Paper No	Paper Name
Paper 1:	**Programming Fundamentals using C++ (DSC)**
CO1:	• Develop structured as well as object-oriented programming skills using C++ programming language.
CO2:	• Focus not only on basic C++ constructs but also cover object-oriented programming features in-depth.
CO3:	• Using Encapsulation, Abstraction, Inheritance and Polymorphism for writing efficient codes.
CO4:	• Demonstrate programming skills in high level language.
CO5:	• Develop robust programs using Exception handling
Paper 2:	**Computer System Architecture (DSC)**
CO1:	• Design Combinational Circuits using basic building blocks. • Simplify these circuits using Boolean algebra and Karnaugh maps. • Differentiate between combinational circuits and sequential circuits.
CO2:	• Represent data in binary form, convert numeric data between different number systems and perform arithmetic operations in binary.
CO3:	• Determine various stages of instruction cycle and describe interrupts and their handling.
CO4:	• Explain how CPU communicates with memory and I/O devices.
CO5:	• Simulate the design of a basic computer using a software tool
Paper 3:	**Introduction to Java Programming (DSC)**
CO1:	• Implement Exception Handling and File Handling.
CO2:	• Implement multiple inheritance using Interfaces.
CO3:	• Logically organize classes and interfaces using packages.
CO4:	• Understanding events, events listeners, and event handling.
CO5:	• Use AWT and Swing to design GUI applications.
Paper 4:	**Discrete Structures Core Course - (DSC)**
CO1:	• Define mathematical structures (relations, functions, sequences, series, and graphs) and use them to model real life situations.
CO2:	• Understand (trace) and construct simple mathematical proofs using logical arguments.t
CO3:	• Solve class room puzzles based on counting principles.
CO4:	• Compare functions and relations with respect to their growth for large values of the input.
CO5:	• Introduce the students to Boolean algebra, sets, relations, functions, principles of counting, and growth functions so that these concepts may be used effectively in other courses.
Paper 5:	**Data Structures (DSC)**
CO1:	• Developing the ability to use basic data structures like array, stacks, queues.
CO2:	• Implement and empirically analyze linear and non-linear data structures like Arrays, Stacks, Queues, Lists, Trees, Heaps and Hash tables as abstract data structures
CO3:	• Developing ability to use lists and trees to solve problems
CO4:	• Introduction to hashing, hash tables and hashing functions
CO5:	• Introduction to heap as a data structure. Analysis and applications to priority queues.
Paper 6:	**Operating system (DSC)**
CO1:	• Identify the services and different kinds of Operating System.
CO2:	• Understanding the concepts of process, threads, CPU scheduling and different scheduling algorithms, synchronization, and deadlocks.
CO3:	• Introducing the role of paging, segmentation and virtual memory in operating systems
CO4:	• Defining I/O systems, Device Management Policies and Secondary Storage Structure and Evaluation of various Disk Scheduling Algorithms.
CO5:	• Implement memory management techniques, multithreading concepts and a basic file system for a small operating system.

Paper 7:	**Computer Networks (DSC)**
CO1:	• Describe the hardware, software components of a network and their interrelations.
CO2:	• Compare OSI and TCP/IP network models.
CO3:	• Describe analyze and compare different data link, network, and transport layer protocols.
CO4:	• Design/implement data link and network layer protocols in a simulated networking environment.
Paper 8:	**Web Design and development (SEC)**
CO1:	• Design and develop a website
CO2:	• Use Front end technologies like HTML, CSS and JavaScript
CO3:	• Use backend technologies like PHP and MySQL
CO4:	• Work on platforms like WAMP/XAMP/LAMP
Paper 9:	**Python (SEC)**
CO1:	• Develop, document, and debug modular python programs to solve computational problems
CO2:	• Select a suitable programming construct and data structure for a situation
CO3:	• Use built-in strings, lists, sets, tuples and dictionary in applications.
