

DHANALAKSHMI SRINIVASAN

INSTITUTE OF RESEARCH & TECHNOLOGY

(Deemed to be University)

THURAIYUR ROAD- PERAMBALUR- 621212.

TAMILNADU- INDIA.

SCHOOL OF ARTS AND SCIENCE

CURRICULUM AND SYLLABUS

B.C.A COMPUTER APPLICATIONS

REGULATION 2026

PROGRAMME:

Vision:

- To stand as a dynamic center of excellence in computer applications, inspiring innovative technologists, analytical thinkers, and ethical digital professionals who harness computing power to drive technological advancement, efficiency, and positive societal transformation.

Mission:

- To offer inclusive, adaptable, and interdisciplinary undergraduate programs in computer applications that combine core computing principles with emerging technologies, programming, and data sciences, building graduates equipped with robust technical, problem-solving, and communication skills for diverse career paths and ongoing professional development.
- To advance cutting-edge research and creative inquiry in software development, digital solutions, artificial intelligence, and cybersecurity, tackling key challenges in industry, healthcare, education, and community services.
- To broaden access to quality computing education for rural and underrepresented students by merging contemporary IT skills with local needs, preserving traditional knowledge through digital tools, and cultivating entrepreneurial abilities for self-reliant career growth.
- To nurture ethical practices, social responsibility, and global outlook via hands-on projects, industry linkages, community initiatives, and experiential learning that connect theoretical concepts with practical real-world computing applications.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

PEO	PROGRAM EDUCATIONAL OBJECTIVES
PEO1	To impart a strong foundation in programming, database management, software engineering, networking, and web technologies, enabling graduates to pursue rewarding careers in IT industry, software development, and allied sectors.
PEO2	To cultivate logical thinking, analytical abilities, and technical proficiency so that graduates can design, develop, test, and maintain efficient software solutions and computer applications for real-life problems.
PEO3	To develop professional ethics, teamwork, communication skills and leadership qualities that prepares graduates to work collaboratively in multidisciplinary environments and contribute responsibly to the IT ecosystem.
PEO4	To encourage continuous learning and adaptability, empowering graduates to keep pace with rapid technological changes and explore higher studies, research, or entrepreneurial opportunities in computer applications.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO	PROGRAM SPECIFIC OUTCOMES
PSO1	Graduates will be capable of applying core concepts of programming, algorithms, data structures, and software development methodologies to create robust, user-friendly applications and digital solutions.
PSO2	Graduates will demonstrate expertise in database handling, web and mobile application development, cyber security basics, and emerging tools, enabling them to contribute effectively to the IT industry, digital entrepreneurship, and community technology projects.

PROGRAM OUTCOMES (Pos)

PO	PROGRAMOUTCOMES
PO1	PO1: Disciplinary Knowledge Demonstrate comprehensive knowledge and understanding of the fundamental concepts, theories, principles, and applications in the chosen discipline and allied fields.
PO2	PO2: Critical Thinking and Problem Solving Identify, analyze, and solve complex academic, professional, and societal problems using logical reasoning, analytical skills, and evidence-based approaches.
PO3	PO3: Research and Inquiry Apply scientific methods, research techniques, and investigative skills to conduct studies, analyze data, interpret results, and draw meaningful conclusions.
PO4	PO4: Creativity and Innovation Develop innovative ideas, creative solutions, and entrepreneurial approaches to address contemporary challenges in academic, professional, and social contexts.
PO5	PO5: Digital and Technological Competence Utilize appropriate digital technologies, software tools, and information resources effectively and ethically for learning, research, communication, and problem-solving.
PO6	PO6: Environment and Sustainability Understand environmental issues and sustainable development goals and apply this knowledge to promote ecological balance and responsible resource utilization.
PO7	PO7: Ethics and Social Responsibility Demonstrate ethical values, integrity, inclusiveness, and social responsibility while contributing positively to society and professional environments.
PO8	PO8: Teamwork and Leadership Work effectively as an individual and as a member or leader of diverse and multidisciplinary teams, demonstrating leadership, collaboration, and interpersonal skills.
PO9	PO9: Communication Skills Communicate effectively through oral, written, visual, and digital means with diverse audiences in academic, professional, and societal settings.
PO10	PO10: Employability and Entrepreneurship Develop professional competencies, managerial abilities, and entrepreneurial skills required for employment, higher education, and self-employment opportunities.
PO11	PO11: Lifelong Learning Recognize the need for continuous learning, adaptability, and self-development to meet evolving personal, professional, and societal demands in a dynamic global environment.

