Nutan Maharashtra Vidya Prasarak Mandal's (NMVPM's)
NUTAN MAHARASHTRA INSTITUTE OF ENGINEERING AND TECHNOLOGY (NMIET)

Under Administrative Support - Pimpri Chinchwad Education Trust (PCET)

(An Autonomous Institute Approved by AICTE and Affiliated to SPPU, Pune)

Curriculum Structure and Syllabus
Of
First Year Master of Computer Applications (BCA)

Effective from Academic Year 2025-26

INDEX

PREAMBLE

Dear Students, teachers and all stakeholders The field of computing is rapidly expanding and changing, especially, since the last decade with continuous emergence of new disruptive technologies such as artificial intelligence, data science, cyber security, Internet of things, robotics and so on. 21st Century has witnessed rapid technological developments in every sector including the field of Computing. Moreover, it has created new job roles and massive job opportunities for budding graduates. Premium Institutes, public and private Universities, autonomous and affiliated colleges in India have always played a crucial role in producing human resources with required skill sets by capturing and monitoring these developments and offered various UG and PG programmes. Nutan Maharashtra Institute of Engineering and Technology (NMIET)**,** established in 2008 under the Nutan Maharashtra Vidya Prasarak Mandal, has consistently pursued its primary objective of providing quality technical education to the developing Maval region around Talegaon, Pune 3. As an AICTE-approved institution, NMIET has built a strong reputation for academic excellence and industry integration, with a recorded highest placement package of 23 LPA and an average of 3.5 LPA, attracting over 300 companies for campus recruitment.

The rapidly evolving field of computing has witnessed continuous emergence of disruptive technologies including artificial intelligence, data science, cybersecurity, Internet of Things, and robotics. These developments have created new job roles and massive employment opportunities for computing graduates. In response to these industry transformations and in alignment with the National Education Policy (NEP) 2020, NMIET is proud to introduce the Bachelor of Computer Applications (BCA) and Master of Computer Applications (MCA) programs starting from Academic Year 2025-26.

Program Overview and NEP 2020 Alignment

2.1 Credit Structure and Duration

- The BCA program follows a six-semester structure with a total of 120 credits, averaging 20 credits per semester
- The MCA program builds upon undergraduate computing education with its own 80-credit framework spread across four semesters

- Both programs incorporate the Multiple Entry/Exit system as recommended by NEP 2020, providing students with flexible academic pathways

2.2 Multiple Entry/Exit Options

- Certificate in Computer Applications: After successful completion of the first year (40 credits)
- Diploma in Computer Applications: After successful completion of the second year (80 credits)
- Bachelor of Computer Applications (BCA): After successful completion of the third year (120 credits)
- Bachelor of Computer Applications (BCA Honours): After successful completion of the entire program (160 credits combined)

This flexible academic structure allows students to tailor their educational journey according to their career aspirations and personal circumstances while receiving appropriate certification at each milestone.

PROGRAMME OUTCOME (PO's)

Graduates will be able to:

1. Scientific Knowledge: Apply the knowledge of mathematics, science fundamentals, and specialization to the solution of complex problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and 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. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern IT tools including prediction and modelling to complex activities with an understanding of the limitations.

6. The Graduate and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional practice.

7. Environment and sustainability: Understand the impact of the professional solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the professional practice.

9. Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. Communication: Communicate effectively on complex activities with the professional community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of the science 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.

12. 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.

COURSE SUMMARY

Department of Bachelor of Computer Applications (BCA)

SEMESTER-WISE COURSE DISTRIBUTION

Course Distribution : Semester Wise				
Sr. No.	**Type of Course**	**No. of Courses / Semester**		**Total**
		1	**2**	
1	Basic Science Course	1	1	2
2	Program Core Course	3	4	7
3	Program Elective Course	0	0	0
4	Open Elective	0	0	0
5	Vocational and Skill Enhancement Course	1	1	2
6	Ability Enhancement Course	1	0	1
7	Entrepreneurship/Economics/ Management Course	0	0	0
8	Experiential Learning Courses	2	2	0
	Total	**8**	**8**	**16**

SEMESTER-WISE CREDIT DISTRIBUTION

Credit Distribution : Semester Wise				
Sr. No.	**Type of Course**	**No. of Credits / Semester**		**Total**
		1	**2**	
1	Basic Science Course	3	3	06
2	Program Core Course	9	12	21
3	Program Elective Course	0	0	00
4	Open Elective	0	0	00
5	Vocational and Skill Enhancement Course	3	3	06
6	Ability Enhancement Course	3	0	03
7	Entrepreneurship/Economics/ Management Course	0	0	00
8	Experiential Learning Courses	4	4	08
	Total	**22**	**22**	**44**

Curriculum Structure

FIRST YEAR

BCA

First Year BCA Semester – I
CURRICULUM STRUCTURE

First Year BCA (With effect from Academic Year 2025-2026)																
Semester-I																
Course Code	**Course Name**	**Credit Scheme**				**Teaching Scheme (Hours/Week)**			**Evaluation Scheme and Marks**							**Total**
									FA		**SA**	**T W**	**PR**	**OR**		
		L	**P**	**T/A**	**Total**	**L**	**P**	**T/A**	**UT**	**CA**						
BCA01PC01	Programming and Problem Solving Using C	2	0	1	3	3	0	1	25	25	50	0	0	0		100
BCA 01PC02	Web Technology	2	0	1	3	2	0	1	25	25	50	0	0	0		100
BCA01PC03	Fundamental of Computer Architecture	2	0	1	3	2	0	1	25	25	50	0	0	0		100
BCA01BS01	Discrete Mathematics	2	0	1	3	3	0	1	25	25	50	0	0	0		100
BCA01AE01	UHV - I: IKS - Concepts and Application in Science	2	0	1	3	2	0	1	25	25	50	0	0	0		100
BCA01SE01	Soft Skills – Business Communication	2	0	1	3	2	0	1	25	25	50	0	0	0		100
BCA01PC01L	Programming and Problem Solving Using C Lab	0	2	0	1	0	4	0	0	0	0	10	30	10		50
BCA01PC02L	Web Technology Lab	0	2	0	1	0	4	0	0	0	0	10	30	10		50
Total		**12**	**4**	**6**	**22**	**14**	**8**	**6**	**150**	**150**	**300**	**20**	**60**	**20**		**700**

L-Lecture, P-Practical, T/A-Tutorial/Activity, FA–Formative Assessment, SA-Summative Assessment, TW-Term Work, OR-Oral, PR-Practical, CA – Course Activity.

