Deccan Education Society's

Fergusson College (Autonomous)
Pune - 411004

Curriculum

as per guidelines of

NEP-2020

for

M.Sc. - I

Industrial Mathematics with Computer Applications (IMCA)

With effect from Academic Year

2023-2024

Deccan Education Society's Fergusson College (Autonomous), Pune

Program Outcomes (POs)

PO1	**Disciplinary Knowledge:** Demonstrate comprehensive knowledge of the discipline that forms a part of a postgraduate programme. Execute strong theoretical and practical understanding generated from the specific programme in the area of work.
PO2	**Critical Thinking and Problem solving:** Exhibit the skill of critical thinking and understand scientific texts and place scientific statements and themes in contexts and also evaluate them in terms of generic conventions. Identify the problem by observing the situation closely, take actions and apply lateral thinking and analytical skills to design the solutions.
PO3	**Social competence:** Exhibit thoughts and ideas effectively in writing and orally; communicate with others using appropriate media, build effective interactive and presenting skills to meet global competencies. Elicit views of others, present complex information in a clear and concise way and help reach conclusions in group settings.
PO4	**Research-related skills and Scientific temper:** Infer scientific literature, build a sense of enquiry and able to formulate, test, analyse, interpret, and establish hypothesis and research questions; and to identify and consult relevant sources to find answers. Plan and write a research paper/project while emphasizing on academics and research ethics, scientific conduct and creating awareness about intellectual property rights and issues of plagiarism.
PO5	**Trans-disciplinary knowledge**: Create new conceptual, theoretical and methodological understanding that integrates and transcends beyond discipline-specific approaches to address a common problem.
PO6	**Personal and professional competence**: Perform independently and also collaboratively as a part of a team to meet defined objectives and carry out work across interdisciplinary fields. Execute interpersonal relationships, self-motivation and adaptability skills and commit to professional ethics.
PO7	**Effective Citizenship and Ethics**: Demonstrate empathetic social concern and equity centred national development, and ability to act with an informed awareness of moral and ethical issues and commit to professional ethics and responsibility.
PO8	**Environment and Sustainability**: Understand the impact of the scientific solutions in societal and environmental contexts and demonstrate the knowledge of and need for sustainable development.
PO9	**Self-directed and Life-long learning:** Acquire the ability to engage in independent and life-long learning in the broadest context of socio-technological changes.

PSO No.	Program Specific Outcomes (PSOs) Upon completion of this Programme the student will be able to
PSO1	**Academic competence** (i) Understand basic facts about Mathematics -annotations, terminology, geometrical figures, graphical displays, and its major subfields (Analysis, Algebra,Applied Mathematics and Statistics). (ii) Design and develop computer programs/computer-based systems in the areas related to algorithms, networking, web design, cloud computing, IoT and data analytics. (iii) Demonstrate unifying structures of Mathematics and the basic constructs of several programming languages and their application areas.
PSO2	**Personal and Professional Competence** (i) Apply mathematical solutions in a variety of contexts related to science, technology, business and industry, and carry out hands-on activities on several programming technologies as well as academic projects. (ii) Analyze the data by selecting and using appropriate mathematical formulae or techniques/programming technology stack in order to draw the relevant conclusion. (iii) Execute the problem-solving skills, equip with Mathematical modelling abilities and develop competent professionals who will be able to address challenges in the field of IT at global level.
PSO3	**Research Competence** (i) Apply advanced knowledge on topics in pure Mathematics and impart analytical skills to develop initiatives and come up with innovative ideas for R & Din various fields. (ii) Integrate the knowledge of Computer Science and Mathematics to solve and authenticate real-time data from various fields.
PSO4	**Entrepreneurial and Social competence** (i) Develop analytical skills required to get distinguishing employment opportunities in several fields including IT, Research and Development, teaching field and gain understanding about the ethical issues related to protection ofintellectual property - copyrights, trademarks and patents. (ii) Execute social competence including communication and effective interactionwith others, listening, speaking, observational skills and presenting skills.

Programme Structure

Semester	Paper Code	Paper Title	Credits
I	MTS-501	Mathematical Analysis	4
	MTS-502	Applied Linear Algebra	4
	MTS-503 **OR**	Theoretical Computer Science	4
	MTS-504	Mathematical Foundations of Computer Science	
	MTS-510	Research Methodology	4
	MTS-520	Practical I	2
	MTS-521	Practical II	2
		Total Semester Credits	**20**
II	MTS-551	Abstract Algebra	4
	MTS-552	Probability & Statistics	4
	MTS-553 **OR**	Graph Theory and its Applications	4
	MTS-554	Numerical Analysis	
	MTS-560	OJT/FP	4
	MTS-570	Practical III	2
	MTS-571	Practical IV	2
		Total Semester Credits	**20**
		Total PG-I Credits	**40**

Teaching and Evaluation (Only for FORMAL education courses)

Course Credits	No. of Hours per Semester Theory/Practical	No. of Hours per Week Theory/Practical	Maximum Marks	CE 40 %	ESE 60%
1	**15 / 30**	**1 / 2**	**25**	**10**	**15**
2	**30 / 60**	**2 / 4**	**50**	**20**	**30**
3	**45 / 90**	**3 / 6**	**75**	**30**	**45**
4	**60 / 120**	**4 / 8**	**100**	**40**	**60**