SEMESTER-I

Semester	Part	Course Category	Title of the Course	Course Code	Lecture	Tutorial	Practical	Credits	Marks		
									CAT	ESE	Total
I	I	AEC	Tamil/Hindi/French-I	TA26C01/ HA26C01/ FA26C01	3	0	0	3	40	60	100
	II	AEC	English-I	EN26C01	3	0	0	3	40	60	100
	III	Core	Problem Solving Techniques using C	UCA26101	3	0	0	3	40	60	100
		Core	Problem Solving Techniques using C Lab	UCA26102	0	0	4	2	40	60	100
		Core	Digital Computer Fundamentals	UCA26103	4	0	0	4	40	60	100
		MDC	MDC -I	-	3	0	0	3	40	60	100
	IV	VAC	Human Values	VA2601	2	0	0	2	40	60	100
								20			700

SEMESTER-II

Semester	Part	Course Category	Title of the Course	Course Code	Lecture	Tutorial	Practical	Credits	Marks		
									CAT	ESE	Total
II	I	AEC	Tamil/Hindi/French-II	TA26C02/ HA26C02/ FA26C02	3	0	0	3	40	60	100
	II	AEC	English-II	EN26C02	3	0	0	3	40	60	100
	III	Core	Java Programming	UCA26204	3	0	0	3	40	60	100
		Core	Java Programming Lab	UCA26205	0	0	4	2	40	60	100
		Core	Fundamentals of Data Structures and Algorithms	UCA26206	4	0	0	4	40	60	100
		MDC	Multi Disciplinary Course -II	-	3	0	0	3	40	60	100
	IV	VAC	Environmental Studies & Sustainability	VA2602	2	0	0	2	40	60	100
								20			700

SEMESTER-III

Semester	Part	Course Category	Title of the Course	Course Code	Lecture	Tutorial	Practical	Credits	Marks		
									CAT	ESE	Total
III	I	AEC	Tamil/Hindi/French-III	TA26C03/ HA26C03/ FA26C03	3	0	0	3	40	60	100
	II	AEC	English-II	EN26C03	3	0	0	3	40	60	100
	III	Core	Relational Database Management Systems	UCA26307	3	0	0	3	40	60	100
		Core	Relational Database Management Systems Lab	UCA26308	0	0	4	2	40	60	100
		MDC	Multi Disciplinary Course -III	-	3	0	0	3	40	60	100
		DEC	Scripting Languages	UCA26001	3	1	0	4	40	60	100
			Open Source Technology	UCA26002							
			PHP	UCA26003							
	IV	VAC	IKS	VA2603	3	0	0	3	40	60	100
								21			700

SEMESTER-IV

Semester	Part	Course Category	Title of the Course	Course Code	Lecture	Tutorial	Practical	Credits	Marks		
									CAT	ESE	Total
IV	I	Language	Tamil/Hindi/French-IV	TA26C04/ H26C04/ F26C04	3	0	0	3	40	60	100
	II	Language	English-II	EN26C04	3	0	0	3	40	60	100
	III	Core	Fundamental of Python Programming	UCA26409	3	0	0	3	40	60	100
		Core	Python Lab	UCA264010	0	0	4	2	40	60	100
		MDC	Multi Disciplinary Course - IV	-	3	0	0	3	40	60	100
		DEC	E-Commerce and M-Commerce	UCA26004	3	1	0	4	40	60	100
			System Analysis and Design	UCA26005							
			Introduction to Web Design	UCA26006							
	IV	VAC	Ethics in AI	VA2604	3	0	0	3	40	60	100
	V	EAA	Extra Academic Activity(NSS-NCC-YRC.....)					1*			
								21			700

SEMESTER-V

Semester	Part	Course Category	Title of the Course	Course Code	Lecture	Tutorial	Practical	Credits	Marks		
									CAT	ESE	Total
V	III	Core	IOT	UCA26511	3	0	0	3	40	60	100
		Core	IOT Lab	UCA26512	0	0	4	2	40	60	100
		Core	Software Engineering	UCA26513	4	0	0	4	40	60	100
		DEC	Operating System with Lab	UCA26007	3	0	2	4	40	60	100
			Cyber Security and Ethical Hacking Lab	UCA26008							
			Internet Programming with HTML	UCA26009							
		SEC	Machine Learning-I	UCA26S01	3	1	0	4	40	60	100
			Digital Marketing - II	UCA26S02	3	1	0	4	40	60	100
		DEC	Information Security	UCA26010	3	0	2	4	40	60	100
			Client Server Technology	UCA26011							
			Generative AI Applications	UCA26012							
	V	EE	Experiential Elective (Internship)	EE26	*			2	*	*	100
								27			800

