

Course Curriculum

B.Sc. (Hons.) Mathematics & Computing

Semester I - 22 Credits

S.No .	Course Title	Course type	L	T	P	Credit s
1	Calculus	Math Core	3	1	0	4
2	Mathematics Foundations	Math Core	3	1	0	4
3	MOOCs (IDE)	IDE	3	0	0	3
4	Modern Indian Language (Preferably other than English)- I (Offline / Online)	AEC	2	0	0	2
5	Modern English Grammar & Pronunciation	SEC	3	0	0	3
6	Indian Knowledge System (MOOCs)	VAC	2	0	0	2
7	Problem Solving and Programming in C	CS (Minor) Core	3	0	2	4
Total Credits						22

ii) Semester II - 22 Credits

S.No.	Course Title	Course Type	L	T	P	Credits
1	Number Theory	Math Core	3	1	0	4
2	MOOCs (IDE)	IDE	3	0	0	3
3	Understanding Disabilities (Offline/Online)	AEC	2	0	0	2
4	Academic Writing	SEC	3	0	0	3
5	Ecology & Environment	VAC	2	0	0	2
6	Introduction to OOPs Using Java	CS (Minor) Core	3	0	2	4
7	Web Application Development	CS (Minor) Core	3	0	2	4
Total Credits						22

iii) Semester III - 21 Credits

S.No.	Course Title	Course Type	L	T	P	Credits
1	Ordinary Differential Equations	Math (Minor) Core	3	1	0	4
2	Probability Theory	Math Core	3	1	0	4
3	MOOCs	IDE	3	0	0	3
4	Modern Indian Language (Preferably other than English)- II (Offline / Online)	AEC	2	0	0	2
5	Introduction to Python Programming	SEC	3	0	2	4
6	Free elective					4
Total Credits						21

iv) Semester IV - 20 Credits

S.No	Course Title	Course type	L	T	P	Credits
1	Linear Algebra	Math (Minor) Core	3	1	0	4
2	Algebra I	Math (Minor) Core	3	1	0	4
3	Building Mathematical Ability and Financial Literacy	AEC	3	1	0	4
4	Fundamentals of Data Science using Python	CS-Minor Core	3	0	2	4

5	Introduction to Machine Learning	CS-Minor Core	3	1	0	4
	Total Credits					20

v) Semester V - 20 Credits

S.No	Course Title	Course type	L	T	P	Credits
1	Real Analysis	Math (Minor) Core	3	1	0	4
2	Algebra II	Math Core	3	1	0	4
3	Operations Research	Math Core	3	1	0	4
4	Introduction to Deep Learning	CS-Minor Core	3	1	0	4
5	Basics of Cloud Computing	CS-Minor Core	3	1	0	4
	Total credits					20

vi) Semester VI - 20 Credits

S.No	Course Title	Course type	L	T	P	Credits
1	Multivariable Calculus	Math Core	3	1	0	4
2	Numerical Analysis	Math Core	3	0	2	4
3		Math Elective	3	1	0	4
4		Math Elective	3	1	0	4
5	Climate Change	VAC	2	0	0	2

6	Summer Internship Project (SIP)	SIP				2
	Total Credits					20

vii) Semester VII - 20 Credits

S.No	Course Title	Course type	L	T	P	Credits
1	MA Elective	Math Elective	3	1	0	4
2	MA Elective	Math Elective	3	1	0	4
3	Dissertation/Industry Project Should have a thesis dissertation of at least 80 pages.					12
	Total credits					20

viii) Semester VIII - 16 Credits

Sr.No	Course Title	Course type	L	T	P	Credits
1	Complex Analysis	Math Core	3	1	0	4
2		Math Elective	3	1	0	4
3		Math Elective	3	1	0	4
4	Big Data Analytics	CS-Minor Core	3	1	0	4

	Total Credits	20
--	----------------------	----

	Course Name	Credits
1	Analysis of functions on metric spaces	4
2	PDE	4
3	Topology	4
4	Field Theory	4
5	Functional Analysis	4
6	Convex Optimization	4

List
of

Maths Electives

iv) Core Syllabus:

Calculus Syllabus (Semester I, Math Core):

Module 1: Sequences and Series: Limit of a sequence, monotone and Cauchy sequences and properties of convergent sequences, examples. Infinite series, positive series, tests for convergence and divergence, integral test, alternating series, Leibnitz test.