Eligibility: As per the rules and regulations of Savitribai Phule Pune University (SPPU)

SEMESTER I

Title of the Course and Course Code	MTS-501 Mathematical Analysis	Number of Credits: 04
Course Outcomes (COs) **On completion of the course, the students will be able to:**		
CO1	Retrieve basic concepts in Metric Topology	
CO2	Interpret properties of Continuous and Differentiable functions	
CO3	Implement Mean Value theorems to real applications and discuss its use	
CO4	Identify series expansions of general continuously differentiable functions for computational needs	
CO5	Evaluate Riemann Integrals and identify its applications	
CO6	Formulate the theory of sequences and series of functions to use it in several applications	

Unit No	Title of the Unit and Contents
1	**Basic Topology** Finite, Countable & Uncountable sets, Metric Spaces, Compact Sets, Perfect Sets Connected Sets
2	**Numerical Sequences and Series** Convergent sequences, sub sequences, Cauchy sequences, Special sequences, Series, Series of non-negative terms, The number e, Root and Ratio tests, Power series, Absolute Convergence
3	**Continuity** Limits of functions, Continuous functions, Continuity and Connectedness, Continuity and Compactness, Monotonic functions, Types of discontinuities
4	**Differentiation** Derivatives and Mean Value Theorems, Taylors theorem, convex functions, Cauchy form of Remainder, Differentiation of Vector Valued functions
5	**Riemann Integral** Concept of Partitions, Refinements, Upper and lower sums, Existence of Integral, Properties of Riemann Integral (without proof), Integral and Differentiation, Fundamental Theorem of Integral Calculus
6	**Uniform Convergence** Concept of uniform convergence, Uniform convergence and continuity, Uniform convergence and integration, uniform convergence and differentiation, Example of a continuous nowhere differentiable function, Equicontinuous family, Stone Weierstrass Theorem (Statement only)

Learning Resources:

1) Walter Rudin, Principles of Mathematical Analysis, McGrawhill India (3rd Edition);
2) Ajit Kumar and S. Kumerasan, A First Course on Real Analysis, CRC Press.
3) J. E. Marsden and M. J. Hoffman, Elementary Classical Analysis, 2nd Edition, W. H. Freeman;

Title of the Course and Course Code	MTS-502 Applied Linear Algebra	Number of Credits: 04

Course Outcomes (COs) On completion of the course, the students will be able to:	
CO1	Describe the key concepts associated with vector spaces
CO2	Illustrate mathematical properties of inner products and solve examples to justify the properties
CO3	Apply Gram-Schmidt process to find an orthogonal basis to use it in applications
CO4	Identify the role of Eigenvalue and Eigenvectors in Matrix Decompositions
CO5	Determine the concept of Linear Transforms to solve real life problems
CO6	Formulate the concept of Matrix Decompositions to discuss the problems in Engineering and Data Science

Unit No.	Title of the Unit and Contents
1	**Vector Spaces** Real Vector Spaces, Subspaces, Span , Linear Independence, Basis and dimension, Fundamental Matrix Subspaces, Kernel, Image, Cokernel, Coimage, Fundamental Theorem of Linear Algebra relating to these subspaces
2	**Inner products, Norms and Orthogonality** Inner products, Inner products on function spaces, Triangle and Cauchy Schwarz Inequality, Norms, Matrix norms, Positive Definite Matrices, Gram Matrices, Cholesky Factorization, Orthonormal Basis, Gram Schmidt Process, QR Factorization, Orthogonal Projections, Orthogonality of Fundamental Matrix subspaces, Orthogonal Polynomials
3	**Linearity** Linear Operators, Space of Linear Functions, Dual Spaces, Composition and Inverses of Linear Transforms, Change of Basis, Affine Transforms, Isometry, Adjoints, Self-Adjoint Operators, Positive definite Linear functions and Minimization
4	**Eigenvalues & Eigenvectors** Introduction, Gerschgorin Circle theorem, Eigenbases, Diagonalization, Invariant Subspaces, Eigenvalues of Symmetric Matrices, Spectral Theorem (Statement only), Jordan Canonical form, Singular Value Decomposition, Matrix Psuedo Inverse, Principal Component Analysis
5	**Applications** Applications of Eigenvalues and Eigenvectors to discrete dynamical systems, Markov Chains, Minimization of Quadratic functions, Closest points, Least Squares, Discrete Fourier Transform and Fast Fourier Transform Algorithm, Matrix Exponential and its applications to Geometry.

Learning Resources:

1) Peter Olver, Chehrzad Shakiban, Applied Linear Algebra, Springer (2nd Edition)
2) Gilbert Strang, Linear Algebra and its applications (4th Edition)
3) David Lay, Linear Algebra with Applications, Pearson Edition
4) Kenneth Hoffman and Ray Kunze, Linear Algebra, PHI India Private Limited
5) M. L. Artin, Algebra, PHI India Private Limited
6) A. G. Hamilton, Linear Algebra, Cambridge University Press
7) Linear Algebra and Learning from Data, Cambridge Press
8) Walter Rudin, Principles of Mathematical Analysis, McGraw Hill India (3rd Edition)
9) Ajit Kumar and S. Kumerasan, A First Course on Real Analysis, CRC Press.
10) J. E. Marsden and M. J. Hoffman, Elementary Classical Analysis, 2nd Edition, W. H. Freeman

Title of the Course and Course Code	MTS-503 Theoretical Computer Science	Number of Credits: 4
Course Outcomes (COs) **On completion of the course, the students will be able to:**		
CO1	Define basic concepts of automata theory and describe various forms of grammar to know functioning, capabilities and limitations of computers.	
CO2	Explain and construct finite state systems and Context Free Grammar for the given language. Construct regular expressions to recognize patterns and PDA	
CO3	Apply various techniques and algorithms to transform computing mod and grammar.	
CO4	Analyze and simplify CFG, classify various grammars according to the Chomsky hierarchy.	
CO5	Evaluate various classes of problems, grammar, languages, and language recognizer machines.	
CO6	Integrate the concepts of finite automata, regular expression and context free grammar in order to generate regular expression for regular languages to recognize patterns.	