SEMESTER-VI

Semester	Part	Course Category	Title of the Course	Course Code	Lecture	Tutorial	Practical	Credits	Marks		
									CAT	ESE	Total
VI	III	Core	Computer Networks	UCA26614	4	0	0	4	40	60	100
		Core	Networks Lab	UCA26615	0	0	4	2	40	60	100
		Core	Project	UCA26616	0	0	12	6	40	60	100
		DEC	Fundamentals of React JS	UCA26013	3	0	2	4	40	60	100
			Front End Technologies	UCA26014							
			GIS and Remote Sensing	UCA26015							
		SEC	Software Testing Tool - I	UCA26S03	3	1	0	4	40	60	100
			Introduction to Artificial Intelligence and Machine Learning - II	UCA26S04	3	1	0	4	40	60	100
	IV	DEC	Multimedia and Applications	UCA26016	3	0	2	4	40	60	100
			System Analysis and Design	UCA26017							
			DevOps Fundamentals	UCA26018							
	V	VAC	Gender Studies	VA2605	2	0	0	2	40	60	100
								30			800

Multidisciplinary Course offered to other Departments:

UCA26901 - Introduction to Computer Applications
UCA26902 - Information Technology Essentials
UCA26903 - Digital Literacy and Office Automation
UCA26904 - Fundamentals of Python Programming
UCA26905 - Web Design and Development Basics
UCA26906 - Data Analytics Using Spreadsheets
UCA26907 - Cyber Security and Digital Safety
UCA26908 - Artificial Intelligence for Beginners
UCA26909 - Cloud Computing and Internet Technologies

Disciplinary Elective Course Offered by the Department:

UCA26001 - Scripting Languages
UCA26002 - Open Source Technology
UCA26003 - PHP
UCA26004 - E-Commerce and M-Commerce
UCA26005- System Analysis and Design
UCA26006 - Introduction to Web Design
UCA26007 - Operating System
UCA26008 - Cyber Security
UCA26009 - Internet Programming
UCA26010 - Information Security
UCA26011 - Client Server Technology
UCA26012 - Generative AI Applications
UCA26013 - Fundamentals of React JS
UCA26014 - Front End Technologies
UCA26015 - GIS and Remote Sensing
UCA26016 - Multimedia and Applications
UCA26017- System Analysis and Design
UCA26018 - DevOps Fundamentals

Semester-wise Categorization of Courses

COURSE CATEGORY	Semester						No of Courses	Total Credits
	I	II	III	IV	V	VI		
Languages	2	2	2	2	-	-	8	24
MSC-Core Courses (Theory & Practicals)	3	3	2	2	3	3	16	55
MDC–Multi Disciplinary Course	1	1	1	1	-	-	4	12
DEC–Disciplinary Elective Course	-	-	1	1	2	2	6	24
SEC–Skill Enhancement Course	-	-	-	-	2	2	4	16
VAC – Value Added Course	1	1	1	1	-	1	5	12
EE–Experiential Elective	-	-	-	-	1	-	1	2
EAA–Extra Academic Activity	-	-	-	1	-	-	1	-
Total Courses Per Semester	7	7	7	8	8	8	45(Courses)	
Total Credits	20	20	21	21	27	30	145(Credits)	

Sem	Course Code	Title of the Course	Course Category	L	T	P	Credits
I	UCA26101	Problem Solving Techniques using C	Core	3	0	0	3
Pre-Requisite:		Co- Requisite:		Progressive Course:			
<ul style="list-style-type: none"> Basic knowledge of computers and operating systems. Familiarity with mathematics (logic, sets, number systems). 		<ul style="list-style-type: none"> A practical lab course in C programming (to reinforce theory with hands-on coding). 		<ul style="list-style-type: none"> Data Structures using C Object-Oriented Programming (C++/Java) Advanced Programming Concepts 			
Course Objectives:							
1. Understand the fundamental concepts of computers, data representation techniques, and input/output devices.							
2. Learn computer memory organization, storage devices, algorithms, and flowcharting techniques for problem-solving.							
3. Acquire knowledge of the C programming language, its structure, syntax, data types, operators, and expressions.							
4. Develop programming skills using control structures, arrays, strings, and input/output operations in C.							
5. Design modular programs using functions, recursion, structures, unions, and pointers for solving computational problems.							
Course Outcomes:							
CO1: Explain computer fundamentals, data representation methods, and the operation of input/output devices. (K2)							
CO2: Apply algorithms and flowcharts to analyze and solve computational problems and describe memory organization and storage devices. (K3)							
CO3: Develop C programs using variables, data types, operators, expressions, and control statements. (K3)							
CO4: Implement solutions using arrays, strings, and input/output functions in C programming. (K4)							
CO5: Design modular and efficient programs using functions, recursion, structures, unions, and							

pointers. (K5)