Module 2: Differential Calculus: Continuity and differentiability of a function of a single variable, statement of Rolle's Theorem, Lagrange's mean value theorem and applications.

Module 3: Integral Calculus: Definite Integrals as a limit of sums, Improper integrals.

Module 4: Functions of several variables: Continuity and differentiability, mixed partial derivatives, local maxima and minima for function of two variables, Lagrange multipliers.

Module 5: Double and Triple Integrals: Calculations, Areas, Volumes, change of variables, Applications.

Module 6: Integrals of Vector Functions: Line integrals, Green's formula, path independence.

Module 7: Surface integral: definition, evaluation, Stoke's formula, Gauss-Ostrogradsky divergence theorem.

Mathematical Foundations Syllabus (Semester I, Math Core):

Module 1: Relations, Equivalence Relations, Partition of a set.

Module 2: Induction principles, The Well-ordering principle, Countability of sets (finite and countable), Schroder-Bernstein theorem, Pigeonhole principle, Inclusion-Exclusion principle.

Module 3: Order Relations, Posets, Zorn's Lemma, Axiom of Choice.

Module 4: Comparing growth rates of functions, Counting and combinatorics.

Module 5: Linear recurrence relations with constant coefficients, Nonlinear recurrence relations (like Fibonacci recurrence, etc.), Generating functions.

Problem Solving and Programming in C Syllabus (Semester I, CS (Minor) Core):

UNIT -I: Introduction to Problem Solving: Problem Solving Aspect, Problem Identification, Problem Understanding, Algorithm Development, Solution Planning, Flowcharts, flow algorithm. Overview of C: History of C, C Language Elements, Basic Structure of C Program, C Tokens- Variables and Data Types, Operators, Expressions and Type Conversions.

UNIT -II: Control Statements: Selection Statements- if and switch statements. Iterative Statements: for, while and do-while statements. Jump Statements: break and continue statements.

UNIT -III: Arrays: Declaration, accessing array elements, Storing values, Operations on arrays, multi-dimensional arrays. Functions: Introduction, Using

Functions, Function declaration, Function definition and Function call, Parameter passing, Passing arrays to functions, Recursion, Storage classes.

UNIT-IV: Pointers: Declaration and Initialization of pointer variables, Pointer arithmetic, Pointers and arrays, Pointer to pointer, Array of pointers, Pointers and functions, Dynamic Memory Allocation. Strings: Introduction to Strings, String handling functions, Preprocessor Directives.

UNIT-V: Structures: Introduction, Nested Structures, Array of Structures, Structures and Functions, Unions. Command-Line Arguments: Command-line Arguments Files: Introduction, File Operations.

Number theory Syllabus (Semester II, Math Core):

Module 1: Primes, Divisibility, Division Algorithm, Arithmetic functions, Congruences, Chinese Remainder Theorem and its applications,

Module 2: Fermat/Euler theorems, Wilson's theorem, Lagrange's theorem, Primitive roots, Pythagorean triplets,

Module 3: Quadratic Reciprocity, Legendre/Jacobi symbols,

Module 4: Counting primes and their distribution, Prime Number Theorem, Riemann zeta function and applications.

Introduction to OOPs Using Java Syllabus (Semester II, CS (Minor) Core):

UNIT I: Introduction to Java Programming

Theory: Introduction to object-oriented programming. Features of Java, JVM, JDK, and JRE. Structure of a Java program, compilation and execution process. Variables, data types, operators, input/output, control statements, loops, arrays, strings, and command-line arguments.

Lab/Practical: Programs on operators, control structures, arrays, strings, and basic problem-solving using Java.

