Master of Science in
Data Science
Course Curriculum

Batch: 2023-2024

Academic Year: 2023-‘24

W.E.F. July 2023

GSFC UNIVERSITY

EDUCATION RE-ENVISIONED

GSFC University, Vigyan Bhavan, P. O.
Fertilizernagar, Vadodara - 391750, Gujarat, India

Teaching Scheme

Semester – I

Sr. No.	Course Code	Course Name	Teaching Scheme (Hours/week)				Teaching Credit				Evaluation Scheme					
			L	P	T	Total	L	P	T	Total	Theory : MS Marks	Theory : CEC Marks	Theory : ES Marks	Theory Marks	Practical Marks	Total Marks
1.	MSDS101	Data structures & Algorithms (DSA)	3	2	0	5	3	1	0	4	20	40	40	100	50	150
2.	MSDS102	Object Oriented Programming using C++ (OOP)	3	2	0	5	3	1	0	4	20	40	40	100	50	150
3.	MSDS103	Programming in "R" (RP)	3	2	0	5	3	1	0	4	20	40	40	100	50	150
4.	MSDS104	Mathematical Foundation for Data Science (MDS)	4	0	1	5	4	0	1	5	20	40	40	100	0	100
5.	MSDS105	Statistics for Data Science-I (SDS)	3	2	0	5	3	1	0	4	20	40	40	100	50	150
6.	MSDS106	Comprehensive VIVA	0	0	0	0	0	0	0	2	0	0	0	0	50	50
7.	SECC101	Internship#*	0	0	0	0	0	2	0	2	0	0	0	0	50	50
		Total	**16**	**8**	**1**	**25**	**16**	**6**	**1**	**25**	**100**	**200**	**200**	**500**	**300**	**800**

Note: L = Lecture, P = Practice, T= Tutorial, MS - Mid Semester, CEC - Continuous Evaluation Component, ES - End Semester

#As per NEP, minimum 60 hours of training has to be taken by the student at respective company/industry for fulfillment of two credits under internship component. This can be equivalent to 2-3 weeks depending on the rules/regulations of the company/industry.

*As per the academic policy of the GSFC University, students have to opt for the industrial internship after completion of their end semester examination.

COURSE CODE MSDS101	COURSE NAME Data Structures and Algorithms	SEMESTER I

Teaching Scheme (Hours)				Teaching Credit			
Lecture	Practical	Tutorial	Total Hours	Lecture	Practical	Tutorial	Total Credit
45	30	0	75	3	1	0	4

Course Prerequisites	Fundamental knowledge of information and data
Course Category	Major Course
Course focus	Employability/Skill development
Rationale	Data structures and algorithms play a crucial role in both local and global contexts. At a local level, efficient data structures and algorithms enable faster processing, retrieval, and storage of information, leading to improved performance of local systems, applications, and databases. Nationally, they contribute to the development of advanced infrastructure, optimizing resource allocation, and enabling the efficient functioning of critical sectors like healthcare, transportation, and finance. Internationally, data structures and algorithms drive global technological advancements, facilitating seamless communication, secure data exchange, and collaborative research. Their relevance lies in enabling innovation, scalability, and problem-solving, benefiting individuals, communities, and societies at various levels.
Course Revision/ Approval Date:	July 30, 2023
Course Objectives **(As per Blooms' Taxonomy)**	To enable the student to: 1. Get an **idea** about data and how it is stored in memory structure. 2. To make students **aware** about arrays and stacks used different programming languages.

	3. **Familiarize** with sorting and searching techniques. 4. **Understand** about tree and graph structures. 5. **Elaborate** testing approach with data structure.

Course Content (Theory)	Weightage	Contact hours
Unit 1: Introduction to Data Structures & Algorithms: Arrays and Strings, AlgorithmDevelopment, Complexity analysis, Recursion.	20%	9
Unit 2: Linear Data Structures: Stacks: Operations and Applications, Queues: Operations and Applications, Circular Queues: Operations and Applications, Links Lists: Operation – Creations, insertion, Deletion, Circular Lists, Doubly Linked.	20%	9
Unit 3: Sorting & Searching: Insertion Sort, Merge Sort, Quick Sort, Binary Search, Linear Search, Selection Sort	20%	9
Unit 4: Non Linear Data Structures: Graphs I: Representation and Traversal, Representation: Matrix, Adjacency list, Traversal: Depth First Search, Breadth First Search, Graphs II: Basic Algorithms, Minimum Spanning Tree, Shortest Path, All pairs Shortest Path, Transitive Closer, Binary Trees, Representation, Operations: Insert, Delete, Traversal: Preorder, In order, Postured, Heap Sort, Method and Complexity, Priority Queue, Search Trees, AVI-trees, B-tree, External Search.	25%	12
Unit 5: Hashing Techniques, String algorithms: Hashing Techniques, Pattern Matching, Text Editor	10%	6

List of Practical	Weightage	Contact hours
1: Introduction to dynamic memory allocation. DMA functions malloc(), calloc(), free() etc.	20%	4
2: Implement a program for stack that performs the following operation using an array. a) PUSH	20%	10

b) POP c) PEEP d) CHANGE e) DISPLAY Implement a program to convert infix notation to postfix notation using stack. Write a program to implement QUEUE using arrays that performs following operations: a)INSERT b) DELETE c) DISPLAY Write a program to implement Circular Queue using arrays that performs following operations: a) INSERT b) DELETE c) DISPLAY Write a menu driven program to implement following operations on singly linked list. a) Insert a node at the front of the linked list b) Insert a node at the end of the linked list. c) Insert a node such that the linked list is in ascending order. d) Delete the first node of a linked list. e) Delete a node before specified position. Write a program to implement stack using linked list		
3: Write a program to implement linear search Write a Program to implement Binary Search Write a program to implement, Bubble sort, Merge sort, Quick sort Write a program to create binary search tree Implement recursive and non recursive tree traversing methods inorder, preorder and post order.	20%	8
4: Solve various examples related to Graphs and Trees.	20%	4
5: Write a program to implement hashing using Linear Probe.	10%	2

Instructional Method and Pedagogy: (Max. 100 words)
Visual Aids and Demonstrations, Hands-On Approach, Active Learning Strategies, Real-World Examples, Project-Based Learning, Continuous Assessment.

Course Outcomes:	**Blooms' Taxonomy Domain**	**Blooms' Taxonomy Subdomain**
After successful completion of the above course, students will be able to:		
CO1: **Understand** and use the process of abstraction using a programming language such as' C++.	Cognitive	Understand
CO2: **Analyze** step by step and develop algorithms to solve real world problems.		Analyze
CO3:**Implement** various data structures viz. Stacks, Queues, Linked Lists, Trees and Graphs		Apply
CO4: **Understand** various searching & sorting techniques.		Understand
CO5:**Identify** the appropriate data structure to design efficient algorithms for the given application.		Analyze

Learning Resources	
1.	Reference Books: 1. ReemaThareja, Data Structures Using C, 2nd Edition 2. Horowitz, Sartaj Sahni, Fundamentals of Data Structures in C++, 2nd Edition 3. Yashwant Kanetkar, Data Structure Through C, 2nd Edition 4. Jean- Paul Tremblay & Paul Sorenson, An Introduction to Data Structures with Application, 2nd edition 5. NarasimhaKarumanchi, Data Structures and Algorithms Made Easy: Data Structures and Algorithmic Puzzles, 5th Edition

2.	Video Tutorial: https://www.youtube.com/playlist?list=PL2_aWCzGMAwI3W_JlcBbTwiQSsOTa6P
3.	Other Electronic Resources: https://nptel.ac.in/courses/106/102/106102064/

