NATIONAL INSTITUTE OF TECHNOLOGY WARANGAL

SCHEME OF INSTRUCTION AND SYLLABI

for two-year M.Sc. (MATHEMATICS & SCIENTIFIC COMPUTING)

(Effective from 2021-22)

DEPARTMENT OF MATHEMATICS

Vision and Mission of the Institute

VISION

Towards a Global Knowledge Hub, striving continuously in pursuit of excellence in Education, Research, Entrepreneurship and Technological services to the society

MISSION

- Imparting total quality education to develop innovative, entrepreneurial and ethical future professionals fit for globally competitive environment.
- Allowing stake holders to share our reservoir of experience in education and knowledge for mutual enrichment in the field of technical education.
- Fostering product-oriented research for establishing a self-sustaining and wealth creating Centre to serve the societal needs.

Vision and Mission of the Department

Department of Mathematics

VISION

To be among the best mathematics departments in the country, to build an international reputation as a centre of excellence in mathematics and computational research, training, and education, and to inculcate mathematical thinking in order to meet the challenges and growth of science and technology, as well as the needs of industry and society, with moral and ethical responsibility.

MISSION

- To attract motivated and talented students by providing a learning environment where they can learn and develop the mathematical and computational skills needed to formulate and solve real-world problems.
- To foster an environment conducive to quality research and to train principled and highly skilled researchers with clear thinking and determination capable of meeting the dynamic challenges of science and engineering.
- To keep up with the rapid advancements of technology while improving academic standards through innovative teaching and learning processes.
- To satisfy the country's human resource and scientific manpower requirements in mathematics through learner-centered contemporary education and research.

Department of Mathematics:

Brief about the Department:

The Department of Mathematics is one of the highly reputed Departments in the institute which functions with excellence as its motto. The Department of Mathematics was established in 1959 along with other engineering departments, expanded in 1984 as Dept. of Mathematics & Humanities and bifurcated in 2009 as Department of Mathematics. The Department is established as a dynamic centre for academic and research activities.

The Department offers basic courses in Mathematics for B.Tech. At post-graduate level, the Department offers well-designed diverse courses for all programmes of M.Tech., M.C.A., M.B.A. and M.Sc. Tech (Engg. Physics) and also offers open electives for all UG, PG and Ph.D. Programmes.

The Department offers two P.G. Programs, M.Sc. (Applied Mathematics) started in the year 1970 and M.Sc. (Mathematics and Scientific Computing) started in 2001. The M.Sc. programs for both streams of Mathematics are designed with one laboratory course in each semester in addition to the regular rigorous theory courses. They inculcate a spirit of practical application of mathematical concept and also instil enthusiasm for research activity. Special emphasis is laid on promoting team spirit and improving the oral communication skills of the students, which enables all-round development of the students.

The Department since its inception in 1959 is known to be an active research centre in Mathematics. The frontier areas of research of the department are Fluid Mechanics, Computational Fluid Mechanics, Bio-mechanics, Numerical Analysis, Finite Element Method, Optimization Techniques, Coding Theory, Cryptography, Differential Equations etc., The Department offers Ph.D. program in Mathematics on regular basis, part-time and also under Quality Improvement Program (QIP) and the Department is the only QIP centre for Mathematics in India. So far about 115 Ph.Ds. have been awarded and several research papers have been published in national and international journals.

The Department has a full-fledged computational laboratory to meet the requirements of the M.Sc. students, research scholars and the faculty. The Department has a well-stocked library for immediate reference of the staff and students.

The Department was recognized as a National Resource Centre in Mathematics by MoE, Govt. of India to conduct Online Refresher Courses for all Mathematics Faculty members (irrespective of their seniority and designation) of all Institutions in the Country.

The department organized three international conference (ICCHMT – 2015, NHTFF – 2018), two GIAN programs. Several National conferences, Summer/refresher courses and Workshops. The Department has successfully completed several research projects funded by various organizations like MHRD, AICTE, UGC, CSIR and DST etc and there are 3 ongoing projects

List of Programs offered by the Department:

Program	Title of the Program
Integrated M.Sc.	Integrated M.Sc., Mathematics
M.Sc.	M.Sc., Applied Mathematics
	M.Sc., Mathematics and Scientific Computing
Minor	Mathematics
Ph.D. (Full time, Part-time and QIP)	Mathematics

Note: Refer to the following weblink for Rules and Regulations of PG programs:
https://nitw.ac.in/main/%20RulesandRegulations/PGProgrammes/

M.Sc. in Mathematics and Scientific Computing
Program Educational Objectives

PEO-1	Provide sufficient understanding of the fundamentals of mathematics with computational techniques, and program core to address challenges faced in mathematics and other related interdisciplinary fields.
PEO-2	Facilitate as a deep learner and progressive careers in teaching, academia, research organizations, national/international laboratories and industry.
PEO-3	Develop models & simulation tools for real life problems by analysing and applying mathematical and computational tools and techniques.
PEO-4	Demonstrate effective communication and interpersonal, management and leadership skills to fulfil professional responsibilities, retaining scientific fervour in day-to-day affairs.
PEO-5	Engage in lifelong learning and adapt to changing professional and societal needs.

Program Articulation Matrix

PEO / Mission Statements	**PEO 1**	**PEO 2**	**PEO 3**	**PEO 4**	**PEO-5**
To attract motivated and talented students by providing a learning environment where they can learn and develop the mathematical and computational skills needed to formulate and solve real-world problems.	**2**	**3**	**2**	**3**	**2**
To foster an environment conducive to quality research and to train principled and highly skilled researchers with clear thinking and determination capable of meeting the dynamic challenges of science and engineering.	**3**	**3**	**3**	**2**	**2**
To keep up with the rapid advancements of technology while improving academic standards through innovative teaching and learning processes.	**2**	**3**	**2**	**2**	**2**
To satisfy the country's human resource and scientific manpower requirements in mathematics through learner-centered contemporary education and research.	**3**	**2**	**3**	**3**	**3**

1-Slightly; 2-Moderately; 3-Substantially

M.Sc.in Mathematics and Scientific Computing
Program Outcomes

At the end of the program, the student will be able to:

PO1	Gain and apply the knowledge of basic scientific and mathematical fundamentals to understand the Nature and apply it to develop new theories and models.
PO2	Design algorithms and derive solutions for complex mathematics problems to meet the specified needs with appropriate consideration for the public health and safety, cultural, societal, and environmental considerations.
PO3	Use of research-based knowledge and research methods including design of physical/computational experiments and evolve appropriate procedures to a given problem.
PO4	Create, select, and apply appropriate techniques, resources, and modern IT tools including prediction and modelling to complex real-life problems with an understanding of the limitations.
PO5	Function effectively as an individual, and as a member or leader in diverse teams to manage projects in multidisciplinary environments
PO6	Use numerical analysis and statistical modelling and interpretation of data, to provide valid conclusions

SCHEME OF INSTRUCTION

M. Sc. (Mathematics & Scientific Computing) - Course Structure

I - Year, I – Semester

S.No	Course code	Course Name	L	T	P	Cr	Cat. Code
1	MA4101	Real Analysis	3	0	0	3	PCC
2	MA4102	Ordinary Differential Equations	3	0	0	3	PCC
3	MA4201	Mathematical Foundations of Computer Science	3	0	0	3	PCC
4	MA4202	Numerical Linear Algebra	3	0	0	3	PCC
5	MA4203	Computer Programming in Python	3	0	0	3	PCC
6	MA4204	Algorithmic Approach to Computational Methods	3	0	0	3	PCC
7	MA4205	Python Lab	0	0	3	1.5	PCC
		Total	**18**	**0**	**3**	**19.5**	

I - Year, II – Semester

S.No	Course Code	Course Name	L	T	P	Cr	Cat. Code
1	MA4151	Probability & Statistics	3	0	0	3	PCC
2	MA4152	Partial Differential Equations	3	0	0	3	PCC
3	MA4153	Complex Analysis	3	0	0	3	PCC
4	MA4251	Optimization Techniques	3	0	0	3	PCC
5	MA4252	Object Oriented Programming with C ++	3	0	0	3	PCC
6		ELECTIVE – I	3	0	0	3	PEC
7	MA4253	OOP WITH C ++ LAB	0	0	3	1.5	PCC
8	MA4298	Seminar - I	0	0	2	1	SEM
		Total	**18**	**0**	**5**	**20.5**	

II - Year, I – Semester

S.No	Course Code	Course Name	L	T	P	Cr	Cat. Code
1	MA5201	Design and Analysis of Algorithms	3	0	0	3	PCC
2	MA5202	Cryptography	3	0	0	3	PCC
3	MA5203	Data Base Management Systems	3	0	0	3	PCC
4	MA5204	Data Structures	3	0	0	3	PCC
5		ELECTIVE – II	3	0	0	3	PEC
6		ELECTIVE – III	3	0	0	3	PEC
7	MA5205	DBMS Lab	0	0	3	1.5	PCC
8	MA5248	Seminar - II	0	0	2	1	SEM
		Total	**18**	**0**	**5**	**20.5**	

II - Year, II – Semester

S.No	Course Code	Course Name	L	T	P	Cr	Cat. Code
1		ELECTIVE – IV	3	0	0	3	PEC
2		ELECTIVE – V	3	0	0	3	PEC
3	MA5251	Computation Lab	0	0	3	1.5	PCC
4	MA5297	Comprehensive Viva	0	2	0	2	CVV
5	MA5299	Dissertation Work	0	5	10	10	DW
		Total	**6**	**7**	**13**	**19.5**	

SCHEME OF INSTRUCTION

Credits in Each Semester					
Cat. Code	**Sem-I**	**Sem-II**	**Sem-III**	**Sem-IV**	**Total**
PCC	19.5	16.5	13.5	1.5	51
PEC	-	3	6	6	15
DW	-	-	-	10	10
SEM	-	1	1	-	2
CVV	-	-	-	2	2
Total	**19.5**	**20.5**	**20.5**	**19.5**	**80**

Program Elective Courses

I - Year, II – Semester: Elective – I	
Course Code	**Course Name**
MA4261	Graph theory
MA4262	Java Programming
MA4263	Advanced Abstract Algebra
II - Year, I – Semester: Elective– II	
MA5211	Symbolic Computing
MA5212	Neural Networks
MA5213	Wavelet Analysis
MA5214	Computer Graphics
II - Year, I – Semester: Elective– III	
MA5215	Data Analysis with R
MA5216	Data Mining
MA5217	Algebraic Coding Theory
MA5218	Parallel Computing
II - Year, II – Semester: Elective–IV	
MA5261	Evolutionary optimization techniques
MA5262	Management Information Systems
MA5263	Computational Number Theory
MA5264	Fuzzy Mathematics and Applications
MA5265	Perturbation Methods
II - Year, II – Semester: Elective– V	
MA5266	Theory of Automata
MA5267	Mathematics of Data Science
MA5268	Elliptic Curves
MA5269	Advanced Theory of Algebraic Codes
MA5270	Approximation Theory
MA5271	Graphical Processing Unit Programming with CUDA

Note:

1. In addition to the above listed electives, a student may register for electives from Applied Mathematics stream on satisfying the minimum pre-requisite of the specific course(s).
2. An elective may be offered to the students, only if a minimum of 15 students opt for it.
3. Students can take maximum of TWO subjects (other than the above listed) from MOOC in any of the Elective slots with the approval of DAC – P G & R.