UNIT II: Classes, Objects and Methods

Theory: Classes and objects, constructors, methods, access specifiers, this keyword, static members, method overloading, object creation and memory management basics. Packages and wrapper classes.

Lab/Practical: Programs using classes, constructors, methods, object interaction, and package implementation.

UNIT III: Core OOP Concepts (Inheritance, Polymorphism, Abstraction, Encapsulation)

Theory: Encapsulation, inheritance and types of inheritance, method overriding, polymorphism, dynamic method dispatch, abstract classes, interfaces, and abstraction principles.

Lab/Practical: Programs implementing inheritance, polymorphism, abstract classes, and interfaces.

UNIT IV: Exception Handling and File Processing

Theory: Exception hierarchy, try-catch-finally, throw and throws, user-defined exceptions. File handling basics, byte and character streams, reading and writing files.

Lab/Practical: Programs on exception handling and file operations.

UNIT V: Collections, Multithreading and GUI Programming

Theory: Introduction to Java Collections Framework: Lists, Sets, Maps, Iterators, and Generics. Multithreading: thread lifecycle, thread creation, synchronization basics. Event-driven programming and introduction to GUI components.

Lab/Practical: Programs using collections, multithreading, and basic GUI/event-driven applications.

Web Application Development Syllabus (Semester II, CS (Minor) Core):

UNIT -I: Course Introduction

Evolution of the Web and Internet, Web Application Architecture, Client-Server Model, HTTP and HTTPS Protocols, Web Browsers and Web Servers, Introduction to URLs, DNS, and Hosting, Static vs Dynamic Websites

UNIT-II: HTML and CSS

HTML: Structure of HTML Documents, Text Formatting and Hyperlinks, Lists, Tables, and Forms, Multimedia Elements; CSS: Using Styles, Simple Examples, Your Own Styles, Properties and Values in Styles, Style Sheet, Formatting Blocks of Information, Layers.

UNIT-III: JavaScript

JavaScript Fundamentals: Variables and Data Types, Operators and Expressions, Control Structures, Functions and Scope, Arrays and Objects, Regular Expressions, Exception Handling.

Advanced JavaScript: DOM Manipulation, Event Handling, Form Validation, JSON, Dynamic HTML

UNIT-IV: Dynamic HTML

Dynamic Content Generation, Interactive User Interfaces, Client-side Validation, Different Frames, Rollover Buttons, Moving Images.

UNIT-V: Backend Development and Database Integration

Server-Side Programming using Node.js and Express.js, API design, basic Database concepts, connecting applications to Databases, Full-stack application development.

Ordinary Differential Equations Syllabus (Semester III, Math (Minor) Core):

Module 1: First order differential equations: Method of separation of variables, homogeneous equations of degree zero, non-homogeneous equations, exact equations, integrating factor, linear equations, Bernoulli's equation

Module 2: First order higher degree equations: solvable for x, y and p. Clairaut's form and singular solutions

Module 3: Picard's method of successive approximations, Picard's theorem for the existence and uniqueness of the solutions of the first order IVPs/Cauchy problems (Statement only). Counter examples related to Picard's theorem

Module 4: Higher order homogeneous linear equations with constant coefficients, second order homogeneous linear equations with variable coefficients

Module 5: Second order linear differential equations: Wronskian, explicit methods to find solutions, method of variation of parameters

Module 6: Power series solutions: Legendre and Bessel functions, properties.

Probability Syllabus (Semester III, Math Core):

Module 1: Random experiments, sample spaces, sets, events, σ -algebras, probability. Elements of combinatorial analysis, classical definition and calculation of probability. Probability Space, Independence and dependence, Bayes theorem and applications.

Module 2: Random variables and distribution Functions, Standard discrete (Bernoulli, Binomial, geometric, negative binomial, Hypergeometric, Poisson

distributions), continuous (Uniform, exponential, Normal, Gamma, beta) distributions. Moment generating functions.

Module 3: Joint distributions, Functions of random variables, Expectation and moments, Conditional expectation, Characteristic functions, Sequences of random variables, Modes of Convergence, Weak and Strong Laws of Large Numbers, Central Limit Theorems (without proofs).