First Year BCA (With effect from Academic Year 2025-2026)															
Semester-II															
Course Code	**Course Name**	**Credit Scheme**				**Teaching Scheme (Hours/Week)**			**Evaluation Scheme and Marks**						**Total**
									FA		**SA**	**T W**	**PR**	**OR**	
		L	**P**	**T/A**	**Total**	**L**	**P**	**T/A**	**UT**	**CA**					
BCA02PC01	Database Management System	2	0	1	3	3	0	1	25	25	50	0	0	0	100
BCA02PC02	Data Structures using C	2	0	1	3	3	0	1	25	25	50	0	0	0	100
BCA02PC03	Software Engineering	2	0	1	3	2	0	1	25	25	50	0	0	0	100
BCA02 PC04	Introduction to Cyber Security and Cyber Laws	2	0	1	3	2	0	1	25	25	50	0	0	0	100
BCA02BS01	Basics of Human Rights	2	0	1	3	2	0	1	25	25	50	0	0	0	100
BCA02SE01	Introduction to Data Science	2	0	1	3	2	0	1	25	25	50	0	0	0	100
BCA02PC01L	Database Management System Lab	0	2	0	1	0	4	0	0	0	0	10	30	10	50
BCA02PC02L	Data Structures using C Lab	0	2	0	1	0	4	0	0	0	0	10	30	10	50
Total		**12**	**4**	**6**	**22**	**14**	**8**	**6**	**150**	**150**	**300**	**20**	**60**	**20**	**700**

L-Lecture, P-Practical, T/A-Tutorial/Activity, FA–Formative Assessment, SA-Summative Assessment, TW-Term Work, OR-Oral, PR-Practical, CA – Course Activity.

BCA Course

FIRST YEAR

Semester I

Program:	BCA						Semester : I	
Course:	Programming and Problem Solving using C						Code : BCA01PC01	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisites: -

1. Basic understanding of computers and operating systems.
2. Logical reasoning and elementary problem-solving skills.

Course Objectives

1. To introduce problem-solving techniques and computational thinking.
2. To familiarize students with the fundamentals of C programming.
3. To enable students to develop modular programs using functions and recursion.
4. To apply structured programming concepts for real-world problem-solving.

Course Outcomes

On completion of the course, students will be able to:

1. Explain the structure and components of C programs.
2. Apply decision-making and looping constructs to solve problems.
3. Use arrays and strings for data manipulation.
4. Implement modular programming with functions and recursion.
5. Utilize pointers, structures, and file handling for problem-solving.

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Introduction & Problem Solving Basics** –Structure of a C Program with Examples, Creating and Executing a C Program, Compilation process in C. Algorithms, flowcharts, C Character Set, Identifiers and Keywords, Variables and constants Data types- Basic data types, Enumerated types, Type casting, Declarations, Expressions, Operators and Expressions Precedence and associativity Input Output Statements: printf, scanf functions, getchar, putchar, getch functions, gets, puts functions, Escape sequence characters, Format specifiers	6
2	**Control Structures and Functions** – Conditional statements (if, if-else, switch); looping constructs (for, while, do-while); break and continue; modular programming, built-in & user-defined functions, function	6

	prototypes, recursion, scope and storage classes.	
3	**Arrays and Strings** – 1D & 2D arrays, applications, memory representation; string operations & library functions; case studies (e.g., matrix operations, text processing).	**6**
4	**Pointers, Structures and File Handling** Basics of pointers, pointer arithmetic, pointers with arrays/functions; structures & unions; introduction to dynamic memory (malloc, free); file operations (create, read, write, update).	**6**
	Total	**24**

Text Books:

1. E. Balagurusamy, Programming in ANSI C, Tata McGraw Hill.
2. The C Programming Language- By Brian W Kernighan and Dennis Ritchie
3. Byron Gottfried, Programming with C, Schaum's Outline Series.

Reference Books

1. Kernighan & Ritchie, The C Programming Language, PHI.
2. Let us C- By Yashwant Kanetkar
3. Yashwant Kanetkar, Let Us C, BPB Publications.

E-Resources:

1. NPTEL: Programming in C – https://nptel.ac.in
2. Spoken Tutorial (IIT Bombay): C Programming – https://spoken-tutorial.org
3. TutorialsPoint: C Programming – [https://www.tutorialspoint.com/cprogramming/]

Program:	BCA						Semester : I	
Course:	Web Technology						Code : BCA01PC02	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Basic knowledge of computers

Course Objectives

This course aims at enabling students:

1. Understand key components and technologies of the World Wide Web.
2. Recall and apply HTML5 elements for creating static web pages.
3. Use CSS for designing structured and responsive layouts.
4. Apply JavaScript for basic interactivity and dynamic web page behavior.

Course Outcomes

After learning the course, the students should be able to:

1. Identify basic concepts of Internet and web technologies.
2. Develop simple static web pages using HTML5.
3. Apply CSS to design clean and responsive page layouts.
4. Implement JavaScript for interactive and dynamic functionality.

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Internet & Web Basics** Web Basics: Browsers, Servers, Static vs. Dynamic Web Pages, Client-side vs. Server-side Scripting, Web Protocols: HTTP, HTTPS, Web Hosting: Domain name, DNS, URL	6
2	**HTML5 Essentials** Document Structure (Head, Body, Metadata), Sections & Grouping Content, Text-level Elements (Headings, Paragraphs, Lists, Links), Images, Tables (basic), Forms (basic text input, buttons),Embedded Content (intro only)	6
3	**CSS for Design & Layout** CSS Basics, Inline/Embedded/External, CSS Properties:Fonts, Colors, Backgrounds, Box Model (Margin, Padding, Border), Positioning & Display (Block, Inline, Relative, Absolute), Responsive Design Basics (Media Queries – introduction only),	6

4	**Introduction to JavaScript & DHTML** JavaScript Basics: Variables, Data Types, Operators, Expressions, Functions & Events, Arrays & Dialog Boxes (alert, prompt, confirm), DOM Basics – Accessing & Modifying Elements, Simple Interactivity: Dynamically Changing Text, Style	**6**
	Total	**24**

Text Books:

1. Learn HTML for Beginners: The Illustrated Guide to Coding Paperback, Jo Foster
2. HTML: A Beginner's Guide, Fifth Edition: A Beginner's Guide, Fifth Edition: Course Load e-book for HTML A BEGINNERS GD 5E, Wendy Willard.
3. HTML and CSS: Design and Build Websites – Jon Duckett
4. Internet Technology and Web Design – IP Innovative Publication

Reference Books

1. JavaScript for Absolute Beginners (Expert's Voice in Web Development) Paperback, by Terry McNavage (Author)
2. Learn JavaScript Quickly: A Complete Beginner's Guide to Learning JavaScript, Even If You're New to Programming by Code Quickly.
3. Responsive Web Design with HTML5 and CSS – Ben Frain

E-Resources:

1. tps://www.w3schools.com/html/
2. https://www.tutorialspoint.com/html5/index.htmhttps://javascript.info/
3. Free Code Camp (freecodecamp.org)
4. MDN Web Docs (developer.mozilla.org)

Program:	BCA						Semester : I	
Course:	Fundamental of Computer Architecture						Code : BCA01PC03	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Basic knowledge of computers Architecture

Course Objectives

This course aims at enabling students:

1. To impart basic concepts of computer architecture and organization,
2. To explain key skills of constructing cost-effective computer systems.
3. To familiarize the basic CPU organization.
4. To help students in understanding various memory devices.

