



Dayanand College, Hisar

Affiliated to Guru Jambheshwar University of Science & Technology, Hisar
Under DAV College Managing Committee, New Delhi
(Accredited with Grade 'A++' by NAAC)

Programme Outcome,
Programme Specific Outcome,
Course Outcome

DEPARTMENT OF COMPUTER SCIENCE

Programme Outcomes:

The Bachelor of Computer Application (BCA) and Bachelor of Science (Computer Science) is a three year undergraduate degree program that is designed to provide students with a strong foundation in computer science and applications. The program focuses on developing analytical and logical reasoning skills, programming knowledge, database management, web development, software engineering, and system analysis.

The following are some of the program outcomes of a Bachelor of Computer Application Bachelor of Science (Computer Science) degree:

- Knowledge and understanding of the fundamental concepts of computer science and its applications.
- Ability to design, develop, and implement computer-based solutions to real-world problems using programming languages like C, C++, Java, Python, etc.
- Skills in data management, including database design, implementation, and querying using database management systems like Oracle, MySQL, etc.
- Ability to develop web-based applications using technologies like HTML, CSS, JavaScript, PHP, and other web development frameworks.
- Knowledge of software engineering principles and methodologies, including software development life cycle (SDLC), software testing, and quality assurance.
- Understanding of computer networks and their components, including protocols, topologies, and network security.
- Skills in system analysis and design, including the ability to analyze user requirements, design system architecture, and implement solutions that meet business needs.
- Awareness of emerging technologies and their potential impact on computer applications.

Programme Specific Outcomes:

- Equip themselves to potentially rich & employable field of computer applications.
- Pursue higher studies in the area of Computer Science/Applications.
- Take up self-employment in Indian & global software market.
- Meet the requirements of the Industrial standards.

Course Outcomes:

Our graduates will have:

- The necessary technical, scientific as well as basic managerial and financial procedures to analyze and solve real world problems within their work domain.
- Clarity on both conceptual and application oriented skills in commerce, Finance & Accounting and IT Applications in Business context.
- Improved communication and business management skills, especially in providing tech support.
- Awareness on ethics, values, sustainability and creativity aspects.
- The ability and the mindset to continuously update and innovate.

BACHELOR OF SCIENCE (COMPUTER SCIENCE)

Three Year Degree Programme

Semester-1

Course Code	Learning Outcomes
C24COS101T (Fundamentals of Computer and Programming in C)	Upon completion of the course, the students will be able to: • Bridge the fundamental concepts of Computer with the present level of knowledge of students. • Understanding the concept of memory devices of Computers • Familiarize operating systems • Write, compile and debug programs in C language and use different data types for writing the programs. • Design programs connecting decision structures, loops and functions. • Explain the difference between call by value and call by address. • Understanding the concept of Strings and pointers.
C24COS101P (Programming in C Lab)	Upon completion of the course, the students will be able to: • Explain the use of formatted and unformatted input and output statements in C and usage of sequence control statements of C. • Enlist the fundamental data types and data structures of C And Explain the usage of arrays and pointers in C.
C24SEC103T (Office Tools)	Upon completion of the course, the students will be able to: • Create, edit, format, and manage documents using word processing software (e.g., Microsoft Word). • Perform calculations, analyze data, and generate charts using spreadsheet software (e.g., Microsoft Excel). • Design and deliver effective presentations using presentation software (e.g., Microsoft PowerPoint).
C24SEC103P (Office Tools Lab)	Upon completion of the course, the students will be able to: • Create, edit, format, and print professional documents using word processing software with appropriate formatting, tables, images, and templates. • Perform practical spreadsheet operations such as data entry, formulas, functions, sorting, filtering, chart creation, and basic data analysis using spreadsheet software. • Develop effective multimedia presentations by applying themes, animations, transitions, charts, images, and presentation delivery techniques.

C24SEC203T (Digital Efficiency Tools)	Upon completion of the course, the students will be able to: • Use digital productivity tools effectively to create, edit, organize, and share documents, spreadsheets, presentations, and other digital content. • Improve workplace efficiency by applying digital tools for task management, scheduling, collaboration, and communication.
C24SEC203P (Digital Efficiency Tools Lab)	Upon completion of the course, the students will be able to: • Demonstrate proficiency in using modern digital tools, software applications, and online platforms effectively. • Problem Solving: Analyze real-world problems and apply appropriate digital technologies to develop efficient solutions. • Effectively collaborate using digital communication platforms and project management tools.