Unit-I		12 Hrs.
Computer Basics: Algorithms- A Simple Model of a Computer- Characteristics of Computers- Problem Solving Using Computers. Data Representation: Representation of Characters in Computers- Representation of Integers- Representation of Fractions- Hexadecimal Representation of Numbers- Decimal to Binary Conversion- Error Detecting Codes. Input / Output Units: Description of Computer Input Units- Other Input Methods- Computer Output Units.		
Unit-II		12 Hrs.
Computer Memory: Memory Cell- Memory Organization- Read Only Memory- Serial Access Memory- Physical Devices Used to Construct memories- Magnetic Hard Disk- Floppy Disk Drives- Compact Disk Read Only Memory (CDROM)- Algorithms-Characteristics of Algorithms- Examples of Algorithms- Advantages and Disadvantages of Algorithms- Flow Charts – Symbols used in Flow Charts.		
Unit-III		12 Hrs.
Introduction to C : -History of C- Structure of a C program- Simple C program- Executing a C Program- Character set in C- C Tokens- Keywords and Identifiers in C- Constants- Variables in C-- Basic Data Types- Type declaration- Operators and Expressions.		
Unit-IV		12 Hrs.
Arrays and String Handling :- Managing Input and output operation- Control statements- Introduction to array- advantages of arrays- array declaration- array initialization- Types of arrays - Single and Multidimensional arrays- Character Arrays - Strings.		
Unit-V		12 Hrs.
Functions:- Introduction to functions - Advantages of functions- Declaring a function- calling a function- Passing arguments for a function- Categories of functions - Nesting of functions-Recursion - Structures and Union- Introduction to Pointers- Pointers and Arrays- Function Returning Pointers- Pointers to functions.		

Text Book(s):

1. Jena- Sisir Kumar and . Jena- Sisir Kumar- “C Programming : Learn to Code”- CRC Press-2021.
2. E. Balagurusamy- “Programming in ANSI C”- Tata McGraw Hill- New Delhi Seventh Edition- 2016.
3. Szuhay- Jeff and Szuhay- Jeff- “Learn C Programming : A Beginner’s Guide to learning C Programming the Easy and Disciplined way”- Packt Publishing- 2020.
4. Yashavant P. Kanetkar, Let Us C, 18th Edition, BPB Publications, 2023.
5. V. Rajaraman, Fundamentals of Computers, 6th Edition, PHI Learning Pvt. Ltd., 2014.

Reference Book(s):

1. YashwantP.Kanetkar (2010)-Let us C- BPB Publications- Sixth Edition.
2. ReemaThareja(2014)-Fundamentals of Computers- Oxford University Press.
3. Pradeep K. Sinha and Priti Sinha, Computer Fundamentals, 8th Edition, BPB Publications, 2017.
4. Herbert Schildt, C: The Complete Reference, 4th Edition, McGraw Hill Education, 2017.
5. Paul Deitel and Harvey Deitel, C How to Program, 8th Edition, Pearson Education, 2016.

e Resources:

1. [GeeksforGeeks C Programming](#)
2. [TutorialsPoint C Programming](#).
3. [Programiz C Programming](#)
4. [W3Schools C Tutorial](#).

Relationship Matrix for COs- POs and PSOs

Semester	Course code		Title of the course									Hours		Credit
I	UCA26101		ProblemSolving Techniques using C									3		3
Course Outcomes (COs)	Programme Outcomes(POs)											Programm e Specific Outcomes (PSOs)		Mean Score of COs
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	
CO1	3	3	2	2	3	-	2	-	2	-	3	3	2	2.50
CO2	3	3	2	2	3	-	-	2	2	-	3	3	2	2.50
CO3	3	3	2	3	3	-	-	2	2	2	3	3	3	2.69
CO4	3	3	2	3	3	-	-	2	2	2	3	3	3	2.69
CO5	3	3	3	3	3	-	2	2	2	3	3	3	3	2.77
Mean Overall Score	3	3	2.2	2.6	3	0	0.8	1.6	2	1.4	3	3	2.6	2.63