CO4:	• Define classes and use them in applications techniques: Iterative, divide-n-conquer
CO5:	• Use files for I/O operations
Paper 10:	**Design and Analysis of Algorithms (DSC)**
CO1:	• To introduce the students to design and analyze algorithms in terms of efficiency and correctness.
CO2:	• Design techniques: Iterative, divide-n-conquer
CO3:	• Introduction to Greedy and Dynamic Programming design techniques using appropriate data structures
CO4:	• Introduction to Advanced Analysis Technique: Amortized analysis
CO5:	• Focus on highlighting difference between various problem-solving techniques for efficient algorithm design.
Paper 11:	**Software Engineering (DSC)**
CO1:	• Analyze and model customer's requirements and model its software design.
CO2:	• Use suitable software model for the problem at hand.
CO3:	• Estimate cost and efforts required in building software.
CO4:	• Analyze and compute impact of various risks involved in software development.
CO5:	• Design and build test cases, and to perform software testing.
Paper 12 :	**Database Management System (DSC)**
CO1:	• Describe major components of DBMS and their functions.
CO2:	• Model an application's data requirements using conceptual modelling tools like ER diagrams and design database schemas based on the conceptual model
CO3:	• Write queries in relational algebra / SQL
CO4:	• Normalize a given database schema to avoid data anomalies and data redundancy
CO5:	• Describe the notion of indexes, views, constraint, and transaction
Paper 13:	**Android Programming (SEC)**
CO1:	• Describe characteristics of Android operating system
CO2:	• Describe components of an android applications
CO3:	• Design user interfaces using various widgets, dialog boxes, menus
CO4:	• Define interaction among various activities/applications using intents, broadcasting, services
CO5:	• Develop Android applications that require database handling

Paper No:	Paper Name
Paper 14:	**Internet Technologies (DSC)**
CO1:	• Design and develop the website by integrating various client-side and server-side technologies i.e., HTML, CSS, JSP, Java, JavaBeans, Custom-Tags and JDBC.
CO2:	• Add dynamic content to the website and perform client-side validations using JavaScript.
CO3:	• Install and configure Tomcat server and create server-side page using JSP that can accept client-side data and generate the response page dynamically on server in addition to performing server-side validations.
CO4:	• Provide database connectivity using JDBC API.
CO5:	• Implement object reusability using JavaBeans.
Paper 15:	**Theory of Computation (DSC)**
CO1:	• Design a finite automaton, pushdown automaton or a Turing machine for a problem at hand.
CO2:	• Apply pumping lemma to prove that a language is non-regular/non-context-free.
CO3:	• Describe limitations of a computing machine.
Paper 16:	**Operational Research for Computer Science (DSE-1)**
CO1:	• Define and formulate linear programming problems and appreciate their limitations.
CO2:	• Solve linear programming problems using appropriate techniques and optimization solvers, interpret the results obtained and translate solutions into directives for action.
CO3:	• Conduct and interpret post-optimal and sensitivity analysis and explain the primal-dual relationship.
CO4:	• Develop mathematical skills to analyze and solve integer programming and network models arising from a wide range of applications.
CO5:	• Understand the mathematical tools that are needed to solve optimization problems and use mathematical software to solve the proposed models
Paper 17:	**Microprocessor (DSE-2)**
CO1:	• Describe the internal architecture of Intel microprocessors
CO2:	• Write assembly language programs
CO3:	• Define and implement interfaces between the microprocessor and the devices.
CO4:	• Describe the type and use of interrupts and direct memory access.
CO5:	• Define and implement various kind of interrupt and I/O controllers. • Define and describe the architecture from single core to multiple core processors.
Paper 18:	**Artificial Intelligence (DSC)**
CO1:	• Learn about intelligent agents and reasoning, heuristic search techniques, game playing, knowledge representation, reasoning with uncertain knowledge.