Course Outcomes

After learning the course, the students should be able to:

1. Identify various components of computer and their interconnection
2. Identify basic components and design of the CPU: the ALU and control unit.
3. Compare and select various Memory devices as per requirement.
4. Compare various types of IO mapping techniques.

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Structure of Computers** Computer types, Functional units, Basic operational concepts, Bus Structures, Software, Performance, Multiprocessors and Multicomputer, Data representation, Error detection and correction codes. Computer Arithmetic Addition and Subtraction, Multiplication and Division algorithms, Floating-point Arithmetic Operations, Decimal arithmetic operations.	6
2	**Basic Computer Organization and Design** Instruction codes, Computer Registers, Computer Instructions and Instruction cycle. Timing and Control, Memory-Reference Instructions, Input-Output and interrupt. Central processing unit: Stack organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Complex Instruction Set Computer (CISC), Reduced Instruction Set Computer (RISC), CISC vs RISC	6

3	**Register Transfer and Micro-Operations**: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro-Operations, Logic Micro-Operations, Shift Micro-Operations, Arithmetic logic shift unit. Micro-Programmed Control: Control Memory, Address Sequencing, Micro-Program example, Design of Control Unit.	**6**
4	**Memory System** Memory Hierarchy, Semiconductor Memories, RAM (Random Access Memory), Read Only Memory (ROM), Types of ROM, Cache Memory, Performance considerations, Virtual memory, Paging, Secondary Storage, RAID. I/O interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA. Multiprocessors: Characteristics of multiprocessors, Interconnection structures,	**6**
	Total	**24**

Text Books:

1. M. Moris Mano (2006), Computer System Architecture, 3rd edition, Pearson/PHI, India.

Reference Books

1. Carl Hamacher, Zvonks Vranesic, SafeaZaky (2002), Computer Organization, 5th edition, McGraw Hill, New Delhi, India.
2. William Stallings (2010), Computer Organization and Architecture- designing for performance,8th edition, Prentice Hall, New Jersy.
3. Anrew S. Tanenbaum (2006), Structured Computer Organization, 5th edition, Pearson Education Inc,
4. John P. Hayes (1998), Computer Architecture and Organization, 3rd edition, Tata McGraw Hill

E-Resources:

1. The Complete Cyber Security Course: Hackers Exposed --- https://www.udemy.com/course
2. Foundations of Cybersecurity----- https://www.coursera.org/
3. NPTEL Computer Architecture - Swayam platform
4. Coursera Cybersecurity Specializations
5. GeeksforGeeks Computer Organization and Architecture Tutorials

Program:	BCA						Semester : I	
Course:	Discrete Mathematics						Code : BCA01BS01	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Basic knowledge of set notation, introductory algebra, and foundational skills in logical reasoning.

Course Objectives

This course aims at enabling students:

1. Build mathematical maturity via logic and proof.
2. Model computing problems using sets, relations, functions, graphs, and trees.
3. Apply counting and recurrence for algorithmic analysis.
4. Connect discrete structures to CS applications (data models, circuits, networks).

Course Outcomes

After learning the course, the students should be able to:

1. Apply concepts of logic, proof techniques, sets, relations, and functions to model and analyze mathematical structures.
2. Employ counting principles, binomial theorem, inclusion–exclusion, and recurrence relations to solve combinatorial and algorithmic problems.
3. Demonstrate understanding of graphs and trees, and apply their properties in representation, traversal, and spanning tree problems.
4. Analyze and simplify Boolean expressions using Boolean algebra laws and Karnaugh Maps for logical circuit design

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Logic & Proof Techniques** Propositional Logic: propositions, truth values, logical connectives, truth tables, tautology, contradiction, contingency. Logical Equivalences: laws of logic, implication and equivalence, normal forms (CNF, DNF). Predicate Logic: predicates, quantifiers (universal/existential), rules of inference. Proof Methods: direct proof, proof by contraposition, proof by contradiction, counterexamples. Mathematical Induction: simple induction, strong induction, applications to divisibility and sequences.	6

2	**Sets, Relations & Functions** Set Theory: definitions, Venn diagrams, operations (union, intersection, complement, difference), algebra of sets, power set, Cartesian product. Relations: definition, representation (matrix & digraph), properties (reflexive, symmetric, transitive), closures, equivalence relations, partial order relations, Hasse diagrams. Functions: definition, domain/range, types (injective, surjective, bijective), composition and inverse functions. Pigeonhole Principle with applications. Algebraic Structures (brief introduction): semigroup, monoid, group with examples.	**6**
3	**Counting & Recurrence** Counting Principles: sum rule, product rule, permutations and combinations. Binomial Theorem and its applications. Principle of Inclusion–Exclusion: two-set and three-set cases. Recurrence Relations: Introduction and formulation of recurrence relations. Solving linear recurrence relations with constant coefficients (first and second order).Applications to counting problems and algorithm analysis (e.g., Fibonacci sequence, divide-and-conquer recurrences)	**6**
4	**Graphs, Trees & Boolean Algebra** Graphs**:** basic terminology, types of graphs, representations (adjacency matrix/list), degree of a vertex, handshaking lemma, subgraphs, paths, cycles, connectedness. Trees**:** definition and properties, binary trees, tree traversal (preorder, inorder, postorder), spanning trees, introduction to minimum spanning trees (Kruskal/Prim concepts only). Boolean Algebra**:** definition, axioms and properties, minterms and maxterms, canonical forms, simplification using Boolean identities, Karnaugh Maps (up to 3–4 variables).	**6**
	Total	**24**

Text Books:

1. R. P. Grimaldi, Discrete and Combinatorial Mathematics, Pearson.
2. C. L. Liu & D. P. Mohapatra, Elements of Discrete Mathematics, McGraw Hill.

Reference Books:

1. Kenneth H. Rosen, Discrete Mathematics and Its Applications, McGraw Hill.
2. Norman L. Biggs, Discrete Mathematics, Oxford University Press.
3. Tremblay & Manohar, Discrete Mathematical Structures with Applications to Computer Science, McGraw Hill.