Semester-2

Course code	Learning Outcomes
C24COS201T (Data structures using C)	By the end of the course a student is expected to be able: • Ability to analyze algorithms and algorithm correctness. • Ability to analyze the time and space complexity of algorithms. • Ability to summarize searching and sorting techniques theoretically and practically using C programming language. • Ability to describe stack, queue and linked list operation and their practical using C language. • Ability to have knowledge of tree and graphs concepts & their implementation using C language. • Ability to write program and step by step approach to solve problems with the help of fundamental data structures using C language.
C24COS201P (Data structures using C Lab)	By the end of the course a student is expected to be able: • Apply the fundamentals of C programming and data structures to solve computational problems. • Analyze problems and select appropriate data structures and algorithms for efficient solutions. • Design, implement, test, and debug programs using linear and non-linear data structures such as arrays, linked lists, stacks, queues, trees, and graphs. • Theoretical analysis, implementation and application of data structures.
C24SEC103T (Office Tools)	By the end of the course a student is expected to be able: • Create, edit, format, and manage documents using word processing software (e.g., Microsoft Word). • Perform calculations, analyze data, and generate charts using spreadsheet software (e.g., Microsoft Excel). • Design and deliver effective presentations using presentation software (e.g., Microsoft PowerPoint).
C24SEC103P (Office Tools Lab)	By the end of the course a student is expected to be able: • Create, edit, format, and print professional documents using word processing software with appropriate formatting, tables, images, and templates. • Perform practical spreadsheet operations such as data entry, formulas, functions, sorting, filtering, chart

	creation, and basic data analysis using spreadsheet software. • Develop effective multimedia presentations by applying themes, animations, transitions, charts, images, and presentation delivery techniques.
C24SEC203T (Digital Efficiency Tools)	By the end of the course a student is expected to be able: • Use digital productivity tools effectively to create, edit, organize, and share documents, spreadsheets, presentations, and other digital content. • Improve workplace efficiency by applying digital tools for task management, scheduling, collaboration, and communication.
C24SEC203P ((Digital Efficiency Tools Lab)	By the end of the course a student is expected to be able: • Demonstrate proficiency in using modern digital tools, software applications, and online platforms effectively. • Problem Solving: Analyze real-world problems and apply appropriate digital technologies to develop efficient solutions. • Effectively collaborate using digital communication platforms and project management tools.

Semester-3

Course Code	Learning Outcomes:
C24COS301T (Database Management System)	By the end of the course a student is expected to be able:. • Understand how data is stored using storage media in a DBMS. • Understand how different indexing techniques work. • Understand why and how data needs to be indexed.
C24COS301P (DBMS Lab)	By the end of the course a student is expected to be able: • Create Database and store information. • Create, update, view and delete table in database, • Learn SQL queries to maintain and access Database.
C24SEC303T (Data analysis using Excel)	By the end of the course a student is expected to be able: • Understand the basics of Microsoft Excel and its interface. • Enter, organize, and format data efficiently. • Use formulas and functions such as SUM, AVERAGE, IF, COUNT, VLOOKUP/XLOOKUP, INDEX-MATCH, and text functions. • Clean and prepare data by removing duplicates, sorting, filtering, and validating data..
C24SEC303P (Data analysis using Excel lab)	By the end of the course a student is expected to be able: • Develop proficiency in using Microsoft Excel for data entry, organization, and formatting. • Apply Excel formulas and functions to perform accurate data analysis. • Clean, sort, filter, and validate datasets for effective analysis.

Course Code	Learning Outcomes:
C24COS401T (Computer Networks)	By the end of the course a student is expected to be able: • Describe the functions of each layer in OSI and TCP/IP model. • Explain the functions of Application layer and Presentation layer paradigms and Protocols. • Describe the Session layer design issues and Transport layer services. • Classify the routing protocols and analyze how to assign the IP addresses for the given network. • Understand network security and define various protocols such as FTP, HTTP, Telnet, DNS.
C24COS401P (Computer Networks Lab)	By the end of the course a student is expected to be able: • Identify and use various networking components • Understand different transmission media and design cables for establishing a network. • Implement any topology using network devices. • Understand the TCP/IP configuration for Windows and Linux. • Implement device sharing on network e) Learn the major software and hardware technologies used on computer networks.