Linear Algebra Syllabus (Semester IV, Math (Minor) Core):

Module 1: System of Linear Equations, Elementary Row Operations, Row-Reduced Echelon Matrices, Gaussian Elimination. Vector Spaces, Subspaces, Direct Sums, Bases and Dimension,

Module 2: Linear Transformations, Kernels, Range Space, Rank-Nullity Theorem, The Matrix of a Linear Transformation, Invertibility, Determinants, Linear Functionals and Adjoints.

Module 3: Eigenvalues and Eigenvectors, Cayley-Hamilton Theorem, Invariant Subspaces, Upper-Triangular Matrices, Diagonal Matrices. The Minimal Polynomial, Diagonalizability, Algebraic, Geometric multiplicities, Generalised Eigenvectors, Rational & Jordan Canonical forms (without proofs).

Module 4: Inner Products, Norms, Orthonormal Bases, Gram - Schmidt process, Orthogonal Projections, Self-adjoint, Normal, and Unitary Operators, Statement of the Spectral Theorem for finite-dimensional operators and its applications.

Algebra-I Syllabus (Semester III, Math (Minor) Core):

Module 1: Binary operation and its properties, Definition of Groups, Basic Examples, Abelian groups, Dihedral Groups, Permutation groups, Symmetric Groups, Matrix Groups.

Module 2: Subgroups, Centralizers and Normalizers, Stabilizers and Normal subgroups, Subgroups Generated by subsets of a Group, Cyclic groups, Coset of a subgroup, Lagrange's theorem.

Module 3: Quotient Groups and Homomorphisms, Automorphisms, Isomorphism Theorems, Composition Series and Jordan-Holder Theorem, Transpositions and the Alternating Group.

Module 4: Group Actions and Permutation Representations, Cayley's Theorem, The Class Equation, Sylow Theorems and its applications, The simplicity of A_n , Direct product of groups, the Fundamental Theorem of finitely generated Abelian groups (without proof).

Fundamentals of Data Science using Python Syllabus (Semester IV, CS (Minor) Core):

UNIT I: The Data Science Ecosystem and Foundations of NumPy

Theory: Introduction to data science, the data science lifecycle, and data privacy/ethics. Roles in data science (Data Scientist vs. Data Engineer vs. ML Engineer). Structured, semi-structured, and unstructured data, Python for Data Science: Overview of Anaconda, Jupyter Notebooks, and Google Colab. The NumPy library: N-dimensional arrays (ndarray), memory layout, and data types. Vectorization, broadcasting rules, indexing, slicing, and boolean masking. Universal functions (ufuncs) and basic statistical functions in NumPy.

Lab/Practical: Setting up the environment; Implementing matrix operations, array manipulation, and performance benchmarking (NumPy vs. native Python loops).

UNIT II: Data Wrangling and Manipulation with Pandas

Theory: Introduction to Pandas. Data structures: Series and Data Frames. Data ingestion from diverse sources (CSV, Excel, JSON, SQL databases, and web APIs). Data cleaning operations: handling missing values (imputation/dropping), removing duplicates, and data type casting. Advanced Wrangling: String manipulation, regular expressions, and date/time data handling. Hierarchical indexing. Data transformation: merging, joining, concatenating, and reshaping (pivot tables, melt). The Split-Apply-Combine paradigm using groupby.

Lab/Practical: Loading a messy real-world dataset (e.g., housing or e-commerce data), cleaning anomalies, and generating aggregated business reports.

UNIT III: Exploratory Data Analysis (EDA) and Visualization

Theory: Philosophy of Exploratory Data Analysis. Visualizing distributions, relationships, and compositions. Principles of effective data storytelling and choosing the right chart type. Avoiding misleading visualizations.

Implementation: Introduction to Matplotlib: figure anatomy, subplots, and customization. Advanced declarative visualization with Seaborn: relational plots (scatterplot, lineplot), categorical plots (boxplot, barplot), and distribution plots (histplot, kdeplot, pairplot). Identifying outliers visually.