Unit. No.	Title of Unit and Contents
I	**Overview of theory of mathematics**: Sets, Functions, Logical statements, Proofs, relations, languages, Mathematical induction, strong principle, Recursive definitions
II	**Introduction to Theory of Computations:** Fundamental concepts, history, Applications, Phases of Compiler, basic terminologies used in theory of computation: automata, symbol, alphabet, String
II	**Introduction to Regular Languages and Finite Automata:** Regular expressions, regular languages, applications, Automata with Output-Moore machine, Mealy machine, Finite automata, memory requirement in a recognizer, definition, union, intersection and complement of regular language, Non Determinism Finite Automata, Conversion from NFA to FA, Non Determinism Finite Automata Conversion of NFA to NFA and equivalence of three Kleene's Theorem, Minimization of Finite automata Regular and Non-Regular Languages - pumping lemma.
III	**Grammar**: Introduction to Grammar, Definition, elements of grammar, Application areas and comparison of different grammars, Comparison
IV	**Context Free Grammar:** Importance of Context free grammar**,** Unions Concatenations and Kleen's of Context free language Regular grammar, Derivations and Languages, Relationship between derivation and derivation trees, Ambiguity Unambiguous CFG and Algebraic Expressions Bacos Naur Form (BNF), Normal Form - CNF

V	**Pushdown Automata, CFL and NCFL:** Pushdown Automata, CFL And NCFL: Definition, deterministic PDA, Equivalence of CFG and PDA, Pumping lemma for CFL, Intersections and Complements of CFL, Non-CFL
VI	**Turing Machine (TM):** TM Definition, Model of Computation and Church Turning Thesis, computing functions with TM, Combining TM, Variations of TM, Non-Deterministic TM, Universal TM, Recursively and Enumerable Languages, Context sensitive languages and Chomsky hierarchy
VII	**Computable Functions**: Partial, total, constant functions, Primitive Recursive Functions, Bounded Mineralization, Regular function, Recursive Functions

Learning References:

1. John Hopcroft, Rajeev Motwani and Jeffrey Ullman, "Introduction to Automata theory, Languages and computation", 3rd edition Pearson Education, 2009.
2. Shirish S. Sane, "Theory of Computer Science", 2nd edition, 2007, Technical publication
3. Daniel I. A. Cohen, John Wiley & Sons, "Introduction to Computer Theory", 2nd edition, 2009
4. John E. Hopcroft and Jeffrey Ullman, "Introduction to Automata theory, Languages andcomputation" Narosa Publishing House,1979.

Title of the Course and Course Code	MTS-504 Mathematical Foundations of Computer Science	Number of Credits: 04

Course Outcomes (COs) On completion of the course, the students will be able to:	
CO1	State basic concepts in Counting and solve examples
CO2	Discuss the concepts of Generating functions
CO3	Apply the concepts of Recurrence relations to model problems in Computer Science
CO4	Analyze the concept of Inclusion Exclusion to count favorable union of events
CO5	Determine the Pigeons and Holes to apply the pigeon hole principle
CO6	Create a set of Boolean identities to model applications in Computer Science

Unit No.	Title of the Unit and Contents
1	**General Counting for Arrangements and Selections** Two basic counting principles, Simple Arrangements and Selections, Arrangements and selections with repetitions, Distributions, Binomial Identities
2	**Generating functions** Generating function mod, Calculating coefficients with generating functions, Partitions, Exponential Generating functions.
3	**Recurrence Relations** Recurrence Relation modulus, Divide and Conquer relations, Solution to Linear Recurrences, Solution to Inhomogeneous Recurrence relations, Solutions with Generating Functions.
4	**Inclusion Exclusion Principle** Counting with Venn Diagrams, Inclusion Exclusion Formula, Restricted Positions and Rook Polynomials
5	**Pigeonhole Principle** Pigeonhole principle and its related problems
6	**Lattice Theory / Boolean Algebra** Poset, Hasse Diagram, Lattices, Complemented Lattices, Bounded and Distributive Lattices, Introduction to Boolean variable, Boolean functions of degree n, Boolean identities, Definition of Boolean Algebra, Representation of Boolean Functions, Minterm Maxterm Disjunctive Normal form, Conjunctive Normal Form, Applications to Computer Science and some practical applications

Learning Resources

1) Alan Tucker, Applied Combinatorics, John Wiley & Sons
2) Course Notes for MATH 239, Introduction to Combinatorics, University of Waterloo
3) Chaun Chong Chen, Khee - Meng, Koh, Principles and Techniques in Combinatorics,
4) Vijay K Garg, Introduction to Lattice Theory with Computer Science Applications, Wiley Publications

Title of the Course and Course Code	MTS-510 Research Methodology	Number of Credits: 4
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Define basic concepts of Research	
CO2	Explain significance of statistical analysis, Exploratory and confirmatory research along with various case studies.	
CO3	Applying various statistical methods and software to solve the case studies in various subjects.	
CO4	Analyzing data by collecting it using various methods such as questionnaires.	
CO5	Evaluate various classes of problems, grammar, languages, and language recognizer machines.	
CO6	Writing research paper / thesis / presentation / research proposal.	

