

Sem. - IV

Vertical – 1 Major

Title of Paper :Core Java

Sr.No.	Heading	Particulars
1	Description the course : Including but Not limited to:	Core Java course focuses on teaching students how to design, develop, and maintain software applications using the Java programming language. The course covers fundamental to advanced concepts of Java, enabling students to understand object-oriented programming (OOP) principles, data structures, algorithms, and real-world application development.
2	Vertical :	Major
3	Type :	Theory
4	Credits :	2 credits (1 credit = 15 Hours for Theory in a semester, Total 30 hours)
5	Hours Allotted :	30 Hr
6	Marks Allotted:	50
7	Course Objectives(CO): CO 1: Understand and Apply Object-Oriented Programming (OOP) Concepts. CO 2: Identify the key components of a class and object in Java, including attributes (fields), methods, and constructors. CO 3: Apply sound software engineering principles in Java by organizing code into classes and methods with proper access control identifiers CO 4: Use tools and techniques like unit testing, as well as IDE debugging tools to find and fix issues within Java programs. CO 5. Effectively use Java's collection framework (e.g., Lists, Sets, Maps) to manage and process groups of related objects. CO 6. Use OOP concepts in designing and building solutions to real-world problems, ensuring the application is modular, maintainable, and reusable.	
8	Course Outcomes (OC): OC1. Understand the basics of Java and its runtime environment. OC2. Be proficient in using Java's data types, control flow statements, and OOP principles such as classes, inheritance, and exception handling. OC3. Creating own classes and objects OC4. Develop mini projects using Class, Interface and exception handling	
9	Modules:- Module 1: Introduction to Java Programming -History of Java and its Evolution,Features of Java (Platform Independence, Object-Oriented),Data Types and Variables,Operators Constants and Literals,Type Casting Decision Making and Loops :If-else Statements,Switch Statement, Loops (For, While, Do-While),Break and Continue Statements Classes and Objects :Array,ArraysString class and String methods, StringBuffer and StringBuilder, Object-Oriented Programming Concepts, Defining Classes and Creating Objects, Instance Variables and Methods, Constructors, this Keyword, super keyword, Types of Classes, Scope Rules, Access Modifier, constants, static members of a class, garbage collection.	
		15 Hrs

	Inheritance: Its types, Superclass and Subclass, Final classes and methods Polymorphism: Compile-time and Runtime Polymorphism	
	Module 2:	
	Interfaces: Defining and Implementing Interfaces, Abstract Classes and Methods, Multiple Interface Implementation Packages: Introduction to predefined packages, User Defined Packages, Access specifier, Java Built-in packages Exception handling- Try, Catch, and Finally Blocks, Throw and Throws Keywords Introduction to Threads: Creating and Running Threads, Thread Lifecycle	15 Hrs
10	Books and References: <ol style="list-style-type: none"> 1. Java: The Complete Reference Herbert Schildt MC-Graw HILL 12th EDITION 2022 2. Core Java, Volume I: Fundamentals Hortsman Pearson 9th 2013 3. Core Java, Volume II: Advanced Features Gary Cornell and Hortsman Pearson 8th 2008 	
12	Internal Continuous Assessment: 40%	Semester End Examination: 60%
13	Continuous Evaluation through: Class test of 1 of 15 marks Class test of 2 of 15 marks Average of the two: 15 marks Quizzes/ Presentations/ Assignments: 5 marks Total: 20 marks	Format of Question Paper: External Examination (30 Marks)– 1 hr duration
14	Format of Question Paper: (Semester End Examination: 30 Marks. Duration:1 hour) Q1: Attempt any two (out of four) from Module 1 (15 marks) Q2: Attempt any two (out of four) from Module 2 (15 marks) Or Q1: Attempt any three (out of five) from Module 1 (15 marks) Q2: Attempt any three (out of five) from Module 2 (15 marks)	

