadership, and Collaboration

Work effectively as an individual and as a member or leader in multidisciplinary teams, demonstrating collaboration, adaptability, and interpersonal skills in diverse environments.

GA 7: Ethics, Professionalism, and Social Responsibility

Adhere to ethical principles, professional standards, and societal responsibilities, ensuring data privacy, cyber security, sustainability, and responsible use of technology.

GA 8: Lifelong Learning and Career Development

Recognize the importance of continuous learning and professional development to adapt to evolving technologies, pursue research, higher education, entrepreneurship, and career advancement.

SDGs for M.C.A. Programme:

SDG 1: Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.

SDG 2: Achieve gender equality and empower all women and girls.

SDG 3: Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation.

SDG 4: Take urgent action to combat climate change and its impacts.

SDG 5: Strengthen the means of implementation and revitalize the global partnership for sustainable development

Master of Computer Applications (MCA 2 Years)

Course Wise Detailed Syllabus for MCA

MCA First Semester

SI. No.	Code	Course Title	HC/SC/OE	L:T:P	Credit Value
1		Object Oriented Programming with C++	HC	3:0:1	**4**
2		Data Indexing Techniques	HC	2:1:1	**4**
3		Data Clustering Algorithms	HC	3:0:1	**4**
Choose atmost three courses as soft core courses					
4		Discrete Mathematical Structure	SC	3:0:1	**4**
5		Digital Image Processing	SC	3:0:1	**4**
6		Distributed Operating System	SC	2:1:1	**4**
7		E-Commerce and E-Governance	SC	3:0:1	**4**

MCA Second Semester

SI. No.	Code	Course Title	HC/SC/OE	L:T:P	Credit Value
1		Linux Programming	HC	2:1:1	**4**
2		Advanced Web Technology	HC	3:0:1	**4**
3		Learning Models	HC	2:1:1	**4**
Choose atmost three courses as soft core courses					
4		Algorithms	SC	2:1:1	**4**
5		Theory of Languages	SC	2:1:1	**4**
6		Fundamentals of Software Engineering	SC	3:0:1	**4**
7		.Net with C#	SC	2:1:1	**4**

MCA Third Semester

SI. No.	Code	Course Title	HC/SC/OE	L:T:P	Credit Value
1		Network Security	HC	2:1:1	**4**
2		Distributed Databases	HC	3:0:1	**4**
Choose atmost four courses as soft core courses					
3		Internet of Things	SC	3:0:1	**4**
4		Deep Learning	SC	2:0:2	**4**
5		Cloud Computing	SC	3:1:0	**4**
6		Artificial Intelligence	SC	2:1:1	**4**
7		Probability and Statistics	SC	3:1:0	**4**

MCA Fourth Semester

SI. No.	Code	Course Title	HC/SC/OE	L:T:P	Credit Value
1		Project	HC	0:2:6	**8**
Choose atmost four courses as soft core courses (Selecting soft core courses not opted in the previous Semesters, and/or including the following two. This shall be exercised if one has shortage of credits in soft core courses.)					
2		Big Data Analytics	SC	3:0:1	**4**
3		Blockchain Technology	SC	3:0:1	**4**

Note: In case of enrolling for an open elective course outside the department, one can enroll for maximum of 24 credits including Open Elective course in a Semester.

Name of Program	**Master of Computer Applications**
Semester No.	**I**
Course No.	**01**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Hard Core**
Course Title	**Object Oriented Programming with C++**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Comprehend the fundamental concepts and syntax of the C++ programming language.
CLO 2	Apply to implement and debug C++ programs using basic programming constructs such as variables, control structures, functions, and arrays.
CLO 3	Implement object-oriented programming principles in C++, including classes, objects, inheritance, polymorphism, and encapsulation.
CLO 4	Utilize advanced features of C++, such as templates, exception handling, and the Standard Template Library (STL).
CLO 5	Create efficient and robust C++ applications, and apply best practices for code organization, documentation, and testing.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures and interactive discussions
- Seminars and student presentations
- Assignments and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Introduction:** Concepts of Object-oriented programming, benefits of OOP, Applications of OOP, Structure of C++ program. Tokens, Keywords, Identifiers and constants, Data Types, Symbolic constants, Type compatibility, Declaration of variables, Dynamic initialization of variables, Reference variables, Operators in C++, Scope resolution operator, Member dereferencing operators, Memory management operators, Manipulators, Type cast operator, Expressions and their types, Special assignment expressions, Implicit conversions, Operator overloading.	**11:0:8**
Unit 2	**Functions:** Function prototyping, Call by Reference, Return by Reference, Inline functions, Default arguments, const arguments, Function overloading, Friend and Virtual functions. **Classes and Objects:** Definitions, defining member functions, Making an Outside function Inline, Nesting of member functions, Private member functions, Arrays within a Class, Static data members, Static member functions, Arrays of Objects, Objects as function arguments, friendly functions, Returning Objects, const member functions. **Constructors and Destructors**: Types of Constructors, Constructors with default arguments, Dynamic initialization of objects, Constructing Two-dimensional arrays, Destructors.	**11:0:8**
	Operator Overloading and Type Conversions: Defining operator overloading, overloading unary operators, Overloading Binary operators, Rules for overloading operators, Type conversions. **Inheritance:** Introduction, defining derived classes,	**11:0:8**

Unit 3	Types of Inheritance, virtual base classes, abstract classes, constructors in derived classes, **Polymorphism:** Introduction, pointers, pointers to objects, this pointer, pointers to derived classes, virtual functions, pure virtual functions.	
Unit 4	**Files and Templates:** C++ streams, C++ stream classes, Unformatted I/O operations, Formatted I/O operations, Console I/O Operations, managing output with manipulators. **Files:** Classes for file stream operations, opening and closing a file, detecting end of file, more about **open():** file modes, file pointers and their manipulations, sequential input and output operations. **Templates:** Function templates, Class templates Exceptions.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Object Oriented Programming with C++ (HC)	✔	✔	✔	—	—	—	✔	—

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Object Oriented Programming with C++ (HC)	✔	✔	—	—	—

Learning Resources:

1. Object Oriented Programming with C++ by E. Balagurusamy.
2. Object Oriented Programming in C++ by Robert Lafore Techmedia Publication.
3. The complete reference C– by Herbert shieldt Tata McGraw Hill Publication.

Name of Program	**Master of Computer Applications**
Semester No.	**I**
Course No.	**02**
Number of Credits	**4**
Credit Pattern	**2:1:1**
Course Category	**Hard Core**
Course Title	**Data Indexing Techniques**
Course Code	

Course Learning Outcomes:

After Completion of the course, the students are having

CLO 1	Thorough Understanding of the fundamental concepts and principles of data indexing.
CLO 2	Ability to identify suitable data indexing techniques for various applications.
CLO 3	Required skill to implement different data indexing structures.
CLO 4	Greater knowledge on evaluating the performance and efficiency of various indexing techniques.
CLO 5	Knowledge on exploring possibility for hybridization and ensembling of various indexing techniques to suit real time spatial applications.

Course Pedagogy:

The course shall be offered with following pedagogy:

- Learning through construction of techniques manually on a dataset and understand the working principles of the technique.
- Learning each topic staring from its design to analysis to implementation to execution towards picking up Problem Solving ability.
- Actively involved in exchanging of ideas and practice with the supervision of the instructor with personal meetings.
- Considering real life examples to explore applications of the indexing techniques.
- Learning through assimilation, aggregation and adaption of the knowledge required to solve a problem.

Units	Course Content	Teaching Hours L:T:P
Unit 1	Introduction, Problem solving, algorithms, Data Indexing, approaches, graph, tree, Binary tree and 2D array based, 1D array based and linked allocation based representation and traversal of binary tree. Heap (Max / Min) structure, operations and applications. Binary Search tree and insertion, deletions, searching operations and applications.	8:8:8
Unit 2	Height balanced binary search tree (AVL) and rotational operations such as LL, LR, RL and RR for balancing tree. Ternary tree, m-ary tree, Balanced Tree (B Tree / B+ tree) and operations. Multidimensional tree, space partition KD tree. Applications.	7:7:7
Unit 3	Hashing, concepts, components (hash functions and properties, hash tables), Type of hash functions. Notion of collision, resolution techniques: probing techniques, open and closed addressing methods, perfect hashing, dynamic hashing, hybrid / ensemble hashing / tree indexing and Applications.	8:8:8

Unit 4	Spatial Data Representation and Indexing (SDRI), stages in SDRI :converting spatial image into abstract / symbolic image, perceiving spatial topology, 2D strings, Limitations of 2D strings, 9DLT matrix and its list based representation, 9DLT + B tree indexing of iconic images, Limitations of 9DLT matrix, concept of Triangular Spatial Relationship (TSR), TSR rules, TSR keys and indexing using B tree and Applications.	**7:7:7**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Data Indexing Techniques(HC)	✓	✓	✓	✓	✓	-	-	✓

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Data Indexing Techniques(HC)	✓	✓	✓	-	✓

Learning Resources:

1. Classic Data Structures, Debasis Samanta, Prentice Hall of India (PHI) Publications.
2. An Introduction to Data Structures with Applications. Jean Paul Tremblay and Paul G Sorenson, McGraw Hill International.
3. Fundamentals of Data Structures, Ellis Horowitz and Sartaj Sahani, Galgotia Booksource.
4. Data Structures, Algorithms and Applications in C++, Sahni, Mc Graw Hill International edition.
5. Data Structures and Program Design, Robert L Kruse, Prentice Hall of India (PHI) Publications.
6. Current research articles and Ph.D thesis on spatial data indexing.

