UNIVERSITY SCHOOL
OF
INFORMATION AND COMMUNICATION TECHNOLOGY

Department of Information Technology

COURSE STRUCTURE

Course: Master of Computer Applications
(Specialization in Data Science)

BATCH: 2026-28

GAUTAM BUDDHA UNIVERSITY

GAUTAM BUDH NAGAR, GREATER NOIDA, UP, INDIA

BRIDGE COURSE

S.No	Course Code	Course Name	Teaching Scheme			Credits	Types
			L	T	P		
1.	MCB001	Computer Fundamental and Programming	3	1	0	4	BC1
2.	MCB003	Introduction to Internet Technology	3	0	0	3	BC2
3.	MCB005	Fundamental of Operating Systems	3	0	0	3	BC3
4	MCB081	Operating System Lab	0	0	4	2	BC-L
5	MCB083	Internet Technology Lab	0	0	4	2	BC-L
		Total Hours and Credits	9	1	8	14	

*__BC__: Bridge Course, BC-L: Bridge Course Lab

SEMESTER I

S.NO.	Course Code	Course Name	Teaching Scheme			Credits	Types
			L	T	P		
1.	MDS101	Computer Fundamental and MS Excel	3	0	0	3	CC -L/FC
2.	MDS103	Software Engineering	3	0	0	3	CC
3.	MDS105	Data Science	3	0	0	3	CC
4.	MDS107	Discrete Mathematics	3	0	0	3	CC
5.	MDS109	Python Programming	3	0	0	3	CC-L
6.	MDS181	MS Excel Lab	0	0	2	1	CC-L
7.	MDS183	Python Programming Lab0	0	0	2	1	CC-L
8.	MDS185	Data Science Lab	0	0	2	1	CC-L
9.	GP	General Proficiency	Non-Credit				
		Total Hours and Credits	15	0	6	18	

*CC: Core Course from IT, L: Lab, OE: Open Electives from other Departments/Schools, AEC: Ability Enhancement Course

SEMESTER II

S.NO	Course Code	Course Name	Teaching Scheme			Credits	Types
			L	T	P		
1.	MDS102	Analysis and Design of Algorithm	3	0	0	3	CC
2.	MDS104	Data Structures	3	0	0	3	CC / SEC
3.	MDS106	Machine Learning	3	0	0	3	CC
4.	MDS108	Database Management System	3	0	0	3	CC
5.	MDS110	Data Modelling and Visualization	3	0	0	3	CC
6.	MDS112	Theory of Computation	3	0	0	3	CC / SEC
7.	MDS186	Machine Learning Lab	0	0	2	1	CC-L
8.	MDS188	Database Management System Lab	0	0	2	1	CC-L
9.	MDS184	Data Modelling and Visualization Lab	0	0	2	1	CC-L
10.	ITV301	Professional Ethics	2	0	0	1	VAC
11.	GP	General Proficiency	Non-Credit				
		Total Hours and Credits	20	0	6	22	

*SEC: Skill Enhancement Course, VAC: Value Added Course

SEMESTER I

Computer Fundamental and MS Excel			
Course Code:	**MDS101**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**U**
Course Year (U / P):	**2P**	**Course Semester (U / P):**	**3P**
No. of Lectures + Tutorials (Hrs/Week):	**03 + 00**	**Mid Sem. Exam Hours:**	**1.5**
Total No. of Lectures (L + T):	**45**	**End Sem. Exam Hours:**	**3**

COURSE OBJECTIVES
1. The course aims to provide students with hand-on experience in using Microsoft Excel for Data Analytics
2. In this course, students will learn how to analyse data
3. visual representations; how to work with data models using multiple worksheets and files
4. and how business calculations can be expressed using the Excel Data Analysis Expressions (DAX).
5. data through flexible data aggregations using Excel tables, pivot tables
COURSE OUTCOMES
At the end of the course the students should be able to:
1. Apply Excel functions and formulas for basic data analysis and interpretation
2. Make use of PivotTables for data aggregation and summarization
3. Build Dashboards for data visualization
4. Build Data Models using multiple worksheets and files
5. Make use of Excel Macros to perform routine tasks

UNIT I EXCEL FUNCTIONS AND TABLES FOR DATA ANALYSIS:
Excel Functions: Text functions, Logical functions, Lookup and Reference functions, Statistical Functions; Excel Tables: Overview, Excel Tables and Pivot Tables, Creating and Formatting Tables,Pivot Tables and Pivot Charts: Overview, Creating Pivot Tables to analyze worksheet data.

UNIT II DATA VISUALIZATION USING EXCEL:
Dashboards: Overview, Using multiple Pivot Tables, Pivot Charts and PivotTable tools to create a dynamic dashboard, Using multiple Pivot Tables, Pivot Charts and PivotTable tools to create a dynamic dashboard.

UNIT III DATA ANALYSIS AND EXPRESSION:
PowerPivot, Power Query, Basics of Data Analysis Expressions (DAX),
DAX in PowerPivot, Calculations in PowerPivot, Data Models:Creating Data Models in Excel, Creating PivotTable/PivotChart through Data Models.

UNIT IV MACROS AND EXCEL VBA :
Creating Macros in Excel, Macros in a single workbook, Absolute References, Relative References, Assigning Macros to Objects; VBA-Excel Macros, Objects, Variables, Conditional statements and loops.

UNIT V POWER BI USING EXCEL :
Introduction to Power BI, Overview of Power BI Desktop, Creating Table visualizations, Formatting Table visualizations Matrix visualizations, Aggregation methods, Score Cards, Multi-row Cards, Percentage calculations, Filtering data – Slicers, Visual filters, Filtering data – Page filters, Drill-through , filters Graphical visualizations – Column graphs, Clustered column graphs, Stacked column graphs,Trend analysis graphs, Area graphs, Ribbon, graphs, Creating dashboards, Using custom visualizations, , Using Power BI

Service, Publishing reports to Power BI Service, Date functions, Date Master tables, Creating Relationships,DAX calculated columns, DAX measures, Power BI Query editor

Books Recommended:

1. Stephen L. Nelson and Elizabeth C. Nelson: “Microsoft Excel Data Analysis for Dummies”, John Wiley and Sons
2. Rob Collie, Avichal Singh: “Power Pivot and Power BI: The Excel User's Guide to DAX, Power Query, Power BI & Power Pivot in Excel 2010-2016”, Holy Macro! Books
3. John WalkenBach: “Excel VBA Programming For Dummies”, John Wiley and Sons
4. Ken Bluttman: “Excel Formulas & Functions For Dummies”, John Wiley and Sons
5. Alberto Ferrari: “Analyzing Data with Power BI and Power Pivot for Excel (Business Skills)”, Microsoft Press
6. “Business Intelligence Practices, Technology and Management”, Rajiv Sabherwal and Irma Becerra-Fernandez, John Wiley and Sons

Online Resources/ Reference Material:

- https://support.microsoft.com
- https://www.tutorialspoint.com
- https://www.excel-easy.com
- https://www.datacamp.com
- “Introducing Microsoft Power BI”, Alberto Ferrari and Marco Russo, Microsoft Press, Available at https://download.microsoft.com/download/0/8/1/0816F8D1-D1A5-4F60-9AF5-BC91E18D6D64/Microsoft_Press_ebook_Introducing_Power_BI_PDF_mobile.pdf

SOFTWARE ENGINEERING			
Course Code:	**MDS103**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**P**
Course Year (U / P):	**P**	**Course Semester (U / P)**	**1 P**
No. of Lectures + Tutorials (Hrs/Week):	**03 + 00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45 + 00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Knowledge of basic SW engineering methods and practices and application.
2. A general understanding of software process models.
3. Understanding of software requirements and the SRS documents.
4. Understanding of software design process.
5. Understanding of software coding, testing and maintenance.

COURSE OUTCOMES

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

1. Basic knowledge and understanding of the analysis and design of complex systems.
2. Ability to apply software engineering principles and techniques.
3. Ability to develop, maintain and evaluate large-scale software systems.
4. To produce efficient, reliable, robust and cost-effective software solutions.
5. Ability to perform independent research and analysis.