Title of Paper: Software Engineering

Sr.No	Heading	Particulars
1	Description the course: Including but Not limited to:	This course provides an in-depth understanding of Scrum, an Agile framework for developing, delivering, and sustaining complex products. Students will learn the principles and practices of Scrum, including roles (Scrum Master, Product Owner, Development Team), events (Sprint, Scrum Meetings), and artifacts (Product Backlog, Sprint Backlog, Increment). The course emphasizes hands-on exercises, real-world scenarios, and collaborative activities to master iterative development and enhance team productivity. By the end, learners will be equipped to implement Scrum in software engineering projects effectively and drive organizational agility.
2	Vertical :	Major
3	Type :	Theory
4	Credits :	2 credits (1 credit = 15 Hours for Theory in a semester, Total 30 hours)
5	Hours Allotted :	30
6	Marks Allotted:	50
7	Course Objectives (CO): CO1: Understand the core principles of Agile and the Scrum framework. CO2: Explore the high-level Scrum process and its key components. CO3: Develop skills in managing the Product Backlog effectively. CO4: Learn techniques for Sprint planning, execution, and tracking. CO5: Gain insights into Scrum-based project, quality, and risk management. CO6: Master the art of writing clear and actionable user stories.	
8	Course Outcomes (OC): OC1: Demonstrate a comprehensive understanding of Agile concepts and Scrum practices. OC2: Apply Scrum processes to effectively manage software development life cycles. OC3: Create and prioritize user stories for efficient Product Backlog management. OC4: Utilize metrics to evaluate and enhance Sprint performance and team productivity. OC5: Implement strategies for cost, customer, and risk management in Scrum projects. OC6: Formulate effective Sprint retrospectives to drive continuous improvement.	
9	Module 1:	
	Software and Software Engineering, Process Models, Introduction to Agile Concepts, All about Scrum, Scrum Process: High-Level View. Product Backlog Management, Sprint Planning, Writing Effective User Stories, Sprint Execution and Tracking, Sprint Review, Sprint Retrospectives	15 Hrs
	Module 2:	
	Measurements and Metrics in Scrum, Software Development Life Cycle and Waterfall Model, Project Management in Scrum and Waterfall, Quality Management in Scrum, Customer Management in Scrum, Risk Management in Scrum, Cost Management in Scrum.	15 Hrs

10	Books and References: <ol style="list-style-type: none"> 1. "Agile Scrum", Rama Bedarkar, Wiley, 1st, 2020 2. "Mastering Professional Scrum: A Practitioner's Guide to Overcoming Challenges and Maximizing the Benefits of Agility" by Stephanie Ockerman and Simon Reindl, Addison-Wesley Professional, 1st edition (2019). 3. "Scrum: A Pocket Guide" by Gunther Verheyen, Van Haren Publishing, 2nd edition (2019). 4. "Software in 30 Days" by Ken Schwaber and Jeff Sutherland, Wiley, 1st edition (2012). 5. "Scrum Insights for Practitioners: The Scrum Guide Companion" by Hiren Doshi, PracticeAgile Solutions, 1st edition (2016). 6. "A Scrum Book: The Spirit of the Game" by Jeff Sutherland and James O. Coplien, Pragmatic Bookshelf, 1st edition (2019). 7. "The Scrum Fieldbook: A Master Class on Accelerating Performance, Getting Results, and Defining the Future" by J.J. Sutherland, Random House Business, 1st edition (2019). 	
12	Internal Continuous Assessment: 40%	Semester End Examination: 60%
13	Continuous Evaluation through: Class test of 1 of 15 marks Class test of 2 of 15 marks Average of the two: 15 marks Quizzes/ Presentations/ Assignments: 5 marks Total: 20 marks	Format of Question Paper: External Examination (30 Marks)– 1 hr duration
14	Format of Question Paper: (Semester End Examination: 30 Marks. Duration:1 hour) Q1: Attempt any two (out of four) from Module 1 (15 marks) Q2: Attempt any two (out of four) from Module 2 (15 marks) Or Q1: Attempt any three (out of five) from Module 1 (15 marks) Q2: Attempt any three (out of five) from Module 2 (15 marks)	