Name of Program	**Master of Computer Applications**
Semester No.	**I**
Course No.	**03**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Hard Core**
Course Title	**Data Clustering Algorithms**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Demonstrate a comprehensive understanding of data types, data representation techniques, and preprocessing methods including normalization for effective clustering analysis.
CLO 2	Analyze and compute similarity and dissimilarity measures using appropriate proximity metrics for different types of data such as binary, real-valued, interval, and multivalued data.
CLO 3	Apply hierarchical clustering techniques and graph-based approaches to identify cluster structures and compute inter-cluster proximities using linkage methods.
CLO 4	Implement and evaluate partition-based clustering algorithms such as K-means and K-medoids, and analyze their effectiveness in solving real-world clustering problems.
CLO 5	Assess the quality of clustering results using validation techniques and performance metrics such as accuracy, precision, recall, F-score, sensitivity, and specificity.

Course Pedagogy:

The course shall be delivered through:

- Learning through manual construction and step-by-step execution of clustering algorithms on datasets to understand their working principles.
- Understanding concepts through a systematic flow from data preprocessing to clustering, implementation, and evaluation for effective problem-solving.
- Analysing real-world applications of clustering techniques to understand their practical relevance and impact.
- Gaining hands-on experience using tools such as Python, WEKA, or MATLAB for implementing and visualizing clustering algorithms.

Units	Course Content	Teaching Hours L:T:P
Unit 1	Learning, types of learning, Notion of data clustering (unsupervised learning), problem solving, Unstructured data, structured data representation, Data types, taxonomy, qualitative v/s quantitative, crisp, interval valued, multi-valued etc., Data representation, pattern / data matrix, features, feature redundancy, data normalization techniques, generalized min-max normalization, z-score normalization.	**11:0:8**
Unit 2	Notion of Proximity and its computation, similarity and dissimilarity proximity, proximity matrix, Computation of proximities when features are binary type (hamming distance, jaccard's coefficient, dot product of vectors), when features are real valued(Minkowski general distance, City block / Manhattan distance, Euclidean and Chess board distance, Correlaton coefficient, cosine similarity), and when features are interval and Multivalued. Metric and Properties, Symmetric and Non symmetric measures on Interval and Multivalued features.	**11:0:8**
Unit 3	Notion of data clustering: Ambiguity in Data clusters, Taxonomy of data clustering, Hierarchical clustering (Agglomerative /	**11:0:8**

	divisive clustering approaches), Algebraic and Graph based representation, Computation of proximity between clusters of samples: single linkage, complete linkage, average linkage, representative linkage. Minimum spanning tree, Kruskal's and Prims algorithms, Computation of cluster representative and different methods.	
Unit 4	Wards clustering, Wards linkage in hierarchical clustering, Compactness of a cluster and measures, Partition clustering algorithms: K- means clustering algorithm, K-Medoids clustering algorithm. Cluster Validation: confusion matrix, accuracy, precision, recall, f-score, sensitivity, specificity.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Data Clustering Algorithms (HC)	✓	✓	✓	✓	✓	-	-	✓

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Data Clustering Algorithms (HC)	✓	✓	✓	-	✓

Learning Resources:

1. Algorithms for Clustering Data *(1988), authored by* Anil K. Jain and Richard C. Dube.
2. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
3. Richard O. Duda, Peter E. Hart, Pattern Classification, Wiley, 2000.

Name of Program	**Master of Computer Applications**
Semester No.	**I**
Course No.	**04**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Soft Core**
Course Title	**Discrete Mathematical Structure**
Course Code	

Course Learning Outcomes:

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

CLO 1	Understand and apply fundamental concepts in discrete mathematics, including sets, relations, functions, and logic.
CLO 2	Analyze and solve problems using principles from combinatorics and discrete probability.
CLO 3	Apply discrete mathematical structures to computer science and engineering applications, such as algorithms, data structures, and cryptography.
CLO 4	Evaluate probability distributions and discrete probabilistic models in computing scenarios.

Course Pedagogy:

The course shall be delivered through:

- Interactive classroom lectures and problem-solving sessions.
- Mathematical proof demonstrations.
- Case studies on discrete structures in computing.
- Student presentations on selected advanced topics.
- Assignments emphasizing analytical and problem-solving skills.
- Mini projects linking discrete mathematics concepts with algorithmic applications.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Set Theory:** Set basics, Venn diagrams, Union, intersection, set difference, complement, Cartesian product, Power sets, Cardinality of finite sets. **Relation:** Reflexivity, symmetry, antisymmetric, transitivity, Equivalence relations, compatibility relation, composite relation.	**11:0:8**
Unit 2	**Function:** Introduction, Domain and range of a function, surjection, injection, bijection, composition, inverse of a function, permutation function Basic **Logic:** Propositional logic, Logical connectives, Compound statements, Truth tables, Tautology, Contradiction, equivalence of statements, Normal forms, Disjunctive normal form, Validity of a well-formed formula, Propositional inference rules, Universal and existential quantifiers and their negations.	**11:0:8**
Unit 3	**Counting:** The basics of counting, the pigeonhole principle, permutations and combinations, recurrence relations, solving recurrence relations, generating functions, inclusion-exclusion principle and application of inclusion-exclusion.	**11:0:8**
Unit 4	**Discrete Probability:** Finite probability space, events, Properties of events, Conditional probability, Bayes‟ theorem, Independence. Statistical Distribution: Discrete Distribution, Binomial distribution, Gamma distribution, Beta distribution, Chi-square distribution, Univariate normal distribution.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Discrete Mathematical Structures (SC)	✓	✓	✓	✓	✓	-	✓	-

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Discrete Mathematical Structures(SC)	✔	-	✔	-	✔

Learning Resources:

1. Edgar Goodaire and Michael Parmenter, Discrete Mathematics with Graph Theory, Third Edition, PHI, ISBN-13-9750131679955.
2. S. Lipschutz, Discreate Mathematics, TMH, ISBN 0-07-066932-0.
3. Bernard Kolman C,Busby and Sharon Ross, Discrete Mathematical Structures, 2007, ISBN 81-203-2082-4, Publication PHI.

Name of Program	**Master of Computer Applications**
Semester No.	**II**
Course No.	**05**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Soft Core**
Course Title	**Digital Image Processing**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Apply the fundamental concepts and principles of digital image processing to solve real life problems.
CLO 2	Analyze and apply image enhancement techniques to improve the quality of digital images.
CLO 3	Evaluate and implement suitable segmentation algorithms to partition image into meaningful regions useful for further image analysis tasks
CLO 4	Analyze and implement morphological image processing techniques to address some image processing tasks.
CLO 5	Utilize feature extraction and pattern recognition techniques for object description, detection and recognition in images.
CLO 6	Develop, test, and execute image processing algorithms using software tools like MATLAB, Python, or OpenCV.

Course Pedagogy:

The course shall be delivered through:

- Pair theoretical lectures (e.g., Fourier Transform, Histogram Equalization, Image Segmentation) with immediate implementation lab sessions.
- Implement a semester-long project where students apply techniques to a real-world problem, such as medical imaging or satellite image analysis.
- Use industry-aligned applications (e.g., defect detection, object recognition, autonomous vehicles) to demonstrate the utility of algorithms.
- Leverage visualization tools to display intermediate image processing steps to enhance understanding.

Units	Course Content	Teaching Hours L:T:P
Unit 1	Introduction to Digital Images, Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Sampling and Quantization, Representing Digital Images, Some Basic Relationships Between Pixels- Neighbors and Connectivity of pixels in image, Applications of Image Processing: Medical imaging, Robot vision, Character recognition, Remote Sensing.	**11:0:8**
Unit 2	**Image Enhancement:** Spatial Domain Enhancement: Some Basic Gray Level Transformations, Histogram Processing, Enhancement using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods. Frequency Domain Enhancement: Introduction, Fourier Transform, Discrete Fourier Transform (DFT), properties of DFT, Discrete Cosine Transform (DCT), Image filtering in frequency domain.	**11:0:8**

Unit 3	**Image Segmentation and Morphological Image Processing:** Image Segmentation: Introduction, Detection of isolated points, Line detection, Edge detection, Edge linking, Region based segmentation-Region growing, split and merge technique, Hough transform, Segmentation using Threshold. Morphological Image Processing: Techniques for manipulating image shapes and structures using erosion, dilation, opening and closing, boundary extraction, hole filling, connected component extraction, thinning and thickening, and grayscale morphology.	**11:0:8**
Unit 4	**Color Image Processing, Image Description and Recognition:** Color Image Processing: Techniques for processing and manipulating color images, including color models and transformations. Representation, Description and Recognition of Objects: Chain codes, polygonal approximation approaches, signatures, boundary segments, boundary descriptors, regional descriptors, matching shape numbers and string matching.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Digital Image Processing (SC)	✔	✔	✔	✔	✔	—	—	—

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Digital Image Processing (SC)	✔	—	✔	—	✔

Learning Resources:

1. A.K. Jain, Fundamentals of Digital Image Processing, Prentice Hall of India, 1995.
2. Rafael C. Gonzalez, Richard E. Woods and Steven L. Eddins, Digital Image Processing using MATLAB, Pearson Education, 2nd Edition, 2009.
3. B. Chanda, D. DuttaMajumder, Digital Image Processing and Analysis, Prentice Hall of India, 2008.
4. William K Pratt, Digital Image Processing, John Willey, 4th edition, 2006.
5. Fundamentals of Digital Image Processing: A Practical Approach with Examples in Matlab by Chris Solomon.
6. Digital Image Processing: An Algorithmic Approach by Madhuri A. Joshi.

