Program Outcomes:

1. **Computer knowledge:** Apply the knowledge of mathematics, science and engineering fundamentals to the solution of complex problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex problems reaching substantiated conclusions using principles of mathematics, natural sciences.
3. **Design/development of solutions:** Design solutions for complex problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern IT tools to complex problems with an understanding of the limitations.
5. **Environment and sustainability:** Understand the impact of the professional solution in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
6. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
7. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
8. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
9. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes:

1. **Computer based systems development**: Ability to apply the basic knowledge of database systems, computing, operating system, digital circuits, microcontroller, computer organization and architecture in the design of computer based systems.
2. **System development**: Ability to specify, design and develop projects, application softwares and system softwares by using the knowledge of data structures, analysis and design of algorithm, programming languages, software engineering practices and open source tools.
3. **System testing:** Ability to debug, verify and validate the systems using various testing methods and tools.

Course Outcomes:

Semester	Course paper	Course Outcomes
1st Semester	Digital Logic and Computer Design	• Describe, Illustrate and analyse Combinational Logic circuits, Simplification of Algebraic Equations using Karnaugh Maps. • Describe the working and Design of Decoders, Digital multiplexers, Adders and Subtractors, and Master-Slave Flip-Flops. • Design different synchronous and asynchronous sequential circuits and their applications. • Analyse the working of ADC and DAC circuits and its applications.
	Mathematical Foundation for Computer Science	• Understand sets, relations, functions and discrete structures. • Apply propositional logic and first order logic to solve problems. • Formulate and solve graph problems.
	Data Structure using C++	• Design programs using a variety of data structures such as stacks, queues, binary trees, heaps, graphs. • Analyze and implement various kinds of searching and sorting techniques. • Discuss the applications of various Data Structures.
	Operating System Principles	• Explain the core structure and functionality of operating system. • Discuss and analyze various inter process communication mechanisms. • Evaluate and analyze the different techniques for solving CPU scheduling problems. • Describe and Apply the knowledge of deadlock concepts to provide wide range of functionality to applications • Identify and analyze the problems that occur in the design of OS based on knowledge gained through process synchronization techniques • Analyze the performance of different memory management techniques and page replacement algorithms
	Problem Solving Techniques using C	• Illustrate and explain the basic computer concepts and programming principles of C language. • Develop C programs to solve simple mathematical and decision making problems. • Develop C programs to solve simple problems using looping constructs.

		• Develop C programs to demonstrate the applications of derived data types such as arrays, pointers, strings and functions.
	Problem Solving Technique using C Lab	• Develop, Debug and Execute programs to demonstrate decision making and looping constructs in C. • Develop, Debug and Execute programs to demonstrate the applications of arrays in C. • Develop, Debug and Execute programs to demonstrate the applications of functions in C. • Develop, Debug and Execute programs to demonstrate the basic concepts of pointers in C.
	Data Structure using C++ Lab	• Design and develop C programs by applying advanced C programming techniques like pointers, structures and files to solve a given problem • Design and develop C programs to implement linear data structures like stack, queue and explore its applications by applying the knowledge of static memory allocation technique. • Design and develop C programs to implement linked lists and its types by applying the knowledge of dynamic memory allocation technique
2nd Semester	Design and Analysis of Algorithm	• Identify the fundamental principles of algorithm analysis and design. • Analyze the complexity of a given algorithm • Apply design techniques such as divide-and-conquer, decrease and conquer to solve a given problem. • Apply the design techniques such as dynamic programming and greedy technique to solve a given problem.
	Database Management System	• Describe the fundamentals of database technologies, Design an ER diagram and transform it to a relational model for a given database specification. • Discuss the relational model concepts and Design relational algebraic expressions for queries. • Explain the various concepts of SQL and Design SQL queries to perform CRUD (Create, Retrieve, Update and delete) operations on database. • Discuss the database design concepts such as functional dependency and solve the problems on minimal set, equivalence set. • Discuss the database design concepts such as Normalization, Relational decomposition and

		concepts of transaction processing. Apply the normalization techniques to improve database design.
	System Software	• Understand the concepts of SIC machine architecture, CISC and RISC machines. • Explain the various concepts of machine dependent and machine independent assembler features and its types. • Discuss on the concepts of loaders, linkers and macro processors features. • Understand the concepts of compiler functions and its design options.
	Object Oriented Analysis and Design using UML	• Apply the fundamental knowledge of object oriented software development, methodologies, UML language, and design patterns to the solution of complex problems. • Analyze, formulate and review and justify a case study to identify classes, attributes, methods and relationships among them in the solutions. • Design UML models for a given case study using object oriented software development methodologies.
	e-Governance	• Understand the concept of e-governance, and the associated benefits and drawbacks • Analyze the National e-Governance plan. • Understand the workflow management in e-Governance. • Case study on some e-Governance bodies.
	Algorithm and DBMS Lab	• Apply the concepts of divide-and-conquer, decrease and conquer for a given problem. • Understand and design the concepts of dynamic programming for a problem. • Design and implement database schema. • Design the queries using DDL, DML, DCL and TCL commands.
	Visual Programming Lab	• Apply the concepts of VB language for a problem. • Design and develop programs to implement procedures, functions, exception handling. • Design and develop programs to create the window forms, menus.
3rd Semester	Programming in Java	• Identify classes, objects, members of a class and the relationships among them needed for a specific problem. • Write java program using threads, event handling and input output utilities. Develop programs