Evaluation Scheme	**Total Marks**
Theory: Mid semester Marks	20 marks
Theory: End Semester Marks	40 marks
Theory: Continuous Evaluation Component Marks (CEC)	Attendance: 05 marks MCQs: 10 marks Open Book Assignment: 15 marks Article Review: 10 marks **Total: 40 Marks**
Practical Marks	Attendance: 05 marks Practical Exam: 20 marks Viva: 10 marks Journal: 10 marks Discipline: 05 marks **Total: 50 Marks**

COURSE CODE MSDS102	COURSE NAME Object oriented Programming with C++	SEMESTER I

Teaching Scheme (Hours)				Teaching Credit			
Lecture	Practical	Tutorial	Total Hours	Lecture	Practical	Tutorial	Total Credit
45	30	0	75	3	1	0	4

Course Prerequisites	Fundamental knowledge of information and data
Course Category	Professional core courses
Course focus	Skill development
Rationale	OOP with C++ is locally relevant as it enhances code organization, reusability, and data security. Nationally, OOP is significant for the software industry, promoting scalability, code maintenance, and employment opportunities. Internationally, OOP in C++ enables collaboration, interoperability with diverse systems, and utilization of popular frameworks and libraries. OOP in C++ empowers developers at all levels to build modular, efficient, and robust software systems, contributing to local development projects, national software initiatives, and global software ecosystems.
Course Revision/ Approval Date:	July 30, 2023
Course Objectives **(As per Blooms' Taxonomy)**	To enable the student to: 1. Be **aware** about the basics of OOP for every O-O based programming language. 2. Be **familiar** with class and object with function. 3. **Understand** working and importance of constructor and destructor. 4. Get a brief **idea** about inheritance. 5. 5: Get an **overview** of file handling and templates.

Course Content (Theory)	Weightage	Contact hours
Unit 1: Principles of OOP: Programming Paradigms, Basic concepts, Benefits of OOP, Applications of OOP. Introduction to C++, History of C++, Structure of C++, Basic data types, Derived data types, Symbolic constants.Dynamic initialization, Type modifiers, Type Casting, Operator and control statements, Input and Output statements in C++.	20%	9
Unit 2: Classes and objects, class specification, member function specification, scope resolution operator, Access qualifiers, Instance creation, Member functions. Function prototyping, Function components, and Passing parameters, call by reference, Return by reference, Inline functions, Default arguments, Overloaded function.	20%	9
Unit 3: Array of objects, pointers to objects, this pointer, Dynamic allocation operators, Dynamic objects. Constructors, parameterized constructors, Overloaded constructors, Constructors with default arguments, copy constructors, static class members and static objects. Operator Overloading, Overloading unary and binary operator, Overloading the operator using friend function, stream operator overloading, data conversion.	20%	9
Unit 4: Inheritance, Defining derived classes, Single inheritance, protected data with private inheritance, multiple inheritance, multi-level inheritance, hierarchical inheritance, hybrid inheritance, multipath inheritance, Constructors in derived and base class, Abstract classes, virtual function and dynamic polymorphism, virtual destructor. Exception Handling, principle of Exception handling, Exception handling mechanism, multiple catch, Nested try, Rethrowing the exception.	20%	9
Unit 5: Streams in C++, Stream classes, Formatted and Unformatted data, manipulators User defined manipulators, file streams, file pointer manipulation, file open and close.	20%	9

Templates, Template functions and Template classes.		

List of Practical	Weightage	Contact hours
1: Write a program to demonstrate simple C++ structure with help of cout and cin. Write a program to demonstrate cascading I/O. Explain the use of name space in C++ with proper examples. Write a different program to demonstrate control statements available in C++. Write a different program to demonstrate loops available in C++.	20%	4
2: Write a program to demonstrate scope resolution operators and reference variables in C++. Write a program to demonstrate implicit type casting and explicit type casting in C++. Write a program to demonstrate const keyword and #define. Write a program to demonstrate different types of user defined function and Function call Write a program to demonstrate class and object creation. Define a member function inside the class and outside the class. Write a program to show the working of different access specifiers . What do you understand about the Inline function? How can you create an Inline function ?	20%	6
3: Why do an array of objects require? Demonstrate an array of objects with proper examples. Write a program to demonstrate concepts and different types of constructor. Write difference between Constructor overloading and overriding. Also apply the concept with proper examples. Write a program to demonstrate friend function. Write a program to demonstrate the concept of copy constructor and static class member.	20%	6
4: Write a program to demonstrate different type of inheritances Write a program to demonstrate the concept of	20%	8

polymorphism and exception handling Explain the behavior of the constructor in derived class using examples.		
5: Perform following operation of file management • Count characters & spaces • Append to a file • Copy contents & change case • Merge two files • Count characters, words & lines • Arrange records in descending order • Add & read contents of file • Create file to store employee details • Display content of file	20%	4

Instructional Method and Pedagogy: (Max. 100 words)

Visual Aids and Demonstrations, Hands-On Approach, Active Learning Strategies, Real-World Examples, Project-Based Learning, Continuous Assessment

Course Outcomes:	**Blooms' Taxonomy Domain**	**Blooms' Taxonomy Subdomain**
After successful completion of the above course, students will be able to:		
CO1: **Understand** object-oriented programming features in C++.	Cognitive	Understand
CO2: **Implement** computer programs to solve real world problems based on object-oriented principles.		Apply
CO3: **Understand** the concept of Array, pointers and Polymorphism.		Understand
CO4: **Analyze** concept of inheritance and exception handling.		Analyze
CO5: **Develop** the applications using object oriented programming with C++.		Create

Learning Resources	
1.	Reference Books:

	1. Barbara E. Moo, JoséeLajoie, Stanley B.Lippman , C++ Primer, 5th Edition. 2. Tony Gaddis, Starting Out with C++ - From Control Structures through Objects. 3. Andrew Koenig Accelerated C++: Practical Programming by Example, 1st Edition 4. E Balagurusamy, Object-Oriented Programming with C++, 7th edition 5. YashwantKanetkar, Let US C++, 2nd Edition
2.	Video Tutorial https://www.studytonight.com/courses/cpp-video-tutorial/
3.	NPTEL MOOC https://nptel.ac.in/courses/106/101/106101208/

Evaluation Scheme	**Total Marks**
Theory: Mid semester Marks	20 marks
Theory: End Semester Marks	40 marks
Theory: Continuous Evaluation Component Marks (CEC)	

Attendance	05 marks
MCQs	10 marks
Open Book Assignment	15 marks
Article Review	10 marks
Total	**40 Marks**

Practical Marks	

Attendance	05 marks
Practical Exam	20 marks
Viva	10 marks
Journal	10 marks
Discipline	05 marks
Total	**50 Marks**

COURSE CODE MSDS103	COURSE NAME R Programming	SEMESTER I

Teaching Scheme (Hours)				Teaching Credit			
Lecture	**Practical**	**Tutorial**	**Total Hours**	**Lecture**	**Practical**	**Tutorial**	**Total Credit**
45	30	0	75	3	1	0	4

Course Pre-requisites	Fundamental Knowledge of any programming language + statistics
Course Category	Major Course
Course focus	Skill Enhancement
Rationale	R Programming has been widely used for assessing diverse data. This being the skill enhancement subject will help students to learn and implement different statistical theories/notions on computers for studying/analysing/assessing large datasets.
Course Revision/ Approval Date:	July 30, 2023
Course Objectives **(As per Blooms' Taxonomy)**	To enable the student to: 1: **Gain** insight to the fundamental notions of R Programming. 2.