Semester-5

Course Code	Learning Outcomes
C24COS501T (Operating System)	By the end of the course the student will be able to: • Operating systems including memory management, networks, processor management, system security, device management, systems management, and file management • Execute process management and scheduling operations. • Apply memory management and resource allocation techniques.
C24COS501P (Operating System Lab)	By the end of the course the student will be able to: • Understand and implement fundamental concepts of operating systems through laboratory experiments. • Develop programs related to process management, process scheduling, and inter-process communication. • Implement and analyze CPU scheduling algorithms such as FCFS, SJF, Priority Scheduling, and Round Robin. • Demonstrate the concepts of memory management, paging, segmentation, and virtual memory through practical exercises.
C24SEI500P (Internship)	By the end of the course the student will be able to: • Apply theoretical knowledge of computer science concepts to real-world industry projects and applications. • Develop practical skills in software development, programming, databases, networking, or emerging technologies. • Analyze, design, and implement solutions to real-world technical problems. • Gain experience with industry-standard tools, technologies, and development methodologies. • Improve programming, debugging, testing, and problem-solving abilities through practical work experience.

Semester-6

Course Code	Learning Outcomes
C24COS601T (Programming in Python)	By the end of the course the student will be able to: • To understand how to write functions and pass arguments in Python. • To understand how to build and package Python modules for reusability. • To understand how to read and write files in Python. • To understand how to design object-oriented programs with Python classes. • To understand how to use class inheritance in Python for reusability
C24COS601P (Programming in Python Lab)	By the end of the course the student will be able to: • Apply fundamental programming concepts and Python syntax to solve computational problems. • Analyze real-world problems and develop appropriate algorithmic solutions using Python. • Design, implement, test, and debug Python programs using structured and object-oriented programming principles. • Perform experiments, analyze program outputs, identify errors, and evaluate the efficiency of Python solutions. • Use Python IDEs, libraries, modules, and development tools to build efficient software applications.

BACHELOR OF COMPUTER APPLICATIONS (BCA)

Three Year Degree Programme

Semenster-1

Course Code	Learning Outcomes
C24CAP101T (Computer fundamental and problem solving through C)	By the end of the course the student will be able to: - Understand the fundamental concepts of computer systems, including hardware, software, operating systems, memory, and number systems. - Apply logical thinking and problem-solving techniques to analyze computational problems and develop appropriate solutions. - Design algorithms and flowcharts for solving programming problems effectively. - Develop C programs using variables, data types, operators, control statements, functions, arrays, pointers, strings, structures, and file handling. - Compile, execute, test, and debug C programs using appropriate programming tools and development environments.
C24CAP101P (Computer fundamental and problem solving through C Lab)	By the end of the course the student will be able to: - Demonstrate proficiency in using computer systems, operating systems, and programming development environments. - Develop, compile, execute, and debug C programs using appropriate programming tools and techniques. - Apply fundamental programming concepts such as variables, data types, operators, decision-making, and looping constructs to solve problems. - Design and implement modular programs using functions, arrays, strings, pointers, and structures. - Analyze programming problems and develop logical algorithms for efficient program implementation.
C24CAP102T (Logical organization of computer)	By the end of the course the student will be able to: - Understand the logical organization and functional units of a digital computer system. - Explain the principles of data representation, number systems, and arithmetic operations used in computer systems. - Analyze the architecture and operation of the CPU, memory, input/output devices, and system buses. - Apply knowledge of instruction formats, addressing modes, and instruction execution cycles to understand processor operations. - Describe the organization and hierarchy of memory, including cache, primary, and secondary storage.