Unit No.	Title of the Unit and Contents
1	**Introduction to Research Methodology**: History of research. Indian, Egyptian, Greek ideas methodologies and research in agriculture, chemistry, metallurgy, medical. Ancient Indian research methodology applications.
2	**Overview of Research Types and Statistical Techniques**: Statistical analysis and its significance, Exploratory and confirmatory research, Planned and ad-hoc methods of data collection, Non-response and methods of recovering the missing response, Various softwares for statistical analysis. The module will consist of case studies of the research performed in various subjects using statistical methods, Error and noise analysis, curve fitting
3	**Data Collection and Research Topic**: Creating questionnaire. Data analysis from answers, Selection of research topic (case study based). Selection of research topic (case study based)
4	**Identification of Research Problem**: Literature search, selection of research topic (case study based), maintaining laboratory records (case study based). Safety in Laboratories, Ethical considerations, effective verbal and nonverbal communication, field data collection, safety in field.
5	**Interpretation and Report Writing:** Writing research paper and/or thesis, making a presentation, writing a research proposal, and patents in Science, technology.
6	**Thesis Writing, Presentation and Visualization Tools:** Latex, Beamer, Introduction to Minitab.

Learning References:

1) 'History of the Scientific Methods' by Martin Shuttleworth, https://explorable.com/history of-the- scientific-method.
2) 'The Statistical Analysis of Experimental Data' by, John Mandel, ISBN: 0486646661

Title of the Course and Course Code	MTS-520 (Practical I) Programming Language I	Number of Credits: 02
Course Outcomes (COs) **On completion of the course, the students will be able to:**		
CO1	Recall the basic concepts of the C programming language.	
CO2	Discuss basic concepts of computers, algorithms and algorithmic thinking.	
CO3	Apply stack for implementation of function calls and parameter passing mechanisms.	
CO4	Analyze and compare usage of arrays, strings, structures and files.	
CO5	Determine the solution for a given problem by distinguishing various memory allocation methods.	
CO6	Write C programs to validate the specifications.	

Unit No.	Title of Unit and Contents
1	**Introduction to Programming** Program and Programming, Programming Languages, Types of Software, Operating Systems, Basic Linux Commands and vi Editor, Compiler, Interpreter, Loader and Linker Introduction to algorithms, flow charts, Background of internal working of compilers.
2	**Basics of C** History and Features of C, Importance of C, Backslash Characters, Character set, Constants, Format Specifiers, Identifiers, Keywords, Variables, Data Types, Comments, const Qualifier, The Structure of a C Program, Building an Executable Version of a C Program, Debugging a C Program, Programming Examples
3	**Applications of C Programming** Demonstration of an application developed using C Note: This unit will not be considered for an assessment of students
4	**Control Statements** Decision Making Statements: if, if-else, switch Loop Control Structures: while, do. while, for Keywords- break and continue, exit () Function, return Statement, Programming Examples
5	**Operators and Expressions** Arithmetic Operators, Increment and Decrement Operators, Relational Operators, Logical Operators, Bitwise Operators, Assignment Operators, Conditional Operator, size of Operator, Comma Operator, Type Casting Operator, Other Operators, Precedence and Order of Evaluation, Programming Examples
6	**Input and Output** Unformatted I/O, Character I/O, String I/O, Formatted I/O, Programming Examples
7	**Functions** Concept, Usage of a Function, Advantages, Function Prototype, Function example, Types of Function, Call by Value and Call by Address, Recursion, Library Functions, Local variable, Global Variable, Storage classes (automatic, static, register, external), Programming Examples

8	**Array** Array Declaration, Initialization, Types of Arrays (1-D, 2-D and Multidimensional), Passing Arrays to Functions, Programming Examples
9	**Pointers** Pointer Declaration and Initialization, Dereferencing Pointers, void Pointer, Pointer Arithmetic, Pointer to Pointer, Arrays and Pointers, Functions and Pointers, Passing Pointers to Function, Function Returning Pointer, Pointer to Function, Dynamic Memory Allocation, Programming Examples
10	**String Handling** Declaration and Initialization, Reading and Writing Strings, Standard String Library Functions, Array of Pointers to String, Command Line Arguments, Programming Examples
11	**Structures and Unions** Overview of Structures, Defining and Using a Structure, typedef Keyword, Nested Structures, Passing Structure to Function, Structure and Pointer, Union, Difference between Structure and Union, Programming Examples
12	**Pre-Processor Directives** Pre-Processor Directives, #define Macro, Conditional Compilation, Pre- defined Macros, #include and Header Files, Programming Examples
13	**File Handling** What is a Stream? Opening and Closing of Files, File Opening Modes, Writing and Reading in Text Format, Writing and Reading in Binary Format, Programming Examples

Learning Resources:

Include Reference Books / e-resources / journals /any other learning material

1. Kernighan Brian W., Ritchie Dennis M., The C Programming Language, PHI Learning Pvt. Ltd., 2nd Edition, 2010
2. Schildt Herbert, C: The Complete Reference, Tata McGraw Hill, 4th Edition, 2006
3. Kanetkar Yashavant, Pointers in C, BPB Publications, 4th Edition, 2013
4. Kanetkar Yashavant, Test your C Skills, BPB Publications, Rev. Edition, 2008

Practical Lab II: Understanding of Data and Databases

Title of the Course and Course Code	MTS-521 (Practical II) Understanding of Data and Databases	Number of Credits: 02
Course Outcome (COs) **On completion of the course, the students will be able to:**		
CO1	Describe and discuss the fundamentals of data, major components of Databases	
CO2	Interpret a problem and recognize the computing requirements appropriate to its solution	
CO3	Implement appropriate database for computer-based systems according to the user requirements, appropriate syntax to write SQL commands to perform various RDBMS operations and NoSQL operations	
CO4	Analyze a problem to find out the computing requirements appropriate to its solution.	
CO5	Discuss the purpose of query processing for optimized solution.	
CO6	Design data requirements of an application with the help of conceptual modelling tools.	

Unit No.	Title of Unit and Contents
I	**Introduction to data and databases** Data – Introduction / Concept, introduction to data structures, Introduction to databases, Significance of Database, System Applications, Data Independence, Entities and their Attributes, Relationship and Relationship Types, E-R Diagram, Data types, Creating tables (without keys)
2	**Overview of RDBMS (PostgreSQL)** Relational Database Management System, RDBMS Properties, Maintaining Integrity and Defining Data Integrity, Integrity Rules and Integrity Constraints, Relational Integrity Rules, Creating tables (with keys)
3	**SQL** Types of SQL, DDL, DML, Basic queries in SQL Single table, Deletion-Insertion and Update in SQL, Simple queries (with insert, delete, and update), Multi table Retrievals, Nested queries (with foreign key and using multi tables), Aggregate-Functions, Joins, GROUPBY - HAVING clause, Nested Sub queries
4	**Views and Stored Functions** View definition, how to write view and its execution, Function definition, how to write function and its execution, Solving some problems with function
5	**Cursors** Cursor definition, how to write cursor and its execution, Solving some problems with cursor, Introduction to triggers and some demo examples
6	**NoSQL** Introduction, Why NoSQL? RDBMS Vs NoSQL, Features of NoSQL, Types: Key-value Pair Based, Column-oriented, Graphs based, Document-oriented, ACID and BASE for reliable database transactions, CAP theorem
7	**Introduction to MongoDB** Basics, Installation and Set Up, CRUD operations
8	Working with online database (Demonstration)

Learning Resources:

1. Henry F. Korth, Abraham Silberschatz, S. Sudarshan Database System Concepts, ISBN:9780071289597, Tata McGraw-Hill Education
2. Korry Douglas, PostgreSQL, ISBN:9780672327568
3. John Worsley, Joshua Drake Practical PostgreSQL (B/CD), ISBN: 9788173663925Shroff / O'reilly
4. Joshua D. Drake, John C Worsley Practical Postgresql, O'Reilly
5. Richard Stones, Neil Matthew Beginning Databases with PostgreSQL, From Noviceto Professional, 2nd Edition

E-Resources -

1. https://www.postgresql.org/docs/current/
2. https://www.mongodb.com/docs/atlas/

Semester II

Title of the Course and Course Code	MTS-551 Abstract Algebra	Number of Credits: 04
Course Outcomes (COs) **On completion of the course, the students will be able to:**		
CO1	Define basic concepts of Group theory with focus on axiomatic theory and key concepts associated with groups.	
CO2	Articulate the fundamental concepts of abstract algebra such as groups and rings and their role in modern Mathematics and applied contexts.	
CO3	Demonstrate capacity for mathematical reasoning through analyzing, proving, and explaining concepts from abstract algebra. Describe the structure of certain finite groups using Sylow's theorems.	
CO4	Compare different types of groups and rings. Explain the notion of rings, ideals.	
CO5	Justify theorems based on group theory, ring theory and articulate problem solving techniques based on them.	
CO6	Construct the structure of finite fields.	

Unit No.	Title of Unit and Contents
I	**Groups, Subgroups and Cyclic Groups:** Definition and Examples of Groups; Properties of Groups; Order of a group; Order of an element in group; Subgroups; Subgroup Tests. Cyclic Groups; Properties of Cyclic Groups; Classification of Subgroups of Cyclic Groups. **Permutation Groups- Isomorphism:** Definition and notation; Cycles; Properties of Permutations; Even and odd permutations; Alternating Group of degree n., Isomorphism of Group; Properties of Isomorphisms; Cayley's Theorem; Automorphisms. **Cosets, Lagrange's Theorem and Normal subgroups, Homomorphisms:** Definition and properties of Cosets; Lagrange's Theorem and consequence; Normal Subgroups; Factor Groups; Application of Factor Groups; Group Homomorphisms; Definition and examples; Properties of Homomorphisms; First Isomorphism Theorem. **Sylow Theorems (Without Proofs):** Fundamental Theorem of Finite Abelian Groups; Isomorphism Classes of Abelian Groups; Proof of the Fundamental Theorem. 5.2 Conjugacy Classes; Class Equation; The Sylow Theorems; Applications of Sylow's Theorems.