Name of Program	**Master of Computer Applications**
Semester No.	**II**
Course No.	**06**
Number of Credits	**4**
Credit Pattern	**2:1:1**
Course Category	**Soft Core**
Course Title	**Distributed Operating System**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Comprehend the fundamental concepts and principles of distributed operating systems.
CLO 2	Analyze and compare different architectures and models of distributed systems, including client- server, peer- to-peer, and cloud computing.
CLO 3	Implement and manage distributed processes, communication protocols, and synchronization.
CLO 4	Design and develop distributed file systems and resource management strategies.
CLO 5	Evaluate performance, fault tolerance, and security issues in distributed operating systems.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures and interactive discussions.
- Seminars and student presentations.
- Assignments and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Introduction to Distributed Systems:** Definition and characteristics of distributed systems, Types of distributed systems, Design goals and challenges, Examples of distributed operating systems. **Architectures of Distributed Systems:** Client-server model, Peer-to-peer systems, Middleware and its role in distributed systems, Case studies of distributed operating systems. **Communication in Distributed Systems:** Inter-process communication (IPC), Remote Procedure Calls (RPC), Message passing mechanisms, Socket programming.	**8:8:8**
Unit 2	**Synchronization and Coordination:** Clock synchronization, Logical clocks (Lamport's and vector clocks), Mutual exclusion algorithms, Election algorithms (Bully and Ring algorithms). **Distributed File Systems and Shared Memory:** Distributed file system design, Case studies: NFS, AFS, Google File System, Distributed shared memory, Consistency models and protocols. **Resource Management:** Distributed scheduling, Load balancing algorithms, Distributed deadlocks, Resource allocation and sharing.	**7:7:7**
Unit 3	**Fault Tolerance and Reliability:** Fault models and failure detection, Replication techniques, Checkpointing and rollback recovery, Consensus protocols (Paxos, Raft), **Security in Distributed Systems:** Security threats and challenges, Cryptographic techniques, Authentication and authorization, Secure communication protocols.	**7:7:7**

Unit 4	**Performance and Scalability:** Performance metrics and benchmarking, Scalability challenges, Techniques for improving performance, Case studies of scalable distributed systems, **Advanced Topics and Emerging Trends:** Cloud computinganddistributedoperatingsystems,InternetofThings(IoT)a ndedge computing, Block chain and distributed ledger technologies, Future directions in distributed systems	**8:8:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Distributed Operating System (SC)	✔	✔	—	✔	✔	✔	✔	✔

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Distributed Operating System (SC)	✔	✔	—	✔	—

Learning Resources:

1. Distributed Systems: Principles and Paradigms" by Andrew S. Tanenbaum and Maarten Van Steen.
2. "Distributed Operating Systems" by Andrew S. Tanenbaum.
3. "Distributed Systems: Concepts and Design" by George Coulouris, Jean Dollimore, and Tim Kindberg.

Name of Program	**Master of Computer Applications**
Semester No.	**I**
Course No.	**07**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Soft Core**
Course Title	**E-commerce and E-Governance**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Explain the fundamentals, models, and technologies of E-commerce and electronic business systems.
CLO 2	Analyze various electronic payment systems, security protocols, and safe transaction mechanisms in E-commerce.
CLO 3	Evaluate consumer-oriented and business-oriented E-commerce models and web-based business strategies.
CLO 4	Assess the role and implementation of E-Governance systems, ICT infrastructure, and public service delivery models.
CLO 5	Design and develop basic E-commerce or E-Governance solutions considering usability, security, and business requirements.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures and interactive discussions on E-commerce models and E-Governance frameworks.
- Case studies on digital business models and government initiatives.
- Demonstrations of electronic payment systems and security mechanisms.
- Group discussions on current trends in digital economy and governance.
- Seminars and student presentations on real-world applications.
- Practical exposure to web-based systems and portals.
- Assignments and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Introduction:** Electronic commerce environment and opportunities: Background, The electronic commerce environment, Electronic market place technologies. Modes of electronic commerce: Overview, Electronic data interchange (EDI), Migration to open EDI, E-commerce with WWW/Internet, Commerce Net advocacy, Web commerce going forward. Approaches to safe E- commerce: Overview, Secure Transport Protocols, Secure Transactions, Secure electronic payment protocol (SEPP), Secure electronic Transaction (SET), Certificates for authentication, Security on web commerce & Enterprise network. Payments and Security: Electronic cash and Electronic payment Schemes: Internet monetary payment and Security requirements, payment and purchase order process, On-line electronic cash. Master card/Visa secured electronic transaction: Introduction, Business Requirements, Concepts, Payment processing.	**11:0:8**
	Consumer-oriented e-commerce: Introduction, Traditional retailing and e-retailing, benefits of e-retailing, Key success factors, Models of e-retailing, features of e-retailing, developing a	**11:0:8**

Unit 2	consumer-oriented e-commerce system, The PASS model. Business-oriented e-commerce: Features of B2B e-commerce, Business models, Integration. Web advertising and web publishing: Traditional versus internet advertising, Internet techniques and strategies, Business models for advertising and their Revenues streams, pricing models and measurement of the effectiveness of Advertisements, web publishing- Goals and criteria, web side development Methodologies, logic design of the user interface.	
Unit 3	**E-Governance –** An introduction, scope, Types of E-Governance- Public, Corporate, Urban, Public-Private Partnership. Models & issues for effective E-Governance. E-Democracy, role of E-Governance, E-Republic, E-Business. The stages of E-Government development, E-Govt Privacy, Security & Accessibility. Mobile security issues for E-Govt. ICT & E-Governance – Role of ICT (Information & Communication Technology). ICT infrastructure, Implementation of ICT policy. CRM (Customer Relationship Model)- Defining CRM in the public-service area. Standards in E-Governance, India Portal-Mission mode project, India development Gateway (InDG).	**11:0:8**
Unit 4	**E-Government in India-** Introduction, Core policies, selection criteria, core infrastructure, support infrastructure, HRD/ Training-Technical assistance, awareness & assessment. National E-Governance Strategy, Implementation approach, governance structure. Draft policy Guidelines on website development-infrastructure, Applicability, Aims & Objectives, Content structure, website features, Administration, maintenance/updating, website promotion, technical aspects, security 7 secrecy of information, infrastructure & Training.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	E-Commerce and E-Governance (SC)	✓	✓	✓	✓	✓	✓	✓	✓

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	E-Commerce and E-Governance (SC)	✓	—	✓	—	✓

Learning Resources:

1. Ravi Kalakota, Andrew B. Frontiers of Electronic Commerce, Addison Wesley 1996.
2. Daniel Minoli, Emma Minoli: web commerce Technology Handbook. Tata McGraw Hill 1999.
3. Henry Chan, Raymond Lee, Tharam Dillon, Elizabeth Cheng: E-Commerce Fundamentals and applications, John Wiley & Sons, 2002.
4. E-Governance by - V.M. RAO
5. E-Governance by - PANKAJ SHARMA.

Name of Program	**Master of Computer Applications**
Semester No.	**II**
Course No.	**01**
Number of Credits	**4**
Credit Pattern	**2:1:1**
Course Category	**Hard Core**
Course Title	**Linux Programming**
Course Code	

Course Learning Outcomes:

At the end of the course, students will be able to:

CLO 1	Understand the fundamentals of Linux operating system architecture and command-line interface.
CLO 2	Learn system programming concepts on Linux.
CLO 3	Analyze Linux libraries and APIs.
CLO 4	Implement system level programming tasks in Linux, such as process management, file handling, and inter-process communication.
CLO 5	Write scripts and automate tasks using shell scripting and other scripting languages on Linux.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures with interactive discussions to students.
- Seminars and student presentations.
- Assignments, skills orientation and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Introduction to Linux Programming:** Overview of Linux operating system, Setting up a Linux development environment, Basic Linux commands and shell scripting, Introduction to GCC and Make file. **Linux System Programming Fundamentals:** System calls and library functions, File I/O operations, Error handling and debugging, Understanding man pages and documentation, File Handling and Filesystem, File descriptors and file control operations, Directory management, Permissions and file attributes, Filesystem hierarchy and navigation.	**8:8:8**
Unit 2	**Process Management:** Process creation and termination, Process states and scheduling, Zombie and orphan processes, Introduction to signals and signal handling. **Inter-Process Communication (IPC):** Pipes and FIFOs, Message queues, Semaphores, Shared memory. **Thread Programming:** Introduction to POSIX threads, Thread creation and management, Synchronization mechanisms (mutexes, condition variables), Thread safety and re-entrancy	**7:7:7**
Unit 3	**Network Programming - Part 1:** Basics of network programming, Sockets API, TCP/IP protocol suite, Creating client-server applications. **Network Programming - Part 2:** Advanced socket programming (non-blocking I/O, select, poll), Multicasting and broadcasting, Secure network programming (SSL/TLS), Debugging and performance tuning. **Advanced Linux Programming Techniques:** Memory management (mmap, malloc, free), Dynamic linking and shared libraries, Handling large files, Introduction to kernel programming.	**8:8:8**

Unit 4	**Debugging and Profiling:** Debugging with GDB, Memory leak detection (Valgrind), Performance profiling (gprof, perf), Analyzing and optimizing code. **Development Tools and Best Practices:** Version control with Git, Automated build systems (CMake, Autotools), Code quality and testing (unit tests, integration tests), Documentation and code review practices.	**7:7:7**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Linux Programming (HC)	✓	✓	✓	-	✓	✓	✓	-

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Linux Programming (HC)	✓	✓	✓	-	-

Learning Resources:

1. Advanced Programming in the UNIX Environment" by W.Richard Stevens, Stephen A. Rago.
2. "Linux System Programming" by Robert Love.
3. "The Linux Programming Interface" by Michael Kerrisk and Online documentation and man pages.