Course Content (Theory)	Weightage	Contact hours
Unit 1: Introduction to R Programming Introduction to R programming History and overview of R, Install and configuration of R programming environment, Installation and demonstration of R-Studio, R Command Prompt, R Script File, Basic language elements and data structures, R - Data Types, Vectors, Lists, Matrices, Arrays, Factors, Data Frames, Variable Assignment, Data Type of a Variable, Finding and deleting Variables, Operators, Reading and writing in R	10%	9
Unit 2: Data Preparation R Data Frame: Create, Append, Select, Subset, List in R: Create, Select Elements with Example, R Sort a Data Frame using Order(), R Dplyr, Tutorial: Data Manipulation(Join) & Cleaning(Spread), Merge Data Frames in R: Full and Partial Match, In-built Functions in R Programming (with Example)	25%	14
Unit 3: Programming in R IF, ELSE, ELSE IF Statement in R, Switch Statement, if else ladder, ifelse function, For Loop in R with Examples for List and Matrix, While Loop in R with Example, Repeat loop, Break and Next Statement, apply(), lapply(), sapply(), tapply() Function in R with Examples.	25%	8
Unit 4: Data Analysis and Visualization in R Import Data into R: Read CSV, Excel Files, How to Replace Missing Values(NA) in R: na.omit & na.rm, R Exporting Data to Excel, CSV, Text File, R Aggregate Function: Summarize & Group by() Example, R Select(), Filter(), Arrange(), Pipeline with Example Scatter Plot in R using ggplot2 , How to make Box plot in R , Bar Chart & Histogram in R, Line graph, Scatter plot, T Test in R: One Sample and Paired , R ANOVA Tutorial: One way & Two way (with Examples).	20%	9
Unit 5: Advance R Concepts Data querying: SQL and R, Writing functions, Reporting, Interactive reporting with Rmarkdown	20%	10

List of Tutorials	Weightage	Contact hours
Unit 1: Problem solving Examples on Graph Theory.	**20%**	**3**
Unit 2: Problem solving on Vector Spaces.	**20%**	**3**

Unit 3: Problem solving on Linear Transformations.	**20%**	**3**
Unit 4: Problem solving on Set Theory.	**20%**	**3**
Unit 5: Problem solving on Propositional Logic and recursive relations.	**20%**	**3**

Instructional Method and Pedagogy: Chalk-Talk, Classroom Discussions, Notes, Use of GeoGebra Toolbox.

Course Outcomes:	**Blooms' Taxonomy Domain**	**Blooms' Taxonomy Subdomain**
After successful completion of the above course, students will be able to: CO1: **Define, classify, explain** the fundamental notions related to graph theory and **solve** relevant problems in the field.	Remember, Understand & Apply	Define, Classify, Draw, Explain & Solve
CO2: **Define, Demonstrate** & **apply** the linear algebra in real-life problems/diverse fields.	Remember, Understand & Apply	Define, Demonstrate & Apply
CO3: **Use** recursion and recurrence relations for **solving** real-life problems.	Apply	Use, Demonstrate & Solve
CO4: **Define** & **Demonstrate** the concepts related to set theory and **apply** the concepts for solving relevant problems.	Understand, Evaluate	Define, Demonstrate & Apply
CO5: **Apply** propositional calculus for **solving** real-life problems.	Apply	Apply & Solve

Learning Resources	
1.	Reference Books: 1. David C. Lay, Steven R. Lay and Judi Donald, Linear Algebra and its applications, Pearson, 5th ed., 2015. 2. Kenneth H. Rosen, Discrete Mathematics and Its Applications, Tata McGraw Hill, 6th edition

	3. Narsingh Deo, Graph Theory and its Applications, PHI learning Private Ltd, 2016 4. Gilbert Strang, Linear Algebra and its applications, 4th ed., Cengage Learning, 2005 5. Jean Paul Trembley, R Manohar, "Discrete Mathematical Structures with Application to Computer Science", Tata McGraw Hill, 1997 6. Johnson Bough R., "Discrete Mathematics", 5th Edition, PEA, 2001. 7. Ronald Graham, Donald Knuth and Oren Patashik, "Concrete Mathematics: A Foundation for Computer Science", Addison- Wesley, 1989.
2.	Other Electronic Resources: GeoGebra Toolbox: https://www.geogebra.org/

Evaluation Scheme	**Total Marks**
Theory: Mid semester Marks	20 marks
Theory: End Semester Marks	40 marks
Theory: Continuous Evaluation Component Marks (CEC)	Attendance: 05 marks MCQs: 10 marks Open Book Assignment: 15 marks Article Review: 10 marks **Total: 40 Marks**
Practical Marks	Attendance: 05 marks Practical Exam: 20 marks Viva: 10 marks Journal: 10 marks Discipline: 05 marks **Total: 50 Marks**

COURSE CODE MSDS104	COURSE NAME Mathematical Foundation for Data Science	SEMESTER I

Teaching Scheme (Hours)				Teaching Credit			
Lecture	Practical	Tutorial	Total Hours	Lecture	Practical	Tutorial	Total Credit
60	0	15	75	4	0	1	5

Course Pre-requisites	Fundamental Knowledge of Mathematics
Course Category	Major Course
Course focus	Employability/Skill Development
Rationale	This course involves the basics of graph theory and its applications. We also study the basics of vector spaces and linear algebra and its applications. Students will study set theory, propositional logic and Recursive relations.
Course Revision/ Approval Date:	July 30, 2023
Course Objectives **(As per Blooms' Taxonomy)**	To enable the student to: 1: **Understand, remember & apply** the fundamental notions and applications of graph theory. 2: **Understand**, **remember** & **apply** the linear algebra concepts in diverse fields.

	3: **Understand & apply** recursion and recurrence relations in real-life problems. 4: **Remember, understand**, **apply** & **analyse** concepts related to set theory. 5: **Understand & apply** propositional calculus to real-life situations.

Course Content (Theory)	Weightage	Contact hours
Unit 1: Graphs And Trees Introduction to graphs, Directed and Undirected graphs, Isomorphic graphs, Subgraphs, Cut points and Bridges, Multigraph and Weighted graph, Paths and circuits, Shortest path in weighted graphs, Eurelian path and circuits, Hamilton paths and circuits, Planar graphs, Euler's formula, Trees, Spanning trees, Binary trees	**20%**	**12**
Unit 2: Linear Algebra, Vector spaces, subspaces, Linear Dependence, Span, Basis, Dimension; Definitions of Norms, Inner Products and metrics. Orthogonality	**20%**	**12**
Unit 3: Representation of linear transformations as a matrix, Similarity transformation Rank-Nullity theorem, Fundamental Theorem of Linear Algebra	**20%**	**12**
Unit 4: Introduction to set theory, Set operations, Algebra of sets, Duality, Finite and Infinite sets, Classes of sets, Power Sets, Multi sets, Cartesian Product, Representation of relations, Types of relation, Equivalence relations and partitions , Partial ordering relations and lattices, Function and its types, Composition of function and relations, Cardinality and inverse relations.	**20%**	**12**
Unit 5: Basic operations: AND (^), OR(v), NOT (~), Truth value of a compound statement, propositions, tautologies, contradictions Recursion and Recurrence Relation: linear recurrence relation with constant coefficients, Homogeneous solutions, Particular solutions.	**20%**	**12**

List of Tutorials	Weightage	Contact hours
Unit 1: Problem solving Examples on Graph Theory.	**20%**	**3**
Unit 2: Problem solving on Vector Spaces.	**20%**	**3**
Unit 3: Problem solving on Linear Transformations.	**20%**	**3**
Unit 4: Problem solving on Set Theory.	**20%**	**3**
Unit 5: Problem solving on Propositional Logic and recursive relations.	**20%**	**3**

Instructional Method and Pedagogy: Chalk-Talk, Classroom Discussions, Notes, Use of GeoGebra Toolbox.