E-Resources:

1. NPTEL: Discrete Mathematics (IITs)
2. MIT Open Course Ware: Mathematics for Computer Science
3. IGNOU MCS-013: Discrete Mathematics Study Material

Program:	BCA						Semester : I	
Course:	IKS – Concept and Application in Science						Code : BCA01AE01	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Basic understanding of Indian history and culture.
2. Fundamental concepts of science at the high school level.
3. General awareness of ancient Indian texts and traditions

Course Objectives

This course aims at enabling students:

1. To introduce the foundations and evolution of the Indian Knowledge System (IKS).
2. To analyze contributions of IKS in mathematics, astronomy, linguistics, architecture, health, and wellness.
3. To develop an appreciation for the scientific temper and logical reasoning present in IKS.
4. To understand the relevance and application of IKS concepts in modern scientific context.

Course Outcomes

After learning the course, the students should be able to:

1. Describe the historical background and key philosophies underpinning IKS.
2. Explain major contributions of IKS to mathematics, astronomy, linguistics, and health.
3. Apply logical frameworks and classification systems derived from IKS to scientific analysis.
4. Evaluate the impact of Indian architectural and urban planning principles.

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Introduction to Indian Knowledge System (IKS) and Vedic Foundations** Definition, significance, and organization of IKS, Historicity and salient aspects of IKS, Overview of Vedic literature: the four Vedas, sub-classifications, messages in Vedas, Basics of Vedāṅgas, Vedic life, introduction to Śikṣā, Vyākaraṇa, Nirukta, Chandas, Kalpa, Jyotiṣa	6
2	**Mathematics, Astronomy, and Knowledge Frameworks in IKS** Historical number systems in India, Bhūta-Saṃkhyā and Kaṭapayādi system, Indian Mathematics: algebra, geometry, trigonometry, combinatorics, contributions of mathematicians, magic squares, Astronomy: Indian	6

	astronomical contributions, calendar systems (Pañcāṅga), celestial coordinates, astronomical instruments (Yantras), Jantar Mantar, Indian schemes of knowledge: Prameya, Dravyas, attributes, logic (deductive & inductive), Pramāṇa, Siddhānta	
3	**Linguistics, Health, and Psychology in IKS** Indian approaches to linguistics: Aṣṭādhyāyī, phonetics, word generation, mnemonics, computational and recursive operations, the role of Sanskrit in NLP, Traditional health sciences: Āyurveda, Sapta-dhātavaḥ, Tri-doṣa, Agni, disease management, Dinacaryā, health and wellness regimen, Indian psychology, consciousness studies	**6**
4	**Architecture, Town Planning, and Applications** Arthaśāstra perspectives on town planning ,Vāstu-śāstra: principles and limbs, architecture of Indian temples, iconography, Relevance and applications of IKS in contemporary domains	**6**
	Total	**24**

Text Books:

1. Indian Knowledge System (IKS): Concepts and Applications in Science

Reference Books:

1. Indian Knowledge Systems: Concepts and Applications Prof. V. N. Jha Central University of Tibetan Studies, Sarnath 2020
2. Science in India: A Historical Perspective, A. Rahman National Book Trust 1999
3. Mathematics in India Kim Plofker, Princeton University Press 2009
4. History of Indian Science, Technology and Culture (Vol. 1-4) A. Rahman (ed.) Oxford University Press 2000
5. Vāstu-Sāstra: The Ancient Science of Architecture and Design - B. V. Doshi, Mapin Publishing 2006
6. Charaka Samhita & Sushruta Samhita – Ayurveda classics Various translators Chowkhamba Press

E-Resources:

1. NPTEL Course Material: Indian Knowledge System (Official Syllabus, e-texts, and videos)

Program:	BCA						Semester : I	
Course:	Business Communication						Code : BCA01SE01	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Basic English comprehension and writing skills; familiarity with common workplace communication.

Course Objectives

This course aims at enabling students:

1. To introduce the fundamentals and significance of business communication.
2. To develop oral, written, and non-verbal communication skills for organizational effectiveness.
3. To enable learners to overcome communication barriers and apply effective strategies in real-life scenarios.
4. To familiarize students with report writing, professional correspondence, and etiquette required in business settings..

Course Outcomes

After learning the course, the students should be able to:

1. On completion of the course, students will be able to:
2. Explain the process and types of communication in business environments.
3. Demonstrate effective oral, written, and non-verbal communication skills.
4. Identify and overcome barriers in business communication.
5. Write and interpret various business documents and participate in interviews using appropriate etiquette.

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Introduction to Communication** Meaning, Significance, Scope; Communication Process; Essentials of Good Communication; Channels of Communication (Formal, Informal); Directions: Upward, Downward, Horizontal	6
2	**Types & Modes of Communication** Types: Verbal (Oral & Written, their characteristics, advantages & limitations); Non-Verbal (Sign language, Body language, Kinesics, Proxemics, Time language, Hepatics); Effective use in business	6

3	**Interpersonal Communication & Barriers** Styles of Communication; Managing Motivation and Emotional Influence; Barriers (Technological, Socio-psychological, Semantic); Strategies to Overcome Barriers, Types of Listening	**6**
4	**Business Communication Skills** Report Writing (Formal Reports); Writing Effective Letters; Types of Business Letters (Inquiry, Order, Sales, Complaint, Adjustment & others); Interview Techniques; Communication Etiquettes	**6**
	Total	**24**

Text Books:

1. Meenakshi Raman and Prakash Singh, "Business Communication," Oxford University Press
2. Rajendra Pal & J S Korlahalli, "Essentials of Business Communication," Sultan Chand & Sons

Reference Books:

1. S. K. Mandal, "Effective Communication and Public Relations," Jaico Publishing
2. P. D. Chaturvedi & Mukesh Chaturvedi, "Business Communication: Concepts, Cases and Applications," Pearson

E-Resources:

1. NPTEL: Business Communication – nptel.ac.in
2. Swayam: Workplace Communication – swayam.gov.in
3. Tutorials Point: Business Communication – tutorialspoint.com

Program:	BCA					Semester : I
Course:	Programming and Problem Solving using C Lab					Code : BCA01PC01L
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks		
	Theory	Practical	Tutorial / Activity	TW	OR	PR	Total
1	-	2	-	10	10	30	50

Prerequisites:

1. Basic knowledge of C programming concepts (parallel to theory course).
2. Ability to construct algorithms and flowcharts for simple problems

Course Objectives

This course aims at enabling students:

1. Introduce the fundamentals of C programming, problem-solving techniques, and program structure.
2. Develop logical thinking through control structures, modular programming, and recursion.
3. Enable efficient use of arrays, strings, and data structures for problem solving.
4. Familiarize students with pointers, structures, and file handling for real-world applications

Course Outcomes

After learning the course, the students should be able to:

1. Apply basic constructs of C programming to solve computational problems.
2. Implement modular programs using control structures, functions, and recursion.
3. Utilize arrays, strings, and structures to develop solutions for data processing applications.
4. Demonstrate the use of pointers and file handling in developing interactive applications..