Title of Paper: Computer Networks

Sr.No.	Heading	Particulars
1	Description the course : Including but Not limited to:	A course on Computer Networks typically focuses on the fundamental principles, technologies, and protocols that enable communication and data exchange between devices in various network environments.
2	Vertical :	Major
3	Type :	Theory
4	Credits :	2 credits (1 credit = 15 Hours for Theory in a semester, Total 30 hours)
5	Hours Allotted :	30
6	Marks Allotted:	50
7	Course Objectives(CO): 1. To understand the basic concepts in OSI Model,distinguishing Factors in TCP/IP ,IP addressing Schemes 2. Understand How the communication happens across the network 3. Understanding of various Routing protocol and their implementation	
8	Course Outcomes (OC): CO 1.Understanding the Transport layer protocols and their utilities CO 2.Various application layer protocols and their implementation CO3:Mailing Services and web services implementation	
9	Modules:- Module 1: 1. Introduction: OSI Model, TCP/IP Protocol Suite, IPV 4 Addresses and Protocol and IPV6 Addresses and Protocol 2. Address Resolution Protocol (ARP), Internet Control 3. Message Protocol Version 4 (ICMPv4), Mobile IP, 4. Unicast Routing Protocols (RIP, OSPF and BGP)	
		15 Hrs
	Module 2: 8. User Datagram Protocol (UDP), Transmission Control Protocol (TCP) 9. Host Configuration: DHCP, Domain Name System (DNS) 10. Remote Login: TELNET and SSH, File Transfer: FTP and TFTP ; World Wide Web and HTTP, 11. Electronic Mail: SMTP, POP, IMAP and MIME	
		15 Hrs
10	Books and References: TCP/IP Protocol Suite, Behrouz A. Forouzan, 4th Edition, Tata McGrawHill (Chapter 2, 5, 7, 26, 27, Chapter 8, 9, 10, 11, Chapter 14,15, Chapter 16, 18, 19, Chapter 20, 21, 22, Chapter 23, 25)	
12	Internal Continuous Assessment: 40%	Semester End Examination: 60%
13	Continuous Evaluation through: Class test of 1 of 15 marks Class test of 2 of 15 marks Average of the two: 15 marks Quizzes/ Presentations/ Assignments: 5 marks Total: 20 marks	Format of Question Paper: External Examination (30 Marks)– 1 hr duration

14

Format of Question Paper: (Semester End Examination : 30 Marks. Duration:1 hour)

Q1: Attempt any two (out of four) from Module 1 (15 marks)

Q2: Attempt any two (out of four) from Module 2 (15 marks)

Or

Q1: Attempt any three (out of five) from Module 1 (15 marks)

Q2: Attempt any three (out of five) from Module 2 (15 marks)

Title of Paper: Major Practical 4

Sr.No.	Heading	Particulars
1	Description the course : Including but Not limited to:	<p>Computer Networking Practical course focuses on providing hands-on experience with various networking concepts and techniques. Students typically practice configuring, troubleshooting, and testing network protocols and hardware in real-world scenarios. The practical component of this course emphasizes network setup, monitoring, and management skills Wireless Networks and Mobile Communications</p> <p>A Java Programming Practical course typically provides hands-on experience in writing, debugging, and executing Java programs. The goal is to help students become proficient in Java programming and apply theoretical concepts to solve real-world problems. The practical sessions in this course usually focus on programming skills and the application of Java principles in various scenario</p>
2	Vertical :	Major
3	Type :	Practical
4	Credits :	2 credits (30 Hours of Practical work in a semester)
5	Hours Allotted :	30
6	Marks Allotted:	50 Marks
7	Course Objectives(CO): <ol style="list-style-type: none">1. Understand core Java programming concepts, including data types, control structures, and object-oriented programming principles.2. Develop the ability to implement inheritance, polymorphism, interfaces, and abstract classes effectively.3. Gain hands-on experience with exception handling, multithreading, and dynamic initialization.4. Learn to apply Java programming to solve real-world problems, such as matrix operations and finding areas/volumes.5. Enhance debugging and problem-solving skills using Java's rich standard libraries and features.6. Basic foundation of LAN7. various command line utilities to be tested8. Practical implementation of IP Subnetting9. Testing of various Routing Protocols	
8	Course Outcomes (OC): <p>CO 1. Write efficient Java programs to perform arithmetic operations, manage control flow, and manipulate strings.</p> <p>CO 2. Demonstrate knowledge of object-oriented concepts by implementing inheritance, polymorphism, and interfaces.</p> <p>CO 3. Apply exception handling mechanisms to create robust Java applications.</p> <p>CO 4. Implement multithreading and explore dynamic initialization for advanced Java programming.</p>	