Name of Program	**Master of Computer Applications**
Semester No.	**II**
Course No.	**02**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Hard Core**
Course Title	**Advanced Web Technology**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Analyze advanced concepts and technologies in web development, including front-end and back-end frameworks.
CLO 2	Design and develop dynamic, responsive, and interactive web applications using modern web technologies such as HTML5, CSS3, JavaScript, and frameworks like React, Angular, or Vue.js.
CLO 3	Implement server-side programming using technologies like Node.js, Django, Ruby on Rails, or ASP.NET, and work with databases such as MySQL, MongoDB, or PostgreSQL.
CLO 4	Utilize web services and APIs, including RESTful and GraphQL, for integrating external services and data sources.
CLO 5	Apply best practices in web security, performance optimization, and scalable architecture design.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures and interactive discussions.
- Seminars and student presentations.
- Assignments and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Advanced Front-End Technologies**: Modern JavaScript Frameworks and Libraries, Advanced CSS Techniques, Single Page Applications (SPAs), Progressive Web Apps (PWAs)	**11:0:8**
Unit 2	**Back-End Development and APIs:** Node.js and Express.js, Database Integration, GraphQL, Web Security.	**11:0:8**
Unit 3	**DevOps and Deployment:** Containerization with Docker, Continuous Integration and Continuous Deployment (CI/CD) Cloud Services and Deployment, Monitoring and Logging.	**11:0:8**
Unit 4	**Advanced Topics and Emerging Trends:** Web Assembly, Integrating Web Assembly with JavaScript, Real-Time Web Applications, Artificial Intelligence and Machine Learning in Web Development, Ethical and Privacy Considerations, Best practices for ethical web development.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Advanced Web Technology (HC)	✓	✓	✓	✓	✓	✓	—	✓

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Advanced Web Technology (HC)	✓	✓	✓	—	—

Learning Resources:

1. "Web Engineering: The Discipline of Systematic Development of Web Applications" by Gerti Kappel, Birgit Pröll, Siegfried Reich, and Werner Retschitzegger, Wiley.
2. "Web Technologies: A Computer Science Perspective" by Jeffrey C. Jackson, Pearson.
3. "Advanced Web Technologies" by Chris Bates, Wiley.
4. "Professional JavaScript for Web Developers" by Nicholas C. Zakas, Wrox.
5. "HTML and CSS: Design and Build Websites" by Jon Duckett, Wiley.
6. "Web Development and Design Foundations with HTML5" by Terry Felke-Morris, Pearson.

Name of Program	**Master of Computer Applications**
Semester No.	**II**
Course No.	**03**
Number of Credits	**4**
Credit Pattern	**2:1:1**
Course Category	**Hard Core**
Course Title	**Learning Models**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Explain the theoretical foundations and principles of various learning models.
CLO 2	Discuss the different types of learning models, including supervised, unsupervised, semi-supervised, and reinforcement learning.
CLO 3	Implement and apply machine learning algorithms and techniques to solve real-world problems and datasets.
CLO 4	Evaluate and compare the performance of different learning models using appropriate metrics and validation methods.
CLO 5	Utilize tools and techniques to build, test, and deploy machine learning models on real-world datasets.

Course Pedagogy:

The course shall be delivered through:

- A project-based approach, where students build ML models from scratch, is crucial. This includes using Jupyter Notebooks to clean data, tune parameters, and evaluate models.
- Instruction should start with foundational knowledge in linear algebra, probability, and statistics, then connect this math to practical ML applications.
- Teaching the complete ML lifecycle—data preparation, model selection, training, evaluation, and deployment—is essential.
- Utilizing weekly mentoring sessions or discussion forums where instructors or industry professionals answer questions helps learners.
- Using tools like the TensorFlow Playground allows students to visualize neural network behaviours.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Overview of Learning:** Supervised, Unsupervised, Semi-supervised, Reinforcement Machine Learning, Applications of Machine Learning, Features, Datasets, Stages in Machine Learning, Error analysis, Notion of proximity measures, validation measures for learning models.	**8:8:8**
Unit 2	**Classification Techniques:** Nearest Neighbor Classifier, Decision Tree Classifier, Naïve Bayes Classifier, Support Vector Machine Classifier, Linear Regression and Logistic Regression. Performance Analysis, Applications of classification with Use Cases.	**7:7:7**
Unit 3	**Dimensionality Reduction Techniques**: Curse of dimensionality, dimensionality reduction, feature selection, filters and wrappers, Feature transformation techniques, Principal Component analysis, Linear Discriminant Analysis.	**7:7:7**

Unit 4	**Neural Networks Classifiers:** Introduction, Perceptron's, Multi-Layer Perceptron's, Backpropagation algorithm, Learning Error-Based Propagation in Neural Networks and Parameter Tuning.	**8:8:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Learning Models (HC)	✔	✔	✔	✔	✔	—	✔	—

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Learning Models (HC)	✔	—	✔	—	✔

Learning Resources:

1. Christopher M. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.
2. Richard O. Duda, Peter E. Hart, Pattern Classification, Wiley, 2000.
3. Yegnanarayana B., Artificial Neural Networks, Prentice-Hall of India Pvt.Ltd, 1999.
4. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, Springer, 2009.
5. Charu C. Aggarwal, Neural Networks and Deep Learning: A Textbook, Springer, 2018.
6. AurélienGéron, Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, O'Reilly Media, 2019 and Current Literature.

Name of Program	**Master of Computer Applications**
Semester No.	**II**
Course No.	**04**
Number of Credits	**4**
Credit Pattern	**2:1:1**
Course Category	**Soft Core**
Course Title	**Algorithms**
Course Code	

Course Learning Outcomes:

At the end of the course, students will be able to:

CLO 1	Explain fundamental concepts of algorithm design and computational problem solving.
CLO 2	Implement classical algorithms for searching, sorting, and graph processing.
CLO 3	Analyze time and space complexity of algorithms using asymptotic notations.
CLO 4	Apply algorithm design paradigms such as divide and conquer, greedy method, and dynamic programming.
CLO 5	Solve combinatorial and optimization problems using backtracking and branch-and-bound techniques.

Course Pedagogy:

The course shall be delivered through:

- Interactive lectures on algorithm design principles.
- Problem-solving sessions focusing on algorithm analysis.
- Demonstrations of algorithm implementation.
- Assignments on algorithm design and complexity analysis.
- Case studies involving real-world optimization problems.
- Mini projects on graph algorithms and optimization techniques.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Introduction to Algorithm**: Algorithm Specification, Analysis Framework, Performance Analysis: Space complexity, Time complexity. Asymptotic Notations: Mathematical analysis of non-recursive and recursive Algorithms with Examples. Important Problem Types: Sorting, Searching, String processing, Graph Problems, Combinatorial Problems. **Fundamental Data Structures**: Stacks, Queues, Graphs, Trees, Sets and Dictionaries.	**8:8:8**
Unit 2	**Divide and Conquer:** General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum, Merge sort, Quick sort, Strassen's matrix multiplication, Advantages and Disadvantages of divide and conquer, **Decrease and Conquer Approach**: Topological Sort.	**7:7:7**
Unit 3	**Greedy Method:** General method, Coin Change Problem, Knapsack Problem, Minimum cost spanning trees, Prim's Algorithm, Kruskal's Algorithm, Dijkstra's Algorithm, Optimal Tree problem: Huffman Trees and Codes. Transform and Conquer Approach: Heaps and Heap Sort. **Dynamic Programming:** General method with Examples, Multistage Graphs. **Transitive Closure:** Warshall's Algorithm, All Pairs Shortest Paths: Floyd's Algorithm, Optimal Binary Search Trees, Knapsack problem, Bellman-Ford Algorithm, Travelling Sales Person problem, Reliability design.	**8:8:8**

Unit 4	**Backtracking:** General method, N-Queens problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles. **Branch and Bound**: Assignment Problem, Travelling Sales Person problem, 0/1. **Knapsack problem:** LC Branch and Bound solution, FIFO Branch and Bound solution. NP-Complete and NP-Hard problems: Basic concepts, nondeterministic algorithms, P, NP, NP-Complete, and NP-Hard classes.	**7:7:7**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Algorithms (SC)	✔	✔	✔	✔	✔	✔	✔	✔

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Algorithms (SC)	✔	-	✔	-	-

Learning Resources:

1. Anany Levitin, Introduction to the Design and Analysis of Algorithms, 2rd Edition, 2009. Pearson.
2. Ellis Horowitz, Satraj Sahni and Rajasekaran, Computer Algorithms/C++, 2nd Edition, 2014, Universities Press.
3. Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, Introduction to Algorithms, 3rd Edition, PHI.
4. S. Sridhar, Design and Analysis of Algorithms, Oxford (Higher Education).

Name of Program	**Master of Computer Applications**
Semester No.	**II**
Course No.	**05**
Number of Credits	**4**
Credit Pattern	**2:1:1**
Course Category	**Soft Core**
Course Title	**Theory of Languages**
Course Code	

Course Learning Outcomes:

At the end of the course, students are having strong knowledge on:

CLO 1	Formal languages, alphabets, strings, and automaton.
CLO 2	Classification of different types of formal languages such as regular, context-free, and context-sensitive languages, and distinguishing their properties and limitations.
CLO 3	Design and development of computational models for solving language recognition problems.
CLO 4	Develop and apply formal grammars (regular and context-free grammars) to generate valid strings and analyze language structures.
CLO 5	Analyzing the computational power of different automata and languages and their properties. Formulating their own formal grammars and languages for an application.

Course Pedagogy:

The course shall be delivered through:

- Concepts of formal languages and automata are taught through structured lectures with examples, diagrams, and step-by-step problem-solving approach.
- Regular practice sessions are conducted to solve problems on automata construction, grammar design, and language classification with direct supervision.
- Students are given assignments and tutorial exercises to reinforce theoretical concepts and improve analytical skills with personal meetings.
- Real-world applications such as compiler design, pattern matching, and language processing are discussed to connect theory with practice.
- Periodic quizzes, tests, and feedback sessions are conducted to monitor student progress and strengthen understanding of key concepts.