UNIT I SOFTWARE ENGINEERING

Introduction to software engineering: definitions, roles, project planning, problem definition, solution development strategies, process planning, paradigms, core principles, and main activities.

.

UNIT II SOFTWARE LIFE CYCLE MODELS

Software Development Life Cycle (SDLC), including various SDLC models such as the waterfall model and its variants, prototype model, iterative enhancement model, spiral model, and RAD model. This encompasses a detailed comparison of these models, the composition and roles of software development teams, the setup of software development environments, processes for validation and traceability, maintenance practices, requirements for prototyping, and principles of software project management.

UNIT III REQUIREMENT ANALYSIS AND DESIGN

Software Requirement Specification (SRS): Introduction, purpose, importance, key features, SRS structure, IEEE SRS standards, functional and non-functional requirements, requirements gathering and analysis, and requirement engineering and management.

UNIT IV SOFTWARE DESIGN PROCESS

Software Design: Introduction, design process activities: architectural design, Abstract specification, Interface design, component design, data structure design, algorithm design modular approach, top-down design, bottom-up design, design methods: data-flow model: data flow diagram, entity-relation-attribute model: E-R diagram

structural model: structure charts, context diagrams, object models: use case modelling, use case diagrams, sequence diagrams, cohesion and coupling.

UNIT V SOFTWARE CODING, TESTING AND MAINTENANCE

Coding, Testing Methods: unit testing, integration testing, system testing, acceptance testing, testing techniques: white box testing, black box testing, thread testing, regression testing, alpha testing, beta testing, static testing, dynamic testing, Evolution of software products, economics of maintenance, category of software maintenance, Role of product development life cycle, deployment model, adaptive maintenance, corrective maintenance, perfective maintenance, enhancement request, proactive defect prevention, problem reporting, problem resolution, software maintenance from customers' perspective, maintenance standard: IEEE-1219, ISO-12207.

Textbooks:

1. Pankaj Jalote. An Integrated Approach to Software Engineering Naroda Publishing House, New Delhi, 1997.
2. Ian Sommerville. Software Engineering Pearson Education, 2009.

Reference books

1. Roger S. Pressman. Software Engineering: A Practitioner's Approach. McGraw-Hill Inc., 2004.
2. Nasib S. Gill. Software Engineering: Software Reliability, Testing and Quality Assurance Khanna Book Publishing Co (P) Ltd., New Delhi, 2002

DATA SCIENCE			
Course Code:	**MDS105**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**P**
No. of Lectures + Tutorials (Hrs/Week):	**03 + 00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45 + 00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Learn concepts, techniques and tools they need to deal with various facts of data science practice, including data collection and integration
2. Understand the basic types of data and basic statistics
3. Identify the importance of data reduction and data visualization techniques
4. Understand and implement vectors
5. Understand different data reduction technique

COURSE OUTCOMES

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

1. Understand basic terms what Statistical Inference means
2. Identify probability distributions commonly used as foundations for statistical modelling. Fit a model to data
3. Describe the data using various statistical measures
4. Utilize R elements for data handling
5. Perform data reduction and apply visualization techniques.

UNIT I INTRODUCTION

Definition of Data Science, Big Data and overcoming its hype, Datafication, current perspectives, statistical inference, populations and samples, statistical modelling, probability distributions, model fitting and overfitting. R basics: introduction, environment setup, programming, and basic data types.

UNIT II DATA TYPES & STATISTICAL DESCRIPTION TYPES OF DATA

Attributes and measurement include the definition of an attribute, various types such as asymmetric, binary, nominal, ordinal, and numeric attributes, and distinctions among these types. Classification of attributes can be based on the number of values they take, and comparisons can be drawn between discrete and continuous attributes. In addition, basic statistical descriptions of data encompass measures of central tendency like mean, median, and mode, as well as data dispersion metrics such as range, quartiles, variance, standard deviation, and interquartile range. Graphic representations are also used to visually convey these basic statistical descriptions of data.

UNIT III VECTORS

Creating and naming vectors, vector arithmetic, and vector sub setting are foundational operations. Matrices involve creating and naming matrices, matrix sub setting, working with arrays, and understanding their class. Factors and data frames include an introduction to factors, factor levels, summarizing a factor, ordered factors, comparing ordered factors, as well as an introduction to data frames, subsetting data frames, extending them, and sorting data frames. creating a list, creating a named list, accessing and manipulating list elements, merging lists, and converting lists to vectors.

UNIT IV CONDITIONALS AND CONTROL FLOW

Relational Operators, Relational Operators with Vectors, Logical Operators, Logical Operators with Vectors, and Conditional Statements. Iterative Programming in R includes topics such as an introduction to loops, the use of while loops, for loops, and looping over lists. The section on Functions in R covers an overview, how to write functions, nested functions, function scoping, recursion, loading R packages, and mathematical functions in R.

UNIT V DATA REDUCTION

Overview of Data Reduction Strategies includes wavelet transforms, principal components analysis, attribute subset selection, regression and log-linear models (parametric data reduction), histograms, clustering, sampling, and data cube aggregation. Data Visualization covers pixel-oriented visualization techniques, geometric projection methods, icon-based visualization approaches, hierarchical visualization strategies, and methods for visualizing complex data and relationships.

TEXTBOOKS:

1. Doing Data Science: Straight Talk from the Frontline – Cathy O'Neil & Rachel Schutt, O'Reilly
2. Data Mining: Concepts and Techniques (3rd ed.) – Jiawei Han, Micheline Kamber & Jian Pei, Morgan Kaufmann
3. Statistical Programming in R – K G Srinivas & G M Siddesh, Oxford Publications

REFERENCE BOOKS:

1. "Introduction to Data Mining" by Pang-Ning Tan, Vipin Kumar, and Michael Steinbach, Pearson Education.
2. Brian S. Everitt, "A Handbook of Statistical Analysis Using R," Second Edition, CRC Press, 2014.
3. Peter Dalgaard, "Introductory Statistics with R," Springer Science & Business Media, 2008.
4. Paul Teetor, "R Cookbook," O'Reilly Media, 2011.

DISCRETE MATHEMATICS			
Course Code:	**MDS107**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**P**
Course Year (U / P):	**P**	**Course Semester (U / P)**	**1 P**
No. of Lectures +Tutorials (Hrs/Week):	**03 + 00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45 + 00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Simplify and evaluate basic logic statements including compound statements, implications, inverses, converses, and contrapositives using truth tables and the properties of logic.
2. Express a logic sentence in terms of predicates, quantifiers, and logical connectives.
3. Apply the operations of sets and use Venn diagrams to solve applied problems; solve problems using the principle of inclusion-exclusion.
4. Determine the domain and range of a discrete or non-discrete function, graph functions, identify one-to-one functions, perform the composition of functions, find and/or graph the inverse of a function, and apply the properties of functions to application problems.
5. Apply rules of inference, tests for validity, proof by contradiction, proof by cases, and mathematical induction and write proofs using symbolic logic and Boolean Algebra.

COURSE OUTCOMES

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

1. To express a logic sentence in terms of predicates, quantifiers, and logical connectives.
2. Apply the rules of inference, proof by contradiction, and mathematical induction.
3. Students will evaluate Boolean functions and simplify expressions using Boolean algebra properties.
4. Students will be able to learn about predicates, quantifiers, and logical connectives.
5. Student will be able to use tree and graph algorithms to solve problems.

UNIT 1 MATHEMATICAL LOGIC

Statements, notations, connectives, well-formed formulas, truth tables, tautology, equivalence, implication, normal forms, predicate logic, free and bound variables, inference rules, consistency, contradiction proofs, automatic theorem proving.

UNIT II SET THEORY

Set Theory covers introductions, set combinations, multisets, ordered pairs, identities, binary relations, equivalence, compatibility, partial orderings, and Hasse diagrams. Functions include operations, inverses, classifications, recursion, lattices, and properties. Algebraic structures focus on systems, general properties, semigroups, monads, groups, subgroups, homomorphism, and isomorphism.

UNIT III ELEMENTARY COMBINATORICS

Counting principles, combinations and permutations (including repetitions and constraints), binomial coefficients, binomial and multinomial theorems, inclusion-exclusion principles, pigeonhole principle, and their applications.