Guidelines:

Students will be assessed based on

The practical work done by them throughout the semester, the Practical Exam, and the Oral Exam. Practical work and practical exam collectively have the weightage of 30 marks and oral exam has the weightage of 20 marks.

Detailed Syllabus

Assign. No.	Suggested List of Assignments
1	Write a C program to display a simple message (Hello World) and demonstrate the structure of a C program.
2	Develop a program to find the largest of three numbers using conditional operators.

3	Write a program to calculate the roots of a quadratic equation (handle real and imaginary roots).
4	Implement a program to convert temperature from Celsius to Fahrenheit and vice versa using type casting.
5	Write a program to find the sum of digits of a given number using while loop.
6	Develop a program to generate the Fibonacci series up to *n* terms using recursion.
7	Write a menu-driven calculator program using switch statement (addition, subtraction, multiplication, division).
8	Implement a program to check whether a number is prime using functions.
9	Write a program to find the sum and average of elements in a 1D array.
10	Implement a program for matrix addition and multiplication using 2D arrays.
11	Write a program to count the number of vowels, consonants, digits, and spaces in a string.
12	Develop a program to check whether a given string is a palindrome using string functions.
13	Write a program to find the sum of all elements of an array using pointers.
14	Implement a structure to store student details (name, roll no, marks) and display the record of students.
15	Write a program to create a file, write some text into it, then read and display the contents of the file.

References:

1. E. Balagurusamy, *Programming in ANSI C*, Tata McGraw Hill.
2. The C Programming Language- By Brian W Kernighan and Dennis Ritchie
3. Byron Gottfried, *Programming with C*, Schaum's Outline Series
4. Kernighan & Ritchie, *The C Programming Language*, PHI.
5. Let us C- By Yashwant Kanetkar
6. Yashwant Kanetkar, *Let Us C*, BPB Publications

Program:	BCA						Semester : I
Course:	Web Technology Lab						Code : BCA01PC02L
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks			
	Theory	Practical	Tutorial / Activity	TW	OR	PR	Total
1	-	2	-	10	10	30	50

Prerequisites:

1. Basic knowledge of Internet protocols such as HTTP and HTTPS.
2. Concept of styling web pages with CSS

Course Objectives

This course aims at enabling students:

1. Fundamental understanding of the web, browsers, and protocols.
2. Ability to create structured HTML content.
3. Competence in applying CSS for page styling and layout.
4. Skills in using JavaScript for interactivity and content manipulation.

Course Outcomes

After learning the course, the students should be able to:

1. Identify web technologies and methodologies.
2. Develop simple static web pages.
3. Design responsive layouts with CSS.
4. Implement interactivity and dynamic content with JavaScript.

Guidelines:

Students will be assessed based on

The practical work done by them throughout the semester, the Practical Exam, and the Oral Exam. Practical work and practical exam collectively have the weightage of 30 marks and oral exam has the weightage of 20 marks.

Detailed Syllabus

Assign. No.	Suggested List of Assignments
1	Demonstrate webpage access using different browsers and basic browser features.
2	Compare (with short report/demo) static and dynamic pages, showing differences.
3	Design a personal profile page with headings, paragraphs, lists, and links.
4	Build a basic registration form (input, radio buttons, checkboxes, submit/reset buttons).

5	Insert images, external/internal links, and image links in an HTML page.
6	Display student marks in an HTML table with proper headers and formatting.
7	Organize content using sections, divs, and semantic tags for better structure.
8	Create a homepage styled using external CSS—set background colors, font styles, margins, and padding.
9	Develop a responsive webpage using CSS media queries, adjusting layout for mobile and desktop.
10	Implement a horizontal navigation bar using unordered lists and style it with CSS.
11	Demonstrate use of margin, padding, border, and box-sizing with at least two sample elements.
12	Write JavaScript code to validate required fields in a form (e.g., name, email) before submission.
13	Create JavaScript that changes the text color or content of a paragraph/button on click.
14	Write a script that accepts input in an array and displays output in a formatted HTML list.
15	Use JavaScript to dynamically add, remove, or modify elements in an HTML document via DOM.

References:

1. HTML and CSS: Design and Build Websites - Jon Duckett
2. JavaScript and JQuery: Interactive Front-End Web Development) -- Jon Duckett
3. Learning Web Design: A Beginner’s Guide to HTML, CSS, JavaScript, and Web Graphics - Jennifer Niederst Robbins
4. WEB TECHNOLOGY: A Developer's Perspective- Gopalan & Akilandeswari
5. HTML Tutorial: https://www.w3schools.com/html/
6. HTML Tutorial: https://www.tutorialspoint.com/html/index.htm
7. HTML Tutorial: https://www.javatpoint.com/html-tutorial

BCA Course

First Year

Semester II

Program:	BCA						Semester : I	
Course:	Database Management System						Code : BCA02PC01	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Basic knowledge of computer operations and programming concepts.

Course Objectives

This course aims at enabling students:

1. To introduce the fundamental concepts, architecture, and features of Database Management Systems.
2. To develop understanding of various data models, especially the relational and ER models, and their roles in database design.
3. To enable students to design relational schemas and construct SQL queries for data manipulation and retrieval.
4. To impart knowledge about normalization, database integrity, and basic security measures in DBMS applications.

Course Outcomes

After learning the course, the students should be able to:

1. Apply the fundamental concepts and architecture of database management systems to real-world data storage and retrieval problems.
2. Implement relational database schemas for a given application scenario.
3. Write SQL queries to perform data definition, manipulation, and transaction processing on databases.
4. Analyze and implement normalization techniques for efficient database design, ensuring data integrity and minimizing redundancy..

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Introduction to Database Management System** What is DBMS? Need and features of DBMS, comparison with file processing system, functions of DBMS. Basic terms: Data, information, database, schema, users, architecture, data dictionary.	6

2	**Data Models and ER Model** Types of data models (Hierarchical, Network, Relational), Entity-Relationship (E-R) Model, attributes, relationships, keys (primary, foreign, candidate, super), normalization concepts.	**6**
3	**Relational Model and SQL** Relational database concepts, relational algebra basics, relational schemas. Structured Query Language (SQL): data definition and manipulation, constraints, aggregate functions. Introduction to PL/SQL and transaction basics.	**6**
4	**Database Design and Security:** Database design process, basics of normalization (1NF, 2NF, 3NF), integrity constraints, basics of security in DBMS, roles/responsibilities of DBA, database backup and recovery.	**6**
	Total	**24**

Text Books:

1. C.J. Date, An Introduction to Database Systems, Pearson Education, 2004.
2. Ramez Elmasri & Shamkant B. Navathe, Fundamentals of Database Systems, Pearson, 2017.
3. Raghu Ramakrishnan & Johannes Gehrke, Database Management Systems, McGraw Hill Education, 2002.