	<p>CO 5. Solve computational problems, such as matrix operations and factorial calculation, using packages and Java constructs.</p> <p>CO 6. Implementation of utility protocols</p> <p>CO 7 Understanding Basic Security features</p> <p>CO 8 Network Traffic and Packet Analysis</p> <p>CO 9 Basic Understanding of Wireless Network</p>	
9	<p>Module 1</p> <ol style="list-style-type: none"> Write a program <ol style="list-style-type: none"> in Java to demonstrate Boolean value. Print a string 10 times using a for loop. Write a program in Java to evaluate $a+b*c\%d$. Write a program <ol style="list-style-type: none"> in Java to find the biggest element among three numbers using if else. Write a program in Java to find the biggest element among three numbers using the ternary operator. Write a program in Java to check the grade of marks using a switch case. Write a program <ol style="list-style-type: none"> in Java to demonstrate dynamic initialization. Write a program in Java to create a class and access all data members and methods using the object and compute the area and perimeter of a circle. Write a program in Java to access member variables using the constructor. Write a program <ol style="list-style-type: none"> in Java to multiply two matrices. Write a program in Java to calculate the area of a rectangle using single inheritance. Write a program in Java to demonstrate multilevel inheritance. Write a program <ol style="list-style-type: none"> in Java to demonstrate hierarchical inheritance. Write a program in Java to find the area and perimeter of a circle using an abstract class. Write a program in Java to show that a private member of a class cannot be inherited. Write a program <ol style="list-style-type: none"> in Java to find the volume of a box using this keyword. Write a program in Java to find the average of three numbers using the method overloading Write a program in Java to find average of three numbers using method overriding. Create a class figure. Create two subclasses rectangle and triangle. Find the area of a rectangle and half the area of the rectangle using the reference of the figure. Write a program <ol style="list-style-type: none"> Create an interface area. Find the area of a circle. Write a program in Java to find the sum and average of three numbers using the super keyword. Write a program 	30 Hrs

	<ul style="list-style-type: none"> a. in Java to find the volume of a box using constructor overloading. b. Write a program in Java to demonstrate exception handling in case of variable/constant divided by zero. <p>13. Write a program in Java</p> <ul style="list-style-type: none"> a. to implement multiple inheritance using the interface. b. Write a program in Java to check if a given string is palindrome or not. <p>14. Write a program in Java</p> <ul style="list-style-type: none"> a. for sorting a given list of strings in ascending order. b. Write a program in Java to find the factorial of a number using the package. <p>15. Write a program in</p> <ul style="list-style-type: none"> a. Java to import the package. b. Write a program in Java to implement thread. c. Write program to implement Flow, Grid and Border Layout using swing. d. Write program to demonstrate following events Action Mouse Key 	
	Module 2	30 Hrs
	<ol style="list-style-type: none"> 1. Configuring LAN setup <ul style="list-style-type: none"> a. Planning and Setting IP networks b. Configuring subnet c. Using, linux-terminal or Windows-cmd, execute following networking commands and note the output: ping, traceroute, netstat, arp, ipconfig, Getmac, hostname, NSLookUp, pathping, SystemInfo 2. IPv4 Addressing and Subnetting <ul style="list-style-type: none"> a. Given an IP address and network mask, determine other information about the IP address such as: <ul style="list-style-type: none"> a. Network address • Network broadcast address • Total b. number of host bits • Number of hosts b. Given an IP address and network mask, determine other information about the IP address such as: c. The subnet address of this subnet • d. The broadcast address of this subnet • e. The range of host addresses for this subnet • f. The maximum number of subnets for this subnet mask • g. The number of hosts for each subnet • h. The number of subnet bits •The number of this subnet 3. Configure Static IP routing using . 4. Configure IP routing using RIP. 5. Configuring Simple and multi-area OSPF 6. Configuring BGP protocol (Multi-Autonomous) 7. Configuring server and client. <ul style="list-style-type: none"> a. Configure DHCP b. Configure DNS c. Configure HTTP d. Configure Telnet e. Configure FTP 8. Configure basic security features for networks 9. Using Wireshark, network analyzer, set the filter for ICMP, TCP, HTTP, UDP, FTP and perform respective protocol transactions to show/prove that the 	