UNIT IV RECURRENCE RELATION

Generating functions, functions of sequences, calculation of coefficients of generating functions, recurrence relations, solving recurrence relations by substitution and generating functions, and characteristic root solutions of inhomogeneous recurrence relations.

UNIT V GRAPH THEORY

Graph representation, tree structures including definitions, binary trees, binary tree traversal, and binary search trees are covered. Topics also include depth-first search (DFS), breadth-first search (BFS), spanning trees, planar graphs, as well as basic concepts in graph theory and its applications. Other subjects discussed are isomorphism, subgraphs, multigraphs, Euler circuits, Hamiltonian graphs, and chromatic numbers.

Textbooks:

1. Discrete and Combinational Mathematics- An Applied Introduction-5th Edition Ralph. Grimaldi, Pearson Education
2. Discrete Mathematical Structures with applications to computer science Trembly J.P. & Manohar, TMH
3. Discrete Mathematics and its Applications, Kenneth H. Rosen, Fifth Edition. MH.

Reference

1. Discrete Mathematical structures Theory and application-Malik & Sen
2. Discrete Mathematics for Computer science, Garry Haggard and others, Thomson Logic

PYTHON PROGRAMMING			
Course Code:	**MDS109**	**Course Credits:**	**3**
Course Category:	**CC-L**	**Course (U / P)**	**P**
Course Year (U / P):	**1 P**	**Course Semester (U / P):**	**1 P**
No. of Lectures + Tutorials (Hrs/Week):	**03+00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45+00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Master the fundamentals of writing Python scripts.
2. Learn core Python scripting elements such as variables and flow control structures.
3. Discover how to work with lists and sequence data.
4. Write Python functions to facilitate code reuse.
5. Use Python to read and write files.

COURSE OUTCOMES

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

1. Problem solving and programming capability.
2. Explain basic principles of Python programming language
3. Implement database and GUI applications.
4. 4. Implement object-oriented concepts

UNIT I PYTHON BASICS, CONDITIONAL &LOOPS

Topics include installing Python and Python Notebook; understanding Python objects such as numbers, Booleans, strings, containers, and concepts of mutability; working with operators including arithmetic, bitwise, comparison, assignment, precedence, and associativity; and mastering control flow structures like if/else, if-Elif-else, loops (while, for), as well as using break, continue, and range functions

UNIT II STRING OBJECTS AND LIST OBJECTS

The basics of string objects, string methods, splitting and joining strings, and string format functions are all important concepts. Also essential are list object basics, list methods, using lists as stacks and queues, and understanding list comprehensions.

UNIT III TUPLES, SET, DICTIONARIES & FUNCTIONS

Tuples, sets, and dictionaries: object basics, dictionary object methods, and dictionary view objects. Functions: basic concepts, parameter passing, iterators, generator functions, lambda functions, as well as map, reduce, and filter functions.

UNIT IV OOPS CONCEPTS & WORKING WITH FILES

OOPS fundamentals, class and object creation, inheritance (including multiple inheritance), file operations (reading, writing, buffered I/O), and additional file methods.

UNIT V MODULES, EXCEPTION HANDLING & DATABASE PROGRAMMING

Using standard modules; creating new modules; handling exceptions with try-except; creating, inserting, retrieving, updating, and deleting table data. Data analysis: NumPy variables and manipulation, SciPy, introductory Pandas, descriptive analysis, Pandas I/O, manipulation, and group by.

Textbooks:

1. Headfirst Python 2e: A Brain-Friendly Guide Paperback – Illustrated, 16 by Paul Barry, Oreilly
2. Python: The Complete Reference Paperback – 20 March 2018 by Martin C. Brown (Author), TMH Publication

Reference

1. Let Us Python by Yash Avant Kanetkar, 1 January 2019, BPB publication
2. Python Programming, A modular approach, First Edition, By Pearson Publication by Taneja Sheetal and Kumar Naveen, 26

MS EXCEL LAB			
Course Code:	**MDS181**	**Course Credits:**	**2**
Course Category:	**CC-L**	**Course (U / P)**	**P**
Course Year (U / P):	**1P**	**Course Semester (U / P):**	**1P**
No. of Lectures + Tutorials +Lab (Hrs/Week):	**0+ 0+3**	**Mid Sem. Exam Hours:**	
Total No. of Lectures (L + T+P):	**0+ 0+ 10**	**End Sem. Exam Hours:**	**3**
COURSE OBJECTIVES			
1. Understand the basics and functions of MS excel.			

2. Clear understanding and use of data validations and templates.
3. Purpose of sorting and filtering features.
4. Use of reports in business organizations.
5. Purpose and advantage of charts for top management in any work place.
COURSE OUTCOME
At the end of the course the students should be able to:
1. Learn to understand the functions in Excel.
2. Understand the validations.
3. Make reports in excel.
4. Learn to work with pivot tables.
5. Learn how to make charts in MS excel.

List of Experiments

1. Create a new workbook and save the file with the name "Payroll". Enter the labels and values in the exact cells locations as desired. Use AutoFill to put the Employee Numbers into cells A6:A8. Set the columns width and rows height appropriately.
2. Create a workbook and enter relatable data of some employees. Calculate the Gross Pay for employee; enter a formula in cell E4 to multiply Hourly Rate by Hours Worked. Calculate the Social Security Tax (S.S Tax), which is 6% of the Gross Pay; enter a formula in cell F4 to multiply Gross Pay by 6%.
3. Create a workbook. Enter data as required. Calculate the Net Pay; enter a formula in cell G4 to subtract Social Security Tax from Gross Pay. Set the work sheet vertically and horizontally on the page.
4. Create a workbook having relatable data of sales of various models of cars in a showroom. Create a 3-dimensional column chart comparing sales data for men and women sales person.
5. Create a pie chart to compare the favourite films data for 15-25 year olds only (be careful not to include any unnecessary blanks rows or columns in your selected data).
6. Create a pivot table from this data, then use the filters within to view the average prices of holidays that have a Travel Method of Plane and a Resort Name that begins with the letter S.

Wise Owl Travel Agents

Country	Resort Name	No of Days	Travel Method	Price	Holiday ID
Australia	Great Barrier Reef	32	Plane	£750	I990AUS
Australia	Perth	28	Plane	£985	AUS112J
Chile	Santiago	21	Plane	£1,259	CH266H
England	London	3	Train	£69	I456UK
England	Bognor	1	Coach	£12	BG726H
France	Lyon	14	Plane	£399	A7995FR
France	Paris - Euro Disney	5	Train	£269	TH789FR
France	Paris - Euro Disney	3	Train	£125	TH788FR

7. Create an If function to calculate whether each movie was a flop or a success. Use the following criteria: If the profit was less than 100,000,000 then the movie is a flop otherwise the movie is a success.

8. Create an If function to rate the players based on the following criteria: If a player scores more than 15 points he has a High score otherwise he must try harder.
9. Convert this data into a pivot table and find the overall average speed of all rides that satisfy the following criteria: The Type is Steel, The Design is Sit Down, The Amusement Park has the word adventure somewhere in the title.

Roller Coaster	Amusement Park	Type	Design	Status	Opened	Speed (mph)
Air	Alton Towers	Steel	Flying	Operating	2002	46.6
Boomerang	Pleasure Island Family Theme Park	Steel	Sit Down	Operating	1993	47
Cobra	Paultons Park	Steel	Sit Down	Operating	2006	31.1
Colossus	Thorpe Park	Steel	Sit Down	Operating	2002	45
Corkscrew	Alton Towers	Steel	Sit Down	Operating	1980	40
Corkscrew	Flamingo Land Theme Park & Zoo	Steel	Sit Down	Operating	1983	40
Crazy Mouse	South Pier	Steel	Sit Down	Operating	1998	29.1
Crazy Mouse	Brighton Pier	Steel	Sit Down	Operating	2000	29.1
Enigma	Pleasurewood Hills	Steel	Sit Down	Operating	1995	34
Express	M&Ds Scotland's Theme Park	Steel	Sit Down	Operating	2006	28
Fantasy Mouse	Fantasy Island	Steel	Sit Down	Operating	2000	29.1

10. Create a worksheet and add desired data. Find TAX (If ITEM PRICE is less than 100, TAX is 50, otherwise it should be 100). TOTAL PRICE BEFORE TAX =NO. OF ITEMS * ITEM PRICE. TOTAL PRICE AFTER TAX = TOTAL PRICE BEFORE TAX + TAX. RATE (If TOTAL PRICE AFTER TAX > 3500 then the rate is “HIGH”, otherwise it is REASONABLE. Find Count of Items, Average of Taxes, Min Item PRICE and Max Item PRICE.