C24CAP102P (Logical organization of computer Lab)	By the end of the course the student will be able to: • Demonstrate an understanding of the logical organization and functional units of a computer system through laboratory experiments. • Perform binary, octal, decimal, and hexadecimal number system conversions and arithmetic operations. • Design and verify the operation of basic combinational and sequential digital circuits using logic gates, flip-flops, multiplexers, decoders, and counters. • Analyze the working of memory organization, registers, and data transfer mechanisms through simulation or hardware experiments.
C24CAP103T (Introduction to Web technologies)	By the end of the course the student will be able to: • Understand the fundamentals of the World Wide Web, internet technologies, web browsers, web servers, and client–server architecture. • Design and develop static web pages using HTML and CSS by applying standard web design principles. • Create interactive web pages using JavaScript for client-side validation and dynamic content.
C24CAP103P (Introduction to Web technologies Lab)	By the end of the course the student will be able to: • Develop and design static web pages using HTML and CSS by applying standard web design principles. • Create interactive and dynamic web pages using JavaScript for client-side scripting and form validation. • Apply responsive web design techniques to develop websites that function effectively across different devices and screen sizes. • Use web development tools, code editors, browsers, and debugging tools to create, test, and maintain web applications.
C24SEC124T (PC Hardware and Networking)	By the end of the course the student will be able to: • To claim proficiency in word, excel and power point. • Independently create professional-looking documents and presentations.
C24SEC124P (PC Hardware and Networking Lab)	By the end of the course the student will be able to: • Understand the architecture, components, and working principles of personal computer hardware and peripheral devices. • Identify, assemble, install, configure, and troubleshoot PC hardware components, operating systems, and device drivers. • Explain the fundamentals of computer networking, including network models, topologies, transmission media, and protocols.

Semester-2

Course code	Learning outcome
C24CAP201T (Object Oriented Programming using C++)	By the end of the course the student will be able to: • Understand the principles of object-oriented programming and the features of the C++ programming language. • Design and implement C++ programs using classes, objects, constructors, destructors, and member functions. • Apply object-oriented concepts such as encapsulation, inheritance, polymorphism, abstraction, and dynamic binding to develop modular and reusable software.
C24CAP201P (Object Oriented Programming using C++ lab)	By the end of the course the student will be able to: • Develop, compile, execute, and debug C++ programs using standard programming tools and integrated development environments (IDEs). • Implement object-oriented programming concepts such as classes, objects, constructors, destructors, encapsulation, inheritance, polymorphism, and abstraction. • Design and develop modular, reusable, and efficient C++ programs to solve real-world problems. • Apply advanced C++ features, including function overloading, operator overloading, templates, exception handling, file handling.
C24CAP202T (Data Structure and applications)	By the end of the course the student will be able to: • Understand fundamental data structures such as arrays, linked lists, stacks, queues, trees, graphs, and their applications. • Analyze computational problems and select appropriate data structures for efficient problem-solving. • Design and implement algorithms using linear and non-linear data structures. • Apply searching and sorting techniques to improve the efficiency of data processing operations. • Analyze the time and space complexity of algorithms and evaluate their performance.

C24CAP202P (Data Structure and applications lab)	By the end of the course the student will be able to: • Implement fundamental data structures such as arrays, linked lists, stacks, queues, trees, and graphs using programming languages. • Develop and execute programs based on various data structure operations such as insertion, deletion, traversal, searching, and sorting. • Apply appropriate data structures and algorithms to solve computational and real-world problems efficiently.
C24CAP203T (Concepts of Operating Systems)	By the end of the course the student will be able to: • Operating systems including memory management, networks, processor management, system security, device management, systems management, and file management • Execute process management and scheduling operations. • Apply memory management and resource allocation techniques.
C24CAP203P (Concepts of Operating Systems Lab)	By the end of the course the student will be able to: • Understand and implement fundamental concepts of operating systems through laboratory experiments. • Develop programs related to process management, process scheduling, and inter-process communication. • Implement and analyze CPU scheduling algorithms such as FCFS, SJF, Priority Scheduling, and Round Robin. • Demonstrate the concepts of memory management, paging, segmentation, and virtual memory through practical exercises.
C24SEC224T (Web Designing Basics using HTML)	By the end of the course the student will be able to: • Understand the fundamentals of web technologies, internet concepts, and the structure of web documents. • Create and design basic web pages using HTML tags, elements, attributes, and document structure. • Develop web pages using HTML features such as text formatting, hyperlinks, images, tables, lists, and forms. • Apply basic principles of web page layout, navigation, and user interface design. • Use web browsers and development tools to create, test, and debug HTML-based web pages.