II	**Rings:** Definitions and examples, simple properties of rings, Commutative rings, ring with unity, integral domain, field, skew field, definitions examples and interrelationship between them. **Subring:** Definition, Examples, Properties. Characteristic of an integral domain. **Ideals and Factor Rings:** Definitions & Examples, Properties of ideals, Prime Ideals, Maximal Ideals, Quotient rings **Homomorphism and Isomorphism of rings:** Definition and examples, properties of ring homomorphisms, fundamental theorem of ring homomorphism and its applications. **Euclidean rings:** Polynomial rings F[X] over a field F, F[X] is Euclidean ring, irreducible polynomial over a field, polynomials over a field of rationals, Gauss lemma and Eisenstein's criterion for irreducibility, Construction of finite fields.

Learning Resources:

1) Joseph Gallian, Contemporary Algebra, 9th Edition, Narosa Publishing House.
2) David S. Dummit, Richard M. Foote, Abstract Algebra, 2nd Edition, John Wiley and Sons (Indian Edition)
3) J. B. Fraleigh, Abstract Algebra, 7th edition
4) I. S. Luthar and I. B. S. Passi, Algebra (Volume 1) Groups (Narosa Publishing House)
5) I. N. Herstein, Topics in Algebra (Wiley-Eastern Ltd.)
6) M. Artin, Algebra (Prentice Hall)
7) N. S. Gopala Krishnan, University Algebra (Wiley-Eastern Ltd.)
8) C. Musili, Rings and Modules, 2nd Revised Edition, Narosa Publishing House.

Title of the Course and Course Code	MTS-552 Probability & Statistics	Number of Credits: 04

Course Outcomes (COs) On completion of the course, the students will be able to:	
CO1	State basic concepts in probability theory and solve examples
CO2	Discuss concepts of distributions and their parameters
CO3	Apply the concepts of distributions to model a given data and check the goodness of fit
CO4	Analyze the concept of correlation and Regression for a given data
CO5	Determine the statistical significance of Null Hypothesis with inference based tests
CO6	Create methodologies to understand Random processes.

Unit No	Title of the Unit and Contents
1	**Introduction to Probability** Classical definition of probability, Mutually Exclusive and Independent Events, Inclusion Exclusion Principle, Conditional Probability, Bayes theorem and related problems
2	**Distribution Theory** Concept of Random variables (discrete & continuous) distribution function, Joint distributions, Marginal distributions, Conditional distributions, Expected value and Variance of a random variables. Also discussion on specific distributions which include Binomial, Poisson, Uniform, Exponential, Normal distributions with specific examples
3	**Special Results** Discussion on Simple Random Walks, Union Bound Inequality, Markov and Chebyshev Inequalities, Chernoff Bounds, Cauchy Schwarz Inequality, Jensen Inequality, Law of Large Numbers, Central Limit theorem and problems related to all of these concepts
4	**Statistical Inference: Classical Methods** Introduction to Random Sampling, Concept of Point Estimation, Point Estimators of Mean and Variance, Maximum Likelihood Estimation, Asymptotic properties of MLE's, Concept of Interval Estimation, Finding Interval Estimators, Confidence intervals for normal Samples, General Concepts in Hypothesis testing, P values and related problems
5	**Correlation and Simple Linear Regression** Concept of Pearsonian Correlation, Simple Linear Regression Model, Method of Least Squares, Various Assumptions of the Simple Linear Regression Model
6	**Introduction to Random processes** Basic Concepts, pdfs and cdfs, Mean and Correlation functions, Multiple Random Processes, Stationary Processes, Gaussian Random Processes

Learning Resources:

1) Sheldon Ross, Introduction to Probability & Statistics for Scientists and Engineers, Academic Press
2) Elliot Tannis and Robert Hogg, Probability & Statistical Inference, Pearson Edition
3) George Casella and Roger Berger, Statistical Inference, Cenage Learning

Title of the Course and Course Code	MTS-553 Graph Theory and Its Applications	Number of Credits: 04
Course Outcomes (COs) On completion of the course, the students will be able to:		
CO1	Define basic concepts of Graph theory with focus on key concepts associated with graphs.	
CO2	Illustrate various mathematical properties of graphs and solve examples to justify the properties.	
CO3	Apply basic graph theory, minimal weighted spanning tree algorithms, graph coloring algorithms.	
CO4	Compare different types of graphs and operations on graphs.	
CO5	Determine algorithms of fusion, matching algorithms, shortest path algorithms Formulate	
CO6	Formulate Matrix representation of graphs, shortest paths for various graphs	