List of Experiments:

DATA STRUCTURE LAB			
Course Code:	**MDS181**	**Course Credits:**	**2**
Course Category:	**CC-L**	**Course (U / P)**	**P**
Course Year (U / P):	**1 P**	**Course Semester (U / P):**	**1 P**
No. of Labs (Hrs/Week):	**04**	**Mid Sem. Exam Hours:**	**-**
Total No. of Labs:	**10**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Introduce the concept of data structures through ADT including List, Stack, Queues.
2. To design and implement various data structure algorithms.
3. Introduce various techniques for representation of the data in the real world.
4. To develop application using data structure algorithms
5. Compute the complexity of various algorithms.

COURSE OUTCOMES

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

1. Select appropriate data structures as applied to specified problem definition
2. Implement operations like searching, insertion, and deletion, traversing mechanism etc. on various data structures.
3. Students will be able to implement Linear and Non-Linear data structures.
4. Implement appropriate sorting/searching technique for given problem.

1) Run time analysis of Fibonacci Series.
2) Study and Application of various data Structure.
3) Study and Implementation of Array Based Program
    a. Searching (Linear Search, Binary Search)
    b. Sorting (Bubble, Insertion, Selection, Quick, Merge etc)
    c. Merging
4) Implementation of Link List.
    a. Creation of Singly link list, Doubly Linked list
    b. Concatenation of Link list
    c. Insertion and Deletion of node in link list
    d. Splitting the link list into two link list

5) Implementation of STACK and QUEUE with the help of (a) Array (b) Link List
6) Implementation of Binary Tree, Binary Search Tree, Height Balance Tree
7) Write a program to simulate various traversing Technique
8) Representation and Implementation of Graph
    a. Depth First Search
    b. Breadth First Search
    c. Prims Algorithm
    d. Kruskal's Algorithms.

PYTHON PROGRAMMING LAB			
Course Code:	**MDS183**	**Course Credits:**	**2**
Course Category:	**CC-L**	**Course (U / P)**	**P**
Course Year (U / P):	**1 P**	**Course Semester (U / P):**	**1 P**
No. of Labs (Hrs/Week):	**04**	**Mid Sem. Exam Hours:**	**-**
Total No. of Labs:	**10**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. To introduce students to the basic concepts and techniques of Machine Learning
2. A general understanding of ML process models
3. To introduce students to the basic concepts and techniques of Machine Learning
4. Understanding of Python Programming and its module
5. Understanding of Deep Learning.

COURSE OUTCOMES

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

1. Installation of python and its module & python notebook.
2. Ability to apply Python principles and techniques.
3. Ability to design ML models and test and train data sets.
4. To Understand working of TensorFlow.
5. Ability to perform deep learning algorithms.

LIST OF PRACTICALS

1. To print the largest/smallest of two numbers
2. To input three numbers and print the greatest of all
3. To read two numbers x and n and print xn (first write with the use of operator and then write with the help of inbuilt function
4. To input the value of x and n and print the sum of the series: 1+x+x2+x3+x4+…….xn
5. To check if a number is a perfect number or not
6. Write a program to compute distance between two points taking input from the user (Pythagorean Theorem)
7. Write a program to count the numbers of characters in the string and store them in a dictionary data structure
8. To print factorial of a number using function
9. To print factorial of a number using recursion
10. To count no of vowels in a string that was given as input by user
11. Write a function to find all duplicates in the list.
12. Write a function unique to find all the unique elements of a list.
13. Write a program to perform addition of two square matrices
14. Write a program to perform multiplication of two square matrices
15. To read from a text file and print each word separated by # symbol, example #vipin #

DATA SCIENCE LAB			
Course Code:	**MDS185**	**Course Credits:**	**2**
Course Category:	**CC-L**	**Course (U / P)**	**P**

Course Year (U / P):	**1 P**	**Course Semester (U / P):**	**1 P**
No. of Lectures+ Tutorials+ Lab (Hrs/Week)	**04 hr**	**Mid Sem. Exam Hours:**	**--**
Total No. of Lectures (L + T+P):	**10**	**End Sem. Exam Hours:**	**3**

COURSE OBJECTIVES

1. Understand the basics and functions of MS excel.
2. Clear understanding and use of data validations and templates.
3. Purpose of sorting and filtering features.
4. Use of reports in business organizations.
5. Purpose and advantage of charts for top management in any workplace.

COURSE OUTCOME

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

1. Learn to understand the functions in Excel.
2. Understand the validations.
3. Make reports in excel.
4. Learn to work with pivot tables.
5. Learn how to make charts in MS excel.

List of Experiments

1) Create a workbook named “Payroll”. Enter labels and values in the specified cells. Use AutoFill for Employee Numbers in A6:A8. Adjust column widths and row heights as needed.

2) Create a workbook and input relevant employee data. Calculate each employee's Gross Pay by entering a formula in cell E4 that multiplies the Hourly Rate by the Hours Worked. Next, determine the Social Security Tax (S.S Tax), which is 6% of the Gross Pay, by entering a formula in cell F4 to multiply Gross Pay by 6%.

3) Create a workbook and enter the necessary data. In cell G4, subtract Social Security Tax from Gross Pay to calculate Net Pay. Set the worksheet orientation to both vertical and horizontal on the page

4) Create a workbook having relatable data of sales of various models of cars in a showroom. Create a 3-dimensional column chart comparing sales data for men and women salespeople.

5) Create a pie chart to compare the favourite films data for 15-25-year-olds only (be careful not to include any unnecessary blanks rows or columns in your selected data).

6) Create a pivot table from this data, then use the filters within to view the average prices of holidays that have a Travel Method of Plane and a Resort Name that begins with the letter S.

Wise Owl Travel Agents

Country	Resort Name	No of Days	Travel Method	Price	Holiday ID
Australia	Great Barrier Reef	32	Plane	£750	I990AUS
Australia	Perth	28	Plane	£985	AUS112J
Chile	Santiago	21	Plane	£1,259	CH266H
England	London	3	Train	£69	I456UK
England	Bognor	1	Coach	£12	BG726H
France	Lyon	14	Plane	£399	A7995FR
France	Paris - Euro Disney	5	Train	£269	TH789FR
France	Paris - Euro Disney	3	Train	£125	TH788FR

7) Create an IF function to classify each movie as a flop if the profit is less than $100,000,000; otherwise, label it a success.

8) Write an If function that rates players: if a player scores over 15 points, they have a high score; otherwise, they need to improve.

9) Convert this data into a pivot table to determine the overall average speed of rides that meet the following conditions: the type is Steel, the design is Sit Down, and the amusement park's name contains the word "adventure".

Roller Coaster	Amusement Park	Type	Design	Status	Opened	Speed (mph)
Air	Alton Towers	Steel	Flying	Operating	2002	46.6
Boomerang	Pleasure Island Family Theme Park	Steel	Sit Down	Operating	1993	47
Cobra	Paultons Park	Steel	Sit Down	Operating	2006	31.1
Colossus	Thorpe Park	Steel	Sit Down	Operating	2002	45
Corkscrew	Alton Towers	Steel	Sit Down	Operating	1980	40
Corkscrew	Flamingo Land Theme Park & Zoo	Steel	Sit Down	Operating	1983	40
Crazy Mouse	South Pier	Steel	Sit Down	Operating	1998	29.1
Crazy Mouse	Brighton Pier	Steel	Sit Down	Operating	2000	29.1
Enigma	Pleasurewood Hills	Steel	Sit Down	Operating	1995	34
Express	M&Ds Scotland's Theme Park	Steel	Sit Down	Operating	2006	28
Fantasy Mouse	Fantasy Island	Steel	Sit Down	Operating	2000	29.1

10) Create a worksheet and enter the required data. Calculate TAX: if ITEM PRICE is less than 100, TAX is 50; otherwise, TAX is 100. Compute TOTAL PRICE BEFORE TAX as NO. OF ITEMS multiplied by ITEM PRICE. Determine TOTAL PRICE AFTER TAX by adding TAX to TOTAL PRICE BEFORE TAX. Assign RATE: if TOTAL PRICE AFTER TAX is greater than 3500, the rate is “HIGH”; otherwise, it is “REASONABLE.” Additionally, find the count of items, the average tax amount, the minimum ITEM PRICE, and the maximum ITEM PRICE.