DETAILED SYLLABUS FOR EACH COURSE
M.Sc. (Mathematics & Scientific Computing):: I Year I Semester

Course Code: MA4101	Real Analysis	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Find whether a given function can be Riemann integrable
CO2	Test whether a given improper integral can be convergent
CO3	Examine uniform convergence of given sequence and /or series of functions
CO4	Expand a given function into Fourier series

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	2	2	3
CO2	3	2	2	3	3	2
CO3	3	2	3	3	2	3
CO4	3	3	3	3	2	3

Syllabus:

Basic Topology: Introduction

Riemann Stieltje's integral: Definition and existence of the integral, Properties of the integral, Integration and differentiation of integral with variable limits.

Improper integrals: Definitions and their convergence, Tests of convergence, β and Γ functions.

Uniform convergence: Tests for uniform convergence, Theorems on limit and continuity of sum functions, Term by term differentiation and integration of series of functions.

Power series: Convergence and their properties.

Fourier series: Dirichlet's' conditions, Existence, Problems, Half range sine and cosine series.

Learning Resources:

Text Books:

1. Principles of Mathematical Analysis**, Walter Rudin,** McGraw Hill, 2017, Third Edition.
2. Real Analysis, **Brian S.Thomson, Andrew M.Bruckner, Judith B.Bruner**, Prentice Hall International, 2008.

Reference Books:

1. Introduction to Real Analysis, **William F. Trench,** Library of Congress Cataloging-in-Publication Data, Free Edition 1.04, April 2010.
2. Real Analysis, N.L. **Carothers**, Cambridge University Press, 2000.
3. Mathematical Analysis, **Tom M. Apostol**, Addison Wesley, 1974, Second Edition.

Course Code: MA4102	Ordinary Differential Equations	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Determine linearly independent solutions and general solution of a non-homogeneous differential equations
CO2	Find power series solution to a differential equation containing variable coefficients
CO3	Discuss the existence and uniqueness of solution for an initial value problem
CO4	Use Green's function to solve a non-homogeneous boundary value problem

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	2	2	3
CO2	3	2	2	3	3	2
CO3	3	2	3	3	2	3
CO4	3	3	3	3	2	3

Syllabus:

First Order Equations: Picard's theorem, Non-Local existence theorem.

Second Order Equations: Linear dependence and independence, A formula for the Wronskian, the non-homogeneous equations, linear equations with variable coefficients, reduction of the order of the homogeneous equation, Sturm comparison theorem, Sturm separation theorem.

Stability: Autonomous Systems. The Phase Plane and Its Phenomena, Types of Critical Points. Stability, Critical Points and Stability for Linear Systems.

Systems of Differential Equations: Existence theorems, homogeneous linear systems, non-homogeneous linear systems, linear systems with constant coefficients, eigenvalues and eigenvectors, diagonal and Jordan matrices,

Boundary value problems: Two-point boundary value problems, Green's functions, construction of Greens functions, non-homogeneous boundary conditions.

Learning Resources:

Text Books:

1. Differential Equations with Applications and Historical Notes, **G.F. Simmons**, McGraw Hill, 2017, Second Edition.
2. An Introduction to Ordinary Differential Equations, **E.A. Coddington**, PHI Learning, 1999.
3. Ordinary Differential Equations, **Tyn Myint U**, Elesvier, North- Holland, 1978.
4. Textbook of Ordinary Differential Equations, **V. Raghavendra, Rasmita Kar, S.G. Deo, V. Lakshmikantham,** McGraw Hill India, 2015, Third Edition.

Reference Books:

1. Differential Equations and Their Applications, **M. Braun**, Springer-Verlag, 1983, Third Edition.
2. Differential and Integral Equations, **P.J. Collins**, Oxford University Press, 2006.
3. Elementary Differential Equations and Boundary Value Problems, **W.E.Boyce** and **R.C. Di-Prima**, John Wiely & Sons, 2001.

Course Code: MA4201	Mathematical Foundations of Computer Science	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Apply Propositional logic and first order logic to determine the validity of the statement
CO2	Construct induction proofs involving summations, inequalities, and divisibility
CO3	Implement the principles of counting, permutations and combinatory to solve real world problems
CO4	Formulate and solve recurrence relations
CO5	Determine whether a given relation is an equivalence relation/poset and will be able to draw a Hasse diagram
CO6	Develop and analyze the concepts of Boolean algebra

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	1		2		
CO2	1	2	2			
CO3		2	2	3		
CO4	1	2		2		3
CO5	2	1		1		1
CO6	2	2	2	2		

Syllabus:

Mathematical Logic: Connectives, Tautologies, equivalence of formulas, duality law, tautological implications, normal forms, theory of inference for statement calculus, methods of proof, predicative logic, statement functions, variables and quantifiers, free and bound variables, inference theory for predicate calculus.

Counting: Basics of counting**,** Permutations and combinations - Generalized Permutations and combinations – Pascal's Identity- Vandermonde's Identity, the Principles of Inclusion–Exclusion, Pigeonhole Principle and its Application.

Recurrence relations: Generating functions, Generating Functions of Permutations and Combinations, Formulation as Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots, Solving Inhomogeneous Recurrence Relations – Applications of recurrence relations.

Relations: Binary relations **-** Properties of binary relations, equivalence relations and partitions, Matrix Representation of relations, Adjacency Matrices, Incidence Matrices, Transitive closure and Warshal's algorithm, Partial and total ordering relations.

Boolean Algebra: Chains, Lattices, principle of duality, basic properties of lattices, distributive and complemented lattices, Boolean lattices and algebras, uniqueness of finite Boolean algebras, Boolean expressions and functions - Representation and Minimizations of Boolean functions.

Learning Resources:

Text Books:

1. Discrete Mathematics for Computer Scientists, **J.R.Mott, A.Kandel and Baker**, PHI, 2006, Third Edition.
2. Elements of Discrete Mathematics, **C.L. Liu**, McGraw Hill, 2017, Fourth Edition.
3. Discrete Mathematical Structures with applications to Computer Science, **J.P.Tremblay and R.Manohar**, McGraw Hill Book Co., 2017.

Reference Books:

1. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, **K. H. Rosen**, Tata McGraw Hill, 2017, Seventh Edition.
2. Discrete Mathematical Structures, **Bernand Kolman**, **Robert C. Busby**, Sharon Cutler Ross, PHI, 2008, Sixth Edition.

Course Code: MA4202	Numerical Linear Algebra	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Understand the basic concepts of linear algebra related to stability, accuracy, etc
CO2	Find QR factorization of a matrix using Householder transformation and study its applications
CO3	Write various algorithms to solve system of linear equations to understand computational issues
CO4	Describe the numerical procedure of eigenvalue problem
CO5	Apply the SVD of a matrix in solving real life problems

Course Articulation Matrix:

	PO	PO2	PO3	PO4	PO5	PO6
CO1	3		2			1
CO2	3		2			1
CO3	3	3	2			1
CO4	3		2			2
CO5	3		2			2

Syllabus:

Review of linear algebra concepts: Vector and Matrix norms and norm-preserving properties of orthogonal matrices; Linear transformations, rank, basis.

Transformations in numerical linear algebra and their applications: Elementary matrices, Householder transformation with applications to LU, QR factorizations and Hessenberg-reduction. Orthonormal bases and orthogonal projections.

Numerical Solutions of Linear Systems: QR factorizations with growth factors and stability; estimation of the condition numbers; iterative methods: Jacobi and Gauss-Seidel methods and their convergence; Least-squares Solutions to Linear Systems: existence and uniqueness, Normal equations; Pseudo Inverse; QR factorization methods for overdetermined systems.

Numerical Matrix Eigenvalue Problems: Gershgorin disk theorem, the Power method, the Inverse Power methods & Rayleigh Quotient Iteration; Basic and Hessenberg QR iterations.

The Singular Value Decomposition: Properties and applications of SVD.

Learning Resources:

Text Books:

1. Numerical Linear Algebra and Applications, **Biswa Nath Datta**, Prentice Hall India/SIAM, 2013/2010, Second Edition.
2. Matrix Analysis and Applied Linear Algebra, **Carl D. Meyer**, SIAM, 2000.

Reference Books:

1. Numerical Analysis: Mathematics of Scientific Computing, **David Kincaid and Ward Cheney**, American Mathematical Society, 2002.
2. Numerical Linear Algebra, **V. Sundarapandian**, Prentice Hall India, 2014.

Course Code: MA4203	Computer Programming in Python	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Introduce the fundamental concepts of Python
CO2	Provide a foundation to use basic building blocks of Python
CO3	Learn to write Python Scripts.
CO4	Explore various exception handling mechanisms
CO5	Develop Python packages

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	1				
CO2	2	2				
CO3	2	2		1		
CO4	3	2	2	2		
CO5	2	3	2	3	1	2

Syllabus:

Introduction to Python: Python variables, Python basic Operators, Python Data Types, variables, Declaring and using Numeric data types: int, float etc., Basic Input-Output Operations, Basic Operators

Conditionals and loops: Boolean Values, if, else and else if, Simple for loops in python, for loop using ranges, string, list and dictionaries. Use of while loops in python, Loop manipulation using pass, continue, break and else.

Strings: Assigning values in strings, String manipulations, String special operators, String formatting operators, Triple Quotes, Raw String, Unicode String, Build-in-String methods,

Lists: Lists Introduction, accessing values in list, List manipulations, List Operations, Indexing, slicing & matrices, use of tuple data type. string, list and Dictionary, string manipulation methods, programming using string, list and dictionary in-built functions.

Functions: Built –in Functions and methods, Functions, writing functions in Python, returning a result from a function, Pass by value & pass by reference, function arguments & its types, recursive functions.

Python packages: Simple programs using the built-in functions of packages matplotlib, numpy, pandas etc.,

Learning Resources:

Text Books:

1. Introduction to Python Programming, **William Mitchell, Povel Solin**, **Martin Novak et al**., NCLab Public Computing, 2012.
2. Introduction to Python Programming, ©**Jacob Fredslund**, 2007.

Reference Books:

1. An Introduction to Python, **John C. Lusth**, The University of Alabama, 2011.
2. Introduction to Python, ©**Dave Kuhlman**, 2008.

Course Code: MA4204	Algorithmic Approach to Computational Methods	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Find the roots of nonlinear equations numerically
CO2	Solve system of equations numerically
CO3	Interpolate the given data
CO4	Find differentiate of a function and definite integrals numerically
CO5	Solve the initial value problems

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2			2		1
CO2	3	2		3		2
CO3	2			2		3
CO4	2			2		
CO5	3	2		3		2

Syllabus:

Representation of integers and fractions: Fixed point and floating point arithmetic, error propagation, loss of significance, condition and instability, computational method of error propagation.

Root finding: bisection method, secant method, regula-falsi method, Newton-Raphson method.

Solution of system of linear equations: Gauss elimination with and without pivoting, LU decomposition, Ill-conditioned equations, Gauss-Jacobi method, Gauss-Seidel method, Jacobi method to find eigenvalues.

Interpolation: Lagrange's interpolation, Newton's divided difference interpolation (forward, backward), Newton-Gregory formulae, Sterling's formula.

Numerical Differentiation: Numerical differentiation with forward, backward and central differences.