Sem	Course Code	Title of the Course	Course Category	L	T	P	Credits
I	UCA26102	Problem Solving Techniques using C Lab	Core	0	0	4	2
Pre-Requisite: Not Required			Co- Requisite:	Progressive Course:			
Course Objectives:							
1. Develop problem-solving skills through the design and implementation of C programs using fundamental programming constructs.							
2. Apply decision-making, looping, and function concepts to solve mathematical and logical problems.							
3. Implement programs using arrays, strings, matrices, and user-defined functions for data processing.							
4. Develop efficient solutions using recursion, pointers, and modular programming techniques.							
5. Utilize file handling operations to store, retrieve, and manage data effectively in C programming							
Course Outcomes:							
CO1: Develop C programs using basic programming constructs, operators, decision-making statements, and looping structures to solve computational problems.(K3)							
CO2: Apply functions and recursion to solve mathematical problems such as factorial, Fibonacci series, GCD, prime number testing, and nPr computations.(K3)							
CO3: Implement programs involving strings and arrays to perform operations such as palindrome checking, character analysis, and data manipulation.(K4)							
CO4: Design and implement matrix operations including addition, subtraction, and transpose using multidimensional arrays.(K5)							
CO5: Develop file processing applications to perform file creation, reading, writing, and storage of structured data records.(K5)							

LIST OF EXPERIMENTS

1. Write a program in C to find whether the given string is Palindrome or not.
2. Write a program in C to count vowels- consonants etc.
3. Write a program in C to find the factorial of a number.
4. Write a program in C to find if the given number is prime or not.
5. Write a program in C to find the value of NPR
6. Write a program in C to find the GCD of two numbers.
7. Write a program in C to find the Fibonacci Series
8. Write a program in C to find Matrix Addition/Subtraction.
9. Write a program in C to find Matrix Transpose.
10. Write a program in C for swapping 2 numbers.
11. Write a program in C to open- read and close the file
12. Write a program in C to read name and marks of n number of students and Store them in a file.

Relationship Matrix for COs- POs and PSOs

Sem	Course Code		Title of the Course									L	T	P	Credit
I	UCA26102		ProblemSolving Techniques using C									4	0	0	2
Course Outcomes (COs)	Programme Outcomes(POs)											Programme Specific Outcomes (PSOs)		Mean Score of COs	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2		
CO1	3	3	2	2	3	-	-	2	2	2	3	3	2	2.58	
CO2	3	3	3	2	3	-	-	2	2	2	3	3	2	2.67	
CO3	3	3	2	3	3	-	-	2	2	2	3	3	3	2.69	
CO4	3	3	2	3	3	-	-	2	2	3	3	3	3	2.77	
CO5	3	3	3	3	3	-	2	2	2	3	3	3	3	2.85	
Mean Overall Score	3	3	2.4	2.6	3	0	0.4	2	2	2.4	3	3	2.6	2.71	

[illegible]

CO5: Analyze the organization and operation of RAM, ROM, programmable memories, flash memories, and secondary storage devices.(K4)		
Unit-I		12 Hrs.
Number System and Codes: Decimal Numbers - Binary Number - Decimal to Binary Conversions - Binary Arithmetic - 1's and 2's complements of Binary Numbers - Signed Numbers - Arithmetic Operations with Signed numbers - Hexadecimal Numbers - Octal Numbers - Digital Codes - Error Detection Codes.		
Unit-II		12 Hrs.
Logic Gates: The Inverter - The AND gate - The OR gate - The NAND gate - NOR gate - The Exclusive – OR gate and Exclusive OR gate: Boolean Algebra and Logic Simplification - Boolean Operations and Expressions - Laws and Rules - De Morgan's Theorems - Boolean Expressions and Truth Tables - The Karnaugh Map - SOP minimizations.		
Unit-III		12 Hrs.
Combinational Logic Analysis: Basic combinational Logic Circuits - Implementing Combinational Logic - The Universal Property of NAND and NOR Gates. Functions of Combinational Logic - Basic Adder - Parallel Binary Adders - Comparators - Decoders - Encoders - Code Converters - Multiplexers - Parity Generator/Checkers.		
Unit-IV		12 Hrs.
Latches and Flip-Flops: Latches - Edge Triggered Flip-Flops - Flip-Flop Operating characteristics - Flip-Flop Applications- Registers - Counters.		
Unit-V		12 Hrs.
Memory and Storage: Memory Basics - The RAM- The ROM - Programmable ROMs - The Flash Memory - Memory Expansion - Special Types of Memories - Magnetic and Optical Storage.		

