Program/ Course Outcomes

Program Educational Objectives

- Graduates of the program will become technically competent to pursue higher studies.
- Graduates of the program will utilize modern and advanced technological tools for performing Investigation, analysis and synthesis by identifying various computer solutions.
- Graduates of the program will collaborate with multidisciplinary teams and will be able to become leaders in their organization, their profession and in society.

Program Outcomes in Computer Science

- Ability to apply knowledge in mathematics, Accounting and science fundamentals to solve problems.
- Ability to use a range of programming languages and tools to develop computer programs to solve problems effectively.
- Design, and analyze precise specifications of algorithms, procedures, and interaction behavior.
- Ability to communicate effectively in both verbal and written form in industry and society.
- Ability to work in teams to build software systems and apply the technologies in various fields of Computer Science, including Mobile applications, Website development and management, databases, and computer networks.
- Ability to select appropriate techniques to tackle and solve problems in the discipline of information security management.
- Understand the basic concepts of system software, hardware and computer graphics

Program outcomes in Data Science

- Students will develop relevant **programming** abilities.
- Students will demonstrate proficiency with statistical **analysis of data**.
- Students will develop the ability to build and assess data-based **models**.
- Students will execute statistical analyses with professional statistical **software**.
- Students will demonstrate skill in **data management**.
- Students will apply data science concepts and methods to **solve** problems in real-world contexts and will **communicate** these solutions effectively

Program outcomes in BCA(Bachelor of Computer Applications)

- Understand the fundamental concepts of computers, software hardware and peripheral devices and evolution of computer technologies
- Familiar with business environment and information technology and its applications in different domains.
- Gain knowledge to identify, explain and apply functional programming and object-oriented programming techniques and use of databases to develop computer programs.
- Analyze, design, implement and evaluate computerized solutions to real life problems, using appropriate computing methods including web applications.
- Understand the front end and backend of software applications.
- Gain expertise in at least one emerging technology.
- Acquire knowledge about computer networks, network devices and their configuration protocols, security concepts at various level etc
- Apply techniques of software validation and reliability analysis to the development of computer programs.
- Acquire technical, communication and management skills to convey or present information, applications, instructions, policies, procedures, decisions, documentations etc. verbally as well as in writing
- Recognize the various issues related to society, environment, health and vivid cultures and understand the responsibilities to contribute in providing the solutions.
- Acquire technical skills to lead a productive life in the society as a professional or as an entrepreneur

Course Outcomes

Course Code	**Course Name**	**Course Outcomes**
	Programming in C	● Explain about the basic concepts of program development statements and its syntax. ● Explain the various types of arrays and its structure ● Discuss about the various types of Functions and String handling mechanisms ● Explain the Concepts of structures and Unions ● Illustrates the various operations performed on different types of files
	C Lab	After Completion of this course the student would be able to • Read, understand and trace the execution of programs written in C language. • Write the C code for a given algorithm. • Implement Programs with pointers and arrays, perform pointer arithmetic, and use the pre-processor. • Write programs that perform operations using derived data types.
	C++	Knowledge gained: ● an ability to incorporate exception handling in object-oriented programs ● an ability to use template classes and the STL library in C++ ● an understanding of the concepts of OOPs including inheritance and polymorphism ● an ability to overload operators in C++ ● an understanding of the difference between function overloading and function overriding Skills gained:

		● Logical thinking ● C++ Programming Competency developed: ● Ability to write object-oriented programs of moderate complexity in C++ Developing real world application using C++
	C++ Lab	Students can learn ● Programming developing concepts on 1) Abstraction 2) Inheritance 3)Constructors 4)Destructors 5)Polymorphism 6)Operator Overloading
	Data Structures with C++	Knowledge gained: ● Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms ● Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs ● Write programs that use arrays, records, linked structures, stacks, queues, trees, and graphs ● Demonstrate different methods for traversing trees ● Illustrate various technique to for searching, Sorting and hashing ● Describe the concept of recursion, give examples of its use, describe how it can be implemented using a stack ● Discuss the computational efficiency of the principal algorithms for sorting, searching, and hashing. ● Summarize different categories of data Structures Skills gained: ● Compare alternative implementations of data structures with respect to performance ● Compare and contrast the benefits of dynamic and static data structures implementations ● Explain the significance of dynamic memory management Techniques ● Identify different parameters to analyze the performance of an algorithm. Competency developed:

		● Choose appropriate data structures to solve real world problems efficiently. ● Design and implement an appropriate hashing function for an application Design algorithms to perform operations with Linear and Nonlinear data structures
	Database Management Systems	Knowledge gained: ● types of databases ● Detailed architecture, define objects, load data, query data and performance tune databases. ● writing SQL queries for the given problem statement Skills gained: ● Establish a basic understanding of the process of Database ● Develop ER diagram for representing conceptual data model ● Convert ER diagram into a set of relations representing logical data model Competency developed: ● Gain ability to handle large volumes of structured, semi-structured, and unstructured data using database technologies. ● Appreciate the need for DB approach and understand the components and roles of DBMS ● Apply DB system development life cycle to business problems ● Implement a set of relations in the chosen DBMS ● Development and Administration using MySQL.
	SQL Lab	● At the end of the course the students are able to: ● Apply the basic concepts of Database Systems and Applications. ● Use the basics of SQL and construct queries using SQL in database creation and interaction. ● Design a commercial relational database system (Oracle, MySQL) by writing SQL using the system. ● Analyze and Select storage and recovery techniques of database system
	Java	Knowledge gained: ● To learn Object Oriented Programming language ● To learn Inheritance, Function Overloading using Java ● To study Function Overriding, Final, Polymorphism

		● To handle abnormal termination of a program using exception handling ● To create flat files Competency developed: ● To develop a game application using multithreading To develop application using multithreading
	Java Lab	Knowledge gained: ● This lab work provides hands-on for Java. ● Java Programming assignments based on class, inheritance, abstraction, encapsulation, dynamic binding, polymorphism, I/O systems, exception handling Skills gained: ● Programming in Java Competency developed: ● Developing application to solve real world problem using Java ● Implement core Java programs to solve simple problems Implement Client and Server end Java programs Knowledge gained
	Web Technologies	● Describe the concepts of markup languages, un order list, table, formatting, Liking and frames. ● Discuss about the creation of cascading style sheets, backgrounds, media types and building a dropdown menu ● Explain the JavaScript, control structure, if structure, switch, do-while and logical operators. ● Describe the javascript functions, javascript arrays and javascript objects. ● Discuss the DOM, javascript events and XML.
	Web Technologies Lab	● Explain the fundamental tags used in HTML. ● Develop the web page in various applications. ● Develop the web page using various ordered and unordered listing commands ● Develop the web page using frame concepts with multi-media handling. ● Develop the web page using java script.
	Operating System	Knowledge gained: ● To understand Complexity of Operating system as a software ● To understand design issues related to process management

		and various related algorithms ● To understand design issues related to memory management and various related algorithms ● To understand design issues related to File management and various related algorithms ● Allocate Main Memory based on various memory management techniques ● Compare Memory allocation using Best fit, Worst fit, and first fit policies ● Apply page replacement policies for dynamic memory management ● Schedule CPU time using scheduling algorithm for processors ● Compare various device scheduling algorithms Skills gained: ● To evaluate, and compare OS components through instrumentation for performance analysis. ● To analyze the various device and resource management techniques for timesharing and distributed systems
	Operating System Lab	To understand the functionalities of various layers of OSI model· ● To explain the difference between hardware, software; operating systems, programs and files. ● Identify the purpose of different software applications. ● Simulate the following CPU scheduling algorithms. a) FCFS b) SJF c) Round Robin d) Priority. ● Write a C program to simulate producer-consumer problem using Semaphores ● Write a C program to simulate the concept of Dining-philosophers problem. ● Simulate MVT and MFT. ● Write a C program to simulate the following contiguous memory allocation Techniques a) Worst fit b) Best fit c) First fit. ● Simulate all page replacement algorithms a)FIFO b) LRU c) OPTIMAL ● Simulate all File Organization Techniques a) Single level directory b) Two level directory ● Simulate all file allocation strategies a) Sequential b) Indexed c)