Units	Course Content	Teaching Hours L:T:P
Unit 1	Introduction to theory of Languages, notions of alphabet, sentence, and language. Grammar, types of grammars and their corresponding automata, Examples of Grammars, Regular grammars and regular Languages, Deterministic and Non-deterministic finite automata (DFA/NFA), Regular grammar to FA and FA to regular grammar and algorithms, examples.	**8:8:8**
Unit 2	Regular Expressions, Regular Expressions and Non-deterministic finite automata, conversion of NFA to DFA, Direct construction of DFA for a regular expressions and Algorithms.	**7:7:7**
Unit 3	Finite automata with multiple outputs, Mealay and Moore machines Properties of Regular languages, Pumping lemma on regular languages, Context-Free Languages, Context free grammars(CFG) and simplifications, CNF of CFG.	**7:7:7**
Unit 4	Notion of Pushdown Automata, deterministic and non-deterministic PDA, PDA accepting by empting stack and by a final state, Properties of Context-Free Languages, Pumping Lemma on Context free languages, Context-Sensitive Grammars and complexity, Chomsky Hierarchy of languages, Notion of Turing Machine and it principles.	**8:8:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Theory of Languages (HC)	✔	✔	✔	✔	-	-	-	✔

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Theory of Languages (HC)	✔	✔	✔	-	✔

Learning Resources:

1. Introduction to Automata Theory, Languages, and Computation. John E. Hopcroft, Jeffery D Ullman and Rajeev, Pearson Edition, 2007.
2. An Introduction to Formal Languages and Automata, Peter Linz. Jones & Bartlett Learning publisher 2022 (Narosa)
3. Formal Languages and Their relation to automata. John E Hopecroft an Jeffery D Ullman, Addison Wesley Publishing company.
4. Introduction to languages and the theory of computation, John C Martin, Tata Mc Graw Hills.

Name of Program	**Master of Computer Applications**
Semester No.	**II**
Course No.	**06**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Soft Core**
Course Title	**Fundamentals of Software Engineering**
Course Code	

Course Learning Outcomes:

At the end of the course, students will be able to:

CLO 1	Describe the typical software development process activities.
CLO 2	Examine the feasibility of the software project to be under taken.
CLO 3	Decide an appropriate software process model to develop the intended software.
CLO 4	Guide the software development team or work in a team very efficiently and effectively.
CLO 5	Apply the knowledge to select coding platform, test environment, debugging process and software maintenance models.

Course Pedagogy:

The course shall be delivered through:

- Students work in teams to build software, simulating real-world project development, including requirements gathering, architecture design, and testing.
- Pre-recorded lectures and readings are reviewed before class, allowing class time to be used for coding labs, peer reviews, and active problem-solving.
- Using concrete examples to demonstrate software failures or best practices, such as analyzing requirement documentation (SRS) or system architecture.
- Practical exposure to industry-relevant tools, such as version control (e.g., Git), modeling software (UML), and testing frameworks.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Introduction:** The role of software engineering in system design, software products, emergence of software engineering, notable changes in software development practices, the changing nature of software, the software engineering challenges, Software processes, desired characteristics of software process, the software life cycle, software development process models, comparison of process models. Requirement analysis and specification need for SRS, characteristics of SRS, organization of SRS document. Techniques for representing complex logic, functional specification with Use Cases, formal system development techniques.	**11:0:8**
Unit 2	**Software Design Concepts:** Introduction to software design: characteristics of a good software design, module level concepts, and software design approaches. Function-oriented software design: Overview of the structured analysis and structured design methodology, data flow diagrams, structured design. Object-oriented software design concepts: Overview, UML, object- oriented design methodology, OOD	**11:0:8**

	metrics and goodness criteria. User-interface design: Characteristics, basic concepts, types of user interfaces, and a GUI design methodology.	
Unit 3	**Coding, Testing and Maintenance**: coding standards, guidelines, code walkthroughs, code inspections, software documentation, unit testing, black box testing, white box testing, integration testing, system testing, general issues associated with testing, debugging, approaches and guidelines, program analysis tools: Software reliability metrics, software reliability specification, software quality factors, quality metrics, software quality management system. The maintenance process, software reverse engineering, software maintenance process models, estimation of maintenance costs, CASE and its scope, CASE support in software life cycle, characteristics of CASE tools, architecture of CASE environment.	**11:0:8**
Unit 4	**Software Project Management**: responsibility of software project managers, project planning, structure of software project management document, Project size estimation metrics, project estimation techniques, project scheduling and staffing, Gantt charts, PERT charts, organization and team structures, attributes of a good software engineer, risk management and configuration management.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Fundamentals of Software Engineering (SC)	✔	✔	✔	✔	✔	✔	✔	✔

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Fundamentals of Software Engineering (SC)	✔	-	✔	-	✔

Learning Resources:

1. Ian Sommerville – Software Engineering, Fifth Edition, Addison-Wesley.
2. Pankaj Jalote – An Integrated Approach to Software Engineering, Third Edition.
3. Rajib Mall – Fundamentals of Software Engineering, PHI.
4. Roger S. Pressman – Software Engineering, Sixth Edition, Mc Graw Hill.
5. Ghezzi, Jazayeri, Mandrioli – Fundamentals of Software Engineering, PHI.

Name of Program	**Master of Computer Applications**
Semester No.	**II**
Course No.	**07**
Number of Credits	**4**
Credit Pattern	**2:1:1**
Course Category	**Soft Core**
Course Title	**.Net With C#**
Course Code	

Course Learning Outcomes:

At the end of the course, students will be able to:

CLO 1	Analyze the fundamentals of the C# programming language, including syntax, data types, control structures, and object-oriented programming principles.
CLO 2	Create console applications, graphical user interfaces (GUIs), and web applications using C# and the .NET framework.
CLO 3	Implement and utilize .NET features such as LINQ (Language Integrated Query), asynchronous programming, and parallel programming.
CLO 4	Design and develop database applications using ADO.NET or Entity Framework for data access.
CLO 5	Apply best practices for software development, including debugging, testing, and deploying C#/.NET applications.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures and interactive discussions.
- Seminars and student presentations.
- Assignments and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Review of OOP Concepts:** Overview of .NET Framework, Basic Elements of C#, Program Structure and simple Input and Output Operations, Operators and Expressions, Statements, Arrays and Structures.	**8:8:8**
Unit 2	**Inheritance:** Namespace, Polymorphism, Interface and Overloading, Multiple Inheritance, Property, Indexes, Delegates, Publish/Subscribe Design Patterns, Operator Overloading Method Overloading.	**7:7:7**
Unit 3	**C# Concepts for creating Data Structures:** File Operation, File Management systems, Stream Oriented Operations, Multitasking, Multithreading, Thread Operation, Synchronization.	**7:7:7**
Unit 4	**Working with XML:** Techniques for Reading and Writing XML Data, Using XPath and Search XML, ADO.NET Architecture, ADO.NET Connected and Disconnected Models, XML and ADO.NET, Simple and Complex Data Binding, Data Grid View Class.	**8:8:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	.NET with C# (SC)	✓	—	✓	—	—	—	—	—

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	.NET with C# (SC)	✓	✓	—	—	—

Learning Resources:

1. S. Thamarai Selvi and R. Murugesan "A Textbook on C# ", Pearson Education, 2003.
2. Stephen C. Perry "Core C# and .NET", Pearson Education, 2006.
3. Jesse Liberty, "Programming C#", Second Edition, O'Reilly Press,2002.
4. Robinson et al, "Professional C#", Fifth Edition, Wrox Press,2002.
5. Herbert Schildt, "The Complete Reference: C#", Tata McGraw Hill,2004.
6. Andrew Troelsen, "C# and the .NET Platform", AI Press,2003.

Name of Program	**Master of Computer Applications**
Semester No.	**III**
Course No.	**01**
Number of Credits	**4**
Credit Pattern	**2:1:1**
Course Category	**Hard Core**
Course Title	**Network Security**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Understand the fundamental concepts and principles of network security.
CLO 2	Recognize emerging trends and challenges in network security.
CLO 3	Apply cryptographic techniques and protocols to secure network communications and data.
CLO 4	Analyze network security requirements and design appropriate security architectures and policies.
CLO 5	Implement and configure security measures.
CLO 6	Evaluate and assess network security risks and vulnerabilities using penetration testing and vulnerability assessment techniques.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures with interactive discussions to students.
- Seminars and student presentations.
- Assignments, skills orientation and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit1	**Introduction:** computer security concepts, attacks, security services, security mechanisms; Classical encryption techniques-symmetric cipher models, substitution techniques, transposition techniques, rotor machines.	**8:8:8**
Unit2	**Symmetric ciphers:** Block cipher principles; DES Algorithm, strengths and weaknesses of DES, attacks on DES and defense, multiple encryptions; Asymmetric Ciphers Essential mathematics, public key cryptography.	**7:7:7**
Unit3	RSA, Diffie Hellman key exchange, random number generation, Data integrity and authentication Hash functions; MAC; Digital signatures.	**7:7:7**
Unit4	**Key management**: Authentication, Web and system security, Web security; IP security; E mail security; System security-intruders, malicious software, firewalls.	**8:8:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Network Security (HC)	✔	✔	✔	-	✔	✔	✔	-

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Network Security (HC)	✔	✔	✔	-	-

Learning Resources:

1. Cryptography and Network Security-Principles and Practice, William Stallings, PEARSON.
2. Cryptography and Network Security, Atul Kahate, Tata McGraw Hill.

Name of Program	**Master of Computer Applications**
Semester No.	**III**
Course No.	**02**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Hard Core**
Course Title	**Distributed Databases**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Understand the fundamental concepts and architecture of distributed databases.
CLO 2	Design and implement distributed database systems, including data distribution and fragmentation.
CLO 3	Apply concurrency control and recovery techniques to ensure data integrity and consistency in distributed environments
CLO 4	Evaluate the performance of distributed database systems and optimize queries for efficient data retrieval.
CLO 5	Address challenges related to distributed database security, including access control and data encryption.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures with interactive discussions to students.
- Seminars and student presentations.
- Assignments, skills orientation and reflective exercises.