CO2:	• Identify problems that are amenable to solution by specific AI methods
CO3:	• Represent knowledge in Prolog and write code for drawing inferences.
CO4:	• Identify appropriate AI technique for the problem at hand
CO5:	• Compare strengths and weaknesses of different artificial Intelligence techniques.
CO6:	• Sensitive towards development of responsible Artificial Intelligence
Paper 19:	**Computer Graphics (DSC)**
CO1:	• Describe Standard raster and vector scan devices as well as Graphical Input and output devices
CO2:	• Implement algorithms for drawing basic primitives such as line, circle and ellipse.
CO3:	• Implement algorithms for line clipping and polygon clipping and filling.
CO4:	• Implement a 3D object representation scheme and carryout 2D and 3D Transformation, 3D projections.
CO5:	• Implement visible surface determination algorithms, Illumination models and surface rendering methods, color models

CO6:	• Implement a simple computer animation algorithm
Paper 20:	**Data Science using R (DSE-3)**
CO1:	• Install, Code and Use R Programming Language in R Studio IDE to perform basic tasks on Vectors, Matrices, Lists, and Data frames.
CO2:	• Define, Calculate, Implement Probability and Probability Distributions to solve a wide variety of problems
CO3:	• Conduct and interpret a variety of Hypothesis Tests to aid Decision Making
CO4:	• Understand, analyse, and interpret correlations to analyze the underlying relationships between different variables.
CO5:	• Query data using SQL and R.
CO6:	• Visualize data attributes using basic R packages.
Paper 21:	**Machine Learning (DSE-3)**
CO1:	• Differentiate between supervised and unsupervised learning tasks.
CO2:	• Differentiate between linear and non-linear classifiers.
CO3:	• Describe theoretical basis of SVM
CO4:	• Implement various machine learning algorithms learnt in the course
Paper 22:	**Data Mining (DSE-4)**
CO1:	• Pre-process the data, and perform cleaning and transformation
CO2:	• Apply suitable classification algorithm to train the classifier and evaluate its performance
CO3:	• Apply appropriate clustering algorithm to cluster data and evaluate clustering quality
CO4:	• Use association rule mining algorithms and generate frequent item-sets and association rules
Paper 23:	**Programming using Python (GE-1)**
CO1:	• Describe the components of a computer and notion of an algorithm.
CO2:	• To explain a basic introduction to object-oriented and procedural programming using Python.
CO3:	• Apply suitable programming constructs and built-in data structures to solve a problem.
CO4:	• Develop, document, and debug modular python programs.
CO5:	• Use classes and objects in application programs and visualize data.
Paper 24:	**Database Management Systems (GE-2)**
CO1:	• Describe major components of DBMS and their functions.
CO2:	• Model an application's data requirements using conceptual modelling tools like ER diagrams and design database schemas based on the conceptual model
CO3:	• Write queries in relational algebra / SQL
CO4:	• Normalize a given database schema to avoid data anomalies and data redundancy
CO5:	• Describe the notion of indexes, views, and constraint
Paper 25:	**Computer Networks (GE-3)**
CO1:	• State the use of computer networks and different network topologies.
CO2:	• Distinguish between LAN, MAN, WAN, and between Intranet, Extranet and Internet.
CO3:	• Compare OSI and TCP/IP architectures
CO4:	• Enumerate different transmission media and describe the use of each of them.
CO5:	• Design web pages using HTML.

Paper No:	Paper Name
Paper 26:	**Information Security and Cyber Laws (GE-4)**
CO1:	• Learn, structure, mechanics and evolution of various crime threats
CO2:	• Learn to protect information systems from external attacks by developing skills in enterprise security, wireless security and computer forensics.
CO3:	• Analyze the risks involved while sharing their information in cyber space and numerous related solutions like sending protected and digitally signed documents
CO4:	• Insights of ethical hacking and usage of password cracking tools
CO5:	• Get an overview of different ciphers used for encryption and decryption.