C24SEC224P (Web Designing Basics using HTML Lab)	By the end of the course the student will be able to: • Develop basic web pages using HTML tags, elements, and attributes according to web standards. • Design web pages incorporating text formatting, hyperlinks, images, tables, lists, and forms. • Create structured and user-friendly web layouts using HTML concepts and proper document organization. • Apply basic web design principles to improve usability, navigation, and presentation of web pages. • Use web browsers and HTML editors/tools to create, test, and debug web documents. • Implement simple web-based solutions for real-world applications using HTML.
---	---

Course Code	Learning Outcomes
C24CAP301T (Java Programming)	By the end of the course the student will be able: - To learn why Java is useful for the design of desktop and web applications. - To learn how to implement object-oriented designs with Java. - To identify Java language components and how they work together in applications.
C24CAP301P (Java Programming Lab)	By the end of the course the student will be able to: - Develop, compile, execute, and debug Java programs using appropriate programming tools and development environments. - Apply object-oriented programming concepts such as classes, objects, inheritance, polymorphism, encapsulation, and abstraction in Java applications. - Implement Java programming concepts including data types, control structures, arrays, strings, methods, and packages. - Develop applications using advanced Java features such as exception handling, multithreading, file handling, and collections.
C24CAP302T (Linux and Shell programming)	By the end of the course the student will be able to: - Understand the fundamentals of Linux operating systems, architecture, file systems, and command-line environments. - Use basic and advanced Linux commands for file management, process management, system administration, and user management. - Develop and execute shell scripts using shell programming concepts such as variables, operators, control structures, loops, and functions. - Apply Linux utilities and scripting techniques to automate routine system tasks and solve computational problems.

C24CAP302P (Linux and Shell programming Lab)	By the end of the course the student will be able to: • Demonstrate practical knowledge of the Linux operating system, command-line interface, and system environment. • Execute and apply Linux commands for file management, directory handling, process control, user management, and system administration tasks. • Develop and execute shell scripts using variables, operators, decision-making statements, loops, functions, and command-line arguments.
C24CAP303T (Database Management System)	By the end of the course the student will be able to: • Understand the fundamental concepts of database systems, database architecture, and data models. • Design and develop efficient database schemas using concepts of entities, relationships, normalization, and constraints. • Apply SQL commands for creating, manipulating, querying, and managing databases. • Implement database operations using relational database concepts, keys, integrity constraints, and transaction management.
C24CAP303P (Database Management System Lab)	By the end of the course the student will be able to: • Implement fundamental database concepts and understand the working of relational database management systems. • Design and create databases using tables, relationships, keys, constraints, and normalization techniques. • Write and execute SQL queries for data definition, data manipulation, data retrieval, and transaction management. • Develop database applications using SQL and database connectivity tools.
C24MIC324T (Theory of Computation)	By the end of the course the student will be able to: • Understand the fundamental concepts of formal languages, automata theory, and computational models. • Analyze and design finite automata, regular expressions, and regular grammars for recognizing formal languages. • Apply concepts of context-free grammars, pushdown automata, and parsing techniques to solve language processing problems. • Understand the principles of Turing machines, computability, and decidability of computational problems.

C24SEC324T (Advance IT Skills)	By the end of the course the student will be able to: • Understand and apply advanced concepts of information technology, computer systems, and digital tools. • Develop proficiency in using advanced software applications, productivity tools, and IT platforms for professional tasks. • Apply problem-solving and analytical skills to handle IT-related challenges and develop effective solutions. • Utilize internet technologies, cloud-based tools, and digital resources for efficient information management.
C24SEC324P (Advance IT Skills Lab)	By the end of the course the student will be able to: • Demonstrate practical skills in using advanced information technology tools, software applications, and digital platforms. • Apply advanced computer skills for document creation, data management, presentations, and professional communication. • Use internet technologies, cloud-based applications, and online collaboration tools effectively.