Numerical integration: Newton-Cotes (closed type formulae)-trapezoidal rule, Simpson's 1/3 rule,Simpson's 3/8 rule. Gaussian Quadrature (Legndre and Chebyshev cases)

Solution of Ordinary differential equations: Initial value problems: Single step methods; Taylor's, Euler's, Runge-Kutta methods, error analysis; System of IVP's and higher order IVP's.

Learning Resources:

Text Books:

1. Computer Oriented Numerical Methods, **V Rajaraman**, Prentice Hall of India, 1993.
2. Numerical Methods for Engineers and Scientists, **M.K.Jain, SRK Iyengar and R.K Jain**, New Age International, 2008.
3. Introduction to Numerical Analysis, **F.B.Hildebrand**, Dover Publications Inc, 2003

Reference Books:

1. Introductory Methods of Numerical Analysis, **S.S.Sastry**, Prentice Hall of India, 2012
2. Elementary *Numerical Analysis*: An Algorithmic Approach, **S.D. *Conte* and C. de Boor**, Mc-GrawHill, 1981.

Course Code: MA4205	Python Lab	Credits 3-0-0: 3

Pre-Requisites: MA4203

Course Outcomes:

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

CO1	Design and test programs to solve mathematical and scientific problems
CO2	Develop and test programs using control structures
CO3	Implement modular programs using functions
CO4	Develop packages

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	3	3	3
CO2	3	3	2	3	2	3
CO3	3	2	3	3	2	3
CO4	3	3	2	3		

Syllabus:

Programs using

1. Conditional control constructs
2. loops
3. Strings
4. Lists
5. user defined functions and library functions
6. built-in functions of packages matplotlib
7. numpy
8. pandas

M.Sc. (Mathematics & Scientific Computing):: I Year II Semester

Course Code: MA4151	Probability & Statistics	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Determine the mean, standard deviation and m^{th} moment of a probability
CO2	Apply theoretical model to fit the empirical data
CO3	Differentiate between large and small sample tests
CO4	Use the method of testing of hypothesis for examining the validity of a hypothesis
CO5	Estimate the parameters of a population from knowledge of statistics of a sample

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	3	3	1	1	2
CO2	2	3	3	1	1	2
CO3	2	3	3			2
CO4	3	3	2			1
CO5	3	3	3			2

Syllabus:

Random variables: Review of probability; Probability distributions with discrete and continuous random variables - Joint probability mass function, Marginal distribution function, Joint density function – Independent random variables - Mathematical Expectation - Moment generating function - Chebyshev's inequality - Weak law of large numbers - Bernoulli trials

Theoretical Probability Distributions: Binomial, Negative Binomial, Geometric, Poisson, Normal, Rectangular, Exponential, Gaussian, Beta and Gamma distributions and their moment generating functions; Fit of a given theoretical model to an empirical data.

Sampling and Testing of Hypothesis: Introduction to testing of hypothesis - Tests of significance for large samples – t, F and Chi-square tests; ANOVA - one-way and two-way classifications.

Theory of estimation: Characteristics of estimation - Minimum variance unbiased estimator - Method of maximum likelihood estimation.

Correlation and Regression: Scatter diagram - Linear and polynomial fitting by the method of least squares - Linear correlation and linear regression - Rank correlation - Correlation of bivariate frequency distribution.

Learning Resources:

Text Books:

1. Fundamentals of Mathematical Statistics, **S.C. Gupta and V.K. Kapur,** S.Chand & Sons, New Delhi, 2008.
2. An Introduction to Probability theory and Mathematical Sciences, **V.K. Rohatgi and A.K. Md. Ehsanes Saleh,** Wiley, 2001.

References

1. Miller & Freund's Probability and Statistics for Engineers, **Richard A. Johnson**, Pearson, 2018, Ninth Edition.

Course Code: MA4152	Partial Differential Equations	Credits 3-0-0: 3

Pre-Requisites: MA4102

Course Outcomes:

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

CO1	Solve linear and nonlinear first order partial differential equations
CO2	Demonstrate the concept of characteristic curves and characteristic strips
CO3	Solve higher order partial differential equations with constant coefficients
CO4	Find canonical forms of second order partial differential equations
CO5	Utilize the knowledge of PDE's in solving various physical problems

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	3			2
CO2	3	2	2			1
CO3	3	3	3			2
CO4	3	1	2			
CO5	3	3	3			1

Syllabus:

Equations of the First Order: Formulation; Classification of first order partial differential equations (PDEs); Lagrange's method, Cauchy problem, and method of characteristics for linear and quasilinear PDEs; Paffian equation, Condition for integrability; First order non-linear equations, Complete integrals, Envelopes and singular solutions, Method of Charpit and Method of characteristics.

Equations of higher order: Method of solution for the case of constant coefficients; Classification of second order equations; Reduction to canonical forms; Method of solution by separation of variables.

Wave equation: Derivation of the wave equation; D'Alembert solution of wave equation, Domain of dependence and range of influence; Method of separation of variables; Inhomogeneous wave equation, Duhamel's principle.

Diffusion equation: Derivation of the heat equation, Method of separation of variables, Solutions of heat equation with homogeneous and non-homogeneous boundary conditions; Inhomogeneous heat equation, Duhamel's principle.

Laplace's equation: Basic concepts; Types of boundary value problems; The maximum and minimum principles; Boundary value problems; Method of separation of variables.

Learning Resources:

Text Books:

1. Elements of Partial Differential Equations, **I. Sneddon,** Dover Publications, 2013.
2. Linear Partial Differential Equations for Scientists and Engineers, **Tyn Myint-U and Lokenath Debnath**, Birkhauser, Bostan, 2007, Fourth Edition.

Reference Books:

1. Partial Differential Equations, **P. Prasad and R. Ravindran,** New Age International (P) Ltd., New Delhi, 2010.
2. An Elementary Course in Partial Differential Equations, **T. Amaranath**, Narosa Publishing House, New Delhi, 2003, Second Edition.

Course Code: MA4153	Complex Analysis	Credits 3-0-0: 3

Pre-Requisites: MA4101

Course Outcomes:

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

CO1	Introduce the analyticity of complex functions and study their applications
CO2	Evaluate complex integrals and expand complex functions
CO3	Determine and classify the zeros and singularities of the complex functions
CO4	Evaluate improper integrals by residue theorem
CO5	Learn the uniqueness of conformal transformation

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2			2
CO2	3	1	2			1
CO3	3	2	1			2
CO4	3	2	3			1
CO5	3	3	3			2

Syllabus:

Functions of Complex Variables: Complex variable, Functions of a complex variable, Continuity, Differentiability, Analytic functions.

Complex Integration: Cauchy's theorem, Cauchy's integral formula, Morera's theorem, Cauchy's inequality, Liouville's theorem.

Series Expansions: Taylor's theorem, Laurent's theorem, Zeros of an analytic function, Singularities.

Contour Integration: Residue, Cauchy's residue theorem, contour integration, the fundamental theorem of algebra, Poisson's integral formula. Analytic continuation, branches of a many-valued function, Riemann surface.

Conformal Mapping: The maximum modulus theorem, mean values of $f(z)$, Conformal representation, Bilinear transformation, Transformation by elementary functions, uniqueness of conformal transformation, representation of any region on a circle.

Learning Resources:

Text Books:

1. Complex Variables and Applications, **R.V. Churchill and J.W. Brown,** McGraw Hill, Tokyo, 2009, Eighth Edition.
2. Theory of Complex Variables, E.T. **Copson,** Oxford University Press, New Delhi, 1974.

References

1. Complex Variables with Applications, **S. Ponnusamy & Herb Silverman**, Birkhauser, Boston, 2006, First Edition.
2. Complex Variable, **Murray Spiegel, Seymour Lipschutz, John Schiller and Dennis Spellman**, Schaum's Outlines Series, McGraw Hill, 2017, Revised Second Edition.

Course Code: MA4251	Optimization Techniques	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Understand optimization models and apply them to real life problems
CO2	Find solution to linear optimization problems
CO3	Determine solution to non-linear optimization problems
CO4	Understand the multistage problems and derive solutions
CO5	Apply search techniques to unconstrained optimization problems

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2		2		
CO2	2	1		2		
CO3	2	1		2		
CO4	2	2		2		
CO5	2	2		2		

Syllabus:

Linear programming problems: Formulation, Graphical solution, Simplex method, Big M method, Two Phase method, Dual problem and duality theory, Dual simplex method, revised simplex method, Sensitivity analysis, Transportation Problem, Assignment problem. Travelling Salesman Problem.

Dynamic programming: Principle of optimality, Recursive relations, Solution of LPP, Simple examples.

Non-Linear programming: Convexity, Classical methods, Lagrange's multipliers method, Kuhn-Tucker conditions. Quadratic programming, Wolfes' Method.

One-dimensional search methods: Sequential search, Fibonacci search, Golden section search.

Multi-dimensional search methods: Gradient methods (Steepest descent/ascent methods), Conjugate gradient method (Fletcher-Reeves's method).

Learning Resources:

Text Books:

1. Operations research: an introduction, **H. A. Taha**, Pearson Education Limited, 2017, Tenth Edition.
2. Nonlinear programming: theory and algorithms, **M. S. Bazaraa, H. D. Sherali, & C. M. Shetty**, John Wiley & Sons, 2013, Third Edition.

3. Linear programming and network flows, **M. S. Bazaraa, J. J. Jarvis, & H. D. Sherali,** John Wiley & Sons, 2009, Fourth Edition.

Reference Books:

1. Introduction to Optimization: Operations Research, **J.C. Pant**, Jain Brothers, 2007, Seventh Edition.
2. An introduction to optimization, **E. K. Chong, & S. H. Zak**, John Wiley & Sons, 2004, Second Edition.
3. Engineering optimization: theory and practice, **S. S. Rao**, John Wiley & Sons, 2019, Fifth Edition.

Course Code: MA4252	Object Oriented Programming with C ++	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

CO1	Implement programs using classes and objects
CO2	Able to understand the overloading concept
CO3	Specify the forms of inheritance and use them in programs
CO4	Analyze polymorphic behavior of objects
CO5	Understand virtual functions and polymorphism

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	3	2		2
CO2	1		1	2		
CO3	2		2			
CO4	3	2	3			
CO5	2		2			

Syllabus:

Basic concepts: Concept of C++; Object oriented languages – Applications of OOP.

Classes and Objects: C++ Program with class-Nesting of member functions-private member functions-Arrays within a class- memory allocation for objects-Static data members-Arrays of objects-objects as Function arguments - Friend functions, inline function, Returning objects.

Constructors and Destructors - Multiple constructors in class-Constructors with default arguments copy constructor-Dynamic constructors.

Overloading: Overloading unary operators-overloading binary operators-overloading binary operators using Friends - Rules for overloading operators – function overloading, Type conversions.

Inheritance: Defining derived classes-Single inheritance - Multilevel inheritance - Multiple inheritance - Hierarchical inheritance -Virtual base classes – Abstract classes, Pointers, Virtual functions and Polymorphism.

Learning Resources:

Text Books:

1. Object-Oriented Programming in C++, **Robert Lafore,** Sams, 2001, Fourth Edition.
2. Object oriented programming with C++, **E. Balaguruswamy,** Tata McGraw Hill, 2008, Fourth Edition.

Reference Books:

1. Object-Oriented programming in C++, **Barkakati Nabajyoti**, PHI, 1991.
2. The C++ Programming Language, **Stroustroup Bjarne,** Addison-Wesley,1991.