	network analyzer is working 10.create a wireless network of multiple PCs using appropriate access point. 11.IPV6 Addressing Basics	
10 & 11	Text Books& References Books : 1. Java: The Complete Reference Herbert Schildt MC-Graw HILL 12th EDITION 2022 2. Core Java, Volume I: Fundamentals Hortsman Pearson 9th 2013 3. Core Java, Volume II: Advanced FeaturesGary Cornell and Hortsman Pearson 8th 2008 4. Cisco CCNA 200-301 Official Cert Guide	
12	Internal Continuous Assessment: 40%	Semester End Examination: 60%
13	Continuous Evaluation through: Students are expected to attend each practical and submit the written practical of the previous session. Performing Practical and writeup submission will be continuous internal evaluation. 2.5 marks can be awarded for each practical performance and writeup submission totaling to 50 marks and can be converted to 20 marks.	30 marks practical exam of 2 hours duration
14	Format of Question Paper: Duration 2 hours. Certified copy of Journal is compulsory to appear for the practical examination Practical Slip: Q1. From Module 1 13 marks Q2. From Module 2 12marks Q3. Journal and Viva 05 marks	

Vertical - 4

VSC

Syllabus

B. Sc. (Information Technology)

(Sem.- III)

Title of Paper APPLIED MATHEMATICS

Sr.No.	Heading	Particulars
1	Description the course : Including but Not limited to:	This course is designed for developing competency of the students in the applications of various mathematical concepts. It is equipped with Complex numbers, Laplace transform, Inverse Laplace transform, Differential equations of first order with first degree and higher degree. This course introduces basic concepts of Algebra and prepares students to study further courses in linear and abstract algebra.
2	Vertical :	Vocational Skill Course
3	Type :	Theory
4	Credits :	2 credits (1 credit = 15 Hours for Theory in a semester, Total 30 hours)
5	Hours Allotted :	30 Hours
6	Marks Allotted:	50 Marks
7	Course Objectives(CO): CO1: Ability to interpret the mathematical results in physical or practical terms for complex numbers. CO2: Know and to understand various types of methods to solve Laplace transform. CO3: Apply the knowledge of Laplace Transforms to solve the problems. CO4: Know and to understand various types of methods to solve differential equation. CO5: Apply the knowledge of differential equations to solve the problems. CO6: Inculcate the habit of Mathematical Thinking through Indeterminate forms.	
8	Course Outcomes (OC): OC 1. Familiar with the various forms and operations of a complex number. OC 2: Find the Laplace transform of a function of using definition. OC 3: Find the Inverse Laplace transform of a function of using definition. OC 4: Solve Differential equations of first degree and first order. OC 5: Solve Differential equations of first degree and higher order.	
9	Modules:- Module 1:	<div> <div> 1.1 Complex Numbers: Complex number, Equality of complex numbers, Graphical representation of complex number (Argand's Diagram), Polar form of complex numbers. Polar form of $x+iy$ for different signs of x,y, Exponential form of complex numbers, Mathematical operation with complex numbers and their representation on Argand's Diagram, Circular functions of complex angles, Definition of hyperbolic function. Relations between circular and hyperbolic functions, Inverse hyperbolic functions. 1.2 The Laplace Transform: Introduction. Definition of the Laplace Transform, Table of Elementary Laplace Transforms. Theorems on Important Properties of </div> <div>15 Hrs</div> </div>