Course Outcomes:	Blooms' Taxonomy Domain	Blooms' Taxonomy Subdomain
After successful completion of the above course, students will be able to:		
CO1: **Define, classify, explain** the fundamental notions related to graph theory and **solve** relevant problems in the field.	Remember, Understand & Apply	Define, Classify, Draw, Explain & Solve
CO2: **Define, Demonstrate** & **apply** the linear algebra in real-life problems/diverse fields.	Remember, Understand & Apply	Define, Demonstrate & Apply
CO3: **Use** recursion and recurrence relations for **solving** real-life problems.	Apply	Use, Demonstrate & Solve
CO4: **Define** & **Demonstrate** the concepts related to set theory and **apply** the concepts for solving relevant problems.	Understand, Evaluate	Define, Demonstrate & Apply
CO5: **Apply** propositional calculus for **solving** real-life problems.	Apply	Apply & Solve

Learning Resources

1.	Reference Books: 1. David C. Lay, Steven R. Lay and Judi Donald, Linear Algebra and its applications, Pearson, 5th ed., 2015. 2. Kenneth H. Rosen, Discrete Mathematics and Its Applications, Tata McGraw Hill, 6th edition 3. Narsingh Deo, Graph Theory and its Applications, PHI learning Private Ltd, 2016 4. Gilbert Strang, Linear Algebra and its applications, 4th ed., Cengage Learning, 2005 5. Jean Paul Trembley, R Manohar, "Discrete Mathematical Structures with Application to Computer Science", Tata McGraw Hill, 1997 6. Johnson Bough R., "Discrete Mathematics", 5th Edition, PEA, 2001. 7. Ronald Graham, Donald Knuth and Oren Patashik, "Concrete Mathematics: A Foundation for Computer Science", Addison- Wesley, 1989.
2.	Other Electronic Resources: GeoGebra Toolbox: https://www.geogebra.org/

Evaluation Scheme	**Total Marks**
Theory: Mid semester Marks	20 marks
Theory: End Semester Marks	40 marks
Theory: Continuous Evaluation Component Marks (CEC)	

Attendance	05 marks
MCQs	10 marks
Open Book Assignment	15 marks
Article Review	10 marks
Total	**40 Marks**

Master of Science in
Data Science
Course Curriculum

Batch: 2023-2024

Academic Year: 2023-‘24

W.E.F. July 2023

GSFC University, Vigyan Bhavan, P. O.
Fertilizernagar, Vadodara - 391750, Gujarat, India

Teaching Scheme

Semester – II

Sr. No.	Course Code	Course Name	Teaching Scheme (Hours/week)				Teaching Credit				Evaluation Scheme					
			L	P	T	Total	L	P	T	Total	Theory : MS Marks	Theory : CEC Marks	Theory : ES Marks	Theory Marks	Practical Marks	Total Marks
1.	MSDS201	Programming using Python	3	2	0	5	3	1	0	4	20	40	40	100	50	150
2.	MSDS202	Database Management System	3	2	0	5	3	1	0	4	20	40	40	100	50	150
3.	MSDS203	Fundamentals of AI & ML	3	2	0	5	3	1	0	4	20	40	40	100	50	150
4.	MSDS204	Numerical Methods & Analysis using MATLAB	3	2	0	5	3	1	0	4	20	40	40	100	50	150
5.	MSDS205	Statistics - II	3	2	0	5	3	1	0	4	20	40	40	100	50	150
6.	AECC203	Working with Spreadsheets	1	2	0	3	1	1	0	2	20	40	40	100	50	150
7.	SECC201	Internship	0	0	0	2	0	0	0	2	0	0	0	0	50	50
8.	SECC202	Comprehensive VIVA	0	0	0	2	0	0	0	2	0	0	0	0	50	50
		Total	**16**	**12**	**0**	**32**	**16**	**6**	**0**	**26**	**120**	**240**	**240**	**600**	**400**	**1000**

Note: L = Lecture, P = Practice, T= Tutorial, MS - Mid Semester, CEC - Continuous Evaluation Component, ES - End Semester

Program Coordinator **Associate Dean**

COURSE CODE MSDS201	COURSE NAME Programming using Python	SEMESTER II

Teaching Scheme (Hours)				Teaching Credit			
Lecture	**Practical**	**Tutorial**	**Total Hours**	**Lecture**	**Practical**	**Tutorial**	**Total Credit**
45	30	0	75	3	1	0	4

Course Prerequisites	Fundamental knowledge of any basic programming language.
Course Category	Major Course
Course focus	Employability
Rationale	The present course will equip students with the skills related to python programming and its applications. Detailed knowledge of libraries and their utility will help students to apply the gained knowledge for performing solution oriented tasks related to data analysis, data cleaning, and, more rigorous tasks like application development can also be done. Small project based learning can equip students with utmost important skill-sets.
Course Revision/ Approval Date:	
Course Objectives **(As per Blooms' Taxonomy)**	To enable the student to: 1: **Understand** the fundamental components of the programming process. 2: **Inculcate** the knowledge related to the control structures of procedural programming languages. 3: **Provide** in-depth knowledge about List, Tuple & Dictionaries. 4: Be **aware** of the utility of python programming and basic functions. 5: Be **familiar** with different libraries and their **utility** in advanced technologies.

Course Content (Theory)	Weightage	Contact hours
Unit 1: Introduction & Control Statements Installation and Working with Python, Essentials of a Python program, Program Structure of Python, Basic Syntax, Variables and Identifiers, Built- In Data Types. Variable definition, Operators And Expressions, Constants And Literals, Basic Input/output Statement. Control Loops and Array Decision Making - Conditions, Relational, Operators, Logical Connectives, If-Else , Nested If-Else Statement, Replacement of switch case Loops: While loop, Do-while, For loop, Nested loops , Break, Continue and pass Statement	20%	9
Unit 2: Python Data Types Arrays – One dimensional and multidimensional array, Array processing. String Manipulation - accessing Strings, Basic Operations, String slices and Function and Methods, Lists: Introduction, accessing list, Operations, Working with lists, Function and Methods. Tuple: Introduction, accessing tuples, Operations,. Dictionaries: Introduction, Accessing values in dictionaries, Working with dictionaries, Properties	20%	9
Unit 3: Functions in python Dates and Time: Basic date and time classes, Different time formats, Converting between formats, Formatting dates and times, Parsing date/time information, Binary Data: What is Binary Data? , Binary vs. text, Using the Struct module. Defining a function, Python Built-in Functions , Calling a function, Types of functions, Function Arguments, Default Argument, Anonymous functions, Global and local variables, Custom Functions vs. Standard Functions, Refactoring, Making Functions Reusable, Functions as Data	20%	9
Unit 4: Python Library Overview & Introduction to Tinker An Introduction to libraries used for AI & ML, IoT & Cyber Security: numpy, matplotlib, pandas, opencv, tinker, tensorflow, keras, tensorflow, Theano, Sci-kit learn, PyTorch , mraa, sockets, mysqldb, requests, paho-mqtt, Scapy, Requests, IMpacket, pwntools, Cryptography, python-nmap, Faker, Twisted, pylibnet, RawSocketpy. Reading from a file – Writing to a file – Other Operations on Files. GUI Programming: Tkinter and its widgets – Overview of other GUIs	20%	9
Unit 5: OOP Concepts & Introduction to Django Classes and Object-Oriented Programming, Abstract Data Types and Classes, Inheritance, Encapsulation and Information Hiding. Features of Django , Django web server, Understanding Django environment, Django architecture, MVC and MTVA , view that displays a hyperlink , Mapping the views to URLs, Improving the views using templates, Template inheritance, Sending data	20%	9

from url to view, Sending data from view to template, Create & Activating the app, Adding background color for web pages, Adding banner, background image to the web site		

