

PROGRAM SPECIFIC OUTCOMES

Bachelor of Computer Applications

- 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 Outcome: Bachelor of Computer Applications

Semester	Paper Name & Code	Course Outcome
Semester I	Computer Fundamentals	After completing this course, students will know about fundamentals of Computer System and Software.
	Introduction to C-Programming	At the end of the course, students will be able to: (a) Understand the basics of C programming like data types and operators (b) Understand and write program in C to implement conditions, loops, functions (c) Work on arrays, strings and basic file operations
	Mathematics I	After successful completion of this course, students will be able to: (a) Learn the concepts of set, relation, and function from Computer Science point of view. (b) Know how to view a table/database as an n-ary relation. (c) Learn what a matrix is and relate it with arrays used in programming. (d) Understand determinants and how determinants are used in solving simultaneous equations. (e) Get familiar with statistical and probabilistic measures that are used in computation related software/packages.
Semester II	Data Structures & Algorithms Using C	At the end of the course, students will be able to: (a) Understand and apply the fundamental data structures and algorithms – such as arrays, linked lists, stacks, queues, trees, sorting and searching algorithms using C programming language. (b) Analyze the time and space complexity of different algorithms and choose the appropriate algorithm for a given problem. (c) Develop efficient algorithms to solve various computational problems by utilizing data structures

		and algorithms covered in the course.
	Digital Logic Fundamentals	After completing this course, students will have grasp of fundamental concepts of digital logic that will make their base to understand the concepts of computer architecture and organization.
	Mathematics II	After successful completion of this course, students will be able to: i. Learn the basic concepts of limit, continuity and derivatives. ii. Understand graphs and its different representations in Computers. How to model real life problems using graphs. Learn a few basic graph traversal algorithms. iii. Understand the basic idea of counting and use it in counting under various constraints. iv. Understand Mathematical Logic from algorithmic point of view.
Semester III	Computer Organization and Architecture	Student will <input type="checkbox"/> be able to learn about the structure, function and characteristics of computer systems. <input type="checkbox"/> understand the design of the various functional units and components of computers. <input type="checkbox"/> identify the elements of modern instructions sets and their impact on processor design. <input type="checkbox"/> able to learn about the function of each element of a memory hierarchy. <input type="checkbox"/> able to learn about identify and compare different methods for computer I/O. <input type="checkbox"/> Student will able to learn about basics of assembly language.
	System Software	After completing this course, students will have understanding of various types of system software.
	Object Oriented Programming through C++	After successful completion of this course, students will be able to: <input type="checkbox"/> Will be able to imagine real-life concepts as objects; derive their properties and functions to operate. <input type="checkbox"/> Develop programs using object- oriented features like data abstraction, polymorphism, inheritance, exception handling. <input type="checkbox"/> Know C++ streams, operators <input type="checkbox"/> Know file handling techniques in C++.
Semester IV	Database Management System	On successful completion of this course, the student should be able to: a. Learn database concepts and its architectural components. b. Describe different data models used for designing a database. c. To create a database using relational models and entity relationships concepts d. Normalize a database into various normal forms e. Design SQL queries to handle a relational database.

	<p align="center">Operating system</p>	<p>After completing this course, students will have understanding of the internal structure and usage of various components related to an operating system.</p>
	<p align="center">Automata Theory and Languages</p>	<p>After completing this course, students will</p> <ul style="list-style-type: none"> <input type="checkbox"/> Understand the Mathematical model of a finite state machine. Know deterministic and nondeterministic versions of Finite automata. <input type="checkbox"/> Grasp the mathematical concepts of languages and grammar. <input type="checkbox"/> Know Pushdown Automata and the associated grammar/language. <input type="checkbox"/> Know the properties of Regular languages and Context free languages.
	<p align="center">Python Programming</p>	<p>After completing this course, students will know about fundamentals of Python Programming and Problem Solving.</p>
<p align="center">Semester V</p>	<p align="center">Software Engineering</p>	<p>On successful completion of this course, the student should be able to:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Determine the primary problems that impact all software development processes. <input type="checkbox"/> Choose relevant software development processes models, methodologies, and strategies for managing a specific software development process, and justify the choices <input type="checkbox"/> Implement different software estimation metrics such as cost, effort size, staffing etc. <input type="checkbox"/> Describe various software design approaches and various coding and testing strategies used in software engineering principles <input type="checkbox"/> Know about software reliability and how to calculate software maintenance cost.
	<p align="center">Web Technologies</p>	<p>At the end of the course, students will be able to:</p> <ul style="list-style-type: none"> <input type="checkbox"/> Understand the basic concept of web applications and web services. <input type="checkbox"/> Design basic well-structured web page using HTML and CSS <input type="checkbox"/> Develop the ability to implement interactive elements and dynamic content using basic JavaScript <input type="checkbox"/> Develop a foundational understanding of server-side scripting using PHP
	<p align="center">Java Programming</p>	<p>After completing this course, students will be familiar with the core concepts of java programming and classes of swing package.</p>
	<p align="center">Computer Networks</p>	<p>After completing this course, students</p> <ul style="list-style-type: none"> <input type="checkbox"/> Student will able to learn about the general principles of data communication. <input type="checkbox"/> Student will able to learn about how computer networks are organized with the concept of layered approach. <input type="checkbox"/> Student will able to learn about how signals are used

		<p>to transfer data between nodes.</p> <ul style="list-style-type: none"> <input type="checkbox"/> Student will able to learn about how packets in the Internet are delivered. <input type="checkbox"/> Student will able to learn about how routing protocols work. <input type="checkbox"/> Student will able to learn about functions of transport layer <input type="checkbox"/> Student will able to learn about functions of application layer
Semester VI	Information Security and Cyber Laws	After the completion of the course, the students will be able to develop basic understanding of security, cryptography, system attack and defences against them.
	Advanced Web Programming	<p>At the end of the course, students will be able to:</p> <ul style="list-style-type: none"> (a) Design basic well-structured web page using HTML and CSS (b) Develop the ability to implement interactive elements and dynamic content using basic JavaScript (c) Develop a foundational understanding of server-side scripting using PHP (d) Create a CRUD web application using HTML, CSS, JavaScript, PHP and MySQL.
	Mobile Application Development	<p>After completing this course, students will know:</p> <ul style="list-style-type: none"> (a) Fundamentals of Mobile Application Development. (b) Difference between Native and Cross Platform Applications. Pros and Cons of Each Approach. (c) To Design and Build a Complete Native Android Application with Both UI and Backend. (d) To Design and Build a Complete Cross Platform Application with Both UI and Backend
	Project	<p>While doing a project:</p> <ul style="list-style-type: none"> • It makes the student confident in designing an Online Project with advanced technologies on their choice • Students are trained to meet the requirements of the Industry