Course Code: MA4253	OOP With C ++ Lab	Credits 0-0-3: 1.5

Pre-Requisites: MA4252

Course Outcomes:

CO1	Ability to develop applications for physical applications using OOP techniques
CO2	Able to understand the overloading concept
CO3	Specify the forms of inheritance and use them in programs
CO4	Analyze polymorphic behaviour of objects
CO5	Understand virtual functions and polymorphism

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1					
CO2	1	2				
CO3				2		
CO4	1					1
CO5	2		2		2	

Syllabus:

Write the C++ program to illustrate

1. nesting of member functions
2. memory allocation for objects
3. objects as function arguments
4. friend functions, inline functions
5. constructors with default arguments
6. copy and dynamic constructors.
7. overloading unary and binary operator, friend functions
8. Function overloading
9. type conversions
10. single, multilevel, multiple and hierarchical inheritance
11. virtual base and abstract classes.

MA4298 – Seminar
Every student has to give a seminar and a committee of staff members from the department shall evaluate the same. The DAC (PG & R) of the Department shall constitute this committee.

M.Sc. (Mathematics & Scientific Computing):: II Year I Semester

Course Code: MA5201	Design and Analysis of Algorithms	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Understand the basic concepts of algorithms and analysis
CO2	Analyze time and space complexity
CO3	Understand algorithm design methodology
CO4	Apply important algorithm methodology to solve problems
CO5	Understand the difference between P and NP classes of problems

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2		1			
CO2	3		2	2		
CO3	2		1			
CO4	3	3		3		2
CO5	2		1			

Syllabus:

Analysing Algorithms: Concepts in Algorithms Analysis, asymptotic complexity of algorithms, Growth functions Recurrences, Master Theorem.

Divide and Conquer Method: Expected Running Time of Randomized Quick Sort, Merge Sort, Strassen's Matrix Multiplication Algorithm.

Data Structures for Set manipulation problems: Binary tree traversal algorithms, disjoint-set union algorithms.

Graph Algorithms: Representations of graphs, Breadth-first search, Depth-first search, Minimum spanning tree, The algorithms of Kruskal and Prim, Shortest paths, Dijkstra's Algorithm

Greedy Method: Activity Selection Problem, Knapsack Problem, single source shortest path Problem.

Dynamic Programming: Solution to 0-1 Knapsack Problem, multistage graphs, TSP using Dynamic Programming Backtracking: Basic examples, N-Queen's Problem, sum of subsets Problem.

Complexity Classes: Example NP-complete problems.

Learning Resources:

Text Books:

1. Introduction to Algorithms, **T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein**, PHI, New Delhi, 2004, Third Edition.
2. Fundamentals of Computer Algorithms, **E. Horowitz, S. Sahni and S. Rajasekaran**, Universities Press, 2011, Second Edition.

Reference Books:

1. The Design and Analysis of Computer Algorithms, **A. V. Aho, J. E. Hopcroft and J.D. Ullman**, Pearson, New Delhi, 2012, Tenth Impression.
2. Computer Algorithms: Introduction to Design and Analysis, **S. Baase and A.V. Gelder**, Addison and Wesley, 2000, THird Edition.
3. Introduction to the Design and Analysis of Algorithms, **A. Levitin**, Pearson Education, New Delhi, 2009, Second Impression.

Course Code: MA5202	Cryptography	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Understand the structure of stream ciphers, DES, AES
CO2	Analyse RSA and ElGamal public key cryptosystems
CO3	Learn algorithms for integer factorization and discrete-log problems
CO4	Learn elliptic curve cryptosystems
CO5	Know the concepts of hash functions and digital signature schemes

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1		3	1	2	1	
CO2		3	1	2	1	
CO3	1	2		3	2	
CO4		3		2	2	
CO5		3	2	2	2	

Syllabus:

Introduction: Review on basic group theory and basic number theory, Historical ciphers and their cryptanalysis, Principles of modern cryptography, perfect secrecy and one-time pad.

Private-key Cryptography: Stream ciphers, Block ciphers - SPN, Feistel design, DES, AES. Introduction to differential and linear cryptanalysis.

Public-key Cryptography: RSA Cryptosystem, Primality testing, Algorithms for factoring, Diffie-Hellman key-exchange protocol, Discrete-Logarithm Problem (DLP), ElGamal Cryptosystem, Algorithms for DLP.

Elliptic curves: Basic facts, elliptic-curve cryptosystem.

Discussion on Hash functions, Digital signatures and other cryptography topics of relevance.

Learning Resources:

Text Books:

1. Cryptography: Theory and Practice, **Douglas R. Stinson**, Chapman & Hall/CRC, 2006, Third Edition.
2. Cryptography and Network Security: Principles and Practice, **William Stallings**, Pearson, 2017, Seventh Edition.

Reference Books:

1. Understanding Cryptography, **Christof Paar and Jan Pelzl**, Springer, 2010.
2. Introduction to Modern Cryptography, **Jonathan Katz and Yehuda Lindell**, CRC Press, Taylor & Francis Group, 2015, Second Edition.

Course Code: MA5203	Data Base Management Systems	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Identify the functional components of database management system with the development of E-R model for real world applications
CO2	Construct the queries using Relational Algebra, Relational Calculus and SQL
CO3	Apply the concepts of SQL and its use to construct the databases
CO4	Design the relational databases using various normal forms and integrity constraints
CO5	Interpret the concepts of authorization, transfer of privileges, and query processing

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2			2		
CO2	3	3	1	3	3	1
CO3	3	3	1	3		1
CO4	2	3	3	2		
CO5			3	2		

Syllabus:

Introduction: Purpose of Data base systems, Data independence, Data models, Database languages, Data storage, Querying and Time management, Database users and administrators.

Entity-Relation Model: Entities, Entities and relationships, Mapping constraints, E-R diagrams; Extended entity-relationship features: Specialization, Generalization and Aggregation; Design of E-R database scheme.

Relational Model: Structure of relational database, Relational algebra; Extended relational-algebra operations; Modifying the database and views; Tuple relational calculus; Domain relational calculus.

Integrity Constraints: Domain constraints, Referential integrity, Assertions, Triggers, Functional dependencies.

Relational Database Design: Pitfalls in relational database design, First, Second, Boyce-Codd, Third, Fourth and Fifth normal forms.

Security and Integrity: Security and integrity violations, Transfer of privileges, Authorization on views and schema.

Query Processing: Overview of query processing, Structure of query optimizer.

Learning Resources:

Text Books:

1. Database System concepts, **A. Silberschatz, H.F. Korth, and S. Sudarshan**, McGraw Hill, NewYork, 2021, Seventh Edition.
2. Database Management Systems, **R. Ramkrishnan, and J. Gehrke**, McGraw Hill, 2014, Third Edition.

Reference Books:

1. Principles of Database Systems, **Jeffery D. Ullman**, Galgotia, 1994, Third Edition.
2. Fundamentals of Database System, **E. Ramez, N. Shamkant**, Pearson, 2017, Seventh Edition.

Online Resources:

1. https://www.youtube.com/watch?v=IoL9Ve2SRwQ&list=PL3pGy4HtqwD3Ov1J2UBTfsLgxUzUktTAM
2. https://www.youtube.com/watch?v=bGyHqvQW6JY&list=PLwZJjHGjgrZqJ9yQZ-WJb5gBJcKMr9iXP

Course Code: MA5204	Data Structures	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Identify data structures suitable to solve problems
CO2	Develop and analyze algorithms for stacks, queues
CO3	Implement operations on queue
CO4	Implement sorting and searching algorithms
CO5	Implement linked lists using pointers

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3					
CO2	2	3		2		
CO3	3		2			2
CO4	2			2		
CO5	3		2	3		2

Syllabus:

Stack: Definition, Array implementation of stack (static stack): Operations PUSH, POP, and TRAVERSE. Applications of Stack: Infix, Prefix, Postfix representation and conversion using stack, Postfix expression evaluation using stack, use of stack in recursion implementation.

Queue: Definition, array implementation of queue (static queue): Operations INSERT, DELETE, TRAVERSE. Applications of queue, Comparisons of array, stack and queue data structures. Introduction to Circular queue, priority queue, Double ended queue, multiple queue.

Linked list: Singly and Doubly Linear link lists, Singly and doubly circular linked list: Definitions, operations INSERT, DELETE, TRAVERSE on all these lists. (Insertion operation includes – insertion before a given element, insertion after a given element, insertion at given position, insertion in sorted linked list), Implementations of Stack and Queue using linked list (Dynamic stack).

Applications of linked list: String representation and string operations like string length, string reverse, string comparison, string concatenation, string copying, convert upper-case to lower and vice-versa, substring using linked list. Polynomial representation and addition of two polynomials using linked list.

Learning Resources:

Text Books:

1. Data Structures (Schaum's Outline series), **S.Lipschutz,** Tata McRaw Hill, 2015.
2. Handbook of Data Structures and Applications, **Dinesh P. Mehta and Sartaj Sahni**, Chapman and Hall/CRC, 2018.

Reference Books:

1. Data Structures and Algorithm in C++, **Adam Drozdek Thomson,** Vikas Publications, 2013.
2. Data Structures & Algorithms, **Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman**, Pearson, 2002.

Course Code: MA5205	DBMS Lab	Credits 0-0-3: 1.5

Pre-Requisites: MA5203

Course Outcomes:

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

CO1	Create the queries using DDL and DML commands
CO2	Construct the queries using the relational constraints, joins, set operations, and aggregate functions
CO3	Implement the integrity constraints on various databases
CO4	Create queries using various data types
CO5	Develop the queries using triggers and assertions

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	2	3	2	2
CO2	2	3	3	3	2	2
CO3	3	2	2	3	2	2
CO4	2	3	2	3	2	2
CO5	3	3	3	2	2	2

Syllabus:

Introduction to SQL: DDL, DML, DCL Statements, Built-in Functions and Aggregate Functions, Sub Query, Nested Sub Queries, Modification of the Database.

Intermediate SQL: Join Expressions, Views, Integrity Constraints, SQL Data Types and Schemes, Authorization.

Advanced SQL: Triggers and Assertions.

PL/SQL, Data types, Control Structures, Error handling mechanism, Subprograms (procedures and functions), Stored procedures, Data base triggers and exception

MA5248 – Seminar
Every student has to give a seminar and a committee of staff members from the department shall evaluate the same. The DAC (PG & R) of the Department shall constitute this committee.

M.Sc. (Mathematics & Scientific Computing):: II Year II Semester

Course Code: MA5251	Computation Lab	Credits 3-0-0: 3

Pre-Requisites: MA4204

Course Outcomes:

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

CO1	Solve nonlinear equations using iterative methods
CO2	Solve Simultaneous Linear Algebraic Equations using direct and iterative methods
CO3	Find the unknown values using the interpolation techniques
CO4	Solve the ordinary differential equations.
CO5	Fit the data using the linear and polynomial regressions models

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2		3	1		1
CO2	2	1	2	2		3
CO3			2			2
CO4	3	2	2	2		2
CO5	1					3

Syllabus:

Write programs for the following using c++

1. Method of successive bisection
2. Method of false position
3. Newton-Raphson iterative method
4. Secant method
5. Gauss elimination method
6. Gauss-Seidel iterative method
7. Lagrange interpolation
8. Linear regression
9. Polynomial regression
10. Simpson's rule
11. Euler's method
12. Taylor series method
13. Runge-Kutta method of second and fourth order formulae

MA5297 – Comprehensive Viva
Each student has to appear for comprehensive viva in front of the panel of examiners.