		Linked. ● Simulate Bankers Algorithm for Dead Lock Avoidance ● Simulate Bankers Algorithm for Dead Lock Prevention.
	RDBMS	● Describe DBMS architecture, physical and logical database designs, database modeling, relational, hierarchical and network models. ● Identify basic database storage structures and access techniques such as file organizations, indexing methods including B-tree, and hashing. ● Learn and apply Structured query language (SQL) for database definition and database manipulation. ● Demonstrate an understanding of normalization theory and apply such knowledge to the normalization of a database. ● Understand various transaction processing, concurrency control mechanisms and database protection mechanisms.
	RDBMS Lab	● Implement Basic DDL, DML and DCL commands ● Understand Data selection and operators used in queries and restrict data retrieval and control the display order ● Write subqueries and understand their purpose ● Use Aggregate and group functions to summarize data ● Join multiple tables using different types of joins ● Understand the PL/SQL architecture and write PL/SQL code for procedures, triggers, cursors, exception handling etc..
	e Commerce	This course provides an introduction to information systems for business and management. It is designed to familiarize students with organizational and managerial foundations of systems, the technical foundation for understanding information systems After Completion of the subject student should able to • Understand the basic concepts and technologies used in the field of management information systems; • Have the knowledge of the different types of management information systems; • Understand the processes of developing and implementing information system
	Cyber Security	Upon successful completion of the programme, candidates will be familiar with cyber security landscapes and able to

		● Analyze and evaluate the cyber security needs of an organization. ● Determine and analyze software vulnerabilities and security solutions to reduce the risk of exploitation. ● Measure the performance and troubleshoot cyber security systems. ● Implement cyber security solutions and use of cyber security, information assurance, and cyber/computer forensics software/tools. ● Comprehend and execute risk management processes, risk treatment methods, and key risk and performance indicators ● Design and develop a security architecture for an organization. g) Design operational and strategic cyber security strategies and policies.
	Python Programming	Knowledge gained: ● To understand why Python is a useful scripting language for developers. ● To learn how to use lists, tuples, and dictionaries in Python programs. ● To learn how to identify Python object types. ● To learn how to use indexing and slicing to access data in Python programs. ● To define the structure and components of a Python program. ● To learn how to write loops and decision statements in Python. ● To learn how to write functions and pass arguments in Python. ● To learn how to build and package Python modules for reusability. ● To learn how to read and write files in Python. ● To learn how to design object-oriented programs with Python classes. ● To learn how to use class inheritance in Python for reusability. ● To learn how to use exception handling in Python applications for error handling. ● To acquire programming skills in core Python. ● To acquire Object Oriented Skills in Python Skills gained: ● To learn how to design and program Python applications. Competency developed: ● To develop the ability to write database applications in Python To develop the skill of designing Graphical user Interfaces in Python

	Design and Analysis of Algorithms	Knowledge gained: ● Analyze the running time and space complexity of algorithms. ● Describe, apply and analyze the complexity of divide and conquer strategy. ● Describe, apply and analyze the complexity of greedy strategy. ● Describe, apply and analyze the complexity of dynamic programming strategy. ● Explain and apply backtracking, branch and bound and string matching techniques to deal with some hard problems. ● Describe the classes P, NP, and NPComplete and be able to prove that a certain problem is NP-Complete. ● Describe analysis techniques for algorithms. ● Identify appropriate data structure and design techniques for different problems ● Identify appropriate algorithm to be applied for the various application like geometric modeling, robotics, networking, etc. ● Appreciate the role of probability and randomization in the analysis of algorithm ● Differentiate polynomial and non-deterministic polynomial algorithms. Skills gained: ● To provide mathematical approach for Analysis of Algorithms ● To solve problems using various strategies ● To provide mathematical approach for Analysis of Algorithms. ● To teach advanced data structures. ● To solve complex problems in real life applications. Competency developed: ● To analyze strategies for solving problems not solvable in polynomial time Analyze various algorithms
	Machine Learning	● Learn the basics of learning problems with hypothesis and version spaces ● Understand the features of machine learning to apply on real world problems

		● Characterize the machine learning algorithms as supervised learning and unsupervised learning and Apply and analyze the various algorithms of supervised and unsupervised learning ● Analyze the concept of neural networks for learning linear and non-linear activation functions ● Learn the concepts in Bayesian analysis from probability models and methods ● Understand the fundamental concepts of Genetic Algorithm and Analyze and design the genetic algorithms for optimization engineering problem