Course Code	Learning Outcomes
C24CAP401T (Cloud Computing)	By the end of the course the student will be able to: • Explain the fundamentals, characteristics, service models (IaaS, PaaS, SaaS), and deployment models (Public, Private, Hybrid, Community). • Design, deploy, and manage cloud-based applications using modern cloud platforms and tools. • Understand virtualization, containerization, and efficient allocation of cloud resources. • Identify security threats, implement security measures, and ensure data privacy and compliance in cloud environments.
C24CAP401P (Cloud Computing Lab)	By the end of the course the student will be able to: • Apply cloud computing concepts to perform practical tasks using cloud platforms and virtualization tools. • Create, deploy, and manage virtual machines, containers, and cloud-based applications. • Configure and utilize cloud storage, networking, and computing resources effectively.
C24CAP402T (Front end Development)	By the end of the course the student will be able to: • Apply the fundamentals of HTML, CSS, and JavaScript to develop responsive and interactive web pages. • Design user-friendly and accessible web interfaces by following modern UI/UX principles. • Develop dynamic web applications using front-end frameworks and libraries such as Bootstrap, React, or Angular. • Implement responsive web designs that work efficiently across different devices and screen sizes.
C24CAP402P (Front end Development Lab)	By the end of the course the student will be able to: • Apply knowledge of HTML, CSS, JavaScript, and web technologies to develop responsive and interactive web pages. • Analyze user requirements and identify appropriate front-end design and implementation techniques. • Design and develop user-friendly, responsive, and accessible web interfaces using modern front-end tools and frameworks. • Evaluate and debug front-end applications using browser developer tools and testing techniques.

C24CAP403T (Computer Graphics)	By the end of the course the student will be able to: ▪ Explain the core concepts of computer graphics, including viewing, projection, perspective, modelling and transformation in two and three dimensions. ▪ Interpret the mathematical foundation of the concepts of computer graphics. ▪ Identify a typical graphics pipeline and apply graphics programming techniques to design and create computer graphics.
C24CAP403P (Computer Graphics Lab)	By the end of the course the student will be able to: ▪ Apply mathematical, computing, and engineering principles to develop computer graphics applications. ▪ Analyze graphical problems and select appropriate algorithms for drawing, transformations, clipping, and rendering. ▪ Design and implement interactive 2D and 3D graphics applications using standard graphics programming techniques. ▪ Evaluate the performance and correctness of graphics algorithms through testing, visualization, and analysis.

Course Code	Learning Outcomes
C24CAP501T (Software Engineering)	By the end of the course the student will be able: • Identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics. • Function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives. • Develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions. • Acquire and apply new knowledge as needed, using appropriate learning strategies.
C24CAP501P (Software Engineering Lab)	By the end of the course the student will be able to: • Apply software engineering principles, programming knowledge, and computing fundamentals to develop quality software solutions. • Analyze software requirements and identify appropriate methods, models, and tools for software development. • Design, develop, test, and maintain software systems that satisfy functional and non-functional requirements.
C24CAP502T (Back-end Development)	By the end of the course the student will be able to: • . Design and develop secure, scalable, and maintainable back-end applications. • Build and consume RESTful APIs for web and mobile applications. • Design, implement, and manage relational and NoSQL databases. • Apply authentication, authorization, and data security best practices. • Integrate back-end services with front-end applications and third-party APIs. • Use version control systems and collaborative development workflows.

C24CAP502P (Back-end Development Lab)	By the end of the course the student will be able to: • Develop server-side applications using appropriate back-end programming languages and frameworks. • Design, create, and manage relational and NoSQL databases for web applications. • Build, test, and document RESTful APIs for client-server communication. • Implement user authentication, authorization, and secure data handling practices. • Perform CRUD (Create, Read, Update, Delete) operations by integrating applications with databases. • Debug, test, and optimize back-end applications for functionality, performance, and reliability.
C24CAP503T (Programming Using Python)	By the end of the course the student will be able: • Understand and apply Python syntax, data types, operators, and control structures to solve programming problems. • Develop modular Python programs using functions, modules, and packages. • Implement object-oriented programming concepts such as classes, objects, inheritance, and polymorphism. • Use Python's built-in data structures (lists, tuples, dictionaries, and sets) efficiently for data manipulation. • Handle files and exceptions to develop reliable and robust applications. • Apply Python libraries and modules to perform common programming tasks and automate workflows.
C24CAP503P (Programming Using Python Lab)	By the end of the course the student will be able to: • To learn how to design and program Python applications. • To learn how to use lists, tuples, and dictionaries in Python programs. • To understand why Python is a useful scripting language for developers.
C24VOC524T (Basics of Data Analysis using Excel)	By the end of the course the student will be able to: • Understand the fundamentals of data analysis and the role of spreadsheets in data-driven decision-making. • Create, organize, and manage datasets using Microsoft Excel effectively. • Apply Excel formulas, functions, and tools to perform data calculations and analysis. • Clean, format, and prepare data for accurate analysis and reporting. • Use sorting, filtering, conditional formatting, and data validation techniques for data management. • Analyze data using charts, graphs, PivotTables, and Pivot Charts to identify trends and patterns.