MA5299 – Dissertation Work
Each student will be allotted to a faculty member by DAC (PG & R) for this term project work. It has to be carried out on a topic chosen by the student in consultation with the supervisor. The project supervisor will periodically review the student progress over the semester. The progress of the work will be evaluated in the middle of the semester by DAC (PG & R). Finally, the student has to submit the project report before the end examinations of the IV semester and present the findings for the final evaluation.

List of Electives for Applied Mathematics

M.Sc. (Mathematics & Scientific Computing):: I Year II Semester

Course Code: MA4261	Graph Theory	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Examine whether the graphs are isomorphic or not
CO2	Determine whether graphs are Hamiltonian and/or Eulerian
CO3	Construct minimal spanning trees and shortest paths
CO4	Determine the matching in a graph and solve the assignment problem
CO5	Construct planar graphs, colouring of graphs and their applications

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1				1	
CO2	1				1	
CO3	3	2	2	3		1
CO4	2	1	2	2		
CO5	3	2	3	3	2	1

Syllabus:

Preliminary Concepts: Graph definition, various kinds of graphs; Incidence matrix; Isomorphism; Decomposition; Special graphs; Paths, cycles and trails - connection in graphs, bipartite graphs, Eulerian Circuits; Vertex degree and counting, Hamiltonian Cycles - necessary and sufficient conditions; Review of digraphs.

Trees: Trees and distance - properties; Spanning trees; Kruskal and Prim algorithms with proofs of correctness; Shortest paths - Dijkstra's algorithm, BFS and DFS algorithms, Application to Chinese postman problem; Trees in Computer science - rooted trees, binary trees, Huffman's Algorithm.

Matchings: Matching in a graph and maximum matchings; Hall's matching theorem; Maximum bipartite matching - Augmenting path algorithm;

Connectivity: Connectivity; Characterizing 2-connected graphs; Menger's theorem; Network flow problems-Ford-Fulkerson labeling algorithm, Max-flow Min-cut Theorem.

Coloring: Chromatic number; Greedy coloring algorithm; Brooks' theorem; Graphs with large chromatic number; Turan's theorem.

Planar Graphs: Planar graphs; Euler's formula, dual of a plane graph; Kuratowski's Theorem; Five Color Theorem; Four Colour Problem.

Learning Resources:

Text Books:

1. Introduction to Graph Theory, **Douglas B**. West, Pearson, 2015, Second Edition.
2. Graph Theory, **R. Diestel**, Springer, 2017, Fifth Edition.

Reference Books:

1. Graph Theory with Applications to Engineering and Computer Science, **Narsingh Deo**, Prentice-Hall, 1979.
2. Graph Theory, **J. A. Bondy and U. S. R. Murty**, Springer, 2008.

Online Resources:

1. https://www.youtube.com/watch?v=E40r8DWgG40&list=PLEAYkSg4uSQ2fXcfrTGZdPuTmv98bnFY5
2. https://www.youtube.com/watch?v=f1JTtMP6NGw&list=PLJ5C_6qdAvBF0v3uOhAeDbuCv-Qq9xKj5

Course Code: MA4262	JAVA Programming	Credits 2-0-2: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Introduce the fundamental concepts of java
CO2	Provide a foundation to use basic concepts in Java
CO3	Learn to write Java Scripts.
CO4	Explore various Exception Handling mechanisms
CO5	Provide the basic knowledge to use Java with OOP Terminology.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2		2			
CO2	3	2		2		
CO3	2	2	2	2		
CO4	2	2	1	2	1	
CO5	2	1	1	1	1	

Syllabus:

Java Basics: History of Java, Java buzzwords, comments, data types, variables, constants, scope and life time of variables, operators, operator hierarchy, expressions, type conversion and casting, enumerated types, control flow-block scope, conditional statements, loops, break and continue statements, simple java program, arrays, input and output, formatting output,

Inheritance: Inheritance concept, benefits of inheritance, Super classes and Sub classes, Member access rules, Inheritance hierarchies, super uses, preventing inheritance: final classes and methods, casting, polymorphism- dynamic binding, method overriding, abstract classes and methods, the Object class and its methods.

Interfaces: Interfaces vs. Abstract classes, defining an interface, implementing interfaces, accessing implementations through interface references, extending interface.

Files: Files – streams- byte streams, character streams, text Input/output, binary input/output, random access file operations, File management using File class, Using java.io.

Networking: Networking in Java – Introduction, Manipulating URLs, Ex. Client/Server Interaction with Stream Socket Connections, Connectionless Client/Server Interaction with Datagrams, Using java.net.

Exception handling: Dealing with errors, benefits of exception handling, the classification of exceptions- exception hierarchy, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, rethrowing exceptions, exception specification, built in exceptions, creating own exception sub classes, Guide lines for proper use of exceptions.

Learning Resources:

Text Books:

1. Core Java, Volume 1-Fundamentals, **Cay S. Horstmann and G. Cornell**, Pearson, 2013, Ninth Edition.
2. Java: the complete reference, **H. Schildt**, McGraw Hill, 2011, Seventh Edition.
3. Java for Programmers, **P.J. Deitel and H.M.Deitel**, Pearson, 2009, Second Edition.

Reference Books:

1. Java Programming, **D.S. Malik**, Cengage Learning, 2009 and Edition.
2. An introduction to Java programming and object-oriented application development, **R.A. Johnson**, Cengage Learning, 2007.

Course Code: MA4263	Advanced Abstract Algebra	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Analyze the structure of groups
CO2	Distinguish the properties among ring structures
CO3	Understand extension of fields and their constructions
CO4	Apply the concepts and results to solve problems of abstract algebra
CO5	Construct proofs that arise in various algebraic structures

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3		2			
CO2	3		2			
CO3	3		2			
CO4	3		2	1		
CO5	3		2	1		

Syllabus:

Groups: Actions on a group; Conjugacy classes, class equation, Sylow Theorems and applications;

Rings: Examples; Quadratic integer rings; ideals, prime and maximal ideals, rings of fractions; Chinese Remainder theorem;

Classes of Rings: Euclidean Domains – norms, division algorithm, field norm on Quadratic integer rings, properties; Principal Ideal Domains – properties and results; Unique Factorizations Domains – prime elements, irreducible elements, associates, properties; Polynomial rings over fields, polynomial rings that are UFDs, irreducibility criteria;

Fields: Algebraic and transcendental elements; Brief introduction to Field extensions, Finite fields;

Learning Resources:

Text Books:

1. Abstract Algebra, **David S. Dummit and Richard M. Foote**, John Wiley & Sons, 2004, Third Edition
2. Algebra, **Michael Artin**, Pearson, 2016, Second Edition

Reference Books:

1. Contemporary Abstract Algebra, **Joseph A. Gallian**, Cengage Learning, 2013, Eighth Edition
2. Topics in Algebra, **I. N. Herstein**, John Wiley & Sons, 1975, Second Edition.

M.Sc. (Mathematics & Scientific Computing):: II Year I Semester

Course Code: MA5211	Symbolic Computing	Credits 2-0-2: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Introduce the fundamental concepts of symbolic computing
CO2	Provide a foundation to use basic building blocks of Mathematica and Matlab
CO3	Learn to write Mathematica and Matlab Scripts.
CO4	Explore various applications of Matlab in Mathematics
CO5	Provide the basic knowledge to use Matlab for programming.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2		3			
CO2	2	3			2	
CO3		2	2			
CO4	3	2	3	2		3
CO5	3				2	2

Syllabus:

Introduction to Mathematica: Difference between Numeric computing and Symbolic computing, Parts of Mathematica, Basics of programming in Mathematica, Built-in functions and constants, Numeric calculation using Mathematica, Symbolic computing with Mathematica,

Built-in function for Matrices and Linear Algebra, solving equations, Calculus with Mathematica, Solving ordinary Differential equations with Mathematica.

Graphics and built-in graphics functions, User defined functions, Conditionals and looping in Mathematica. Modules, Simple programs using Mathematica**.**

Introduction to MATLAB, Advantages and disadvantages of MATLAB, MATLAB environment, MATLAB basics, Programming in MATLAB, Built-in functions.

Application to Linear algebra, curve fitting and interpolation, numerical integration and solving Ordinary differential equations. Branching statements, loops and program design,

User defined functions, Input and output functions, introduction to plotting, handling Graphics.

Learning Resources:

Text Books:

1. Mathematica, **Paul R.Wellin**, Wolfram Research Inc., 2005.
2. Numerical Computing with MATLAB, **Cleve Moler**, SIAM, 2004.

Course Code: MA5212	Neural Networks	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Learn different types of neural networks and different types of learning models
CO2	Determine the mathematical foundations of neural network models
CO3	Implement of neural networks using training algorithms such as the feed-forward, back-propagation algorithm
CO4	Design neural networks for practical purposes
CO5	Build neural networks for practical purposes

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2			1		
CO2	3	2		1		
CO3				3		
CO4		2		3		
CO5	2			3		

Syllabus:

Introductory Concepts: 'Neurons' and their basic function- Math review- Mathematical Machinery and Review- How and Why Perceptrons Can Compute Logic Statements- Training Perceptrons Using Supervised Learning Techniques- Training Multi-layer

Neural Networks Using Supervised Learning Techniques: Recurrent Neural Networks and Unsupervised Learning: Optimization Techniques-Implementation and Performance Considerations-Variations on the Hopfield Network-A Stochastic Version of the Hopfield Network:

The Boltzmann Machine-A Stochastic Version of the Binary Associative Memory: Restricted Boltzmann Machines-Competitive Learning and Self-Organizing Maps-Neural Network Modifications and Applications-Cellular Neural Networks and the Future of Massively Parallel Computation

Introduction to Machine Learning Techniques: Types of learning, hypothesis space and inductive bias, evaluation, cross-validation. Linear regression, Decision trees, overfitting. Support Vector Machine, Kernel function and Kernel SVM. Neural network: Perceptron, multilayer network, backpropagation, introduction to deep neural network.

Learning Resources:

Text Books:

1. Neural Networks - A Systematic Introduction, **Raul Rojas**, Springer-Verlag, Berlin, New-York,1996.

2. Biophysics of Computation: Information Processing in Single Neurons, **Koch, Christof**, Oxford University Press, 2004.

Reference Books:

1. Neural Networks Methodology and Applications, **G. Dreyfus, Springer, 2004.**
Neural Networks Algorithms, Applications, and Programming Techniques, **James A. Freeman David M. Skapura,** Addison-Wesley Publishing Company, 1991.