Reference Books:

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, McGraw Hill Education, 2019.
2. Peter Rob, Carlos Coronel, Database Systems*:* Design, Implementation, and Management, Cengage Learning, 2015.

E-Resources:

1. IBM Developer Team, "IBM DBMS Resources", IBM Developer, 2025.
2. Microsoft, "SQL Guide", Microsoft Learn, 2025

Program:	BCA						Semester : I	
Course:	Data structure using C.						Code : BCA02PC02	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Basic Programming Concepts.
2. Basic Mathematics.
3. Introduction to Algorithms.
4. Memory Management.

Course Objectives

This course aims at enabling students:

1. Understand and explain fundamental concepts of data structures and algorithms. To understand the fundamentals of array data structures.
2. Apply data structures and algorithms to solve computational problems.
3. Analyze and compare searching and sorting algorithms for efficiency and applicability.
4. Design and construct efficient solutions using advanced data structures like trees and graphs

Course Outcomes

After learning the course, the students should be able to:

1. Understand basic data structures and algorithms.
2. Implement Linked Lists and apply appropriate searching techniques on Arrays to solve computational problems.
3. Apply fundamental Stack and Queue operations and apply them to solve computational problems
4. Design efficient solutions using trees and graphs.

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Introduction to Data Structure:** Introduction, Basic concepts, Data types and data objects, Abstract Data Types (ADT),Types of Data Structures: Linear and non -linear ,Algorithm analysis: Frequency counts, Space and Time complexity	6
2	**Arrays & Linked Lists:** Intro. To Matrix representation, techniques with time complexity, searching techniques with time Complexity, Introduction, Types of linked lists, Real world applications of Link list.	6
3	**Stacks and Queues:** Introduction ,Representation of Stack: Using arrays and Linked Lists , Operations on stack: push, pop , Applications of Stack ,	6

	Representation of Queues, Operations on queue, Types of queues: Circular queue and Priority queue	
4	**Trees & Graphs:** Introduction and terminologies ,Types of Binary Trees - Representation of Trees using arrays and linked lists, Applications of Binary trees , Binary Search Tree (BST), operations on BST, Introduction and Graph terminologies ,Representation of a Graph, Graph Traversals.	**6**
	Total	**24**

Text Books:

1. Horowitz, Ellis and Sahani Sartaj, "Fundamentals of Data Structures",1st Edition, Galgotia,1984 2.
2. Kamthane, Ashok N., "Introduction to Data Structures using C",1st Edition, Pearson,2004
3. Bandopadhya, S. K. and Dey, K. S. "Data Structures using C", 1 st Edition, Pearson, 2004

Reference Books:

1. Srivastava, S. K. and Srivastava, D., "Data Structures using C",1st Edition, BPB Publication, 2004
2. Gilberg, Richard F. and Forouzan, Behrouz A., "Data Structures: A Pseudocode approach with C", 2nd Edition, Cengage Learning, 2007

E-Resources:

1. https://www.geeksforgeeks.org/dsa/data-structure-meaning/
2. https://www.programiz.com/c-programming/examples
3. https://www.w3schools.com/c/

Program:	BCA						Semester : I	
Course:	Software Engineering.						Code : BCA02PC03	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Students Should know basic about Software Development process, Programming Fundamentals.

Course Objectives

This course aims at enabling students:

1. Explain software engineering concepts and its importance in software development.
2. Analyze the fundamental structure and phases of the Software Development Life Cycle (SDLC)
3. To analyze the Software Requirements Engineering (SRE) process, distinguishing between system needs and implementation details.
4. To enable students to analyze software requirements and create a well-documented software blueprint

Course Outcomes

After learning the course, the students should be able to:

1. Understand the key differences between software development and traditional engineering,
2. Compare and contrast the Waterfall and Iterative models
3. Apply requirements elicitation techniques.
4. Design a modular software component from a set of requirements

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Introduction to Software Engineering:** Definition, importance, and goals, Software Development vs. Traditional Engineering, Key differences and why a systematic approach is necessary. Characteristics of Good Software: Reliability, efficiency, maintainability, usability, Software Evolution: How software changes over time and the challenges involved.	6
2	**Software Development Life Cycle (SDLC)** Software Development Life Cycle (SDLC) Models: Sequential vs. Iterative, Waterfall Model: A traditional SDLC model (requirements, design, implementation, testing, deployment, and maintenance), Iterative Models.	6

3	**Software Requirements Engineering:** Introduction to Requirements Engineering, Elicitation Techniques, Requirements Analysis & Specification, Modeling Requirements Non-Functional Requirements & Validation, Requirements Management	**6**
4	**Software Design** Translating Requirements into a Blueprint, Applying Design Principles, Employing Design Approaches, Utilizing Design Tools and Notations ,Importance of Well-Documented Designs.	**6**
	Total	**24**

Text Books:

1. Software Engineering: A Practitioner's Approach" by Roger S. Pressman.
2. Software Engineering" by Ian Sommerville.
3. Fundamentals of Software Engineering" by Rajib Mall

Reference Books

1. Software Requirements" by Karl Wiegers
2. Software Testing: Principles and Practices" by Srinivasan Desikan and Gopal swamy Ramesh
3. Lessons Learned in Software Testing" by Cem Kaner, James Bach, and Bret Pettichord

E-Resources:

1. GeeksforGeeks: https://www.geeksforgeeks.org/ software engineering/software-engineering/
2. Stack Overflow: https://stackoverflow.com/ Stack Overflow
3. freeCodeCamp: https://www.freecodecamp.org/news/tag/online-courses/ freeCodeCamp.org
4. Coursera: https://www.coursera.org/ Coursera

Program:	BCA						Semester : I	
Course:	Introduction to Cyber Security						Code : BCA02PC04	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Basic knowledge of computers, use of the internet

Course Objectives

This course aims at enabling students:

1. To introduce core concepts and importance of cyber security in the modern digital world.
2. To explain different cyber threats—malware, phishing, ransomware, social engineering, hacking—and real-world examples.
3. To teach fundamental cyber safety practices for securing computers, mobile devices, networks, and online identities.
4. To provide understanding of Indian laws such as the IT Act, data privacy and basics of international laws (GDPR).

Course Outcomes

After learning the course, the students should be able to:

1. Identify and describe various cyber-attacks, threats, and vulnerabilities that affect individuals and organizations.
2. Demonstrate steps to secure personal devices and accounts, including installing antivirus, updating software, and creating strong passwords.
3. Recognize the risks of public Wi-Fi, unsafe downloads, and online scams, practice safe browsing.
4. Explain the Indian Cyberspace, IT Act.