List of Practical	Weightage	Contact hours
1: a. Demonstrate installation of python. b. Working of variables and identifiers with simple programs. c. Create different variables to show different types of data, operators and expressions. d. Demonstrate working of constants. e. Usage of input output statements.	20%	6
2: a. Demonstrate practicals based on conditional statements. b. Working of decision statements. c. Show replacement of switch statements. d. Demonstrate working of different loops. e. Usage of break , continue and pass. f. Show working of arrays. g. Show different string manipulations.	20%	6
3: a. Demonstrate List and accessing of list. b. Working of tuples. c. Show working of dictionaries.	20%	6
4: a. Demonstrate date and time functions. b. Working of conversion of different date format manipulations. c. Create functions and its types.	20%	6
5: a. Demonstrate working of listed libraries for real time applications.	20%	6

Instructional Method and Pedagogy: (Max. 100 words)

Utilizing Powerpoint Presentations, Chalk-Board demonstration, videos on various topics, group discussions and seminars are some of the methods adopted to improve the student ability to grasp the fundamental knowledge related to the subject. The discussion during the theoretical sessions and hands-on sessions during the practical laboratory sessions will enable students to apply the knowledge gained on real-life problems. Additional activities like minor projects on the said subject will work as a stepping stone in enhancing the student's performance.

Course Outcome:	Blooms' Taxonomy Domain	Blooms' Taxonomy Subdomain
After successful completion of the above course, students will be able to:		
CO1: Understand basics of python programming.	**Cognitive**	**Understand**
CO2: Understand the concepts of loops and control structures for different purposes.		**Understand**
CO3: Comprehend about working of list and dictionaries.		**Comprehend**
CO4: Design python application with the use of date-time and other functions.		**Design**
CO5: Apply in development of real time applications.		**Apply**

Learning Resources	
1.	Reference Books: 1. Python Crash Course: A Hands-On, Project-Based Introduction to Programming (2nd Edition) 2. John V Guttag. "Introduction to Computation and Programming Using Python", Prentice Hall of India 3. R. Nageswara Rao, "Core Python Programming", dreamtech 4. Wesley J. Chun. "Core Python Programming - Second Edition", Prentice Hall 5. Kenneth A. Lambert, "Fundamentals of Python – FirstPrograms", CENGAGE Publication 6. Luke Sneeringer, "Professional Python", Wrox 7. John Paul Mueller, Luca Massaron, Python for Data Science For Dummies , WILEY 8. Wes McKinney,Python for Data Analysis Data Wrangling with Pandas, NumPy, and IPython, 2nd Edition , O'REILLY
2.	Other Electronic Resources: 1. https://onlinecourses.nptel.ac.in/noc22_cs32/preview 2. https://onlinecourses.swayam2.ac.in/aic20_sp33/preview

<table>
<tr><th>Evaluation Scheme</th><th>Total Marks</th></tr>
<tr><td>Theory:
Mid semester Marks</td><td>20 marks</td></tr>
<tr><td>Theory:
End Semester Marks</td><td>40 marks</td></tr>
<tr><td>Theory:
Continuous Evaluation Component Marks (CEC)</td><td>
<table>
<tr><td>Attendance</td><td>05 marks</td></tr>
<tr><td>MCQs</td><td>10 marks</td></tr>
<tr><td>Open Book Assignment / Skill enhancement activities / case study / Assignment</td><td>15 marks</td></tr>
<tr><td>Article Review /Presentation/ miscellaneous activities / Projects</td><td>10 marks</td></tr>
<tr><td>Total</td><td>40 Marks</td></tr>
</table>
</td></tr>
<tr><td>Practical Marks</td><td>
<table>
<tr><td>Attendance</td><td>05 marks</td></tr>
<tr><td>Practical Exam</td><td>20 marks</td></tr>
<tr><td>Viva</td><td>10 marks</td></tr>
<tr><td>Journal</td><td>10 marks</td></tr>
<tr><td>Discipline</td><td>05 marks</td></tr>
<tr><td>Total</td><td>50 Marks</td></tr>
</table>
</td></tr>
</table>

COURSE CODE MSDS202	COURSE NAME Database Management System	SEMESTER II

Teaching Scheme (Hours)				Teaching Credit			
Lecture	**Practical**	**Tutorial**	**Total Hours**	**Lecture**	**Practical**	**Tutorial**	**Total Credit**
45	30	0	75	3	1	0	4

Course Prerequisites	Fundamental knowledge of any basic programming language.
Course Category	Major Course
Course focus	Employability
Rationale	The present course will equip students with the skills related to data categorization, data classification and formation of databases using SQL. This will help students to design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS. It also provides students with theoretical knowledge and practical skills in the use of databases and database management systems in information technology applications.
Course Revision/ Approval Date:	
Course Objectives **(As per Blooms' Taxonomy)**	To enable the student to: 1: **Understand** basic principles of classification and categorization of data and databases. 2: **Understand** the basic elements of a relational database management system. 3: **Understand** & **Remember** the fundamentals of a database of collected data & **apply** the data models for relevant problems. 4: **Understand** the entity relationship and convert the entity relationship. diagrams into RDBMS and **apply** SQL queries on the respective data into RDBMS and **apply** SQL queries on the data. 5: **Understand** the basic understanding related to query evaluation and optimization techniques.

Course Content (Theory)	Weightage	Contact hours
Unit 1: Basic Concepts - Purpose of database systems-Components of DBMS – DBMS Architecture and Data Independence- Data modeling - Entity Relationship Model, Relational – Network- Hierarchical and object-oriented models, Data modeling using the Entity Relationship Model.	25%	12
Unit 2: Relational databases, Structure of relational databases — relational algebra- tuple relational calculus, Data definition with SQL, insert, delete and update statements in SQL– data manipulation with SQL, –Introduction to views, joins and types of queries.	15%	6
Unit 3: Introduction to Transaction Processing- Transaction and System, Concepts Desirable properties of Transactions-Schedules and Recoverability- Serializability of Schedules-Query processing and Optimization-Concurrency Control - assertions – triggers.	15%	6
Unit 4: Database Design– Design guidelines– Relational database design – Integrity Constraints – Domain Constraints-Referential integrity, ACID properties, Functional Dependency Normalization using Functional Dependencies, Normal forms based on primary keys- general definitions of Second and Third Normal Forms. Boyce Codd Normal Form– Multivalued Dependencies and Fourth Normal Form – Join Dependencies and Fifth Normal Form – Pitfalls in Relational Database Design.	25%	12
Unit 5: Distributed databases: Distributed Database Concepts-Data Fragmentation, Replication and Allocation Techniques-Different Types- Query Processing – semi-join - Concurrency Control and Recovery.	20%	9

List of Practical	Weightage	Contact hours
1: Demonstrate various SQL based software**.**	9%	4
2: Understand the working of ER Model.	7%	2
3: Demonstrate creation of database and table.	7%	2
4: Working of insert, update, delete queries.	7%	2
5: Create a view for the SQL table.	7%	2
6: Demonstrate working of different view manipulation.	7%	2
7: Demonstrate working of Oracle.	7%	2
8: Demonstrate the working of PL/SQL.	7%	2
9: Demonstrate advanced database concepts**.**	7%	2
10: Demonstrate working of different join operations.	7%	2
11: Creation of primary key and foreign key.	7%	2
12: Demonstrate working of relational databases in real time applications.	7%	2

13: Demonstration of distributed databases**.**	7%	2
14: Understand the working of semi-join**.**	7%	2

Instructional Method and Pedagogy: (Max. 100 words)

Utilizing models, Powerpoint Presentations, Chalk-Board demonstration, videos on various topics, group discussions and seminars are some of the methods adopted to improve the student ability to grasp the fundamental knowledge related to database management systems. The discussion during the theoretical sessions and hands-on sessions during the practical laboratory sessions will enable students to apply the knowledge gained.