Text Book(s):

1. Floyd, Thomas L,1997, University Book Stall, 10thEdition“Digital Computer Fundamentals”.
2. Malvino, Paul Albert and Leach, Donald P, 2000,TMH, 4thEdition, “Digital Principles and Applications”.
3. Malvino, Paul Albert and Leach, Donald P,1995,TMH, 3rd Edition, “Digital Computer Fundamentals”.

Reference Book(s):

1. Malvino, Paul Albert and Leach, Donald P,1995,TMH, 3rd Edition, “Digital Computer Fundamentals”.
2. Bartee, Thomas C,1995, TMH,6th Edition, “Digital Computer Fundamentals”.

e Resources:

1. [Electronics Tutorials – Digital Electronics](#)
2. GeeksforGeeks – Digital Logic Design
3. TutorialsPoint – Digital Electronics Tutorial
4. [CircuitVerse Digital Logic Simulator](#)
5. [Logicly Digital Logic Simulator](#)

Relationship Matrix for COs- POs and PSOs

Semester	Course code		Title of the course									L	T	P	Credit
I	UCA26103		Digital Computer Fundamentals									4	0	0	4
Course Outcomes (COs)	Programme Outcomes(POs)											Programme Specific Outcomes (PSOs)		Mean Score of COs	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2		
CO1	3	3	2	2	3	-	-	2	2	2	3	3	2	2.58	
CO2	3	3	3	2	3	-	-	2	2	2	3	3	2	2.67	
CO3	3	3	2	3	3	-	-	2	2	3	3	3	3	2.77	
CO4	3	3	2	3	3	-	-	2	2	2	3	3	3	2.69	
CO5	3	3	3	2	3	-	2	2	2	3	3	3	3	2.77	
Mean Overall Score	3	3	2.4	2.4	3	0	0.4	2	2	2.4	3	3	2.6	2.69	

[illegible]

CO3: Implement packages, interfaces, exception handling mechanisms, multithreading, and inter-thread communication in Java applications.(K3)		
CO4: Apply Java I/O streams, file handling techniques, applets, string manipulation, and utility classes for software development.(K4)		
CO5: Design and develop network-based and GUI applications using socket programming, URLs, datagrams, AWT controls, layouts, and menus.(K5)		
Unit-I		12 Hrs.
Introduction to Java : Introduction to Java – Features of Java – Object Oriented Concepts – Lexical issues – Data Types – Variables – Arrays – Operators – Control Statements.		
Unit-II		12 Hrs.
Classes, Objects and Methods : Classes – Objects – Constructors – Overloading methods – Access control – Static and fixed Methods – String Class – Inheritance – Overriding Methods – Using Super – Abstract Class.		
Unit-III		12 Hrs.
Packages : Packages – Access Protection – Importing Packages – Interfaces – Exception Handling – Thread – Synchronization – Messaging – Runnable Interface– Inter Thread Communication.		
Unit-IV		12 Hrs.
I/O Streams : I/O Streams – File Streams – Applets – String Buffer – Char Array – Java Utilities – Random, Vector, Calendar and Properties.		
Unit-V		12 Hrs.
Network Basics : Network Basics – Socket Programming – Proxy Server – URL – Datagrams – Working With Windows Using AWT Classes. AWT Controls – Layout Management and Menus.		

Text Book(s):

1. P. Naughton& H. Schildt, Java2-The Complete Reference, 5th Edition, Tata McGraw Hill, 2002.
2. Cay S. Horstmann, Core Java Volume I – Fundamentals, 12th Edition, Pearson Education, 2022.
3. Herbert Schildt, Java: The Complete Reference, 11th Edition, McGraw Hill Education, 2019.
4. E. Balagurusamy, Programming with Java: A Primer, 6th Edition, McGraw Hill Education, 2019.