Unit No.	Title of Unit and Contents
I	**Graph Theory:** **Graph:** Definition, vertex, edge, Terminal vertices, self-loop, parallel edges,incidence, adjacent, degree of vertex, isolated vertex, pendent vertex, null graph, hand shaking lemma, regular graph, bipartite graph, complete graph, complete bipartite graph. **Matrix Representation:** Incidence matrix, adjacency matrix, properties.Subgraph, Isomorphism and examples of isomorphic graphs. **Operations on graphs:** Union, intersection, deletion of vertex, deletion ofedge, ring sum, fusion. **Connected graphs:** Walk, paths, circuit, Theorems on connected graphs. **Euler graph:** Definition, examples, Chinese postman problem, Fleury's algorithm, Theorems on Eulerian graphs. **Trees:** Definition, pendent vertex in a tree, distance and centres in a tree, rooted and binary trees, spanning trees and rank nullity, fundamental circuits, fundamental cutset, vertex connectivity, edge connectivity, spanning trees, weighted graphs, Kruskal's algorithms, Prim's algorithm, Breadth first search algorithm, depth first search algorithm, Dijkstra's algorithm, Warshall Floyd algorithm, Theorems on trees. **Directed graphs:** Incident out of a vertex, incident into a vertex, indegree, out degree, isolated vertex, pendent vertex, types of digraphs, arborescence definition. **Networks** Flows and cuts, Max ow and min cut theorem, The Ford and FulkersonAlgorithm **Graph Coloring:** **Vertex Coloring:** K-coloring, K-colorable, Chromatic Number, K - Chromatic.Vertex coloring Algorithm: Simple Sequential Coloring, Largest - First Sequential Algorithm (Welsh and Powell) Smallest–Last Sequential Algorithm. **Edge Coloring:** Definition and Concept Only. **Planar Graphs:** Introduction Kuratowski's two graphs (K5, K3)

	Euler'stheorem, Examples based on Euler's theorem. **Matching and Factors:** Matching in bipartite graphs, maximum matchings, Hall`s matching conditions, Min-Matching in bipartite graphs, sets, applications and algorithms, maximum bipartite matching, weighted bipartitematching, Tutte`s 1 factor theorem, factors of graphs.

Learning Resources:

1) John Clark, D. A. Holton: A First Look at Graph Theory, World Scientific, 1991.
2) N. Deo, Graph theory with Applications to Computer Science and Engineering, PHI
3) Douglas B. West, Introduction to Graph Theory, Pearson Education, Second Edition Purna Chandra Biswal, Discrete Mathematics and Graph Theory, Fourth Edition (PHI.).

Title of the Course and Course Code	MTS-554 Numerical Analysis	Number of Credits: 04

Course Outcomes (COs) On completion of the course, the students will be able to:	
CO1	State and apply different methods of Numerical Differentiation and Integration
CO2	Explain the basic principles and theory of interpolation
CO3	Implement all the standard curve fitting techniques
CO4	Explain basic methods of solving Linear and Non-Linear Equations
CO5	Test different methods of solving differential equations, compute and evaluate differential equations numerically
CO6	Develop the knowledge to compute the errors in all of the techniques mentioned in the course

Unit No.	Title of the Unit and Contents
1	**Solution to Non-Linear Equations** Introduction to Big O notation, Types of Errors, Fixed point iterative method, Bracketing method for roots, initial approximation and convergence criteria, Newton Raphson and Secant method, Error Analysis, rate of convergence, Accelerated Newton Raphson Method, Mullers Method
2	**Solutions to Linear and Non-Linear Systems** Upper triangular linear systems, Gaussian Elimination and Pivoting, Triangular Factorization, Iterative methods to solve Linear Systems and their convergence, Newton's method to solve nonlinear systems, Computing Eigenvalues numerically using Power Method, computing Eigenvalues of Symmetric matrix using House holders' method.
3	**Interpolation and Polynomial Approximation** Introduction to the concept of Interpolation, Lagrangian Interpolation, Newton's Interpolation and its various forms (divided difference, forward and backward), Error Analysis with these approximations, Pade Approximations
4	**Curve Fitting** Least Square Line, Curve Fitting, Interpolation by Spline Functions, Fourier Series and Trigonometric Polynomials
5	**Numerical differentiation and Integration** Approximating the derivatives, various Numerical differentiation formulas, Introduction to Quadrature, Composite Trapezoidal and Simpson Rule, Error Computations in Numerical Integration
6	**Solutions to Differential Equations** Introduction to Ordinary differential equations, Euler's method, Heun method, Taylor series method, Runge Kutta methods of various orders, Predicator- Corrector Methods with precision and Error analysis

Learning Resources

1) John Mathews & Kurtis Fink, Numerical Analysis using Matlab, Prentice Hall.
2) Kendall Atkinson, Numerical Analysis, Wiley Publications.
3) S. S. Sastry, Numerical Analysis, Prentice Hall India Publication.

Practical Lab III: MTS-570 Introduction to Web Technologies

Title of the Course and Course Code	MTS570 (Practical III) Introduction to Web Technologies	Number of Credits: 02
Course Outcome (COs) On completion of the course, the students will be able to:		
CO1	State design process for single-page applications using ReactJS.	
CO2	Illustrate the use of JavaScript in webpages.	
CO3	Demonstrate Bootstrap5 features	
CO4	Analyze the importance of user experience in designing of the websites through various web based technologies.	
CO5	Test and validate web applications using JavaScript.	
CO6	Write web pages using various web technologies.	