	Laplace Transformation, First Shifting Theorem, Second Shifting Theorem, Convolution Theorem, Laplace Transform of Derivatives. 1.3 Inverse Laplace Transform: Shifting Theorem, Partial fractions Methods, Use of Convolution Theorem, Solution of Ordinary Linear Differential Equations with Constant Coefficients, Laplace Transformation of Special Function, Periodic Functions, Heaviside Unit Step Function, Dirac-delta Function (Unit Impulse Function).	
	Module 2:	
	2.1 Equation of the first order and of the first degree: Separation of variables, Equations homogeneous in x and y, Non-homogeneous linear equations, Exact differential Equation, Integrating Factor, Linear Equation and equation reducible to this form, Method of substitution. 2.2 Differential equation of the first order of a degree higher than the first: Introduction, Solvable for p (or the method of factors), Solve for y, Solve for x, Clairaut's form of the equation, Method of Substitution. 2.3 Linear Differential Equations with Constant Coefficients: Introduction, The Differential Operator, Linear Differential Equation $f(D) y = 0$, Different cases depending on the nature of the root of the equation $f(D) = 0$, Linear differential equation $f(D) y = X$, The complimentary Function, The inverse operator $1/f(D)$ and the symbolic expression for the particular integral,	15 Hrs
10	Books and References: 1. A text book of Applied Mathematics Vol I, P. N. Wartikar and J. N. Wartikar, Pune Vidyathi Griha, 7 th , 1995 2. A text book of Applied Mathematics Vol II, P. N. Wartikar and J. N. Wartikar, Pune Vidyathi Griha, 7 th , 1995 3. Higher Engineering Mathematics, Dr. B. S. Grewal, Khanna Publications.	
12	Internal Continuous Assessment: 40%	Semester End Examination: 60%
13	Continuous Evaluation through: Class test of 1 of 15 marks Class test of 2 of 15 marks Average of the two: 15 marks Quizzes/ Presentations/ Assignments: 5 marks Total: 20 marks	Format of Question Paper: External Examination (30 Marks)– 1 hr duration
14	Format of Question Paper: (Semester End Examination : 30 Marks. Duration:1 hour) Q1: Attempt any two (out of four) from Module 1 (15 marks) Q2: Attempt any two (out of four) from Module 2 (15 marks) Or Q1: Attempt any three (out of five) from Module 1 (15 marks) Q2: Attempt any three (out of five) from Module 2 (15 marks)	

SEC

Syllabus

B. Sc. (Information Technology)

(Sem.- IV)

Title of Paper: Computer Graphics

Sr.No	Heading	Particulars
1	Description the course : Including but Not limited to:	Computer Graphics Practical is a hands-on course designed to introduce students to the foundational principles of computer graphics, including 2D transformations, graphical modelling, and basic animations. The course focuses on applying mathematical concepts like translation, rotation, scaling, and shearing to create and manipulate graphical objects. Students will learn to use programming tools such as Python (Matplotlib, Pygame, OpenCV) or C/C++ libraries to implement these concepts. Through practical assignments, they will develop the skills to create simple 2D animations, simulate real-world objects, and design graphical scenes. The course bridges the gap between theoretical concepts and real-world applications, fostering creativity and problem-solving in visual computing.
2	Vertical :	Skill Enhancement Course
3	Type :	Practical
4	Credits :	2 credits (30 Hours of Practical work in a semester)
5	Hours Allotted :	30
6	Marks Allotted:	50 Marks
7	Course Objectives(CO):	<p>CO1: Introduce foundational concepts of 2D transformations, geometric modelling, and rendering techniques in computer graphics.</p> <p>CO2: Develop skills to apply 2D transformations (translation, rotation, scaling, shearing, reflection) and basic animations.</p> <p>CO3: Enable students to simulate real-world objects and create simple animations.</p> <p>CO4: Equip students to understand graphics pipelines, coordinate systems, and basic rendering principles.</p> <p>CO5: Foster creativity and logical thinking by implementing graphical scenes and animations.</p>
8	Course Outcomes (OC):	<p>OC1: Understand and apply 2D transformation matrices to graphical objects.</p> <p>OC2: Implement simple graphics primitives and manipulate them using transformations.</p> <p>OC3: Create basic 2D animations (e.g., bouncing ball, rotating shapes).</p> <p>OC4: Understand and utilize color models, coordinate systems, and graphical libraries.</p> <p>OC5: Develop basic graphical applications using lightweight tools and programming languages.</p>