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Fundamentals of Cyber Security** Need for cyber security, types of threats—viruses, Trojans, worms, spyware, adware, phishing, ransomware, case studies of real cyber-attacks, essential definitions(attack, vulnerability, exploit, risk),Cybercrime and Cyber terrorism.	6
2	**Protecting Systems and Networks** System vulnerabilities, firewalls and antivirus tools, network security basics, safe use of Wi-Fi and Bluetooth, understanding two-factor authentication,	6

	password management tools, software updates and security patches. Antispyware, IDS, Virtual Private Networks.	
3	**Safe Online Behavior and Privacy** Recognizing malicious emails, safe browsing habits, risks of social media, privacy settings and protecting digital footprints, handling cyberbullying, online harassment, and cyberstalking, protecting sensitive data.	**6**
4	**I.T. ACT** Introduction, Cyber Security Regulations, Roles of International Law, the state and Private Sector in Cyberspace, Cyber Security Standards. The INDIAN Cyberspace, I.T. Act	**6**
	Total	**24**

Text Books:

1. S. Godbole & A. Belapure, Cyber Security: Understanding Cyber Crimes, Computer Forensics, and Legal Perspectives, Wiley, 1st Edition, 2011.
2. Anirudh Rastogi, Cyber Law and Ethics, Bharat Law House, 1st Edition, 2014.
3. William Stallings, Cryptography and Network Security: Principles and Practice, Pearson, 7th Edition, 2017.

Reference Books

1. Sushma Bansal, Cyber Laws in India, Allahabad Law Agency, 3rd Edition, 2023.
2. Mary Manjikian, Cybersecurity Ethics, Taylor & Francis, 2nd Edition, 2023.
3. DSCI, Information Security Best Practices, Latest Edition.
4. NASSCOM, Cybersecurity Frameworks Guide, 2023.

E-Resources:

1. National CERT-In: https://www.cert-in.org.in
2. Information Security Awareness (CDAC): https://www.infosecawareness.in
3. University of Mumbai e-resources: https://mu.ac.in
4. SANS Institute: https://www.sans.org
5. NIST Cybersecurity Framework: https://www.nist.gov/cyberframework

Program:	BCA						Semester : I	
Course:	Basics of Human Rights						Code : BCA02BS01	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. General Awareness recommended

Course Objectives

This course aims at enabling students:

1. To introduce the concept, evolution, and historical significance of human rights.
2. To make students aware of national and international human rights frameworks.
3. To develop an understanding of rights and duties, and their relationship.
4. To discuss the Indian Constitution's provisions on human rights, fundamental rights, and duties.

Course Outcomes

After learning the course, the students should be able to:

1. Understand the basic philosophy, values, characteristics, and evolution of human rights.
2. Identify important human rights declarations and conventions.
3. Analyze the provisions of the Indian Constitution relating to human rights and duties.
4. Examine and discuss current human rights issues in India and globally.

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)
1	**Introduction to Human Rights** Definition, concept, characteristics, importance, philosophical and historical background (Indian and global), classification of rights, correlation between rights and duties.	6
2	**Evolution and Development of Human Rights** Major historical milestones: Magna Carta, Bill of Rights, French Revolution, UN Charter, Universal Declaration of Human Rights (UDHR), International Covenants, natural and legal rights, classification of human rights (civil, political, economic, social and cultural, collective rights).	6
3	**Human Rights in Indian Constitution** Preamble, Fundamental Rights and Duties, Directive Principles of State Policy, special provisions for vulnerable groups, role of judiciary, National Human Rights Commission and State Commissions.	6

4	**Major Issues and Challenges** Human rights issues: poverty, illiteracy, gender violence, social inequality, custodial violence, child rights, rights of minorities, weaker sections, sustainable development.	**6**
	Total	**24**

Text Books:

1. Manikrao Jadhav, Teaching of Human Rights, Himalaya Publishing, 1st Edition, 2011
2. Satvinder Kaur, Human Rights and Duties, Unistar Books, 1st Edition, 2012
3. Paras Diwan, Human Rights and the Law, Universal Law Publishing, 1998
4. D.D. Basu, Introduction to the Constitution of India, LexisNexis, 24th Edition, 2020

Reference Books

1. Mohanti M., Peoples' Rights, Sage Publications, 2000
2. Pal R.M., Human Rights Education, PUDR, 2001

E-Resources:

1. National Human Rights Commission: https://nhrc.nic.in
2. University of Minnesota Human Rights Library: https://www1.umn.edu/humanrts/
3. OHCHR Human Rights Education: https://www.ohchr.org/en/resources/educators/human-rights-education-training

Program:	BCA						Semester : II	
Course:	Introduction to Data Science						Code : BCA02SE01	
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks				
	Lecture	Practical	Tutorial / Activity	FA		TW	SA	Total
				UT	CA			
3	2	-	1	25	25	-	50	100

Prerequisite:

1. Foundational Mathematics & Statistics
2. Database Concepts
3. Computer Fundamentals
4. Problem-Solving & Analytical Thinking
5. Curiosity & Eagerness to Learn

Course Objectives

This course aims at enabling students:

1. Remember the definitions and scope of Data Science, its process from data collection to deployment, and its real-world applications.
2. Apply SQL commands (SELECT, FROM, WHERE, JOIN) and fundamental database concepts to retrieve and manipulate data within a DBMS like MySQL.
3. Create various types of charts and graphs using tools like Excel and Tableau to visualize and communicate data findings effectively.
4. Explain the core concepts of Machine Learning, differentiating between supervised and unsupervised learning, and the principles behind simple algorithms like linear regression or decision trees.

Course Outcomes

After learning the course, the students should be able to:

1. Define and recall foundational data science concepts and their applications.
2. Differentiate and classify various types of data and assess data quality issues.
3. Apply SQL commands to extract and manipulate data from relational databases.
4. Analyze data and generate visual representations using appropriate tools and techniques.
5. Explain fundamental machine learning concepts and distinguish between supervised and unsupervised learning approaches.
6. Evaluate the ethical implications of data science and demonstrate an awareness of emerging trends and technologies. (Evaluating)

Detailed Syllabus

Unit No.	Description	Duration (Hrs.)

1	**Introduction to Data Science, Data and its Characteristics**: Definition and scope of Data Science, The Data Science process: from data collection to deployment, Real-world applications of Data Science, Types of Data: structured, unstructured, and semi-structured.	**6**
2	**Introduction to Databases and SQL** Basic database concepts: tables, rows, columns, keys, SQL fundamentals: SELECT, FROM, WHERE, JOIN, etc. Working with a database management system (DBMS) like MySQL.	**6**
3	**Data Analysis and Visualization:** Introduction to exploratory data analysis (EDA), Basic statistical concepts relevant to data analysis, Introduction to visualization tools: Excel and Tableau, Creating charts and graphs to represent data.	**6**
4	**Machine Learning** Introduction to Machine Learning concepts, Supervised and unsupervised learning, Simple machine learning algorithms like linear regression or decision trees.	**6**
	Total	**24**

Text Books:

1. Data Science from Scratch: First Principles with Python by Joel Grus.
2. Python Data Science Handbook: Essential Tools for Working with Data by Jake VanderPlas.
3. Introduction to Data Science by Laura Igual and Santi Seguí.