SEMESTER II

ANALYSIS AND DESIGN OF ALGORITHM			
Course Code:	**MDS102**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**P**
Course Year (U / P):	**1 P**	**Course Semester (U / P):**	**2 P**
No. of Lectures + Tutorials (Hrs/Week):	**03+00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45+00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. To introduce fundamental concepts of algorithm analysis.
2. To familiarize students with major algorithmic design paradigms.
3. To demonstrate the application of algorithms to solve real-world problems.
4. To develop the ability to design and analyse efficient algorithms.
5. This objective aims to provide an understanding of the limitations of algorithms and the challenges of solving computationally difficult problems.

COURSE OUTCOMES

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

1. Ability to analyse algorithm performance.
2. Competence in solving common algorithmic problems.
3. Understanding of computational complexity.
4. Proficiency in applying algorithmic design techniques.
5. Skill in evaluating and comparing algorithms.

UNIT I BASIC CONCEPT OF ALGORITHMS

Topics include definition and fundamentals of algorithms, mathematical foundations (sets, functions, relations, vectors, matrices), linear inequalities and equations, analysis frameworks, efficient algorithms, case analyses (average, best, worst), asymptotic notation, control statement analysis, loop invariants, and algorithm correctness.

UNIT II MATHEMATICAL ASPECTS AND ANALYSIS OF ALGORITHM

Mathematical analysis of non-recursive and recursive algorithms, including examples like Fibonacci numbers; empirical analysis and visualization of algorithms.

UNIT III ANALYSIS OF SORTING AND SEARCHING ALGORITHM

Sorting algorithms: Bubble, Selection, Insertion, Shell, and Heap sort; linear-time sorts: Bucket, Radix, and Counting sort. Also covers sequential search, brute-force string matching, divide and conquer, merge sort, binary search, binary trees (including traversal and properties), as well as depth-first and breadth-first search.

UNIT IV ALGORITHM TECHNIQUES

Transform and conquer, pre-sorting, balanced search trees (including AVL trees), heaps and heap sort, dynamic programming (Wars hall's and Floyd's algorithms), optimal binary search trees, greedy methods (Prim's, Kruskal's, Dijkstra's algorithms), and Huffman trees.

UNIT V ALGORITHM DESIGN METHODS

Algorithms of Backtracking: n-Queens, Hamiltonian circuit problem, subset-sum problem. Branch and bound assignment problem, knapsack problem, traveling salesman problem.

Textbooks:

1 . Anany Levitin, "Introduction to the Design and Analysis of Algorithm", Pearson Education Asia, 2003

2 . T.H. Carmen, C.E. Leiserson, R. L. Rivest and C. Stein, "Introduction to Algorithm", PHI Pvt. Ltd., 2001

3 . Sara Baase and Allen Van Gelder, "Computer Algorithms-Introduction to the Design and Analysis "Pearson Education Asia, 2003.

4. V. Aho, J.E. Hopcroft and J.D. Ullman, "the Design and Analysis of Computer Algorithms", Pearso Education Asia, 2003.

DATA STRUCTURE			
Course Code:	**MDS104**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**P**
No. of Lectures + Tutorials (Hrs/Week):	**03 + 00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45 + 00**	**End Sem. Exam Hours:**	**3 Hrs**
COURSE OBJECTIVES			

To emphasize the importance of appropriate data structure in developing and implementing efficient algorithms
Understand basic data structures such as arrays, stacks, queues, hash tables and linked list
To analyze the asymptotic performance of various algorithms
Solve problems using graphs, trees and heaps
Apply important algorithmic design paradigms and methods of analysis
COURSE OUTCOMES
At the end of the course the students should be able to:
1. Define basic static and dynamic data structures and relevant standard algorithms for them.
2. Select basic data structures and algorithms for autonomous realization of simple programs or program parts.
3. Determine and demonstrate bugs in program, recognise needed basic operations with data structures
4. Formulate new solutions for programming problems or improve existing code using learned algorithms and data structures
5. Evaluate algorithms and data structures in terms of time and memory complexity of basic operations.

UNIT I INTRODUCTION TO DATA STRUCTURES

Abstract data types, sequences as value definitions, data types in C, pointers in C, data structures and C, arrays in C, array as ADT, one dimensional array, Implementing one dimensional array, array as parameters, two dimensional array, structures in C, implementing structures, Unions in C, implementation of unions, structure parameters, allocation of storage and scope of variables, recursive definition and processes: factorial function, fibonacci sequence, recursion in C, efficiency of recursion, hashing: hash function, open hashing, closed hashing: linear probing, quadratic probing, double hashing, rehashing, extendible hashing.

UNIT II STACK, QUEUE AND LINKED LIST

Stack definition and examples, primitive operations, example -representing stacks in C, push and pop operation implementation, queue as ADT, C Implementation of queues, insert operation, priority queue, array implementation of priority queue, inserting and removing nodes from a list-linked implementation of stack, queue and priority queue, other list structures, circular lists: stack and queue as circular list - primitive operations on circular lists, header nodes, doubly linked lists, addition of long positive integers on circular and doubly linked list.

UNIT III TREES

Binary trees: operations on binary trees, applications of binary trees, binary tree representation, node representation of binary trees, implicit array representation of binary tree, binary tree traversal in C, threaded binary tree, representing list as binary tree, finding the Kth element, deleting an element, trees and their applications: C representation of trees, tree traversals, evaluating an expression tree, constructing a tree.

UNIT IV SORTING AND SEARCHING

General background of sorting: efficiency considerations, notations, efficiency of sorting, exchange sorts: bubble sort; quick sort; selection sort; binary tree sort; heap sort, heap as a priority queue, sorting using a heap, heap sort procedure, insertion sorts: simple insertion, shell sort, address calculation sort, merge sort, radix sort, sequential search: indexed sequential search, binary search, interpolation search.

UNIT V GRAPHS

Application of graph, C representation of graphs, transitive closure, Warshall's algorithm, shortest path algorithm, linked representation of graphs, Dijkstra's algorithm, graph traversal, traversal methods for graphs, spanning forests, undirected graph and their traversals, depth first traversal, application of depth first traversal, efficiency of depth first traversal, breadth first traversal, minimum spanning tree, Kruskal's algorithm, round robin algorithm.

Text Books:

1. Aaron M. Tenenbaum, Yeedidyah Langsam, Moshe J. Augenstein, 'Data structures using C', Pearson Education, 2004 / PHI.
2. E. Balagurusamy, 'Programming in Ansi C', Second Edition, TMH, 2003.

Reference books:

1. Robert L. Kruse, Bruce P. Leung Clovis L.Tondo, 'Data Structures and Program Design in C', Pearson Education, 2000 / PHI.

MACHINE LEARNING			
Course Code:	**MDS106**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**P**
Course Year (U / P):	**1 P**	**Course Semester (U / P):**	**2 P**
No. of Lectures + Tutorials (Hrs/Week):	**03+00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45+00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Understand the foundational theory of Machine Learning.
2. Formulate Machine Learning Problems.
3. Master Various Algorithms and Their Limitations.
4. Solve Problems of Moderate Complexity.
5. Apply and Optimize Models.

COURSE OUTCOMES

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

1. Design and Implement ML Solutions.
2. Analyse Algorithm Performance.
3. Apply ML Techniques to Real-World Problems.
4. Preprocess and Understand Data.