Course Outcome:	**Blooms' Taxonomy Domain**	**Blooms' Taxonomy Subdomain**
After successful completion of the above course, students will be able to:		
CO1: **Demonstrate** the basic elements of a relational database management system.	**Understand** & **Apply**	Define, Describe & Demonstrate
CO2: **Describe** the fundamental elements of relational database management systems.	**Understand** & **Apply**	Define, Describe & Demonstrate
CO3: **Design** entity relationship and convert entity relationship diagrams into RDBMS and **formulate** SQL queries on the respective data into RDBMS and **formulate** SQL queries on the data**.**	**Evaluate** & **Create**	Analyze & Design
CO4: **Demonstrate** the fundamental understanding related to query evaluation and optimization techniques.	**Understand**	Describe
CO5: **Extend** normalization for the development of application softwares.	**Understand** & **Remember**	Define, Describe & Demonstrate

Learning Resources	
1.	Reference Books: 1. Elmasri and Navathe, Fundamentals of Database System, (4th Edition), Pearson Education Asia (2008). 2. Henry F Korth, Abraham Silbershatz, Database System Concepts, McGraw Hill 2nd edition. (2005).

	3. C.J.Date, An Introduction to Database Systems, (7th Edition) Pearson Education Asia (2006). 4. Bibin C. Desai, An Introduction to Database Systems, Galgotia Publications, (2000).
2.	Other Electronic Resources: 1. https://onlinecourses.nptel.ac.in/noc22_cs91/preview 2. https://www.coursera.org/learn/database-structures-and-management-with-mysql

Evaluation Scheme	**Total Marks**
Theory: Mid semester Marks	20 marks
Theory: End Semester Marks	40 marks
Theory: Continuous Evaluation Component Marks (CEC)	

Attendance	05 marks
MCQs	10 marks
Open Book Assignment / Skill enhancement activities / case study / Assignment	15 marks
Article Review /Presentation/ miscellaneous activities / Projects	10 marks
Total	**40 Marks**

Practical Marks	

Attendance	05 marks
Practical Exam	20 marks
Viva	10 marks
Journal	10 marks
Discipline	05 marks
Total	**50 Marks**

COURSE CODE MSDS203	COURSE NAME Fundamentals of AI & ML	SEMESTER II

Teaching Scheme (Hours)				Teaching Credit			
Lecture	**Practical**	**Tutorial**	**Total Hours**	**Lecture**	**Practical**	**Tutorial**	**Total Credit**
45	30	0	75	3	1	0	4

Course Prerequisites	Basic programming proficiency (Python/Java/R), foundational knowledge in calculus, linear algebra, and probability.
Course Category	Major Course
Course focus	Employability
Rationale	This course offers a comprehensive exploration of Artificial Intelligence (AI) and Machine Learning (ML) principles, equipping students with practical knowledge and skills essential for modern data-driven environments. Through a structured curriculum, students delve into the historical evolution, diverse algorithms, and ethical considerations surrounding AI and ML applications. By fostering critical thinking and hands-on experience, the course aims to empower learners to leverage AI and ML techniques effectively, contributing to innovative solutions across various domains while promoting responsible and ethical use of emerging technologies.
Course Revision/ Approval Date:	
Course Objectives **(As per Blooms' Taxonomy)**	To enable the student to: 1: **Understand** the development of Machine Learning and its practical applications in various fields. 2: **Explore** different algorithms used in supervised learning and their practical implications. 3: **Investigate** unsupervised learning techniques and **understand** how reinforcement learning contributes to decision-making processes.

	4: **Examine** the principles behind neural networks and their applications in pattern recognition and data classification. 5: **Develop** skills in **validating** and **evaluating** machine learning models while considering ethical implications.

Course Content (Theory)	Weightage	Contact hours
Unit 1: **Introduction** Overview of AI and ML, Historical perspective, Types of Machine Learning: Supervised, Unsupervised, Reinforcement Learning, Applications of AI and ML in Data Science	10%	8
Unit 2: **Supervised Learning** Linear Regression, Logistic Regression, Decision Trees and Random Forests, Support Vector Machines, Decision Boundary	20%	12
Unit 3: **Unsupervised Learning & Reinforcement Learning** Clustering: K-Means, Dimensionality Reduction: PCA, Association Rules: Apriori algorithm	20%	10
Unit 4: **Neural Network Learning Principles & Models** Neural Networks basics, Perceptron Learning, Feedforward Networks and Back Propagation, Gradient Descent Learning Algorithm, Application of ANN for classification, Multi-layer NN and Deep Learning Network principles	25%	15
Unit 5: **Model Validation & Evaluation** Training-Set, Test-Set, Validation-Set, Cross-validation, Ethical considerations in AI and ML	25%	15

List of Practical	Weightage	Contact hours
1: Data Exploration and Preprocessing a. Importing datasets using Python libraries like Pandas. b. Exploring dataset characteristics: dimensions, data types, missing values. c. Data cleaning techniques: handling missing data, outlier	12%	4

detection, and removal.		
2: Supervised Learning a. Implementing Linear Regression for predicting continuous variables. b. Building Classification models using Logistic Regression, Decision Trees, and Random Forests. c. Evaluating model performance using metrics like accuracy, precision, recall, and F1-score.	20%	6
3: Unsupervised Learning a. Applying K-Means Clustering for grouping similar data points. b. Performing Dimensionality Reduction using Principal Component Analysis (PCA). c. Mining association rules with the Apriori algorithm for market basket analysis.	12%	4
4: Neural Network Modeling a. Building a simple Perceptron for binary classification tasks. b. Constructing a Feedforward Neural Network using TensorFlow or PyTorch. c. Training and optimizing Multi-layer Neural Networks for image classification tasks.	20%	6
5: Model Validation and Evaluation a. Splitting datasets into training, validation, and test sets. b. Implementing Cross-validation techniques to assess model generalization. c. Performing hyperparameter tuning using techniques like Grid Search or Random Search.	20%	6
6: Data Visualization a. Creating various types of plots and charts using libraries like Matplotlib and Seaborn. b. Visualizing dataset distributions, feature importance, and model performance metrics.	6%	2
7: Ethical Considerations a. Discussing ethical implications of AI and ML applications. b. Identifying biases in datasets and models, and strategies for mitigation. c. Debating real-world case studies involving AI ethics and societal impact.	6%	2

Instructional Method and Pedagogy: (Max. 100 words) Powerpoint presentation, Interactive lectures, Hands-on coding sessions, Group discussions and case studies, Peer review and presentations, Guest lectures and industry interactions.