Reference Books

1. Data Science for Dummies by Lillian Pierson
2. SQL Quick Start Guide: The Simplified Beginner's Guide to Managing, Analyzing, and Manipulating Data With SQL by Walter Shields
3. Mastering Data Visualization with Tableau by Arpana Chaturvedi and Praveen Malik.

E-Resources:

1. https://www.coursera.org/courses?query=data%20science&topic=Data%20Science
2. https://developers.google.com/machine-learning/crash-coursehttps://www.mygreatlearning.com/data-science/courses
3. https://www.geeksforgeeks.org/mysql/mysql-tutorial/
4. https://www.tableau.com/community/public

Program:	BCA						Semester : II
Course:	Database Management System Lab						Code : BCA02PC01L
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks			
	Theory	Practical	Tutorial / Activity	TW	OR	PR	Total
1	-	2	-	10	10	30	50

Prerequisites:

1. Basic knowledge of computer operations and programming concepts.

Course Objectives

This course aims at enabling students:

1. To introduce the fundamental concepts, architecture, and features of Database Management Systems.
2. To develop understanding of various data models, especially the relational and ER models, and their roles in database design.
3. To enable students to design relational schemas and construct SQL queries for data manipulation and retrieval.
4. To impart knowledge about normalization, database integrity, and basic security measures in DBMS applications.

Course Outcomes

After learning the course, the students should be able to:

1. Apply the fundamental concepts and architecture of database management systems to real-world data storage and retrieval problems.
2. Implement relational database schemas for a given application scenario.
3. Write SQL queries to perform data definition, manipulation, and transaction processing on databases.
4. Analyze and implement normalization techniques for efficient database design, ensuring data integrity and minimizing redundancy.

Guidelines:

Students will be assessed based on

The practical work done by them throughout the semester, the Practical Exam, and the Oral Exam. Practical work and practical exam collectively have the weightage of 30 marks and oral exam has the weightage of 20 marks.

Detailed Syllabus

Assign. No.	Suggested List of Assignments

1	Install and set up a Database Management System such as MySQL, Oracle, or PostgreSQL.
2	Create a new database and define tables with appropriate data types.
3	Insert data into database tables using SQL INSERT statements.
4	Update existing data in database tables using SQL UPDATE statements.
5	Delete data from database tables using SQL DELETE statements.
6	Retrieve specific records from tables using SQL SELECT queries with WHERE clauses.
7	Use ORDER BY and DISTINCT clauses in SQL queries to sort data and remove duplicates.
8	Apply aggregate functions such as COUNT, SUM, AVG, MIN, and MAX in SQL queries.
9	Perform INNER JOIN and LEFT JOIN operations to combine data from multiple tables.
10	Define and enforce primary key and foreign key constraints on tables.
11	Create and use views to simplify complex data retrieval.
12	Normalize database tables up to the Third Normal Form (3NF).
13	Write and execute basic PL/SQL or procedural code blocks
14	Set up user roles and assign privileges to maintain database security.
15	Perform database backup and recovery using appropriate DBMS utilities.

References:

1. Abraham Silberschatz, Henry F. Korth, S. Sudarshan, Database System Concepts, McGraw Hill Education, 2019.
2. Peter Rob, Carlos Coronel, Database Systems: Design, Implementation, and Management, Cengage Learning, 2015.

Program:	BCA						Semester : II
Course:	Data structure using C Lab						Code : BCA02PC02L
Credits	Teaching Scheme (Hrs. / Week)			Evaluation Scheme and Marks			
	Theory	Practical	Tutorial / Activity	TW	OR	PR	Total
1	-	2	-	10	10	30	50

Prerequisites:

1. Basic Programming Concepts.
2. Basic Mathematics.
3. Introduction to Algorithms.
4. Memory Management.

Course Objectives

This course aims at enabling students:

1. Understand and explain fundamental concepts of data structures and algorithms. To understand the fundamentals of array data structures.
2. Apply data structures and algorithms to solve computational problems.
3. Analyze and compare searching and sorting algorithms for efficiency and applicability.
4. Design and construct efficient solutions using advanced data structures like trees and graphs

Course Outcomes

On completion of the course, learners should be able to understand the

1. Understand basic data structures and algorithms.
2. Apply appropriate data structures to solve computational problems.
3. Analyze various searching and sorting algorithms for performance.
4. Design efficient solutions using trees and graphs.

Guidelines:

Students will be assessed based on

The practical work done by them throughout the semester, the Practical Exam, and the Oral Exam. Practical work and practical exam collectively have the weightage of 30 marks and oral exam has the weightage of 20 marks.

Detailed Syllabus

Assign. No.	Suggested List of Assignments
1	Generates an array of 20 random integers.
2	Implement the Insertion Sort algorithm-Test it on three types of input arrays: a randomly shuffled array, an already sorted array, and a reverse-sorted array.

3	Modify the merge function to also count the number of *inversions* in the original array. (An inversion is a pair of indices (i, j) such that i < j and arr[i] > arr[j]).
4	Write a function partition that places the pivot in its correct position and returns its index.
5	Implement a singly linked list with the following functions-insertAtEnd(data),insertAtBeginning(data),deleteNode(key) (deletes the first occurrence),search(key),display()
6	Implement a doubly linked list with the following operations-Insertion at the front, end, and after a given node.
7	Implement a stack using an array (static size)-Use this stack to implement an algorithm that converts an infix expression (e.g., A+B*(C-D)) to postfix notation (ABCD-*+).
8	Implement a circular queue using an array to avoid wasted space-Use the standard operations: enqueue, dequeue, isFull, isEmpty, and display.
9	Build a binary tree (not necessarily a BST) by inserting nodes level-by-level.
10	Implement a Binary Search Tree (dynamic nodes) with the following operations-insert(node),search(key),deleteNode(key)
11	Read a set of edges (vertex pairs) from a user or a file – To represent the graph using an Adjacency Matrix.
12	Implement a Breadth-First Search (BFS) traversal starting from a given vertex, printing the order of visited nodes.

References:

1. Data Structures and Algorithm Analysis in C by Mark Allen Weiss.
2. Data Structures and Algorithms Made Easy by Narasimha Karumanchi.
3. Algorithms in C by Robert Sedgewick