Course Code: MA5213	Wavelet Analysis	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Expand a function in Haar wavelets
CO2	Construct Meyer wavelets to a given function
CO3	Find Daubechies wavelet series to a given function
CO4	Analyse two or more dimensional problems using wavelets

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	2	2	3
CO2	3	3	3	3	2	3
CO3	3	3	3	2	3	2
CO4	3	3	3	3	3	3

Syllabus:

Haar Wavelets: Heuristic treatment of the wavelet transform – Wavelet transform – Haar wavelet expansion: Haar functions and Haar series, Haar sums and Dyadic projections, Completeness of the Haar functions, Haar series in C_0 and L_p spaces, Point wise convergence of Haar series, Construction of standard Brownian motion, Haar function representation of Brownian motion

Multi-resolution analysis: Orthogonal systems, Scaling functions, from scaling function to MRA, Meyer wavelets, from scaling function to ortho-normal wavelet; Wavelets with compact support - From scaling filter to scaling function, Explicit representation of compact wavelets, Daubechies recipe, Hernandez-Weiss recipe, Smoothness of wavelets

Convergence properties of wavelet expansions: Wavelet series in L^p spaces, Large scale analysis, Almost everywhere convergence, Convergence at a pre-assigned point; Wavelets in several variables - Tensor product of wavelets, general formulation of MRA and wavelets in R^d, Examples of wavelets in R^d.

Learning Resources:

Text Books:

1. Introduction to Fourier Analysis and Wavelets, **Mark A. Pinsky**; Cenage Learning India Pvt. Ltd, 2002.
2. Wavelets Theory: Applications Implementation, **M. V. Altaisky**; University Press, 2009

Reference Books:

1. An Introduction to Wavelet Analysis, **Walnut**, **David F,** Springer Nature Switzerland AG 2021
2. Wavelet Analysis Basic Concepts and Applications, **Sabrine Arfaoui, Anouar Ben Mabrouk, Carlo Cattani**, Chapman and Hall/CRC, 2021.
3. An Introduction to Wavelets Through Linear Algebra, **Michael W. Frazier,** Springer-Verlag New York, 1999, First Edition.

Course Code: MA5214	Computer Graphics	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Understand the structure of an interactive computer graphics system, and the separation of system components
CO2	Develop and analyses the algorithms for generation lines and polygons
CO3	Apply the geometrical transformations on objects
CO4	Implement the techniques for segmentation
CO5	Differentiate different techniques for windowing and clipping

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3					
CO2		2		2		
CO3						2
CO4					1	
CO5			3			2

Syllabus:

Introduction: Pixels and frame buffers - Coordinate systems - vector generation - line drawing and circle generation - algorithms and initializing of lines - thick line segments - character generation - display file and its structure**.**

Polygons: Polygon representation - inside test - filling of polygon - 2D Transformations: Matrices - coordinate transformations - rotation about an arbitrary point - other transformations and inverse transformations.

Segments: Segment table - operations on segments - image transformation and other display file structures.

Windowing and Clipping: The viewing transformations - clipping - the cohen sutherland outcode algorithm - the sutherland Hodgman algorithm - clipping of polygons and multiple windowing.

3D Transformations: Rotation about an arbitrary axis - parallel projection - respective projection - Clipping in three dimensions - clipping planes and 3D viewing transformations.

Hidden surfaces and Lines: Back face algorithms, Z buffers - scan line algorithms - the painter's algorithms - warnock's algorithm - Franklin algorithm and hidden line methods

Shading: Shading equations - smooth shading - Gouraud and phong shading methods - shadows. Curves and Fractals: Curve generation - interpolation - B-Splines - Benzier curves - fractal lines and fractal surfaces.

Learning Resources:

Text Books:

1. Computer Graphics - A Programming Approach, **S.Harrington**, McGraw Hill, New York, 1983.
2. Mathematical Elements of Computer Graphics, **D.F.Rogers & J.A.Adams**, McGraw Hill, New York, 1990.

Reference books:

1. Computer Graphics, **Desai, Apurva A,** PHI, New Delhi, 2008.
2. Computer Graphics, **SAMIT BHATTACHARYA**, Oxford University Press, 2018.

M.Sc. (Mathematics & Scientific Computing):: II Year I Semester

Course Code: MA5215	Data Analysis with R	Credits 2-0-2: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	manage and manipulate a data in R
CO2	extract data from various sources and store
CO3	compare and analyse data
CO4	build statistical models for the data

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	2	2	2	3
CO2	3	2	3	3	2	2
CO3	3	3	3	3	2	3
CO4	3	3	3	3	3	3

Syllabus:

Basics of R: Variables, Data types, vectors, calling functions, missing data.

Advanced Data Structures: Data frames, Lists, Matrices and Arrays.

Reading Data into R: Reading CSVs, Excel Data, Data from statistical tools, Data from websites.

Functions and control statements: function arguments, return values, do.call, if and else, switch, for loops, while loops.

Group Manipulation: Apply family, aggregate, plyr, data.table.

Probability and Statistics: Normal, Binomial and Poisson distributions. dbinom(), pbinom(), qbinom(), rbinom() for probability distributions. Correlation and covariance, t test, ANOVA.

Linear Models: Simple linear regression, multiple regression, logistic regression, Poisson regression, survival analysis.

Modal diagnostics: residuals, comparing modals, cross validation, bootstrap.

Learning Resources:

Text Books:

1. R for every one- Advanced analysis and graphics, **Jared P Lander,** Pearson Education 2014
2. Introductory Statistics with R, **Peter Dalgaard**, Springer Science Business Media, LLC 2008

Reference Books:

1. Hands on programming with R, **Garrett Grolemund,** Oreilly, SPD (Shroff Publications and Distributors Pvt Ltd), 2014.
2. Understanding Statistics Using R, **Randall Schumacker and Sara Tomek**, Springer Science Business Media New York, 2013.

Course Code: MA5216	Data Mining	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

At the end of the course, the student will be able to

CO1	Interpret data mining tasks and issues
CO2	Apply different techniques for data mining
CO3	Analyze multi-dimensional modelling techniques
CO4	Implement of the clustering techniques
CO5	Evaluate the performance of algorithms for association rules

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2			2		
CO2			1	3		
CO3				2	1	2
CO4	2	2		2		
CO5		2	1			

Syllabus:

Introduction: Basic Data Mining Tasks, Data Mining Issues, Data Mining Metrics, Data Mining from a Database Perspective.

Data Mining Techniques: A Statistical Perspective on Data Mining, Similarity Measures, Decision Trees, Neural Networks, Genetic Algorithms.

Classification: Statistical-Based Algorithms, Distance-Based Algorithms, Decision Tree-Based Algorithms, Neural Network-Based Algorithms, Rule-Based Algorithms, Combining Techniques.

Clustering: Similarity and Distance Measures, Hierarchical Algorithms, Partitional Algorithms, Clustering Large Databases, Clustering with Categorical Attributes.

Association Rules: Basic Algorithms, Incremental Rules, Advanced Association Rule Techniques, Measuring the Quality of Rules.

Advanced Techniques: Web Mining, Spatial Mining and Temporal Mining.

Learning Resources:

Text Books:

1. Data Mining: Concepts and Techniques, **J. Han, M. Kamber and Jian Pei**, Elsevier. 2007, Third Edition.
2. Data Mining: Introductory and Advanced Topics, **M. H. Dunham**, Pearson Education. 2006.

Reference Books:

1. Data Mining: Practical Machine Learning Tools and Techniques, **I. H. Witten and E. Frank**. Morgan Kaufmann. 2008.
2. Principles of Data Mining, **D. Hand, H. Mannila and P. Smyth**, Prentice-Hall. 2009.
3. Data mining methods and models, **Daniel Larose**, Wiley, 2006.

Course Code: MA5217	Algebraic Coding Theory	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Use the main algebraic tools utilized in coding theory
CO2	Classify the linear codes based on their equivalence
CO3	Understand the geometry of linear codes
CO4	Apply abstract algebra concepts in encoding and decoding of linear codes
CO5	Analyze the parameters governing the codes for identification in real-life applications.

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	3	1	2
CO2	3	3	2	3	1	2
CO3	3	3	2	3	1	2
CO4	3	3	2	3	2	2
CO5	3	3	3	3	2	2

Syllabus:

Linear Codes: Codes as a subspace of vector space over finite fields, Hamming weight, Generator and Parity-check matrices, Equivalence of linear codes, Encoding, decoding of linear codes, cosets and decoding.

Bounds: Sphere-covering bound, Gilbert-Varshamov bound, Hamming bound, Binary and q-ary Hamming codes, Perfect codes, Singleton bound and MDS codes, non-linear codes.

Cyclic Codes: Generator polynomial, Cyclic codes as an ideal of quotient ring, Decoding of cyclic codes.

Special Codes: BCH codes and their parameters, decoding of BCH codes, Reed-Solomon (RS) codes, Quadratic-residue codes.

Goppa Codes: Generalized Reed-Solomon codes, Alternant codes, Goppa Codes, Post-quantum cryptography - Application in McEliece Public Key Cryptosystem.

Learning Resources:

Text Books:

1. Coding Theory: A First Course, **San Ling and Chaoping Xing**, Cambridge University Press, 2004
2. Introduction to the Theory of Error-Correcting Codes, **Vera Pless**, Wiley-Interscience, 1998, Third Edition

Reference Books:

1. Fundamentals of Error Correcting Codes, **W. Cary Huffman and Vera Pless**, Cambridge University Press, 2010.
2. Applied Abstract Algebra, **Rudolf Lidl and Gunter Pilz**, Springer-Verlag, 2004, Second Edition.
3. The Theory of Error-Correcting Codes, **F. J. MacWilliams and N. J. A. Sloane**, North-Holland Publishers, 1985, Revised Edition.

Course Code: MA5218	Parallel Computing	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

CO1	Understand the fundamentals of parallel computing
CO2	Design and analyze the parallel algorithms for solving linear system of equations
CO3	Design parallel algorithms for eigenvalue problems
CO4	Develop parallel algorithm code for iterative methods
CO5	Apply parallel computing method for solving simple differential equations

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	1				
CO2		2	2			
CO3						
CO4				2		2
CO5	3					

Syllabus:

Introduction to parallel programming: Data parallelism, functional parallelism, pipelining, Flynn's taxonomy, parallel algorithm design, task/channel model, Foster's design methodology, case studies: boundary value problem, finding the maximum, *n*-body problem, Speedup and efficiency, Amdahl's law, Gustafson-Barsis's Law, Karp-Flatt Metric, Isoefficiency metric.

Shared Memory: Fundamentals of Shared Memory Programming, Basic OpenMP concepts, PARALLEL directive, Data scoping rules, Basic OpenMP constructs/directives/calls, Examples, parallelizing an existing code using OpenMP, More advanced OpenMP directives & functions, OpenMP Performance issues, Fundamentals of Distributed Memory Programming, MPI concepts - Point to Point Communications.

The message-passing model: the message-passing interface, MPI standard, basic concepts of MPI: MPI_Init, MPI_Comm_size, MPI_Comm_rank, MPI_Send, MPI_Recv, MPI_Finalize timing the MPI programs: MPI_Wtime, MPI_Wtick, collective communication: MPI_Reduce, MPI_Barrier, MPI_Bcast, MPI_Gather, MPI_Scatter, case studies: the sieve of Eratosthenes, Floyd's algorithm, Matrix-vector multiplication.

Monte Carlo methods: parallel random number generators, case studies, Matrix multiplication, row wise block-stripped algorithm, Cannon's algorithm, solving linear systems, back substitution, Gaussian elimination.

Sorting algorithms: quicksort, parallel quick sort, hyper quick sort, sorting by regular sampling, Fast Fourier transform, combinatorial search, divide and conquer, parallel backtrack search.