Lab/Practical: Performing full-scale EDA on a public dataset (e.g., Kaggle Titanic or COVID-19 tracking), generating a distribution profile, and detecting correlation patterns.

UNIT IV: Statistical Inference and Hypothesis Testing

Theory: Descriptive vs. inferential statistics. Measures of central tendency, dispersion, and shape (skewness, kurtosis). Core probability distributions (Normal, Binomial, Poisson). The Central Limit Theorem (CLT) and its practical significance.

Hypothesis Testing: Formulating null and alternative hypotheses. Type I and Type II errors, p-values, and confidence intervals. Parametric and non-parametric tests: Z-test, T-test (independent and paired), Chi-Square test for independence, and ANOVA. Introduction to the concept of A/B testing.

Lab/Practical: Implementing statistical tests using scipy.stats to validate experimental data, analyze user behaviour, and interpret A/B testing results.

UNIT V: Introduction to Predictive Modelling and Machine Learning

Theory: Transitioning from data analysis to predictive modelling. Overview of supervised vs. unsupervised learning. The machine learning workflow: feature engineering, feature scaling (standardization vs. normalization), data splitting (train-test split, cross-validation), and data leakage prevention. Algorithms & Evaluation: Linear Regression for continuous variables; Logistic Regression and k-Nearest Neighbours (k-NN) for classification. Performance metrics: R-squared, Mean Absolute Error (MAE), Mean Squared Error (MSE), Accuracy, Precision, Recall, F1-score, and the Confusion Matrix.

Lab/Practical: Building a complete end-to-end predictive pipeline using Scikit-Learn: preprocessing data, training a classification/regression model, tuning hyperparameters, and evaluating performance on unseen test data.

Introduction to Machine Learning Syllabus (Semester IV, CS (Minor) Core):

UNIT -I: Introduction: Learning problems, perspectives and issues, concept learning, version spaces and candidate eliminations, inductive bias, decision tree learning, representation, algorithm, Ensemble Learning and Random Forest

UNIT -II: Linear Regression – The Normal equation, Computational complexity, Gradient Descent, Polynomial Regression, Learning Curves, Regularized Linear Models, Logistic Regression, Regularized Logistic Regression, Multi class classification using Logistic Regression, K-Nearest neighbour learning, locally weighted regression, radial basis functions, case-based learning.

UNIT-III: Support Vector Machine: functional and geometric margins, optimum margin classifier, constrained optimization, Lagrange multipliers, primal/dual problems, KKT conditions, dual of the optimum margin classifier, soft margins, kernels, SVM multi class classification, SVM Regression.

UNIT-IV: Bayes theorem, concept learning, maximum likelihood, minimum description length principle, Bayes optimal classifier, Gibbs Algorithm, Naïve Bayes Classifier, Bayesian belief network, EM algorithm, probability learning, sample complexity.

UNIT-V: Hidden Markov Model: Introduction, discrete Markov processes, hidden Markov models, three basic problems of HMMs evaluation problem, finding the state sequence, learning model parameters, continuous observations, the HMM with input, model selection in HMM.

Real Analysis Syllabus (Semester V, Math (Minor) Core):

Module 1: Real number system and its structure: Archimedean property; infimum and supremum; inequalities such as Bernoulli inequality and Cauchy-Schwarz inequality; dense subset; distance between a point and a set; distance between two sets; nested interval theorem.

Module 2: (Sequences and series of real numbers and their convergence): sequences and subsequences of real numbers; convergence and limit of some standard sequences; sequences related to exponential and log functions; monotone sequences; series of real numbers and their convergence; absolute and unconditional convergence; geometric series; tests for convergence such as comparison test, ratio test, and root test; power series and radius of convergence.

Module 3: Limits and Continuity: Algebra of limits, continuous functions, uniform and Lipschitz continuity; Compactness, Heine-Borel theorem; continuous functions and compactness; connectedness; intermediate value theorem.