Course Outcome:	**Blooms' Taxonomy Domain**	**Blooms' Taxonomy Subdomain**
After successful completion of the above course, students will be able to:		
CO1: **Demonstrate** an understanding of the historical background and real-world applications of Machine Learning.	**Cognitive**	**Understand**
CO2: **Analyze** and compare various supervised learning algorithms in practical contexts.		**Analyze**
CO3: **Apply** unsupervised learning techniques and **comprehend** the role of reinforcement learning in decision-making.		**Apply**
CO4: **Demonstrate** proficiency in understanding neural network principles and their applications.		**Understand**
CO5: **Evaluate** machine learning models effectively and consider ethical implications in their applications.		**Evaluate**

Learning Resources	
1.	Reference Books: 1. "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron 2. "Pattern Recognition and Machine Learning" by Christopher M. Bishop 3. Online resources, research papers, and documentation for relevant libraries 4. Artificial Intelligence – Guide to Intelligent Systems by Michael Negnevitsky 5. Data Science from Scratch – First Principle with Python – Joel Grus O'Reilly 6. Introduction to Machine Learning by Tom Mitchel MIT Press 7. Machine Learning for Absolute Beginners – Oliver Theobald
3.	Other Electronic Resources: 1. https://onlinecourses.nptel.ac.in/noc23_ge40/preview 2. https://onlinecourses.swayam2.ac.in/cec21_cs08/preview

<table>
<tr><th>Evaluation Scheme</th><th>Total Marks</th></tr>
<tr><td>Theory:
Mid semester Marks</td><td>20 marks</td></tr>
<tr><td>Theory:
End Semester Marks</td><td>40 marks</td></tr>
<tr><td>Theory:
Continuous Evaluation Component Marks (CEC)</td><td>
<table>
<tr><td>Attendance</td><td>05 marks</td></tr>
<tr><td>MCQs</td><td>10 marks</td></tr>
<tr><td>Open Book Assignment / Skill enhancement activities / case study / Assignment</td><td>15 marks</td></tr>
<tr><td>Article Review /Presentation/ miscellaneous activities / Projects</td><td>10 marks</td></tr>
<tr><td>Total</td><td>40 Marks</td></tr>
</table>
</td></tr>
</table>

COURSE CODE MSDS204	COURSE NAME Numerical Methods and Analysis using MATLAB	SEMESTER II

Teaching Scheme (Hours)				Teaching Credit			
Lecture	**Practical**	**Tutorial**	**Total Hours**	**Lecture**	**Practical**	**Tutorial**	**Total Credit**
45	30	0	75	3	1	0	4

Course Prerequisites	Basic knowledge of algebra and calculus. Basic knowledge of derivatives and integration.
Course Category	Minor
Course focus	Employability
Rationale	This course involves solving the system of linear equations, approximation of the root of nonlinear equations. Students also learn interpolation methods for equi-distant and non equi-distant data. Also they develop the skill to find numerical integration and differentiation and techniques to solve IVP.
Course Revision/ Approval Date:	
Course Objectives **(As per Blooms' Taxonomy)**	To enable the student to: 1: **Solve:** Make the students familiarize with the ways of solving complicated mathematical problems numerically 2:**Understand:** To help them become familiar with MATLAB and other convenient numerical software such as Microsoft Excel and with simple programming 3: **Recognize, Find:** Obtain numerical solutions to non-algebraic equations and systems of linear equations. 4: **Understand:** Describe and understand of the several errors and approximation in numerical methods 5: **Understand, Find:** Understanding of several available Solution techniques for differential Equations in One Variable. Study of Curve Fitting and Interpolation.

Course Content (Theory)	Weightage	Contact hours
Unit 1: **Errors:** Notions of round off, truncation and other errors, Errors in numerical computations **Solution of Algebraic and Transcendental Equations:** Bisection, False position, Iterative Method, Newton Raphson Method, Secant Method. Solution using Matlab	**20%**	**9**
Unit 2: **Solution of system of Linear Equations:** Gauss Elimination method, Gauss Jordan Method, LU decomposition method, Gauss Jacobi Method, Gauss Seidel method. Solution using Matlab	**20%**	**9**
Unit 3: **Interpolation:** Newton's forward and backward interpolation, Newton's divided difference interpolating polynomials, Lagrange Interpolating polynomials. Solution using Matlab.	**20%**	**9**
Unit 4: **Numerical Differentiation:** First and second order differentiation Equations of Equally Spaced Data. Solution using Matlab. **Numerical Integration:** Trapezoidal rule, Simpson's one third and 3/8th rule. Solution using Matlab	**20%**	**9**
Unit 5: **Numerical methods for Solution of ordinary differential equations:** Taylor's series method, Euler's method, Modified Euler's method, Runge Kutta forth order method, Milne's Predictor Corrector Method. Finite element method to solve second order ODE. Solution using Matlab. **Curve Fittings:** General Linear Least Squares, Fitting of quadratic and exponential curves.Solution using Matlab.	**20%**	**9**

List of Practicals	Weightage	Contact hours
1: Introduction to MATLAB, Matrix algebra, functions.	7%	2
2: Loops: For, if else , while Programme for Bisection Method.	7%	2
3: Programme for Regula-falsi and Secant Method.	7%	2
4: Programme for Newton-Raphson's Method.	7%	2
5: Programme for Difference Table.	7%	2
6. Programme for Newtons's Forward and Backward Interpolation.	8%	3
7. Programme for Newton's Divided Difference Interpolation Method.	7%	2

8. Programme for Lagrange's Method.	7%	2
9. Use of Curve fitting Toolbox.	7%	2
10. Programme for Numerical integrations (Trapezoidal and simpson's rules).	8%	3
11. Solving system of linear equations.	7%	2
12. Plotting 2D and 3D graphs.	7%	2
13. Programme for Euler's Method.	7%	2
14. Practice test and Revision.	7%	2

Instructional Method and Pedagogy: Chalk board, Powerpoint presentations, Use of Mathlab, Excel and Geogebra. Group Discussion, Case Study, Quizziz application.

Course Outcomes:	**Blooms' Taxonomy Domain**	**Blooms' Taxonomy Subdomain**
After successful completion of the above course, students will be able to:		
CO1: **Apply, solve:** Apply numerical methods to find out solutions of non-algebraic equations using different methods under different conditions, and numerical solutions of systems of algebraic equations.	**Apply**	Describe, Find
CO2: **Demonstrate, find:** Apply various interpolation methods and finite difference concepts.	**Understand, Remember**	Demonstrate & Examine, Find
CO3: **Demonstrate:** Work out numerical differentiation and integration whenever and wherever routine methods are not applicable	**Understand, Remember**	Demonstrate & Examine, Find
CO4: **Solve:** Work numerically on the ordinary differential equations using different methods through the theory of finite differences.	**Evaluate**	Examine, Find
CO5: **Understand:** Work numerically on the partial differential equations using different methods through the theory of finite differences.	**Understand, Remember, Apply & Analyze**	Define, Classify, Describe, Demonstrate & Examine

Learning Resources	
1.	**Reference Books:** 1. S. S. Sastry, Introductory methods of Numerical Analysis, 5th Edition, Prentice-Hall India, 2012. 2. G. Shankar Rao, Numerical Analysis, New Age International Pvt. Ltd., 2006. 3. P.C. Biswal, Numerical Analysis, Prentice-Hall India, 2008.
2.	**Journals & Periodicals:** Mathematics Open
3.	**Other Electronic resources:** 1. MATLAB, Microsoft Excel, Geogebra Toolbox

Evaluation Scheme	Total Marks
Theory: Mid semester Marks	20 marks
Theory: End Semester Marks	40 marks
Theory: Continuous Evaluation Component Marks	

Attendance	05 marks
MCQs	10 marks
Open Book Assignment / Skill enhancement activities / case study / Assignment	15 marks
Article Review /Presentation/ miscellaneous activities / Projects	10 marks
Total	**40 Marks**

Practical Marks	

Attendance	05 marks
Practical Exam	20 marks
Viva	10 marks
Journal	10 marks
Discipline	05 marks
Total	**50 Marks**

COURSE CODE MSDS205	COURSE NAME Statistics - II	SEMESTER II

Teaching Scheme (Hours)				Teaching Credit			
Lecture	**Practical**	**Tutorial**	**Total Hours**	**Lecture**	**Practical**	**Tutorial**	**Total Credit**
45	30	0	75	3	1	0	4