Units	**Course Content**	**Teaching Hours L:T:P**
Unit1	**Introduction to Distributed Databases:** Fundamentals of Distributed Databases, Distributed Database Architecture, Distributed Data Independence, Distributed Database Design.	**11:0:8**
Unit2	**Distributed Database Models and Languages:** Distributed Database Models, Distributed Data Storage and Access, Distributed Database Languages, Distributed Transactions.	**11:0:8**
Unit3	**Distributed Database Management Systems (DDBMS):** Distributed DBMS Architecture, Data Replication and, Distributed Query Processing and Optimization, Distributed Data Security and Privacy.	**11:0:8**
Unit4	**Advanced Topics and Applications in Distributed Databases:** Big Data and Distributed Databases, Cloud-based Distributed Databases, NoSQL and NewSQL Databases, Future Trends and Research Directions.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	**Course Title (HC/SC only)**	**PLO1**	**PLO2**	**PLO3**	**PLO4**	**PLO5**	**PLO6**	**PLO7**	**PLO8**
01	Distributed Databases (HC)	✔	✔	✔	-	✔	✔	✔	-

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Distributed Databases (HC)	✔	✔	✔	-	-

Learning Resources:

1. "Principles of Distributed Database Systems" by M. Tamer Özsu and Patrick Valduriez, Springer.
2. "Distributed Database Management Systems: A Practical Approach" by Saeed K. Rahimi and Frank S. Haug, Wiley.
3. “Distributed Databases: Principles and Systems" by Stefano Ceri and Giuseppe Pelagatti, McGraw-Hill.
4. "Distributed and Cloud Computing: From Parallel Processing to the Internet of Things" by Kai Hwang, Jack Dongarra, and Geoffrey Fox, Morgan Kaufmann.
5. "Principles of Distributed Systems" by Vijay K. Garg, Springer.
6. "Advanced Database Systems" edited by Carlo Zaniolo, Stefano Ceri, Christos Faloutsos, Richard Snodgrass, V. S. Subrahmanian, and Roberto Zicari, Morgan Kaufmann.

Name of Program	**Master of Computer Applications**
Semester No.	**III**
Course No.	**03**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Soft Core**
Course Title	**Internet of Things**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Understand the fundamental concepts and components of the Internet of Things (IoT), including sensors, actuators, and IoT platforms.
CLO 2	Design and develop IoT systems and applications using microcontrollers, sensors, and communication protocols.
CLO 3	Implement IoT communication protocols such as MQTT, CoAP, and HTTP(S).
CLO 4	Analyze and interpret IoT data streams using cloud-based IoT platforms and edge computing techniques.
CLO 5	Evaluate security and privacy issues in IoT systems and implement appropriate measures for securing IoT deployments.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures and Practical lab session.
- Case studies and Project based learning.
- Programming exercises and assignments.

Units	Course Content	Teaching Hours L:T:P
Unit1	**Introduction to IoT:** Definition and Characteristics, Physical Design of IoT, Logical Design of IoT, IoT Enabling Technologies. **M2M and IoT:** Introduction to M2M, Difference between IoT and M2M, SDN and NFV for IoT. **IoT Protocols:** IEEE 802.15.4, BACNet Protocol, Modbus, KNX, Zigbee Architecture, 6LoWPAN, RPL.	**11:0:8**
Unit2	**Developing Internet of Things:** IoT Platforms Design Methodology, Python packages of Interest for IoT, IoT Physical Devices and Endpoints **IoT and Cloud**: IoT Physical Servers and Cloud Offerings, **IoT Tools:** Chef, Puppet.	**11:0:8**
Unit3	**Data Analytics for IoT:** Big Data Platforms for the IoT, Hadoop Map Reduce for Batch Data Analysis, Apache Oozie Workflows for IoT Data Analysis, In-Memory Analytics using Apache Spark, Apache Storm for Real Time Data Analysis, Sustainability Data and Analytics in Cloudbased M2M Systems, **Fog Computing**: A Platform for IoT and Analytics.	**11:0:8**
Unit4	**Domain Specific IoTs:** Home Automation, Cities, Environment, Energy, Retail, Logistics, Agriculture, Industry, Health and Lifestyle, VirtualReality Internet Advertising, Intelligent Transportation Systems, **Health Information System:** Genomics Driven Wellness Tracking and Management System (Go-WELL).	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	INTERNET OF THINGS (SC)	✓	✓	✓	✓	✓	✓	–	✓

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	INTERNET OF THINGS (SC)	✓	–	✓	✓	✓

Learning Resources:

1. Arshdeep Bahga, Vijay Madisetti, Internet of Things: A Hands-on Approach,2015,1st Edition, Universities Press.
2. Olivier Hersent, David Boswarthick, Omar Elloumi, The Internet of Things–Key applications and Protocols,2012, Wiley Publication.
3. Honbo Zhou, The Internet of Things in the Cloud: A Middleware Perspective,2012, CRC Press.
4. Dieter Uckelmann; Mark Harrison; Florian Michahelles Architecting the Internet of Things,2011, Springer.
5. Internet of Things: Architecture and Design Principles-Raj Kamal,1stEdition, McGraw Hill Education.
6. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things-David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry,1st Edition, Pearson Education.
7. Internet of Things-Srinivasa K G, CENGAGE Leaning India.

Name of Program	**Master of Computer Applications**
Semester No.	**III**
Course No.	**04**
Number of Credits	**4**
Credit Pattern	**2:0:2**
Course Category	**Soft Core**
Course Title	**Deep Learning**
Course Code	

Course Learning Outcomes:

At the end of the course, students will be able to:

CLO 1	Design and implement deep learning architectures using modern frameworks.
CLO 2	Apply CNNs, RNNs, LSTMs, Transformers, and generative models for domain-specific problems.
CLO 3	Evaluate and optimize deep learning models using advanced optimization and regularization techniques.
CLO 4	Assess performance, interpretability, scalability, and ethical implications of deep learning systems.
CLO 5	Develop AI-based solutions for real-world applications in computer vision, NLP, and reinforcement learning.

Course Pedagogy:

The course shall be delivered through:

- Concept-oriented lectures with mathematical derivations.
- Hands-on programming using TensorFlow / PyTorch.
- Case studies on real-world AI systems.
- Research paper presentations.
- Mini-project involving end-to-end model development.
- Industry dataset-based assignments.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Deep Feed forward Networks**: Gradient-Based Learning, Hidden Units, Architecture Design, Back Propagation. **Regularization:** Parameter Norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi Supervised Learning, Multi-Task Learning.	**8:0:15**
Unit 2	**Optimization for Training Deep Models**: How Learning Differs from Pure Optimization, Challenges in Neural Network Optimization, Basic Algorithms. Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates. **Convolutional Networks**: The Convolution Operation, Pooling, Convolution and Pooling as an Infinitely Strong Prior, Variants of the Basic Convolution Function, Efficient Convolution Algorithms, Random or Unsupervised Features.	**7:0:15**
Unit 3	**Sequence Modelling**: Recurrent and Recursive Nets, Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks. Long Short-Term Memory (LSTM).	**7:0:15**

Unit 4	**Introduction to Reinforcement Learning:** State of the art applications in Atari, Alpha Go, relation to other problems in artificial intelligence, **Markov Decision Processes (model based):** Formulation, Value Iteration (VI), Policy Iteration (PI), Linear Programming, and Approximate.	**8:0:15**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Deep Learning(SC)	✔	✔	✔	✔	✔	✔	✔	✔

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Deep Learning(SC)	✔	-	✔	✔	✔

Learning Resources:

1. Deep Learning, Lan Good fellow and Yoshua Bengio and Aaron Courville, MIT Presshttps://www.deeplearn ingbook.org/, 2016.
2. Richard S. Sutton and Andrew G. Barto, Introduction to Reinforcement Learning, 2nd Edition, MIT Press. 2017. ISBN-13 978-0262039246.
3. Dimitri Bertsekas and John G. Tsitsiklis, Neuro Dynamic Programming, Athena Scientific. 1996. ISBN-13: 978-1886529106.
4. Neural Networks, A systematic Introduction, Raúl Rojas, 1996.
5. Pattern Recognition and machine Learning, Chirstopher Bishop, Springer, 2007.
6. V. S. Borkar, Stochastic Approximation: A Dynamical Systems Viewpoint, Hindustan Book Agency, 2009. ISBN-13: 978-0521515924.
7. Deep Learning. Ian Goodfellow and Yoshua Bengio and Aaron Courville. MIT Press. 2016.ISBN-13: 978-0262035613.