Module 4: Differentiation of one-variable functions: Properties of differentiation; Product rule and Chain rule; Darboux theorem; Mean value

theorems; Test for local extrema; L'Hospital's rule; Taylor's theorem; Convex functions and differentiation.

Module 5: Riemann Integration: Upper and lower Darboux sums, monotone, continuous functions as integrable functions, Riemann's definition of integration, Fundamental theorem of Calculus, Mean value theorem.

Algebra-II Syllabus (Semester V, Math Core):

Module 1: Definition of Rings, Examples and basic properties, Zero divisors, Integral domains, Fields, Characteristic of a ring, Quotient field of an integral domain.

Module 2: Subrings, Ideals, Prime, Maximal ideals, Quotient rings, Isomorphism theorems. Prime, Irreducible elements, Unique Factorisation Domains, Principal Ideal Domains, and Euclidean domains. Prime avoidance theorem, Chinese remainder theorem.

Module 3: Modules, Free modules, Cartesian products and direct sums of modules, quotient modules, Simple and semisimple modules, isomorphism theorems. Modules over PIDs, Structure theorem for finitely generated modules over PIDs (without proof) and applications.

Module 4: Noetherian and Artinian rings/Modules, Hilbert basis theorem, Jordan-Hölder theorem.

Operations Research Syllabus (Semester V, Math Core):

Module 1: Introduction to Vector spaces, Matrix algebra, Formulation of LPP, Some illustrative examples of LPP formulation. Graphical method of solving LPP in two variables

Module 2: Linear programming model, various forms of LPP. Foundations and theories of LPP: convex sets, feasible solution, basic solution, basic feasible solution, optimal solution. Simplex methods, geometry of polyhedral simplex methods, and its demonstration.

Module 3: Dual of LPP, fundamental theorems of duality, Dual simplex method.

Module 4: Formation and solving assignment Problem, Transportation problem, Dynamic programming problem.

Introduction to Deep Learning and Generative AI Syllabus (Semester V, CS (Minor Core)):

UNIT I: Foundations of Deep Learning & Optimization Theory: Transition from shallow to deep learning. Anatomy of a Neural Network: Perceptrons, Multi-Layer Perceptrons (MLPs), and hidden layers. Activation functions (Sigmoid, Tanh, ReLU, Leaky ReLU, Softmax) and the vanishing/exploding gradient problems. Mathematical derivation of Backpropagation. Optimization & Regularization: Loss functions (MSE, Cross-Entropy). Optimizers: SGD, Momentum, RMSprop, and Adam. Overfitting remedies: L1/L2 regularization, Dropout, Early Stopping, and Batch Normalization. Weight initialization strategies (Xavier, He).

Lab/Practical: Building an MLP from scratch in PyTorch; benchmarking different optimizers and activation functions on the MNIST/Fashion-MNIST dataset.

UNIT II: Deep Learning for Computer Vision (CNNs) Theory: The visual cortex inspiration. Components of a Convolutional Neural Network (CNN): convolution operations, filters/kernels, strides, padding, and pooling layers (Max, Average). Feature map visualization. Classic Architectures: Evolution of CNN architectures—LeNet-5, AlexNet, VGG, and the breakthrough of residual connections in ResNet. Transfer Learning: leveraging pre-trained backbones for custom computer vision tasks.

Lab/Practical: Implementing a custom CNN; fine-tuning a pre-trained ResNet model for a specialized image classification dataset (e.g., medical imaging or defective product detection).

UNIT III: Sequence Models and the Attention Revolution Theory: Handling sequential data (text, time-series). Recurrent Neural Networks (RNNs) and their limitations with long-term dependencies. Gated architectures: Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs). Encoder-Decoder (Seq2Seq) architectures and the informational bottleneck. The Transformer: The Attention mechanism (Scaled Dot-Product Attention). The Transformer architecture: Multi-Head Attention, positional encoding, feed-forward sub-layers, and layer normalization.

Lab/Practical: Implementing an LSTM for sentiment analysis; building a basic text-generation model using a custom-trained Transformer block.