Course Prerequisites	Basic concepts related to Statistics
Course Category	Minor
Course focus	Employability
Rationale	The syllabus covers foundational and advanced statistical concepts essential for data science. Beginning with hypothesis testing, students learn to draw conclusions from data. They progress to understanding standard error, sampling distribution, and significance testing techniques for accurate estimation and analysis. Advanced statistical tests like chi-square and ANOVA equip students to analyze complex datasets effectively. The study of advanced indexing techniques enhances multidimensional data evaluation. Time series analysis provides skills for forecasting and trend identification, preparing students for real-world data challenges across various industries.
Course Revision/ Approval Date:	
Course Objectives **(As per Blooms' Taxonomy)**	To enable the student to: 1. **Develop** proficiency in conducting hypothesis tests and **interpreting** their results. 2. **Understand** the significance of standard error and sampling distribution in statistical **analysis**. 3. **Explore** advanced statistical tests and their **applications** in data science. 4. **Introduce** weighted composite index numbers and quality indexing methods for representing and **assessing** complex data structures. 5. **Develop** skills in **analyzing** time series data and identifying trends, seasonal variations, and cyclical patterns.

Course Content (Theory)	Weightage	Contact hours
Unit 1: Statistical Inference for Data Science Introduction to Hypothesis Testing: Basics of hypothesis testing, null and alternative hypotheses. Case studies/examples related to hypothesis testing.	10%	4
Unit 2: Standard Error and Sampling Distribution: Understanding standard error, sampling distribution, and estimation techniques. Test of Significance: Two-tailed test, one-tailed test, Z-test for significance of coefficient of correlation.	15%	8
Unit 3: Advanced Statistical Tests Advanced Tests of Goodness: Introduction to advanced chi-square tests and their applications in data science, additional tests used for checking goodness of fit and independence in categorical data. ANOVA/MANOVA: Introduction to ANOVA/MANOVA and their application in analyzing the variance. Utility of ANOVA/MANOVA in data science.	20%	9
Unit 4: Advanced Indexing Techniques Composite Index Numbers: Weighted composite index numbers, composite indices in representing complex data structures. Quality Indexing: Introduction of quality indexing methods and their application in assessing data quality, role of quality indices in data science.	22%	12
Unit 5: Time Series Analysis Introduction, Components of Time series analysis, Measurement of trends-Graphical method, semi-average method, Moving average method, least square method, Measurement of seasonal variation-simple average, Ratio to trend and Ratio to moving average method, cyclical variation.	22%	12

List of Practicals	Weightage	Contact hours
1: a. Use Excel/R to conduct t-tests and z-tests for population mean. b. Perform hypothesis testing for population proportions using Excel/R functions.		

c. Analyze real-world datasets to formulate and test hypotheses. d. Conduct hypothesis tests on sample data provided in Excel sheets or imported into R. e. Analyze and interpret results using descriptive statistics and graphical representations in Excel/R. f. Compare outcomes of hypothesis tests for different scenarios.	20%	6
2: a. Generate sampling distributions using Excel/R functions. b. Calculate standard error for sample means and proportions. c. Visualize sampling distributions and standard error using Excel/R plots. d. Perform two-tailed and one-tailed tests using Excel/R functions. e. Conduct Z-tests for correlation coefficients in Excel/R. f. Interpret p-values and make decisions based on test results.	20%	6
3: a. Use Excel/R to perform chi-square tests for independence and goodness of fit. b. Apply additional tests for categorical data analysis, such as Fisher's exact test and McNemar's test. c. Interpret test results and assess the goodness of fit in Excel/R. d. Conduct ANOVA/MANOVA analysis using excel and/or built-in functions in R. e. Compare group means and test for statistical significance. f. Visualize multivariate data using plots and graphs in R.	20%	6
4: a. Calculate weighted composite index numbers using Excel formulas or R functions. b. Compare different weighting schemes and their impact on index values. c. Interpret composite indices and assess trends over time. d. Implement quality indexing methods in Excel/R to assess data quality. e. Evaluate data integrity and identify areas for improvement based on quality indices. f. Develop quality assurance strategies for data science projects.	20%	6
5: a. Use Excel/R to plot time series data and identify trends. b. Calculate trend measures like moving averages and least		

squares in Excel/R. c. Analyze seasonal variation using Excel/R functions and visualize seasonal patterns. d. Detect the cyclical patterns in time series data using Excel/R. e. Apply statistical methods to measure cyclical variation and assess its significance. f. Interpret cyclical trends and their implications for forecasting and decision-making.	20%	6

Instructional Method and Pedagogy: Chalk board, Powerpoint presentations, demonstrations, Hands-on sessions, discussion, Quiz, online materials, Small projects, etc.

Course Outcomes:	**Blooms' Taxonomy Domain**	**Blooms' Taxonomy Subdomain**
After successful completion of the above course, students will be able to:		
CO1: **Demonstrate** the ability to formulate null and alternative hypotheses, conduct hypothesis tests using appropriate statistical methods, and interpret the results effectively.	**Apply**	Formulate Conduct Interpret
CO2: **Analyze** sampling distributions and understand the role of standard error in estimating population parameters, enabling them to conduct **tests** of significance confidently.	**Understand**	Analyze Understand Estimate Conduct
CO3: **Apply** advanced chi-square tests and ANOVA/MANOVA techniques to analyze complex datasets, **deriving** meaningful insights and making informed decisions in data science contexts.	**Apply**	Apply Analyze Derive Make
CO4: **Construct** weighted composite index numbers and quality indices accurately, **demonstrating** proficiency in representing data structures and assessing data quality effectively in data science applications.	**Create**	Construct Demonstrate Represent Assess
CO5: **Apply** graphical and statistical methods to **analyze** time series data, measure trends, **assess** seasonal variations, and identify cyclical patterns accurately, facilitating informed decision-making in data science.	**Apply**	Apply Analyze Measure Assess Identify Facilitate

Learning Resources	
1.	**Reference Books:** 1. D.C. Montogomery and G.C.Runger, Applied Statistics and Probability for engineers, New Jersey, John Wiley and Sons, 3rd edition, 2003. 2. David. M. Levin, David. F. Stephen, and Cathryn. A. Szadat, (2013), Statistics for managers using MS-Excel, 7Th Edition, Pearson Education (India) **Text Books:** 1. S. P. Gupta, 2014, Business Statistics and Statistical Methods, S. Chand Publication, New Delhi 2. L. Mayes & Keying, (2005), Probability Statistics for Engineers and Scientists, Pearson Education. 3. Goon, Gupta and Dasgupta – Fundamentals of Statistics, Vol. I.
2.	**Journal**
3.	**Other Electronic resources:** 2. https://nptel.ac.in/courses/110/108/110108068/ 3. https://nptel.ac.in/courses/111/108/111108061/ 4. https://swayam.gov.in/nd2_cec20_ma07/preview 5. https://swayam.gov.in/nd1_cec20_ma02/preview 6. https://www.coursera.org/learn/statistics 7. https://www.coursera.org/specializations/jhu-data-science

Evaluation Scheme	Total Marks
Theory: Mid semester Marks	20 marks
Theory: End Semester Marks	40 marks
Theory: Continuous Evaluation Component Marks	

Attendance	05 marks
MCQs	10 marks
Open Book Assignment / Skill enhancement activities / case study / Assignment	15 marks
Article Review /Presentation/ miscellaneous activities / Projects	10 marks
Total	**40 Marks**

Practical Marks	

Attendance	05 marks
Practical Exam	20 marks
Viva	10 marks
Journal	10 marks
Discipline	05 marks
Total	**50 Marks**