5. Develop Intelligent Systems.

UNIT I INTRODUCTION TO MACHINE LEARNING

What is Machine Learning, Types of Machine Learning, Supervised Learning, Unsupervised Learning, Reinforcement Learning, Applications of Machine Learning –Stock Price Prediction, Face Recognition, Handwriting Recognition, Image Recognition, Virtual Personal Assistants, Medical Diagnosis, Online Fraud Detection.

UNIT II SUPERVISED LEARNING (REGRESSION/CLASSIFICATION)

Basic methods: Distance-based methods, Nearest-Neighbours, Decision Trees, Naive Bayes Linear models: Linear Regression, Logistic Regression, Generalized Linear Models, Support Vector Machines, Nonlinearity and Kernel Methods.

UNIT III UNSUPERVISED LEARNING

Clustering: K-means/Kernel K-means, Self-Organizing Maps Dimensionality Reduction: PCA and kernel PCA Matrix relation and Matrix Completion Generative Models (mixture models and latent factor models).

UNIT IV ARTIFICIAL NEURAL NETWORKS

Biological Neurons and Biological Neural Networks, Artificial Neural Network, Types of Neural Network, Perceptron, History behind Perceptron, Importance of Perceptron, Working of Perceptron, Perceptron Learning, Perceptron Learning Rule, Perceptron Learning of AND & OR gate, XOR gate, Activation functions, Binary Activation function, Re LU, Sigmoid, Hyperbolic, SoftMax Activation function, Multilayer Perceptron's, Back propagation Neural Networks, and Feed-Forward Neural Networks, Applications and Future of Neural Networks.

UNIT V SELECTED TOPICS

Ensemble Methods (Boosting, Bagging, Random Forests), Sparse Modelling and Estimation, Modelling Sequence/Time-Series Data, Deep Learning and Feature Representation Learning, Recent trends in various learning techniques of machine learning and classification methods, Case studies in interdisciplinary domains.

Textbooks:

1. Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow by Aurélie Geron, O'Reilly publication.
2. An Introduction to Statistical Learning with Applications in R by Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, Springer publication (springer.com)

Reference Books

1. Python Machine Learning: Machine Learning and Deep Learning with Python, scikit- learn, and Tensor Flow 2, Publisher: Packet Publishing (December 12, 2019), Language: English, ISBN-10: 1789955750, ISBN-13: 978-1789955750
2. Machine Learning: The Absolute Complete Beginner's Guide to Learn and Understand Machine Learning From Beginners, Intermediate, Advanced, To Expert Concepts by Steven Samelson Publisher: Independently published (May 5, 2019) Language: English, ISBN-10: 1096853205, ISBN-13: 978-109685320

DATABASE MANAGEMENT SYSTEM

Course Code:	**MDS108**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**P**
Course Year (U / P):	**1P**	**Course Semester (U / P):**	**2P**
No. of Lectures + Tutorials (Hrs./Week):	**03 + 00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45 + 00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Provide a solid foundation in basic database concepts, terminology, and applications.
2. Teach Entity-Relationship (ER) modelling and relational model principles for designing databases to meet user requirements.
3. Equip students with the ability to write effective Structured Query Language (SQL) statements for data definition, manipulation, and retrieval.
4. Introduce database normalization techniques to optimize database design for data integrity and efficiency.
5. Explain the concepts of transaction processing , concurrency control, and recovery for reliable database management.

COURSE OUTCOMES

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

1. Ability to analyse user requirements and design logical ER diagrams and relational schemas for databases.
2. Ability to construct and execute complex SQL queries to perform data operations in relational databases.
3. Ability to apply normalization techniques to normalize database schemas, improving data integrity and reducing redundancy.
4. Understanding and ability to apply principles of transaction management and Familiarity with database storage structures and access techniques, including indexing and hashing , to optimize performance

UNIT I DATA BASE SYSTEM

Data base system vs file system, view of data, data abstraction, instances and schemas, data models, ER model, relational model, database languages, DDL, DML, database access for applications programs, data base users and administrator, transaction management, data base system structure, storage manager, query processor, history of data base systems, data base design and ER diagrams, beyond ER design entities, attributes and entity sets, relationships and relationship sets, additional features of ER model, concept design with the ER model, and conceptual design for large enterprises.

UNIT II RELATIONAL DATA BASE MODEL

Introduction to the relational model, integrity constraint over relations, enforcing integrity constraints, querying relational data, and logical database design, destroying altering tables and views relational algebra and calculus: relational algebra, selection and projection set operations, renaming, joins, division, relational calculus, tuple relational calculus, domain relational power of algebra and calculus.

UNIT III SQL QUERY

Examples of basic SQL queries, nested queries, correlated nested queries sets comparison operators, aggregative operators, NULL values, comparison using null values, logical connectivity's AND, OR and NOTR, impact on SQL constructs, outer joins, disallowing NULL values, complex integrity constraints in SQL triggers and active data bases.

UNIT IV NORMAL FORM

Problems caused by redundancy, decompositions, problem related to decomposition, reasoning about FDS, FIRST, SECOND, THIRD normal form, BCNF, forth normal form, fifth normal form, lossless join decomposition, dependency preserving decomposition, schema refinement in data base design, multi valued dependencies

UNIT V TRANSACTION MANAGEMENT

ACID properties, transactions and schedules, concurrent execution of transaction, lock based concurrency control, performance locking, and transaction support in SQL, crash recovery, concurrency control, Serializability and recoverability, lock management, lock conversions, dealing with dead locks, specialized locking techniques, concurrency without locking, crash recovery: ARIES, log, other recovery related structures, the write, ahead log protocol, check pointing, recovering from a system crash, media recovery, other approaches and interaction with concurrency control.

Textbooks:

1. Elmasri Navrate, Data Base Management System, Pearson Education, 2008.
2. Raghuraman Krishnan, Johannes Gehrke, Data Base Management Systems, TMH, 3rd edition, 2008.
3. C. J. Date, Introduction to Database Systems, Pearson Education, 2009.

Reference books

1. Silberschatz, Korth, Database System Concepts, McGraw hill, 5th edition, 2005.
2. Rob, Coronel & Thomson, Database Systems Design: Implementation and Management, 2009.

DATA MODELING AND VISULIZATION			
Course Code:	**MDS110**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**U**
Course Year (U / P):	**1P**	**Course Semester (U / P):**	**2P**
No. of Lectures + Tutorials (Hrs/Week):	**03 + 00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45+00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Understand basic of data handling.
2. Understand the various visualization technologies.
3. Understand and verify the underlying assumptions of a particular analysis.
4. Understanding & Visualizing Bar, grouped Plots & stacked plots.
5. Understand histograms, distribution analysis, statistics analysis.

COURSE OUTCOMES

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

1. Understand basics of Data Visualization.
2. Implement visualization of distributions.
3. Write programs on visualization of time series, proportions & associations.
4. Apply visualization on Trends and uncertainty.
5. Explain principles of proportions

UNIT I INTRODUCTION TO VISUALIZATION

Visualizing Data-Mapping Data onto Aesthetics, Aesthetics and Types of Data, Scales Map Data Values onto Aesthetics, Coordinate Systems and Axes- Cartesian Coordinates, Nonlinear Axes, Coordinate Systems with Curved Axes, Colour Scales-Colour as a Tool to Distinguish, Colour to Represent Data Values ,Colour as a Tool to Highlight, Directory of Visualizations- Amounts, Distributions, Proportions, x–y relationships, Geospatial Data.