Reference Book(s):

1. Cay S. Horstmann, Gray Cornell, Core Java 2 Volume 1 Fundamentals, Addison Wesley, 2003
2. K. Arnold and J. Gosling, The Java Programming Language, Second Edition
3. Y. Daniel Liang, Introduction to Java Programming and Data Structures, 12th Edition, Pearson Education, 2019.

e Resources:

1. https://www.w3schools.com/java/java_intro.asp
2. [https://en.wikipedia.org/wiki/Java_\(programming_language\)](https://en.wikipedia.org/wiki/Java_(programming_language))

Relationship Matrix for COs- POs and PSOs

Semester	Course code	Title of the course										L	T	P	Credit
II	UCA26204	Java programming										3	0	0	3
Course Outcomes (COs)	Programme Outcomes(POs)											Programm e Specific Outcomes (PSOs)		Mean Score of COs	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2		
CO1	3	3	2	2	3	-	-	2	2	2	3	3	2	2.58	
CO2	3	3	2	3	3	-	-	2	2	2	3	3	3	2.69	
CO3	3	3	3	3	3	-	-	2	2	3	3	3	3	2.85	
CO4	3	3	2	3	3	-	-	2	2	3	3	3	3	2.77	
CO5	3	3	3	3	3	-	2	2	2	3	3	3	3	2.92	
Mean Overall I Score	3	3	2.4	2.8	3	0	0.4	2	2	2.6	3	3	2.8	2.76	

Sem	Course Code	Title of the Course	Course Category	L	T	P	Credits
II	UCA26205	Java Programming Lab	Core	0	0	4	2
Pre-Requisite: Not Required			Co- Requisite:	Progressive Course:			
Course Objectives:							
1. To understand the basic structure and syntax of Java programming and develop simple Java programs.							
2. To learn object-oriented programming concepts such as classes, objects, constructors, methods, and inheritance.							
3. To apply Java programming constructs including operators, arrays, control statements, and method overloading.							
4. To develop programs using advanced Java concepts such as packages, exception handling, and multithreading.							
5. To design and implement graphical user interfaces and applets using Java graphics programming techniques.							
Course Outcomes:							
CO1: Develop simple Java programs using basic syntax, operators, arrays, and control structures.(K3)							
CO2:Apply object-oriented concepts (classes, objects, constructors, inheritance, method overloading/overriding).(K3)							
CO3: Implement packages, exception handling, and multithreading in Java applications.(K4)							
CO4: Design programs using applets, graphics, and Java utilities.(K5)							
CO5: Create GUI-based applications integrating event handling and graphics programming.(K5)							

LIST OF EXPERIMENTS

1. Program to define a structure of a basic JAVA program.
2. Program to define operators, arrays and control structures.
3. Program to define class and constructors. Demonstrate constructors.
4. Program to define class, methods and objects.
5. Program to demonstrate method overloading.
6. Program to define inheritance and show method overriding.
7. Program to demonstrate Packages.
8. Program to demonstrate Exception Handling.
9. Program to demonstrate Multithreading.
10. Program to demonstrate Applet structure
11. Program to demonstrate Graphics programming

Relationship Matrix for COs- POs and PSOs

Semester	Course code	Title of the course										Hours	Credit	
I	UCA26205	Java Programming Lab										4	2	
Course Outcomes (COs)	Programme Outcomes(POs)											Programme Specific Outcomes (PSOs)	Mean Score of COs	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1		PSO2
CO1	3	3	2	2	3	-	-	2	2	2	3	3	2	2.58
CO2	3	3	2	3	3	-	-	2	2	2	3	3	3	2.69
CO3	3	3	3	3	3	-	-	2	2	3	3	3	3	2.85
CO4	3	3	2	3	3	-	-	2	2	3	3	3	3	2.77
CO5	3	3	3	3	3	-	2	2	2	3	3	3	3	2.92
Mean Overall Score	3	3	2.4	2.8	3	0	0.4	2	2	2.6	3	3	2.8	2.76

Sem	Course Code	Title of the Course	Course Category	L	T	P	Credits
II	UCA26206	Fundamentals of Data Structures and Algorithm	Core	4	0	0	4
Pre-Requisite: <ul style="list-style-type: none"> Problem Solving Techniques using C (Theory + Lab), Digital Computer Fundamentals. 			Co- Requisite: <ul style="list-style-type: none"> Data Structures Lab (if offered separately). 	Progressive Course: <ul style="list-style-type: none"> Design and Analysis of Algorithms, Advanced Data Structures, Database Management Systems. 			
Course Objectives:							
1. Understand algorithm analysis, complexity, and asymptotic notations.							
2. Implement linear data structures (arrays, linked lists) and their applications.							
3. Apply restricted linear structures (stacks, queues) in problem-solving.							
4. Design and analyze non-linear structures (trees, graphs, hashing).							
5. Apply sorting, searching, and algorithmic paradigms (divide & conquer, greedy, dynamic programming).							
Course Outcomes:							
CO1: Analyze algorithms using time and space complexity, and apply asymptotic notations.(K4)							
CO2: Implement linear data structures (arrays, linked lists) and solve related problems.(K3)							
CO3: Apply stacks and queues to expression evaluation, recursion management, and scheduling problems.(K3)							
CO4: Design and implement non-linear structures (trees, graphs, hashing) for efficient data organization.(K5)							
CO5: Apply searching, sorting, and algorithmic paradigms to solve computational problems.(K4)							
Unit-I							12Hrs.
Introduction to Algorithms and Linear Data Structures (Arrays & Lists) : Algorithm analysis, Time and Space Complexity, Asymptotic Notations ,Abstract Data Types (ADTs).							