Unit No	Title of Unit and Contents
1	**HTML5** What is HTML? Tags, Heading, paragraph, anchor, image, table, Lists (ordered, unordered, description), form, label, input, button, br, hr, script (simple JavaScript code) Audio, Video, Progress, Data list Tag, Header Tag, Footer Tag, Article Tag, Aside Tag, Canvas, SVG, Google Maps, Geolocation, Web Storage
2	**CSS3** What is CSS? Syntax, Selector, Types, Comments Background, colors, Border, Margin, Padding, Height/Width, Box Model, Outline, Text, Fonts, Icons, Links, Lists, Tables, Display, Max-width, Position, Overflow, Inline-block, Align, Combinators, Pseudo-classes (link, visited, hover, active, focus), Pseudo- elements (first-line, first letter, before, after), Tooltips, Attribute Selectors, Forms Rounded Corners, Backgrounds, Colors, Text, Fonts, Images, Buttons, Multiple Columns, User Interface, Box Sizing, Flexbox, Media Queries, MQ Examples
3	**JavaScript** Introduction, JavaScript BOM, Comments, Variables, Operators, Data Types, Functions, Objects, Scope, Events (onclick, onchange, onmouseover, onmouseout, onkeydown, onload), Strings, String Methods (indexOf(), lastIndexOf(), search(), slice(), substring(),substr()), Numbers, Number Methods (toString(), toExponential(), toFixed(), toPrecision(), valueOf()), Math (min (), max (), pow (), random (), sqrt(), ceil(), floor() methods), Array, Array methods (toString(), join(), pop(), push(), shift(), unshift(), splice(),concat()), Date & its methods, Booleans, Comparisons, Conditions, Switch, for loop, while loop, break, Type Conversion, Debugging, Class, Objects, Object Properties, Object Methods Asynchronous Programming Callbacks, Promises. Async-Await

4	**Bootstrap 5** What is Bootstrap, Container, Jumbotron, Button, Grid, Table, Form, Alert, Badges, Cards, Pagination, Image, Carousel, Progress Bar, List Group, Dropdown, Collapse, Tabs / Pills, Navbar, Input Types (Check Box, Radio Button), Modals, Popover, Flex
5	**Introduction to ReactJS** Overview, Virtual DOM, JSX, Components, Styling, Props, State management, Forms, Routing, React Redux

Learning Resources:

1) Beginning HTML5 and CSS3 by Christopher Murphy, Divya Manian, Oliver Stud Holme andRichard W. Clark (APress)
2) Beginning JavaScript by Jeremy McPeak and Paul Wilton (Wrox)
3) Beginning jQuery by Jack Franklin (APress)
4) Bootstrap by Jake Spurlock (O'Reilly)
5) Head First HTML5 Programming, Building Web Apps with JavaScript By Eric Freeman,Elisabeth Robson (O'Reilly)
6) Head First JavaScript Programming By Eric T. Freeman, Elisabeth Robson (O'Reilly)
7) Head First Ajax By Rebecca M. Riordan (O'Reilly)
8) Head First jQuery, A Brain-Friendly Guide By Ryan Benedetti, Ronan Cranley (O'Reilly)
9) Roger S. Pressman. *Software Engineering: A Practitioner's Approach* (Sixth Edition, International Edition). McGraw-Hill, 2005.
10) Ian Sommerville. *Software Engineering* (Seventh Edition). Addison -Wesley, 2004.

Important URLs: https://www.w3schools.com/, https://getbootstrap.com/

Practical Lab IV: MTS-571 Programming and Problem solving with Python

Title of the Course and Course Code	MTS571 (Practical IV) Programming and Problem solving with Python	Number of Credits: 02
Course Outcome (COs) **On completion of the course, the students will be able to:**		
CO1	Recall the basic concepts programming languages	
CO2	Discuss fundamentals of data structures	
CO3	Demonstrate Bootstrap5 features	
CO4	Apply fundamental concepts of functions using Python	
CO5	Determine the solution for a given problem by implanting various libraries.	
CO6	Integrating all the python features to solve the case studies.	

Unit No.	Title of the Unit
1	**Introduction** What is Python? Features, History, Version, Applications, Install Python, Python Path, Python Example, Execute Python, Variables, Keywords, Identifiers, Literals, Operators, Comments, Control Statements
2	**Strings and Data Structures** Accessing Strings, Basic Operators, Membership Operators, Relational Operators, Slice Notation, String functions and Methods Data Structures: List - Accessing Lists, List Operations, Functions and Methods of Lists, Tuple - Accessing Tuple, Tuple Operations, Functions and methods of Tuples, why use Tuple? Python Dictionary - Accessing Values, Functions & Methods
3	**Functions** Built-in Functions, User defined Functions, invoking a Function, return, Statement, Argument and Parameter, Positional Argument (Required Argument), Default Argument, Keyword Argument, Anonymous Function, Difference between Normal Functions and Anonymous, Function, Scope of a Variable
4	**Files I/O and Modules** Input from Keyboard, File Handling, Attributes of File, Modes of File, File Handling Methods-What is a Module? Importing a Module, Built in Modules in Python, Package
5	**Python-database Connectivity** Establishing a Connection, CRUD (Create, Read, Update, Delete) Operations
6	**Introduction to data processing and visualization** Concept, Overview of the data processing, Various applications and branches, what is dataset? importing dataset, various sources of datasets, visualization - concept and use
7	**Python Libraries** Overview of few important Python Libraries for data processing and visualization - NumPy, Pandas: importing dataset, dataframe, series different functions - Scikit-Learn etc, matplotlib
8	Case Studies implementing python libraries

Learning Resources:

1) Beginning-Python, Second Edition by Magnus Lie Hetland
2) The Complete Reference Python by Martin C. Brown
3) Head First Python by Patrick Barry
4) Learning Python, O'Reilly by Mark Lutz
5) Python in a Nutshell, O'Reilly by Alex Martelli

Important URL:

- https://www.python.org