Name of Program	**Master of Computer Applications**
Semester No.	**III**
Course No.	**05**
Number of Credits	**4**
Credit Pattern	**3:1:0**
Course Category	**Soft Core**
Course Title	**Cloud Computing**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Explain the fundamental concepts, architectures, and service models of cloud computing, including NIST architecture, deployment strategies, and resource provisioning mechanisms.
CLO 2	Analyze and apply virtualization technologies and cloud storage systems, including hypervisors, containers, distributed storage, and cloud database solutions.
CLO 3	Implement and evaluate parallel and distributed programming models such as MapReduce and Hadoop for large-scale data processing in cloud environments.
CLO 4	Assess cloud security challenges and design appropriate security mechanisms, including access control, risk management, and virtual machine security.
CLO 5	Design and deploy scalable cloud-based solutions using cloud networking concepts, DevOps practices, CI/CD pipelines, and Infrastructure as Code (IaC) tools.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures and interactive discussions on cloud architecture, models, and services.
- Hands-on laboratory sessions using cloud platforms.
- Demonstrations of virtualization, containerization.
- Case studies on real-world cloud deployments and industry practices.
- Mini-projects on cloud application development and deployment.
- Seminars and student presentations on emerging cloud technologies.
- Assignments and reflective exercises

Units	Course Content	Teaching Hours L:T:P
Unit1	**Introduction:** Cloud models-Evolution of Cloud Computing – System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture – On-demand Provisioning – Elasticity in Cloud – deployment models – service models-cloud service providers. **Cloud Infrastructure:** Architectural Design of Compute and Storage Clouds – Layered Cloud Architecture Development – Design Challenges - Inter Cloud Resource Management – Resource Provisioning and Platform Deployment – Global Exchange of Cloud Resources.	**11:8:0**
Unit2	**Virtualization and Storage Systems:** Advanced Virtualization Concepts, Types of Virtualization (Server, Network, Storage), Hypervisors (VMware, Hyper-V, KVM – Architectural View), Containers vs Virtual Machines, Edge and Fog Computing, Software Defined Infrastructure (SDN, SDS), Distributed Storage Systems, Object Storage (S3, Azure Blob, Google Cloud Storage), Cloud Databases: AWS RDS, Azure SQL, Google Cloud SQL, NoSQL Databases (DynamoDB, Azure Cosmos DB, Firestore.	**11:8:0**
	Programming Model: Parallel and Distributed Programming Paradigms – Map Reduce, Twister and Iterative Map Reduce – Hadoop Library from Apache – Mapping Applications -	**11:8:0**

Unit3	Programming Support. **Security in the Cloud:** Security Overview – Cloud Security Challenges – Access control mechanisms – Security Governance – Risk Management – Security Monitoring – Security Architecture Design – Virtual Machine Security.	
Unit4	**Cloud Networking, Security and DevOps:** Cloud Networking Basics, Virtual Private Cloud (VPC), Subnets and Routing, Load Balancing and Auto Scaling, Content Delivery Networks (CDN), Identity and Access Management (IAM), Cloud Security Models and Threat Mitigation, DevOps Principles and Practices, CI/CD Pipelines, Infrastructure as Code (IaC) using Terraform/CloudFormation, Monitoring and Logging Tools, Case Study and Mini Project on Cloud Deployment.	**11:8:0**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Cloud Computing (SC)	✓	✓	✓	✓	✓	✓	✓	✓

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Cloud Computing (SC)	✓	—	✓	—	✓

Learning Resources:

1. Rajkumar Buyya, Christian Vecchiola, S. Thamarai Selvi, Mastering Cloud Computing: Foundations and Applications Programming, Elsevier, 2nd Edition.
2. Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, Morgan Kaufmann, 2012.
3. Thomas Erl, Ricardo Puttini, Zaigham Mahmood, Cloud Computing: Concepts, Technology & Architecture, Pearson/Prentice Hall, 2013.
4. Gautam Shroff, Enterprise Cloud Computing: Technology, Architecture, Applications, Cambridge University Press, 2010.
5. George Reese, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, O'Reilly, 2009.
6. John W. Rittinghouse, James F. Ransome, Cloud Computing: Implementation, Management, and Security, CRC Press, 2010.

Name of Program	**Master of Computer Applications**
Semester No.	**III**
Course No.	**06**
Number of Credits	**4**
Credit Pattern	**2:1:1**
Course Category	**Soft Core**
Course Title	**Artificial Intelligence**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Analyze the foundational concepts and principles of artificial intelligence.
CLO 2	Apply AI algorithms and techniques, including search algorithms, knowledge representation, and reasoning.
CLO 3	Create AI systems capable of learning from data using machine learning algorithms such as supervised, unsupervised, and reinforcement learning.
CLO 4	Apply natural language processing (NLP) techniques for text analysis and generation.
CLO 5	Evaluate ethical and societal implications of AI technologies and apply responsible AI practices.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures with interactive discussions to students.
- Seminars and student presentations.
- Assignments, skills orientation and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit1	**Foundations and Problem-Solving in AI**: Introduction to AI: History of Artificial Intelligence, State of the Art Artificial Intelligence, Risks and Benefits of AI. Intelligent Agents: Agents and Environments, Nature of Environments, Structure of Agents. **Problem-Solving by Searching:** Problem-Solving Agents, Search Algorithms, Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Heuristic Functions.	**8:8:8**
Unit2	**Advanced Search Techniques and Game Playing**: Search in Complex Environments, Local Search and Optimization Problems, Local Search in Continuous Spaces, Search with Nondeterministic Actions. **Adversarial Search and Games:** Game Theory, Optimal Decisions in Games, Heuristic Alpha–Beta Tree Search and Limitations of Game Search Algorithms.	**7:7:7**
Unit3	**Knowledge Representation and Reasoning**: Logical Agents: Knowledge Based Agents, Propositional Logic, Agents Based on Propositional Logic, Predicate Logic, Knowledge Representation: Categories and Objects, Events, Reasoning Systems for Categories, Reasoning.	**7:7:7**
Unit4	**Planning, Uncertainty, and Learning:** Planning, Definition of Classical Planning, Algorithms for Classical Planning, Heuristics for Planning, Hierarchical Planning, Uncertain Knowledge and Reasoning: Bayes' Rule and Its Use, Naive Bayes Models, Probabilistic Reasoning. **Expert Systems and Learning**: Rote Learning, Learning by Advice, Learning by Analogy, Macro Learning.	**8:8:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Artificial Intelligence (SC)	✔	✔	—	✔	—	✔	—	✔

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Artificial Intelligence (SC)	✔	✔	✔	✔	✔

Learning Resources:

1. Russell, S., & Norvig, P. (2021). Artificial Intelligence: A Modern approach (4th ed.). Pearson.
2. Nilsson, N. J. (1998). Artificial Intelligence: A New Synthesis. Morgan Kaufmann.
3. Rich, E., & Knight, K. (1991). Artificial Intelligence. McGraw-Hill.
4. Luger, G. F. (2008). Artificial Intelligence: Structures and Strategies for Complex Problem Solving (6th ed.). Addison-Wesley.
5. Poole, D., &Mackworth, A. (2010). Artificial Intelligence: Foundations of Computational Agents. Cambridge University Press.
6. Russell, S., &Norvig, P. (2021). Artificial Intelligence: A Modern approach (4th ed.). Pearson.

Name of Program	**Master of Computer Applications**
Semester No.	**III**
Course No.	**07**
Number of Credits	**4**
Credit Pattern	**3:1:0**
Course Category	**Soft Core**
Course Title	**Probability and Statistics**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Understand the concepts in probability theory, including probability distributions, random variables, and mathematical expectation.
CLO 2	Evaluate statistical methods for data analysis, including descriptive statistics, hypothesis testing, and confidence intervals.
CLO 3	Analyze and interpret data using appropriate statistical techniques, such as regression analysis and ANOVA.
CLO 4	Apply probability and statistics to real-world problems in various domains, including business, engineering, and social sciences.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures with interactive discussions to students.
- Seminars and student presentations.
- Assignments, skills orientation and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit1	**Probability:** The concept of probability, the axioms and theorems, conditional probability, Independent Event's, Bayes Theorem. Random Variables and Probability Distributions: Random variables, discrete probability distributions and Distribution functions: Bernoulli, Binomial, Hyper Geometric, Geometric, Poisson, and Uniform.	**11:8:0**
Unit2	**Continuous Probability distribution:** Exponential, Normal, Uniform, Concepts of Chi squared, T, joint Distributions, Independent random variables, Functions of random Variables.	**11:8:0**
Unit3	**Mathematical Expectation:** Definition, Functions of Random variables. The variance and Standard Deviation, Moments, Moment Generating Functions, Covariance, Correlation Coefficient. Sampling Theory & Estimation: Population and sample, Sampling with and without replacement, the sample mean, sampling distribution of means, proportions, differences and sums. The sample variance, the sample distribution of variances, Point estimates, Interval estimates. Variance analysis.	**11:8:0**
Unit4	**Tests of Hypotheses and Significance:** Statistical Decisions, Statistical hypotheses, Null Hypotheses, Tests of hypotheses and significance, Type I and Type II errors, level of significance, Tests involving the Normal distribution, One-Tailed and Two-tailed, Special tests of Significance for large and small samples, The Chi square test for goodness of fit. Introduction to regression and curve fitting.	**11:8:0**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Probability and Statistics (SC)	✓	✓	-	✓	-	✓	✓	✓

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Probability and Statistics (SC)	✔	✔	-	-	✔

Learning Resources:

1. K S Trivedi, “Probability and Statistics with Reliability, Queuing and Computer Applications”, Prentics Hall of India, 2000.
2. Papoulis and S. UnnikrishnaPillai, “Probability, Random Variables and Stochastic Processes”, McGraw Hill, 4th Edition.
3. Richard A Johnson, Probability and Statistics for Engineers Prentice Hall, India 2002.

Name of Program	**Master of Computer Applications**
Semester No.	**IV**
Course No.	**01**
Number of Credits	**8**
Credit Pattern	**0:2:6**
Course Category	**Hard Core**
Course Title	**Project Work**
Course Code	

Course Learning Outcomes:

At the end of the course, students will be able to:

CLO 1	Identify and formulate real-world computing problems by applying domain knowledge and analytical skills.
CLO 2	Design and develop robust, scalable, and efficient software solutions using appropriate tools and technologies.
CLO 3	Apply software engineering principles, programming skills, and data management techniques in project implementation.
CLO 4	Demonstrate teamwork, professional ethics, and effective project management practices in a collaborative environment.
CLO 5	Evaluate, document, and communicate project outcomes through technical reports, presentations, and demonstrations.

Course Pedagogy:

The course shall be delivered through:

- Execution of real-time projects/internships with hands-on development.
- Regular monitoring, feedback, and evaluation by faculty/industry mentors.
- Team-based project work to enhance coordination and problem-solving skills.
- Independent exploration of modern tools, frameworks, and emerging technologies.
- Periodic evaluation through progress reviews, demonstrations, and final viva.