UNIT II VISUALIZATION TECHNIQUES AND ASSOCIATIONS

Visualizing Amounts-Bar Plots, Grouped and Stacked Bars, Dot Plots and Heatmaps, Visualizing Distributions: Histograms and Density Plots- Visualizing a Single Distribution, Visualizing Multiple Distributions at the Same Time, Visualizing Distributions: Empirical Cumulative Distribution Functions and Q-Q Plots-Empirical Cumulative Distribution Functions, Plots, Visualizing Many Distributions at Once Visualizing Distributions Along the Vertical Axis, Visualizing Proportions-A Case for Pie Charts, A Case for Side-by-Side Bars, A Case for Stacked Bars and Stacked Densities, Visualizing Proportions Separately as Parts of the Total ,Visualizing Nested Proportions- Nested Proportions Gone Wrong, Mosaic Plots and Tree maps, Nested Pies ,Parallel Sets. Visualizing Time Series and Other Functions of an Independent Variable-Individual Time Series Multiple Time Series and Dose–Response Curves, Time Series of Two or More Response Variables

UNIT III PRINCIPLE OF PROPORTIONALINK

The Principle of Proportional Ink-Visualizations Along Linear Axes, Visualizations Along Logarithmic Axes,Direct Area Visualizations, Handling Overlapping Points-Partial Transparency and Jittering, 2D. Histograms, Contour Lines, Common Pitfalls of Colour Use-Encoding Too Much or Irrelevant Information, Using Nonmonotonic Colour Scales to Encode Data Values.

UNIT IV VISUALIZING UNCERTAINTY

Visualizing Trends-Smoothing, Showing Trends with a Defined Functional Form, Detrending and Time Series Decomposition, Visualizing Geospatial Data-Projections, Layers, Choropleth Mapping Cartograms, Visualizing Uncertainty-Framing Probabilities as Frequencies, Visualizing the Uncertainty of Point Estimates, Visualizing the Uncertainty of Curve Fits, Hypothetical Outcome Plots

UNIT V DATA HANDLING AND VISUALIZATION USING TABLEAU

Introduction to tableau, Tableau products suite, file type, Connection to data source, creating basic charts and graphs, handling filter data sorting grouping data in tableau, working with dates, waterfall chart and bump chart in tableau, heat, and tree map in tableau.

Textbooks

1. Claus Wilke, “Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures”1st edition, O’Reilly Media Inc, 2019.
2. Ryan Sleeper “Practical Tableau: 100 Tips, Tutorials, and Strategies from a Tableau Zen Master “O’Reilly Media

Reference Books

1. Tony Fischetti, Brett Lantz, R: Data Analysis and Visualization, O’Reilly ,2016
2. OssamaEmbarek, Data Analysis and Visualization Using Python: Analyse Data to Create Visualizations for BI Systems, Press, 2018 Joshua N. Milligan: Learning Tableau.
3.

THEORY OF COMPUTATION			
Course Code:	**MDS112**	**Course Credits:**	**3**
Course Category:	**CC/SEC**	**Course (U / P)**	**P**
Course Year (U / P):	**1 P**	**Course Semester (U / P):**	**2 P**
No. of Lectures + Tutorials (Hrs/Week):	**03+00**	**Mid Sem. Exam Hours:**	**1.5 Hrs**
Total No. of Lectures (L + T):	**45+00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Understand Abstract Models of Computation.
2. Analyse Formal Language Hierarchies.
3. Comprehend Decidability and Undecidability.
4. Apply Mathematical Proof Techniques.
5. Apply Mathematical Proof Techniques.

COURSE OUTCOMES

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

1. Students will be able to design finite automata (DFA, NFA) and pushdown automata (PDA) for specified regular and context-free languages.
2. Students will be able to construct context-free grammars to describe formal languages and understand different normal forms.
3. Students will be able to design Turing Machines and understand their capability to address computational problems.
4. Students will be able to differentiate between decidable and undecidable problems and understand the significance of computability.
5. Students will gain foundational knowledge essential for advanced topics in computer science, such as compilers, artificial intelligence, and formal verification

UNIT I INTRODUCTION

Introduction; alphabets, strings and languages; automata and grammars, deterministic finite automata (DFA)-formal definition, simplified notation: state transition graph, transition table, language of DFA, Nondeterministic finite Automata (NFA), NFA with epsilon transition, language of NFA, equivalence of NFA and DFA, minimization of finite automata, distinguishing one string from other, Myhill-Nerodia Theorem.

UNIT II REGULAR EXPRESSIONS

Regular expression (RE), definition, operators of regular expression and their precedence, algebraic laws for regular expressions, Kleen's theorem, regular expression to FA, DFA to regular expression, Arden theorem, non-regular languages, pumping lemma for regular languages. application of pumping lemma, closure properties of regular languages, decision properties of regular languages, FA with output: Moore and mealy machine, equivalence of Moore and mealy machine, applications, and limitation of FA.

UNIT III CFG

Context Free Gramma (CFG) and Context Free Languages (CFL): definition, examples, derivation, derivation trees, ambiguity in grammar, inherent ambiguity, ambiguous to unambiguous CFG, useless symbols, simplification of CFGs, normal forms for CFGs: CNF and GNF, closure properties of CFLs, decision properties of CFLs: emptiness, finiteness and membership, pumping lemma for CFLs.

UNIT IV PUSH DOWN AUTOMATA

Push Down Automata (PDA): description and definition, instantaneous description, language of PDA, acceptance by final state, acceptance by empty stack, deterministic PDA, equivalence of PDA and CFG, CFG to PDA and PDA to CFG, two stack PDA.

UNIT V TURING MACHINES (TM)

Basic model, definition and representation, instantaneous description, language acceptance by TM, variants of Turing machine, TM as computer of integer functions, universal TM, church's thesis recursive and recursively enumerable languages, halting problem, introduction to undecidability, undecidable problems about TMs. Post Correspondence Problem (PCP), modified PCP, introduction to recursive function theory.

Textbooks:

1. Hopcroft, Ullman, "Introduction to Automata Theory, Languages and Computation", Pearson Education
2. K.L.P. Mishra and Chandrasekaran, "Theory of Computer Science Automata, Languages and Computation", PHI

References

1. Mart. introduction to Languages and Theory of Computations", TMH
2. Papadimitriou, C. and Lewis, C.L., "Elements of the Theory of Computation", PHI

MACHINE LEARNING LAB			
Course Code:	**MDS184**	**Course Credits:**	**2**
Course Category:	**CC-L**	**Course (U / P)**	**P**
Course Year (U / P):	**1P**	**Course Semester (U / P):**	**2P**
No. of Lectures + Tutorials (Hrs./Week):	**04+ 00**	**Mid Sem. Exam Hours:**	
Total No. of Lectures (L + T):	**10 + 00**	**End Sem. Exam Hours:**	**3 Hrs**

COURSE OBJECTIVES

1. Introduce students to the basic concepts and techniques of Machine Learning
2. A general understanding of ML process models.
3. Introduce students to the basic concepts and techniques of Machine Learning
4. Understanding of Python Programming and its module
5. Understanding of Deep learning.

COURSE OUTCOMES

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

1. Installation of python and its module &I python notebook.
2. Ability to apply Python principles and techniques.
3. Ability to design ML models and test and train data set.
4. Understand working of TensorFlow.
5. Ability to perform deep learning algorithms.

LIST OF PRACTICALS

1. Installation of Python and python Notebook.
2. Implement- Data Types and Containers in Python.
3. A scatter plot is a diagram where each value in the data set is represented by a dot.
4. Implement Regression to find the relationship between variables.
5. Machine Learning - Train/Test- Evaluate Your Model
6. Implement polynomial regression - R-squared, Predict the future and Bad fit.
7. Implement - Machine Learning - Decision Tree.
8. Implement - Machine Learning - Random Forest.
9. Introduction to Deep Learning - Deep Learning basics with Python, TensorFlow and Kera’s p.1
10. Optimizing Models with Tensor Board - Deep Learning basics with Python, TensorFlow.

DATA BASE MANAGEMENT SYSTEM LAB			
Course Code:	**MDS186**	**Course Credits:**	**2**
Course Category:	**CC-L**	**Course (U / P)**	**P**
Course Year (U / P):	**1P**	**Course Semester (U / P):**	**2P**
No. of Lectures + Tutorials (Hrs/Week):	**04+ 00**	**Mid Sem. Exam Hours:**	**--**
Total No. of Lectures (L + T):	**10+ 00**	**End Sem. Exam Hours:**	**3**

COURSE OBJECTIVES

1. Understand the basics and functions of MS excel.
2. Clear understanding and use of data validations and templates.
3. Purpose of sorting and filtering features.
4. Use of reports in business organizations.
5. Purpose and advantage of charts for top management in any workplace.