UNIT IV: Classical Generative AI (VAEs and GANs) Theory: Introduction to Generative vs. Discriminative modelling. The concept of latent space

mapping. Autoencoders (AEs) for dimensionality reduction. Variational Autoencoders (VAEs): the reparameterization trick and the Kullback-Leibler (KL) divergence loss. Generative Adversarial Networks (GANs): The minimax game framework. The Generator and Discriminator dynamics. Training challenges: Mode collapse and vanishing gradients. Overview of advanced GAN architectures (DCGAN, CycleGAN).

Lab/Practical: Implementing a VAE to reconstruct and generate handwritten digits; training a DCGAN to synthesize realistic human faces or structural patterns.

UNIT V: Modern Generative AI (LLMs and Diffusion Models) Theory: Large Language Models (LLMs): Autoregressive models (GPT family) vs. Masked Language Models (BERT). Pre-training, Instruction Tuning, and Reinforcement Learning from Human Feedback (RLHF). Parameter-Efficient Fine-Tuning (PEFT) techniques like LoRA (Low-Rank Adaptation). Retrieval-Augmented Generation (RAG) conceptual framework. Diffusion Models: The forward noise injection process and the reverse denoising process (DDPM). Introduction to text-to-image synthesis architectures (Stable Diffusion, Midjourney). AI Ethics: deepfakes, copyright alignment, bias mitigation, and safety guardrails.

Lab/Practical: Fine-tuning an open-source LLM (e.g., LLaMA or Mistral) using LoRA for a domain-specific chatbot; building a local RAG pipeline using a vector database to query institutional documents.

Basics of Cloud Computing Syllabus (Semester V, CS (Minor) Core):

UNIT -I: Introduction to Cloud Computing Fundamentals Cloud computing at a glance, defining a cloud, cloud computing reference model, types of services (IaaS, PaaS, SaaS), cloud deployment models (public, private, hybrid), utility computing, cloud computing characteristics and benefits, cloud service providers (Amazon Web Services, Microsoft Azure, Google AppEngine).

UNIT-II: Cloud Enabling Technologies: Ubiquitous Internet, parallel and distributed computing, elements of parallel computing, hardware architectures for parallel computing (SISD, SIMD, MISD, MIMD), elements of distributed computing, Inter-process communication, technologies for distributed computing, remote procedure calls (RPC), service-oriented architecture (SOA), Web services, virtualization.

UNIT-III: Virtualization and Containers: Characteristics of virtualized environments, taxonomy of virtualization techniques, virtualization and cloud Computing, pros and cons of virtualization, technology examples (XEN, VMware), building blocks of containers, container platforms (LXC, Docker), container orchestration, Docker Swarm and Kubernetes, public cloud VM (e.g. Amazon EC2) and container (e.g. Amazon Elastic Container Service) offerings.

UNIT-IV: Cloud computing challenges Economics of the cloud, cloud interoperability and standards, scalability and fault tolerance, energy efficiency in clouds, federated clouds, cloud computing security, fundamentals of computer security, cloud security architecture, cloud shared responsibility model, security in cloud deployment models.

UNIT -V: Advanced concepts in cloud computing: Serverless computing, Function-as-a-Service, serverless computing architecture, public cloud (e.g. AWS Lambda) and open-source (e.g. OpenFaaS) serverless platforms, Internet of Things (IoT), applications, cloud-centric IoT and layers, edge and fog computing, DevOps, infrastructure-as-code, quantum cloud computing.

Multivariable Calculus Syllabus (Semester VI, Math Core):

Module 1: Functions of several variables, Limits and Continuity, directional derivative, partial derivative, total derivative, Jacobian, chain rule and mean value theorems, higher derivatives, interchange of the order of differentiation, Taylor's theorem.

Module 2: Inverse mapping theorem, implicit function theorem, Extremum problems, extremum problems with constraints, Lagrange's multipliers method.