Units	Note	Teaching Hours L:T:P
	Candidate has to take up a problem to work for internship/ Project Work as stipulated in CBCS regulations of Mysore University.	**0:15:45**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Project Work (HC)	✓	✓	✓	✓	✓	✓	✓	✓

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Project Work (HC)	✓	✓	✓	✓	✓

Learning Resources:

1. Text Books (General Reference)
2. E-Learning Platforms / Online Resources
3. Development & Collaboration Tools
4. Research & Data Resources
5. Industry & Documentation Resources

Name of Program	**Master of Computer Applications**
Semester No.	**IV**
Course No.	**02**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Soft Core**
Course Title	**Big Data Analytics**
Course Code	

Course Learning Outcomes:

At the end of the course, students will be able to:

CLO 1	Understand the key issues in big data management and its associated applications in intelligent business and scientific computing.
CLO 2	Acquire fundamental enabling techniques and scalable algorithms like Hadoop, Map Reduce and NO SQL in big data analytics.
CLO 3	Interpret business models and scientific computing paradigms, and apply software tools for big data analytics.
CLO 4	Achieve adequate perspectives of big data analytics in various applications like recommender systems, social media applications etc.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures and Practical lab sessions.
- Case studies and Project based learning.
- Programming exercises and assignments.
- Analytical and programming skills.

Units	Course Content	Teaching Hours L:T:P
Unit 1	**Introduction to Big Data Analytics:** Big Data Overview, State of practice in analytics, Role of Data Scientists, Examples of Big Data Analytics, Data Analytics Lifecycle, Components of Hadoop, Analyzing Big data with Hadoop, Design of HDFS, developing a Map reduce Application.	**11:0:8**
Unit 2	**Map Reduce**: Distributed File System (DFS), Map Reduce, Algorithms using Map Reduce, Communication cost Model, Graph Model for Map Reduce Problem.	**11:0:8**
Unit 3	**Hadoop Environment:** Setting up a Hadoop Cluster, Hadoop Configuration, Security in Hadoop, Administering Hadoop, Hadoop Benchmarks, Hadoop in the cloud. **Big Data Analytics Methods using R**: Introduction to R-Attributes, R Graphical user interfaces, Data import and export, attribute and Data Types, Descriptive Statistics, Exploratory Data Analysis.	**11:0:8**
Unit 4	**Statistical methods for evaluation:** Hypothesis Testing, Difference of Means, Wilcoxon Rank-Sum Test, Type I and Type II errors, power and sample size, ANOVA. Advanced Analytics - technologies and tools: Analytics for unstructured data, The Hadoop ecosystem – pig – Hive- HBase- Mahout- NoSQL.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Big Data Analytics (SC)	✓	✓	✓	✓	✓	—	—	✓

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Big Data Analytics (SC)	✓	—	✓	—	✓

Learning Resources:

1. Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data by EMC Education Services, 2015, publishing.
2. Anand Raja Raman and Jeffrey David Ullman, Mining of Massive Datasets, 2012, Cambridge University Press.
3. Tom White, Hadoop: The Definitive Guide, 3rd Edition, O'Reilly Media.
4. Data mining and Data warehousing by Parteek Bhatia.

Name of Program	**Master of Computer Applications**
Semester No.	**IV**
Course No.	**03**
Number of Credits	**4**
Credit Pattern	**3:0:1**
Course Category	**Soft Core**
Course Title	**Blockchain Technology**
Course Code	

Course Learning Outcomes:

On successful completion of this course, the learners will be able to:

CLO 1	Develop a foundational understanding of blockchain technology, covering its principles, components, and decentralized nature.
CLO 2	Explore various blockchain types and their applications across industries, study consensus algorithms used for securing blockchain networks.
CLO 3	Analyze and compare different blockchain platforms and architectures.
CLO 4	Implement and develop smart contracts using blockchain programming languages.
CLO 5	Evaluate security, scalability, and regulatory aspects of blockchain implementations.

Course Pedagogy:

The course shall be delivered through:

- Classroom lectures with interactive discussions to students.
- Seminars and student presentations.
- Assignments, skills orientation and reflective exercises.

Units	Course Content	Teaching Hours L:T:P
Unit1	**Introduction to Blockchain Technology:** History and Evolution, Origins of Blockchain, Bitcoin and the rise of crypto currencies, Key milestones in blockchain development.	**11:0:8**
Unit2	**Blockchain Architecture:** Structure of a Blockchain, Blocks, transactions, and chains, Merkle trees, Distributed ledger technology, Consensus Algorithms, Proof of Work (PoW), Proof of Stake (PoS), Delegated Proof of Stake (DPoS), Practical Byzantine Fault Tolerance (PBFT).	**11:0:8**
Unit3	**Cryptography and Security:** Cryptographic Fundamentals, Public-key cryptography, Private keys and addresses, Digital signatures, Security in Blockchain, Double-spending problem, 51% attack, Security of smart contracts.	**11:0:8**
Unit4	**Blockchain Platforms:** Bitcoin, Bitcoin protocol and network, Bitcoin transactions and scripting, Wallets and mining, Ethereum: Ethereum Virtual Machine (EVM), Smart contracts and Solidity, Decentralized applications (DApps), Other Platforms, Hyper ledger Fabric, Ripple, EOS, Finance Smart Chain.	**11:0:8**

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

Sl. No.	Course Title (HC/SC only)	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
01	Blockchain Technology (SC)	✓	✓	✓	-	✓	✓	✓	-

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

Sl. No.	Course Title (HC/SC only)	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
01	Blockchain Technology (SC)	✔	✔	✔	-	-

Learning Resources:

1. “Mastering Bitcoin" by Andreas M. Antonopoulos, Published by Author's Republic, 2021.
2. "Mastering Ethereum" by Andreas M. Antonopoulos and Gavin Wood, 2018.
3. . "Blockchain Basics: A Non-Technical Introduction in 25 Steps" by Daniel Drescher, Apress, 2017.

PROGRAM LEARNING OUTCOMES (PLOs) – COURSE COMPLIANCE MATRIX

SI. No.	Course Title	PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8
1	Object Oriented Programming with C++ (HC)	✓	✓	✓	—	—	—	✓	—
2	Data Indexing Techniques (HC)	✓	✓	✓	✓	✓	—	—	✓
3	Data Clustering Algorithms (HC)	✓	✓	✓	✓	✓	—	—	✓
4	Linux Programming (HC)	✓	✓	✓	—	✓	✓	✓	—
5	Advanced Web Technology (HC)	✓	✓	✓	✓	✓	✓	—	✓
6	Learning Models (HC)	✓	✓	✓	✓	✓	—	✓	—
7	Network Security (HC)	✓	✓	✓	—	✓	✓	✓	—
8	Distributed Databases (HC)	✓	✓	✓	—	✓	✓	✓	—
9	Project Work (HC)	✓	✓	✓	✓	✓	✓	✓	✓
10	Discrete Mathematical Structure (SC)	✓	✓	✓	✓	✓	—	✓	—
11	Digital Image Processing (SC)	✓	✓	✓	✓	✓	—	—	—
12	Distributed Operating System (SC)	✓	✓	—	✓	✓	✓	✓	✓
13	E-Commerce and E-Governance (SC)	✓	✓	✓	✓	✓	✓	✓	✓
14	Algorithms (SC)	✓	✓	✓	✓	✓	✓	✓	✓
15	Theory of Languages (SC)	✓	✓	✓	✓	—	—	—	✓
16	Fundamentals of Software Engineering (SC)	✓	✓	✓	✓	✓	✓	✓	✓
17	.Net with C# (SC)	✓	—	✓	—	—	—	—	—
18	Internet of Things (SC)	✓	✓	✓	✓	✓	✓	—	✓
19	Deep Learning (SC)	✓	✓	✓	✓	✓	✓	✓	✓
20	Cloud Computing (SC)	✓	✓	✓	✓	✓	✓	✓	✓
21	Artificial Intelligence (SC)	✓	✓	—	✓	—	✓	—	✓
22	Probability and Statistics (SC)	✓	✓	—	✓	—	✓	✓	✓
23	Big Data Analytics (SC)	✓	✓	✓	✓	✓	—	—	✓
24	Blockchain Technology (SC)	✓	✓	✓	—	✓	✓	✓	—

SUSTAINABLE DEVELOPMENT GOALS (SDGs) – COURSE MAPPING

SI. No.	Course Title	SDG 1	SDG 2	SDG 3	SDG 4	SDG 5
1	Object Oriented Programming with C++ (HC)	✓	✓	—	—	—
2	Data Indexing Techniques (HC)	✓	✓	✓	—	✓
3	Data Clustering Algorithms (HC)	✓	✓	✓	—	✓
4	Linux Programming (HC)	✓	✓	✓	—	—
5	Advanced Web Technology (HC)	✓	✓	✓	—	—
6	Learning Models (HC)	✓	—	✓	—	✓
7	Network Security (HC)	✓	✓	✓	—	—
8	Distributed Databases (HC)	✓	✓	✓	—	—
9	Project Work (HC)	✓	✓	✓	✓	✓
10	Discrete Mathematical Structure (SC)	✓	—	✓	—	✓
11	Digital Image Processing (SC)	✓	—	✓	—	✓
12	Distributed Operating System (SC)	✓	✓	—	✓	—
13	E-Commerce and E-Governance (SC)	✓	—	✓	—	✓
14	Algorithms (SC)	✓	—	✓	—	—
15	Theory of Languages (HC)	✓	✓	✓	—	✓
16	Fundamentals of Software Engineering (SC)	✓	—	✓	—	✓
17	.Net with C# (SC)	✓	✓	—	—	—
18	Internet of Things (SC)	✓	—	✓	✓	✓
19	Deep Learning (SC)	✓	—	✓	✓	✓
20	Cloud Computing (SC)	✓	—	✓	—	✓
21	Artificial Intelligence (SC)	✓	✓	✓	✓	✓
22	Probability and Statistics (SC)	✓	✓	—	—	✓
23	Big Data Analytics (SC)	✓	—	✓	—	✓
24	Blockchain Technology (SC)	✓	✓	✓	—	—