COURSE OUTCOMES

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

1. Learn to understand the functions in Excel.
2. Understand the validations.
3. Make reports in excel.
4. Learn to work with pivot tables.
5. Learn how to make charts in MS excel.

LIST OF EXPERIMENTS:

1. Write the queries for Data Manipulation and Data Definition Language.
2. Write SQL queries using logical operations and operators.
3. Write SQL query using group by function. 4 Write SQL queries for sub queries, nested queries 5 Write SQL queries to create views.
4. Write an SQL query to implement JOINS.
5. Write a query for extracting data from more than one table.
6. Write a query to understand the concepts for ROLL BACK, COMMIT & CHECK POINTS.
7. Create tables according to the following definition.
8. Create table deposit (actno varchar2(5) ,cname varchar2(18), bname varchar2(18), amount number (8,2) ,a date date).
9. Create table branch (name varchar2(18), city varchar2(18));
10. Create table customers (cname varchar2(19), city varchar2(18));
11. Create table borrow(loanno varchar2(5), cname varchar2(18), b name varchar2(18), amount number
12. (8,2));
13. Retrieve all data from employee, jobs, and deposit.
14. Give details of account no and deposited rupees of customers having account opened between dates 01-01-06 and 25-07-06.
15. Display all jobs with minimum salary is greater than four thousand.
16. Display name and salary of employee whose department no is 20 Give alias name to name of employee.

Data Modelling and Visualization			
Course Code:	**MDS 110**	**Course Credits:**	2
Course Category:	**CC-L**	**Course (U / P)**	**P**
Course Year (U / P):	**2P**	**Course Semester (U/ P):**	**3P**
No. of Lectures + Tutorials (Hrs/Week):	**04 + 00**	**Mid Sem. Exam Hours:**	1.5
Total No. of Lectures (L + T):	**10**	**End Sem. Exam Hours:**	3

COURSE OBJECTIVES

1. Understand the importance of data visualization for business intelligence and decision making.
2. Know approaches to understand visual perception
3. Learn about categories of visualization and application areas
4. Familiarize with the data visualization tools
5. Gain knowledge of effective data visuals to solve workplace problems

COURSE OUTCOMES

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

1. Use Python, R and Tableau for data 5.
2. Apply data visuals to convey trends in data over time using tableau
3. Explore and work with different plotting libraries
4. Learn and create effective visualizations
5. Construct effective data visuals to solve workplace problems

List Of Experiments:

1. Getting Started - Tableau Workspace, Tableau terminologies, basic functionalities.
2. Connecting to Data Source – Connecting to Database, Different types of Tableau Joins.
3. Creating a View - formatting charts, adding filters, creating calculated fields and defining parameters.
4. Dashboard Design and Storytelling – Components of Dashboard, understanding how to place worksheets in Containers, Action filters and its types.
5. Introducing Power BI –Components and the flow of work. Power BI Desktop Interface The Report has five main areas.
6. Querying Data from CSV - Query Editor, Connecting the data from the Excel Source, Clean, Transform the data.
7. Creating Reports & Visualizations - Different types of charts, Formatting charts with Title, Colors.
8. Dashboards - Filters in Power BI, Formatting dashboards.
    1. Analysis of revenue in sales dataset:
9. Create a choropleth map (fill the map)to spot the special trends to show the state which has the highest revenue.
10. Create a line chart to show the revenue based on the month of the year.
11. Create a bin of size 10 for the age measure to create a new dimension to show the revenue.
12. Create a donut chart view to show the percentage of revenue per region by creating zero access in the calculated field.
13. Create a butterfly chart by reversing the bar chart to compare female & male revenue based on product category.
14. Create a calculated field to show the average revenue per state & display profitable & non-profitable state.
15. Build a dashboard
16. Analysis of GDP dataset:

a. Visualize the countries data given in the dataset with respect to latitude and longitude along with country name using symbol maps.
b. Create a bar graph to compare GDP of Belgium between 2006 – 2026.
c. Using pie chart, visualize the GDP of India, Nepal, Romania, South Asia, Singapore by the year 2010.
d. Visualize the countries Bhutan & Costa Rica competing in terms of GDP.
e. Create a scatter plot or circle views of GDP of Mexico, Algeria, Fiji, Estonia from 2004 to 2006.
f. Build an interactive dashboard

PROFESSIONAL ETHICS			
Course Code:	**ITV301**	**Course Credits:**	**3**
Course Category:	**CC**	**Course (U / P)**	**U**
Course Year (U / P):	**2U**	**Course Semester (U / P):**	**3 U**
No. of Lectures + Tutorials (Hrs/Week):3	**03 + 00**	**Mid Sem. Exam Hours:**	**1**
Total No. of Lectures (L + T):45	**45 + 00**	**End Sem. Exam Hours:**	**3**

COURSE OBJECTIVES

1.Describe the Professional ethics and morals
2.Explain the basic concepts of Global business Ethics
3.Explain the ethical theory
4.workplace ethics in changing domain of research
5.Improve the safety and moral leadership

COURSE OUTCOMES

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

1. Understand How to live life peacefully by following Professional ethics
2.Social responsibilities of Global business
3. Understand the Emotional Intelligence as social Experimentation.
4. Execute Safety, Responsibilities and Rights
5.Explain the principle of Professional ethics and Environmental responsibility.

UNIT I: Individual and Professional Ethics

Individual and Professional Ethics Introduction to Professional Ethics, Morals, Values and Ethics. Personal and Professional sense towards society, Code of Ethics by Morals, values. Integrity, right of equality, Academic integrity, Work Ethics, Society Service, Respect for others, living peacefully, Caring and Sharing, Honesty, courage, Cooperation commitment, Empathy, Self Confidence, Social Expectations.

UNIT II Business Ethics

Philosophical approaches to Business Ethics, ethical reasoning, ethical issues in business, Social Responsibility of Business conflict of interest, cultural relativism, Ethical leadership, Resisting un-ethical authority and domination, Global Business Ethics, Models of professional roles, Theories about right action, Self-interest, Customs and Religion, Uses of Ethical Theories.

UNIT III Psychological Approaches

Ethical Theories, Psychological and Philosophical approaches, Myths about Morality, conflict of interest in psychological perspective, Courage, Integrity, Emotional Intelligence as social Experimentation, Plagiarism, A balanced outlook on law, Challenges case study, Bhopal gas tragedy.

UNIT IV Workplace Ethics

Ethics in changing domains of Research, academic integrity, intellectual honesty, Role of Engineers and Managers, Ethical issues in Diverse workplace, competition, free will Confidentiality, employee rights, Intellectual property rights, discrimination Responsibilities and Rights. Role of confidentiality in moral integrity, Conflicts of interest, Occupational crime Professional rights Employee right.

UNIT V Safety, Responsibilities and Rights

Ecology, Engineering, Economy, Risk benefit analysis, Corporate social responsibility and Corporate Sustainability in India, Sustainability, Case Studies, Global Ethical Issues. Environmental Ethics, Business Ethics, Computer Ethics, Role in Technological Development and Moral leadership

Textbooks:

1. Dr. B.R. Ambedkar (1979). Dr. Babasaheb Ambedkar, writings and speeches, Volume 1. Education Dept., Govt.of Maharashtra, Pp.33-87.
2.Subramanian.R. Professional Ethics, Oxford Publication, 2013.
3. Nagarasan. R.S. Professional Ethics and Human Values. New Age International Publications, 2006.
4. Mike W Martin and Roland Schinzinger, Ethics in Engineering,4th edition, Tata McGraw Hill Publishing Company Pvt Ltd, New Delhi,2014

References Books:

1. Mike W Martin and Roland Schinzinger, Ethics in Engineering,4th edition, Tata McGraw Hill Publishing Company Pvt Ltd, New Delhi,2014.
2. Charles D Fleddermann, Engineering Ethics, Pearson Education/ Prentice Hall of India, New Jersey,2004.
3. Charles E Harris, Michael S Pritchard and Michael J Rabins, Engineering Ethics- Concepts and cases, Wadsworth Thompson Learning, United states,2005.