		using the Java Collection API as well as the Java standard class library. • Demonstrate the ability to use Threads and synchronization in java. • Explain and write input – output programming in java and applications using Applets.
	Data Communication & Computer Network	• Apply basics of data communication and its components to understand computer networks technology. Enumerate the layers of TCP/IP and explain the functions of each layer. • Experiment with error detection and correction techniques and explain various transmission media. • Analyze various data link and network layer services. • Analyze various transport and application layer services. • Discuss the various security oriented protocols in internet.
	Computer Graphics	• Apply basics of graphics to create interactive applications using one or more graphics application programming interfaces. • Design and implement programs to demonstrate 2D image processing techniques. • Demonstrate the 2D and 3D transformations, creating polygons, solid modeling.
	Distributed Operating Systems	• Understand the concepts of Distributing computing environment. • Discuss the message passing mechanism and synchronization concepts in distributed computing environment. • Explain the concept of Remote Procedure Call and its implementation. • Discuss the concepts of Distributed Shared memory, Resource Management and Distributed File System.
	Information Technology	• Understand the concepts of basic networking concepts, different transmission media. • Enumerate the layers of OSI and TCP/IP and explain the functions of each layer. • Discuss on various IP addressing methods, internet services and search engines.
	Java Programming And Computer Networks Lab	• Implement Object oriented features using Java • Apply the concept of polymorphism and inheritance.

		• Implement exception handling • Implement Applet event handling
	Computer Graphics Lab	• Understand fundamental principles of computer graphics. • Design and implement the program for generating geometric pictures and 2D translation transformations. • Design and implement the program for 2D scaling, polygon fill, projections.
4th Semester	Internetworking & Web Design	• Recall programming skills on internet based applications. • Design and develop sophisticated web sites and applications. • Compare web projects developed with traditional projects • Critique procedures of internet programming • Implement the important HTML tags for designing static pages and separate design from content using Cascading Style sheet.
	Software Engineering	• Analyze the process model chosen for the development of software and its merits and demerits. • Identify the clear, correct and consistent requirements for the project. • Design suitable data, architecture and user interface that copes with the requirements. • Estimate the cyclomatic complexity and design the corresponding test cases. • Conduct various integration testing approaches and note down pit falls in requirements, design and test cases.
	Bioinformatics	• Knowledge and awareness of the basic principles bioinformatics and its applications. • To get exposed to computational methods, tools and algorithms employed for Biological Data Interpretation. • Describe about the different types of Biological databases. • Overview about biological structure for prediction methods. • Understand the concepts of genomics and proteomics.
	Artificial Intelligence	• Apply the knowledge of Artificial Intelligence to write simple algorithm for agents.

		• Apply AI knowledge to solve problem on search algorithm. • Develop knowledge base sentences propositional logic and first order logic. • Apply first order logic to solve knowledge engineering process. • Apply and analyze the knowledge of machine learning
	Neural Networks and Fuzzy Systems	• Describe and analysis of neural network architectures. • Design neural network approach to a particular problem. • Understand the concepts of Fuzzy systems and its applications.
	Theory of Computation	• Understand formal machines, languages. • Design finite state machines for acceptance of strings. • Design context free grammars for formal languages. • Develop pushdown automata acceptance strings. • Design Turing machine.
	Pattern Recognition	• Defining the concepts of pattern recognition. • Distinguish procedures, methods and algorithms related to pattern recognition. • Design and develop a pattern recognition system for the specific application. • Able to apply various decision making and clustering algorithms in pattern recognition systems.
	Data Warehousing and Mining	• Understanding the fundamental concepts of data mining and its stages. • Apply preprocessing techniques on the data. • Understand and apply association rules for large databases. • Analyze classification and prediction algorithms.
	Embedded Systems	• Able to demonstrate knowledge of basic principles of embedded systems. • Understand the various important parameters required for the development of embedded software. • Understand the software and hardware requirements for the development of embedded systems.

		• Study on case studies and applications of embedded systems.
	Advanced Computer Architecture	• Understand the concepts of computer architecture. • Understand the concept of Array Processing, Parallel algorithms for SIMD. • Analyze various multiprocessor architecture and its software requirements for multiprocessors.
	Mobile Communications	• To make students familiar with various generations of mobile communications • To understand the concept of cellular communication • To understand the basics of wireless communication • Knowledge of GSM mobile communication standard, its architecture, logical channels, advantages and limitations. • Knowledge of 4G mobile standards and their comparison with 3G technologies
	Digital Image Processing	• Review the fundamental concepts of digital image processing system. • Analyze images in the frequency domain using various transforms. • Analyze the concepts of image compression and segmentation.
	SC Lab and Web Design Lab	• Apply the validation and try real world outputs. • Apply the concepts of HTML for developing static webpage. • Demonstrate the concepts of Javascript and Java. • Implement the advanced concepts of java such as servlets and JSP to create dynamic web pages.
	Project Work	• Conduct literature survey on specified area of research. • Identify requirements of hardware/software for performing specified project. • Design and develop the software using various Software Engineering principles and practices. • Analyzing professional issues, ethics, legal and security issues related to projects. • Demonstrate the ability to communicate effectively in speech and writing. • Learn to work as a time and focus on developing the project within stipulated cost and time.