Linear Structures: Arrays, Sequential storage implementation. Single Linked Lists, Doubly Linked Lists, Circular Linked Lists—Operations (Insertion, Deletion, Traversal). Applications of linked lists (Polynomial addition).

Unit-II

12Hrs.

Restricted Linear Data Structures (Stacks & Queues): ADT Stack, Array and Linked List implementation. Operations: Push, Pop, Peek. Applications: Infix to Postfix conversion, Evaluation of postfix expressions, Recursion management. **Queues:** ADT Queue, Array and Linked List implementation. Operations: Enqueue, Dequeue. Circular Queues, Deque (Double Ended Queue), Priority Queues. Applications of queues.

Unit-III

12Hrs..

Non-Linear Data Structures (Trees): **Trees:** Basic terminology, Binary Trees, Binary Tree Representations (Array and Linked list). Tree Traversals (Inorder, Preorder, Postfix). **Advanced Trees:** Binary Search Trees (BST) Insertion, Deletion, and Searching operations. Expression Trees. Balanced Trees: AVL Trees (Rotations and Balancing), Introduction to B-Trees and B+ Trees.

Unit-IV

12Hrs.

Non-Linear Data Structures (Graphs) and Hashing: **Graphs:** Terminology, Representation of Graphs (Adjacency Matrix, Adjacency List). Graph Traversals: Breadth First Search (BFS) and Depth First Search (DFS). **Minimum Spanning Trees:** Prim's and Kruskal's algorithms. **Shortest Path:** Dijkstra's Algorithm. Topological Sort. **Hashing:** Hash Functions, Collision Resolution Techniques: Separate Chaining, Open Addressing (Linear Probing, Quadratic Probing, Double Hashing).

Unit-V

12Hrs.

Sorting, Searching, and Algorithmic Paradigms: **Searching & Sorting:** Linear Search, Binary Search. Insertion Sort, Bubble Sort, Quick Sort, Merge Sort, Heap Sort. **Algorithmic Paradigms:** Divide and Conquer (Merge Sort), Greedy Method (Prim's and Kruskal's algorithms for MST), Dynamic Programming (0/1 Knapsack problem).

Text Book(s):

- Ellis Horowitz, Sartaj Sahni, and Sanguthevar Rajasekaran – Fundamentals of Computer Algorithms, Universities Press.
- Mark Allen Weiss – Data Structures and Algorithm Analysis in C, Pearson Education.
- Robert Lafore – Data Structures and Algorithms in Java, Pearson Education.

Reference Book(s):

- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein – Introduction to Algorithms (CLRS), MIT Press.
- Narasimha Karumanchi – Data Structures and Algorithms Made Easy, CareerMonk Publications.
- Jean-Paul Tremblay and Paul G. Sorenson – An Introduction to Data Structures with Applications, McGraw Hill.

e Resources:

- GeeksforGeeks – Data Structures
- TutorialsPoint – Data Structures and Algorithms
- Coursera – Data Structures and Algorithms Specialization
- MIT OpenCourseWare – Algorithms
- HackerRank – Data Structures Practice

Relationship Matrix for COs- POs and PSOs

Semester	Course code	Title of the course										L	T	P	Credit
II	2UCA26206	Fundamentals of Data Structures and Algorithms										4	0	0	4
Course Outcomes (COs)	Programme Outcomes(POs)											Programme Specific Outcomes (PSOs)		Mean Score of COs	
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2		
CO1	3	3	3	2	3	-	-	2	2	2	3	3	2	2.67	
CO2	3	3	2	3	3	-	-	2	2	2	3	3	3	2.69	
CO3	3	3	3	3	3	-	-	2	2	3	3	3	3	2.85	
CO4	3	3	3	3	3	-	2	2	2	3	3	3	3	2.92	
CO5	3	3	3	3	3	-	2	2	2	3	3	3	3	2.92	
Mean Overall Score	3	3	2.8	2.8	3	0	0.8	2	2	2.6	3	3	2.8	2.81	