Learning Resources:

Text Books:

1. Parallel Computing – Theory and Practice, **M. J. Quinn,** Tata McGraw-Hill Publishing Company Ltd., 2002, Second Edition.
2. Parallel Programming–Techniques and applications using networked workstations and parallel computers, **B. Wilkinson and M. Allen,** Pearson Education, 2005, Second Edition.

Reference Books:

1. Parallel Programming in C with MPI and OpenMP, **Michael J. Quinn,** Tata McGraw-Hill Publishing Company Ltd., 2003.

M.Sc. (Mathematics & Scientific Computing):: II Year II Semester

Course Code: MA5261	Evolutionary Optimization Techniques	Credits 3-0-0: 3

Pre-Requisites: MA4251

Course Outcomes:

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

CO1	Differentiate and classify traditional and non-traditional optimization methods
CO2	Formulate an optimization problem to solve complex problems
CO3	Design GA for various optimization problems
CO4	Apply PSO and ACO algorithms for various optimization problems

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	1	2		
CO2	2	2	1	2		
CO3	3	2	2	2		
CO4	3	2	2	2		

Syllabus:

Problem Solving Methods: Problem Space, Problem solving, State space, Algorithm's performance and Complexity, Search Algorithms, Depth first search method, Breadth first search method, Branch and Bound search method, Introduction to P type, NP complete and NP Hard problems. Classical methods versus Non-traditional methods.

Evolutionary Methods: Principles of Evolutionary Processes and genetics, Introduction to evolutionary algorithms, Evolutionary strategy, Evolutionary programming.

Genetic Algorithm: Basic concepts, working principle, procedures of GA, flow chart of GA, Genetic representations, (encoding) Initialization and selection, Genetic operators, Mutation, Generational Cycle, Genetic programming, Simple applications.

Swarm Optimization: Introduction to Swarm intelligence, Ant colony optimization (ACO), Metaheuristic, Algorithm for Travelling Salesman Problem, Particle swarm optimization (PSO), Other variants of swarm intelligence algorithms, Simple problems and applications.

Learning Resources:

Text Books:

1. Multi-objective Optimization using Evolutionary Algorithms, **K. Deb**, John Wiley and Sons, 2001, First Edition.
2. An introduction to metaheuristics for optimization, **B. Chopard & M. Tomassini**, Springer International Publishing, 2018, First Edition.

Reference Books:

1. Metaheuristics for hard optimization: methods and case studies, **J. Dréo, A. Pétrowski, P. Siarry & E. Taillard**, Springer Science & Business Media, 2006, First Edition.
2. Engineering optimization: theory and practice, **S. S. Rao**, John Wiley & Sons, 2019, Fifth Edition.

Course Code: MA5262	Management Information Systems	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Determine the key terminologies and concepts of MIS
CO2	Design Information Technology solutions for business problems
CO3	Develop Information Technology solutions for business problems
CO4	Understand ethical issues that occur in business
CO5	Plan projects, work in team settings and deliver project outcomes in time

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1				1		
CO2				3	2	
CO3				3		
CO4				2		
CO5					3	

Syllabus:

Management Information System: Meaning, Nature, Need, Role, Importance, Evolution of Management through Information system. Relatedness of MIS with management activities, Management functions and decision making. Concept of 'Balanced MIS' effectiveness and efficiency criteria.

Development of MIS: Information System Planning – Methodology and Tools/Techniques for systematic identification, evaluation, modification of MIS. A study of major financial, production, manpower and marketing MIS. End user computing and development.

Advanced MIS: concept, need and problems in achieving advanced MIS, Decision support systems, Export systems. Rationale of computer application.

Learning Resources:

Text Books:

1. Management Information Systems- Strategy and Action, **Charles S.Parker**, McGraw Hill Intl. Edition., 1989.
2. Management Information Systems, **James O'Brian**, McGraw Hill, 1998.

Reference books:

1. Principles of Management Information Systems, **George M. Scott**, McGraw-Hill, 2003.
2. Management Management Information Systems: Conceptual Foundations, Structure and Development, **Gordon B. Davis, Margrethe H. Olson,** McGraw-Hill, 1985, Second Edition.

Course Code: MA5263	Computational Number Theory	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Solve the system of congruences
CO2	Test quadratic residuosity
CO3	Know the basic arithmetic of finite fields
CO4	Apply Lenstra-Lenstra-Lovasz (LLL) algorithm for factoring polynomials with integer coefficients
CO5	Apply CFRAC method or the elliptic curve method for integer factorization

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1					
CO2			3			
CO3	2					
CO4			3	1		
CO5			3	1		

Syllabus:

Algorithms for integer arithmetic: Divisibility, gcd, prime number theorem, modular arithmetic, modular exponentiation, congruence, Chinese remainder theorem (CRT), Hensel lifting, orders and primitive roots, quadratic residues, modular square roots, continued fractions and CFRAC method for integer factoring.

Representation of finite fields: Prime and extension fields, representation of extension fields, polynomial basis, finite field arithmetic, primitive elements, normal basis, optimal normal basis, irreducible polynomials.

Algorithms for polynomials: Root-finding and factorization, Lenstra-Lenstra-Lovasz (LLL) algorithm, polynomials over finite fields.

Elliptic curves: The elliptic curve group, elliptic curves over finite fields, pairing on elliptic curves, elliptic curve method for integer factoring.

Learning Resources:

Text Books:

1. A Computational Introduction to Number Theory and Algebra: **Victor Shoup;** Version 2, Cambridge University Press, 2008. (available at https://www.shoup.net/ntb/)
2. Computational number theory: **Abhijit Das**, CRC press, 2015.
3. Introduction to Elliptic Curves and Modular Form; **Neal Koblitz**; Springer, 1984, Second Edition.

Reference Books:

1. An Introduction to the Theory of Numbers: **I. Niven, H. S. Zuckerman and H. L. Montgomery**; Fifty Edition, John Wiley & Sons, 1991.
2. Elementary Number Theory & Its Applications: **Kenneth H. Rosen;** Sixth Edition, Pearson, 2011.
3. The Arithmetic of Elliptic Curves; **Joseph H. Silverman**; Second Edition, Springer, 2009.

Course Code: MA5264	Fuzzy Mathematics and Applications	Credits 3-0-0: 3

Pre-Requisites: MA4251

Course Outcomes:

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

CO1	Understand the relation between imprecise data and fuzzy sets
CO2	Deal with arithmetic operations of fuzzy numbers
CO3	Understand fuzzy rule-based implications and approximate reasoning
CO4	Distinguish between the possibility and probability
CO5	Apply fuzzy tools to solve optimization problems

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2		2		
CO2	1	1		2		
CO3	2	1		2		
CO4	2	2		2		
CO5	3	3		3		

Syllabus:

Fuzzy sets and uncertainty: Basic concepts of fuzzy sets and fuzzy logic, Motivation, Fuzzy sets and their representations, Membership functions and their designing, Operations on fuzzy sets, Convex fuzzy sets, Alpha-level cuts, Geometric interpretation of fuzzy sets. Fuzzy extension principle and its application.

Fuzzy arithmetic: Fuzzy numbers, Fuzzy numbers in the set of integers, Arithmetic operations on fuzzy numbers.

Fuzzy Relations: Linguistic variables, Linguistic modifiers, Fuzzy rules, Fuzzy relations, Basic properties of fuzzy relations, Composition of fuzzy relations.
Fuzzy reasoning: Fuzzy mapping rules and fuzzy implication rules, Fuzzy rule-based models for function approximation.

Possibility theory: Fuzzy logic, Truth, Propositions of fuzzy logic, Fuzzy logic and probability theory, Possibility and Necessity, Possibility versus probability, Probability of a fuzzy event, Bayes' theorem for fuzzy events, Probabilistic interpretation of fuzzy sets.

Fuzzy optimization: Decision making in Fuzzy environment, Fuzzy Multi criteria decision making, Fuzzy Linear programming.

Learning Resources:

Text Books:

1. Fuzzy set theory and its applications, **H. J. Zimmermann**, Springer Science & Business Media, 2011, Fourth Edition.
2. First course on fuzzy theory and applications, **K. H. Lee**, Springer Science & Business Media, 2005, First Edition.

Reference Books:

1. Fuzzy Systems Engineering: Toward Human-Centric Computing, **W. Pedrycz & F. Gomide,** Wiley – IEEE, 2007, First Edition.
2. Fuzzy logic with engineering applications, **T. J. Ross**, John Wiley & Sons, 2016, Fourth Edition.
3. Fuzzy Sets and Fuzzy Logic: Theory and Applications, **G. J. Klir & B. Yuan**, Prentice-Hall of India Pvt. Limited, 2008, First Edition (Reprint).

Course Code: MA5265	Perturbation Methods	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Solve perturbation problems in differential equations
CO2	Understand boundary layer in fluid flow problems
CO3	Understand regular and singular perturbation theory
CO4	Use asymptotic expansions to solve perturbation problems

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	3			2
CO2	3	2	3			1
CO3	3	1	2			2
CO4	3	2	2			2

Syllabus:

Introduction: Parameter perturbations, Coordinate perturbations, Order symbols and Gauge functions, Asymptotic expansions and sequences, Convergent versus asymptotic series, Nonuniform expansions, Elementary operations on asymptotic expansions.

Straight forward expansions and sources of nonuniformity: Infinite domains, A small parameter multiplying highest derivative. Type change of partial differential equations, The presence of singularities, The role of coordinate systems.

The method of strained coordinates: The method of strained parameters, Lighthill's technique, Temple's technique, Renormalization technique, Limitations of the method of strained coordinates.

The methods of Matched and Composite Asymptotic Expansions: The methods of matched asymptotic expansions, The methods of composite asymptotic expansions.

Variation of Parameters and Methods of Averaging: Variation of parameters, The method of averaging, Struble's technique, The Krylov-Bogoliubov-Mitropolski technique, The method of averaging by using canonical variables, Von-Zeipel's procedure, averaging by using the Lie series and transforms, averaging by using Lagrangians.

The method of Multiple Scales: Description of method, Applications of the derivative, Expansion method, The two variable expansion procedure, Generalized method.

Learning Resources:

Text Books:

1. Perturbation Methods, **A. H. Nayfeh,** Wiley, New York, 2008.
2. Introduction to Perturbation Techniques, **A.H. Nayfeh,** John Wiley & Sons, 2011.

Reference Books:

1. Advanced Mathematical Methods for Scientists and Engineers I: Asymptotic Methods and Perturbation Theory, **Carl M. Bender Steven A. Orszag**, Springer-Verlag New York, 1999, First Edition.
2. Perturbation Methods in Applied Mathematics, **J. Kevorkian and J. D. Cole**, Springer, New York, 1981, Revised Edition.
3. Perturbation Methods in Fluid Dynamics, **Milton Van Dyke,** Academic Press, 2008, Digitized Version.

M.Sc. (Mathematics & Scientific Computing):: II Year II Semester

Course Code: MA5266	Theory of Automata	Credits 3-0-0: 3

Pre-requisites: NIL

Course Outcomes:

At the end of the course, the student will be able to

CO1	Formulate computer languages
CO2	Construct Finite automata to accept given language
CO3	Construct grammar for a given language
CO4	Classify the languages and arrange in hierarchy
CO5	Design the Turing machines

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	3	2		
CO2	3	3		3		2
CO3	2	2		2		
CO4	2			2		
CO5	3	3	3	2		

Syllabus:

Preliminaries: Sets, Relations, Equivalence relation, partition, Transitive closures, Kleene' closure*, Strings, Alphabets, Languages, Recursive definitions.