C24VOC524P (Basics of Data Analysis using Excel Lab)	By the end of the course the student will be able to: • Create, enter, format, and organize datasets using Microsoft Excel effectively. • Apply Excel formulas and functions to perform data calculations and analysis tasks. • Perform data cleaning, sorting, filtering, and validation to improve data quality. • Use PivotTables, PivotCharts, and analytical tools to summarize and interpret data. • Generate meaningful charts, graphs, and dashboards for data visualization and reporting. • Apply basic statistical functions in Excel to analyze datasets and derive insights. • Perform practical data analysis tasks using real-world datasets and spreadsheet techniques.
C24SEC500P (Skill Enhancement Course)	By the end of the course the student will be able to: • Apply theoretical knowledge of computer science concepts to real-world industry projects and applications. • Develop practical skills in software development, programming, databases, networking, or emerging technologies. • Analyze, design, and implement solutions to real-world technical problems. • Gain experience with industry-standard tools, technologies, and development methodologies. • Improve programming, debugging, testing, and problem-solving abilities through practical work experience. • Work effectively in teams using collaboration, communication, and project management skills.

•

Course Code	Learning Outcomes
C24CAP601T Network and Data Communications Technologies	By the end of the course the student will be able : • Describe the functions of each layer in OSI and TCP/IP model. • Explain the functions of Application layer and Presentation layer paradigms and Protocols. • Describe the Session layer design issues and Transport layer services. • Classify the routing protocols and analyze how to assign the IP addresses for the given network. • Understand network security and define various protocols such as FTP, HTTP, Telnet, DNS.
C24CAP601P Network and Data Communications Technologies Lab	By the end of the course the student will be able to: • Identify and use various networking components Understand different transmission media and design cables for establishing a network. • Implement any topology using network devices. • Understand the TCP/IP configuration for Windows and Linux. • Implement device sharing on network e) Learn the major software and hardware technologies used on computer networks.
C24CAP602T (Data Analysis Using Python)	By the end of the course the student will be able to: • Understand the fundamentals of data analysis and the role of Python in data-driven decision-making. • Use Python programming concepts, libraries, and tools for data analysis tasks. • Import, clean, transform, and preprocess datasets using Python libraries such as Pandas and NumPy. • Perform exploratory data analysis (EDA) to identify trends, patterns, and relationships in data. • Apply statistical methods and analytical techniques to interpret datasets effectively. • Visualize data using Python libraries such as Matplotlib and Seaborn to communicate insights.
C24CAP602P (Data Analysis Using Python Lab)	By the end of the course the student will be able to: • Apply programming, mathematical, and statistical concepts to analyze and interpret data using Python tools. • Identify data-related problems, preprocess datasets, and select suitable analytical techniques for solving problems. • Develop Python-based solutions for data cleaning, visualization, statistical analysis, and predictive modeling. • Perform data exploration, experimentation, and interpretation using Python libraries such as NumPy, Pandas, and Matplotlib.

C24CAP603T (Artificial Intelligence)	By the end of the course the student will be able to: • Compare AI with human intelligence and traditional information processing and discuss its strengths and limitations as well as its applications to complex and human-centered problems. • Apply the basic principles, models, and algorithms of AI to recognize model and solve problems in the analysis and design of information system. • Analyze the structures and algorithms of a selection of techniques related to searching, reasoning, machine learning, and language processing.
C24CAP603P (Artificial Intelligence Lab)	By the end of the course the student will be able to: • Apply mathematical, computational, and AI concepts to develop intelligent solutions. • Analyze real-world problems and identify suitable AI techniques and algorithms for solving them. • Design and implement AI models such as search algorithms, machine learning models, and expert systems.