Module 3: Multiple integrals, existence of integrals, properties of integrals, iterated integrals, Fubini's theorem (without proof), change of variables (without proof)

Module 4: Curl, gradient, div, Laplacian in cylindrical and spherical coordinates, line integrals, surface integrals, Theorems of Green, Gauss, and Stokes.

Numerical Analysis Syllabus (Semester VI, Math Core):

Module 1: Solution of nonlinear equations: Bisection method, secant method, Regula Falsi method, Newton-Raphson method, fixed point iteration, convergence of methods

Module 2: Solution of linear systems: Error and residual of an approximate solution. condition number, Theorems of Gershgorin and Brauer, Jacobi method, Power method for Hermitian matrices, inverse power method.

Module 3: Polynomial interpolation: Existence and uniqueness of an interpolating polynomial, Lagrange interpolation, Newton's divided difference, Newton's divided difference formula, forward difference formula, backward difference formula, Gauss' central difference formula for interpolating polynomial. Error of the interpolating polynomials.

Module 4: Numerical integration: Newton-Cotes closed and open type formulae, error, composite rules, adaptive quadrature, extrapolation to the limit, properties of orthogonal polynomials, Gaussian quadrature.

Module 5: Solution of O.D.E.: Difference equations, explicit and implicit methods, single and multi-step methods - forward, backward Euler methods, mid-point formula, modified Euler's method, Runge-Kutta methods (up to 2nd order O.D.E.).

Complex Analysis Syllabus (Semester VIII, Math Core):

Module 1: Complex plane, Stereographic projection, Analytic Functions, Harmonic Conjugates, Elementary Functions

Module 2: Complex integration, Cauchy Theorem, and Integral Formula, Power Series, Taylor's theorem, Zeros of Analytic functions, Maximum Modulus Principle, Hurwitz Theorem

Module 3: Classification of singularities, Laurent's Theorem, Residues, Residue theorem and applications, Casorati-Weierstrass theorem, Argument Principle, Theorem of Rouché and its applications

Module 4: Linear fractional transformations and Conformal mappings.

Big Data Analytics Syllabus (Semester VIII, CS-Minor Core):

UNIT -I: Introduction to Big Data: Data, Characteristics of Data and Types of Digital Data: Unstructured, Semi-Structured and Structured - Sources of Data. Big Data Evolution - Definition of Big Data-Characteristics and Need of Big Data-Challenges of Big Data. Big Data Analytics, Overview of Business Intelligence.

UNIT-II: Big Data Technologies and Databases Distributed data processing. Hadoop – Requirement of Hadoop Framework -Design Principle of Hadoop –Comparison with Other System SQL And RDBMS-Hadoop Components – Architecture -Hadoop 1 Vs Hadoop 2. Distributed file system and HDFS. Introduction To YARN, Architecture, Managing Resources and Applications with Hadoop YARN.

UNIT-III: MapReduce and YARN Framework Introduction to MapReduce, Processing Data with Hadoop Using Map Reduce.How to design MapReduce algorithms and Hadoop usage patterns. MapReduce in Information Retrieval, information retrieval cycle, Boolean retrieval, ranked retrieval, TFIDF with MapReduce, Graph Data processing with MapReduce.

UNIT-IV: New SQL: Overview of New SQL - Comparing SQL, NoSQL and NewSQL. Mongo DB: Introduction – Features – Data Types – Mongo DB Query Language – CRUD Operations – Arrays – Functions: Count – Sort – Limit – Skip – Aggregate – Map Reduce. Cursors – Indexes – Mongo Import – Mongo Export. Cassandra: Introduction – Features – Data Types – CQLSH – Key Spaces – CRUD, Operations – Collections – Counter – TTL – Alter Commands – Import and Export – Querying System Tables.

UNIT -V: In-memory data processing: Introduction to Apache Spark, Resilient Distributed Datasets (RDD), RDD actions (load, store, extract etc.), RDD transformations (FlatMap, GroupByKey, ReduceByKey etc.), fault tolerance, Spark Lineage, DataFrames, using Spark SQL language for data processing. Graph data processing with Pregel.