Regular Languages and Finite Automata: Regular Expressions, Regular Languages, Finite State Machines, Deterministic finite automata (DFA), Non-deterministic finite automata (NFA), Non-deterministic finite automata with ε moves (NFA- ε), ε -closure, Equivalence of DFA, NFA and NFA- ε, Language accepted by Finite Automata, Kleene's Theorem

Properties of Regular Sets: Properties of the Languages accepted by finite automata, Regular and non-regular languages, Minimal finite automata, Pumping lemma, Myhill - Nerode theorem. Closure properties of Regular languages,

Context Free Languages and Pushdown Automata: Context free grammars (CFG), context free languages (CFL), closure properties of context free languages, Chomsky normal form, Greibach normal form, Pumping lemma for CFL, parsing, Pushdown automata (PDA), CFG for PDA, PDA for CFG, phrase structured grammars and languages and context sensitive grammars and languages.

Turing Machines: Turing machine model, example, Modification of Turing machines, Church's hypothesis and Non-deterministic Turing machines.

Learning Resources:

Text Books:

1. Introduction to Automata Theory, Languages and Computation, **Hopcroft J. and Ullman J.D.**, Narosa Publishing, 1989.
2. Introduction to Languages and the Theory of Computation, **Martin, J.C.**, Tata McGraw Hill, 2009.

Reference Books:

1. Theory of finite automata with an introduction to formal languages, **Carrel J. and Long D.**, Prentice Hall, 1989.
2. An introduction to formal languages and automata, **Peter Linz**, Jones & Bartlett, 1997.

Course Code: MA5267	Mathematics of Data Science	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

At the end of the course, the student will be able to

CO1	Analyze the basics of data science
CO2	Apply PCA
CO3	Analyze spectral clustering
CO4	Compute dimension reduction and clustering of random graphs
CO5	Apply approximation algorithms

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2				1	2
CO2	2	1		2		2
CO3		1	2	2	1	2
CO4			2			2
CO4	2	2				

Syllabus:

Introduction and Algebra: Introduction of Data Science, Visualization of data, Resampling, Distributions, Linear Model & Baysian Model, Simple examples, Gradients of Vector-Valued Functions, Gradients of Matrices, Useful Identities for Computing Gradients, Backpropagation and Automatic Differentiation.

Statistics: Parameter Estimation, Bayesian Linear Regression, Maximum Likelihood as Orthogonal Projection, Principal Component Analysis (PCA), Spectral Clustering, Cheeger's inequality, Concentration of measure and tail bounds in probability. Dimension reduction through Johnson-Lindelstrauss Lemma and Gordon's Escape through a Mesh Theorem.

Graph Theory: Approximation algorithms in Theoretical Computer science and the Max-cut problem, Clustering of random graphs: Stochastic Block model, Synchronization, Inverse problems on graphs.

Optimization: Continuous Optimization, Basics of duality in Optimization, Convex Optimization.

Learning Resources:

Text Books:

1. Data science from scratch, **Joel Grus**, O'Reilly Media, 2015.
2. Getting started with data science, **Murtaza Haider**, IBM Press, 2016.

Reference Books:

1. Data science for Dummies, **Lillian Pierson**, Wiley, 2017, Second Edition.
2. Data Visualization Handbook, **J Koponen & J Hidden**, CRC Press, 2019.
3. Mathematics for Machine Learning, **Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong**, Cambridge University Press, 2019.

Online Resources:

1. https://ocw.mit.edu/courses/mathematics/18-s096-topics-in-mathematics-of-data-science-fall-2015/
2. https://ocw.mit.edu/courses/sloan-school-of-management/15-075j-statistical-thinking-and-data-analysis-fall-2011/

Course Code: MA5268	Elliptic Curves	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Know the basic arithmetic of elliptic curves
CO2	Understand the concept of torsion points and pairings in elliptic curves
CO3	Compute the order of a point and the order of the group of points in elliptic curves over finite fields
CO4	Apply the elliptic curve method for factorization and primality testing
CO5	Learn about elliptic curves over rational and complex fields

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1		1		1	1	
CO2			2	1		
CO3	1					
CO4		2		2	2	
CO5			1		1	

Syllabus:

Basic theory and tools: Weierstrass Equations; Group law on elliptic curves; a little of projective geometry; projective space and the point at infinity; other equations for Elliptic curves and other coordinate systems; the j-invariant; elliptic curves in characteristic 2; Endomorphisms, Frobenius map; Singular curves; Elliptic curves mod n; Torsion points; Division polynomials; Weil pairing; Tate-Lichtenbaum pairing.

Elliptic curves over Finite Fields: Frobenius endomorphism; Hasse theorem; Baby step, Giant step for finding order of a point; Schoof's algorithm for finding order of the group of points; supersingular curves.

Applications: A cryptosystem based on Weil Pairing; Lenstra's factorization algorithm using elliptic curves; Primality testing using elliptic curves.

Number theoretic perspective: An introductory treatment of the topics - Elliptic curves over $\mathbb{Q}$, Elliptic curves over $\mathbb{C}$, Divisors, Pairings, Isogenies.

Learning Resources:

Text Books:

1. Elliptic Curves: Number Theory and Cryptography, **Lawrence C. Washington**, Chapman & Hall/CRC, 2008, Second Edition.
2. The Arithmetic of Elliptic Curves, **Joseph H. Silverman**, Springer, 2009, Second Edition.

Reference Books:

1. Introduction to Elliptic Curves and Modular Form, **Neal Koblitzm**, Springer, 1984, Second Edition.
2. Elliptic Curves, **Anthony W. Knapp**, Princeton University Press, 1992.

Course Code: MA5269	Advanced Theory of Algebraic Codes	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Use the main algebraic tools utilized in coding theory
CO2	Compare the linear codes based on their metric
CO3	Understand the geometrical properties of codes with different metrics
CO4	Make use of partial orders to understand inherent properties of codes with different metrics
CO5	Characterize codes with specific properties

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	2	3	1	2
CO2	3	3	2	3	1	2
CO3	3	3	3	3	2	2
CO4	3	3	2	3	1	2
CO5	3	3	3	3	2	2

Syllabus:

Foundations to Codes over Rings: Modules – basics, free modules, modules over principle rings. Galois rings, finite chain rings, Frobenius rings, Codes and linear codes, Codes over rings, codes as a subset of a module.

Codes with Different Metrics: Lee metric, homogeneous metric and Hamming metric, some bounds, rank metric, RT metric, poset metrics, pomset metric.

Geometric Properties: Minimum distance, Weight distributions, MacWilliams Identities, Covering radius, Codes over Z4.

Codes in Poset Metrics: Partial orders over finite sets, some finite posets including hierarchical posets, metrics induced by posets, linear isometries; some characterization and classification results.

Learning Resources:

Text Books:

1. Codes and Rings: Theory and Practice, **Minjia Shi, Adel Alahmadi and Patrick Sole**, Academic Press, 2017.
2. Poset Codes: Partial Orders, Metrics and Coding Theory, **Marcelo Firer, Marcelo M. S. Alves, J. A. Pinheiro and L. Panek,** Springer, 2018.

Reference Books:

1. Fundamentals of Error Correcting Codes, **W. Cary Huffman and Vera Pless**, Cambridge University Press, 2010.
2. Algebra, **Serge Lang**, Springer, 2002, Third Edition.

Course Code: MA5270	Approximation Theory	Credits 3-0-0: 3

Pre-Requisites: MA4101, MA4204

Course Outcomes:

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

CO1	Understand approximation in normed linear spaces
CO2	Use the properties of Chebyshev polynomials in approximation problems
CO3	Find Generalized approximations
CO4	Approximate function by special functions
CO5	Apply the results in determining the best approximation

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3		2	1		1
CO2	3		2	1		1
CO3	3	1	2	1		1
CO4	3		2	1		1
CO5	3	2	2	1		2

Syllabus:

Linear Chebyshev Approximation: Approximation in normed linear spaces – Existence, uniqueness – approximation of vector-valued functions – Haar subspaces - approximation of real Valued functions on an interval

Chebyshev polynomials: Properties - extremal properties of Chebyshev polynomials - Strong Uniqueness and continuity of metric projection - discretization - discrete best Approximation.

Interpolation: Introduction - algebraic formulation of finite interpolation - Lagrange form - Extended Haar subspaces and Hermite interpolation - Hermite-Fejer interpolation.

Best approximation in normed linear spaces: Introduction - approximative properties of sets - characterization and duality.

Projection: Continuity of metric projections - convexity, solarity and Chebyshevity of sets - best Simultaneous approximation.

Learning Resources:

Text Books:

1. Fundamentals of Approximation Theory, **Hrushikesh N. Mhaskar and Devidas V. Pai**, Narosa Publishing House, 2000.

Reference Books:

1. A Course in Approximation Theory, **Ward Cheney and Will Light**, American Mathematical Society, 2000
2. Introduction to Approximation Theory, **E. W. Cheney**, AMS Chelsea Publication, 1966
3. Best Approximation in Normed Linear Spaces by Elements of Linear Subspaces, **Ivan Singer**, Springer-Verlag, 1970

Course Code: MA5271	Graphical Processing Unit Programming with CUDA	Credits 3-0-0: 3

Pre-Requisites: NIL

Course Outcomes:

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

CO1	Write the code to dynamically allocate the memory.
CO2	Write the CUDA programming
CO3	Able to effectively use the shared memory
CO4	Apply the GPU sorting techniques
CO5	Apply the CUDA streams

Course Articulation Matrix:

	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3					
CO2		2				
CO3			2			2
CO4	2					
CO5	2					

Syllabus:

Introduction: Course Context, Supercomputing and software, Module Details.

Introduction to C: Functions and scoping, Arrays, strings, IO – Pointers, Advanced use of pointer, Dynamically managed memory, Structures and binary files.

Optimisation: Overview, Compute bound code, Memory bound code.

OpenMP: Overview, Loops and Critical Sections, Scoping and Tasks, Parallel Reduction, Scheduling, Nesting and Summary.

GPU Architecture: Introduction to GPUs, Programming GPUs, GPU Hardware.

Introduction to CUDA: The CUDA Programming Model, CUDA Device Code, CUDA Host Code and Memory Management.

CUDA Memory: Memory Overview, Global and Constant Memory, Read only and Texture Memory, Introduction to Shared Memory, Shared Memory Bank Conflicts, Boundary Conditions. Shared Memory Bank Conflict Calculator CUDA Performance, Memory Coalescing, The L1 Cache, Occupancy.

Warp Level CUDA: Scheduling and Divergence, Advanced Divergence - Atomic and Warp Operations.

Parallel Patterns: Overview - Reduction - Scan.

Sorting and Libraries: Sorting (Networks) - Libraries and Thrust - Applications of GPU Sort.

CUDA Streams: Synchronous and Asynchronous Execution - Synchronisation - Multi GPU Programming.

Learning Resources:

Text Books:

1. Understanding and Using C Pointers: Core Techniques for Memory Management, **Richard Reese,** Oreilly publications, 2013.
2. CUDA by Example: An Introduction to General-Purpose GPU Programming, **Edward Kandrot, Jason Sanders**, Addison Wesley 2010.

Reference Books:

1. NVIDIA, CUDA C Programming Guide