9	Module 1	30 Hrs
	<p>Module 1: Basic Setup and 2D Graphics Fundamentals</p> <ol style="list-style-type: none"> 1. Installing Required Software <ul style="list-style-type: none"> • Objective: Install and configure Python, Matplotlib, OpenCV, or Pygame for computer graphics. • Task: Verify the installation and create a "Hello, Graphics!" window. 2. Drawing Basic Shapes <ul style="list-style-type: none"> • Objective: Draw lines, circles, rectangles, and polygons using graphical primitives. • Tool: Python with Matplotlib or OpenCV. 3. Line Drawing Algorithms <ul style="list-style-type: none"> • Objective: Implement the DDA (Digital Differential Analyzer) algorithm. • Tool: Python or C++. 4. Bresenham's Line Drawing Algorithm <ul style="list-style-type: none"> • Objective: Implement Bresenham's line drawing algorithm. • Tool: Python or C++. 5. Circle Drawing Algorithms <ul style="list-style-type: none"> • Objective: Implement the Midpoint Circle algorithm. • Tool: Python or C++. 6. Polygon Filling <ul style="list-style-type: none"> • Objective: Implement the boundary-fill and flood-fill algorithms. • Tool: Python or C++. 7. Translation Transformation <ul style="list-style-type: none"> • Objective: Shift a 2D object using translation matrices. • Tool: Python with Matplotlib. 8. Rotation Transformation <ul style="list-style-type: none"> • Objective: Rotate a 2D object about a fixed point or origin. • Tool: Python with Matplotlib. 9. Scaling Transformation <ul style="list-style-type: none"> • Objective: Scale a 2D object up or down using scaling matrices. • Tool: Python with Matplotlib. 10. Reflection Transformation <ul style="list-style-type: none"> • Objective: Reflect a 2D object across x-axis, y-axis, and diagonal. • Tool: Python with Matplotlib. 11. Shearing Transformation <ul style="list-style-type: none"> • Objective: Apply x-axis and y-axis shearing to a 2D object. • Tool: Python with Matplotlib. 12. Composite Transformations <ul style="list-style-type: none"> • Objective: Combine translation, rotation, and scaling on a 2D object. • Tool: Python with Matplotlib. 13. Clipping Algorithms <ul style="list-style-type: none"> • Objective: Implement the Cohen-Sutherland line clipping algorithm. 	

	<ul style="list-style-type: none"> • Tool: Python or C++. <p>14. Window-to-Viewport Transformation</p> <ul style="list-style-type: none"> • Objective: Map a 2D object from a window to a viewport. • Tool: Python or C++. <p>15. Basic Interactive Graphics</p> <ul style="list-style-type: none"> • Objective: Create a simple interactive graphics program (e.g., moving a rectangle with arrow keys). • Tool: Python with Pygame. 	
	Module 2	30 Hrs
	<p>1. Simple Animation</p> <ul style="list-style-type: none"> • Objective: Animate a moving ball across the screen. • Tool: Python with Pygame. <p>2. Bouncing Ball Animation</p> <ul style="list-style-type: none"> • Objective: Create a bouncing ball with collision detection. • Tool: Python with Pygame. <p>3. Rotating Object Animation</p> <ul style="list-style-type: none"> • Objective: Animate a rotating triangle or square. • Tool: Python with Pygame or Matplotlib. <p>4. Scaling Animation</p> <ul style="list-style-type: none"> • Objective: Create an animation showing pulsating objects (grow/shrink). • Tool: Python with Matplotlib. <p>5. Multiple Object Animation</p> <ul style="list-style-type: none"> • Objective: Animate multiple objects moving independently. • Tool: Python with Pygame. <p>6. Color Models</p> <ul style="list-style-type: none"> • Objective: Experiment with RGB and HSI color models. • Tool: Python with OpenCV. <p>7. Bezier Curves</p> <ul style="list-style-type: none"> • Objective: Draw and animate a Bezier curve. • Tool: Python with Matplotlib. <p>8. 2D Game Development Basics</p> <ul style="list-style-type: none"> • Objective: Create a simple 2D game (e.g., a ball avoiding obstacles). • Tool: Python with Pygame. <p>9. Scene Creation</p> <ul style="list-style-type: none"> • Objective: Design a basic 2D scene (e.g., a house, tree, and sun). • Tool: Python with Matplotlib. <p>10. Parallax Scrolling Animation</p> <ul style="list-style-type: none"> • Objective: Implement parallax scrolling for a background in 2D graphics. • Tool: Python with Pygame. <p>11. Path Animation</p> <ul style="list-style-type: none"> • Objective: Animate an object moving along a predefined path. 	

	<ul style="list-style-type: none"> • Tool: Python with Matplotlib. <p>12. Collision Detection</p> <ul style="list-style-type: none"> • Objective: Implement collision detection between 2D objects. • Tool: Python with Pygame. <p>13. Interactive Graphics with Mouse Input</p> <ul style="list-style-type: none"> • Objective: Create an interactive program where shapes follow mouse clicks. • Tool: Python with Pygame. <p>14. Text Rendering</p> <ul style="list-style-type: none"> • Objective: Render and animate text in a 2D graphical environment. • Tool: Python with Pygame. <p>15. Final Project</p> <ul style="list-style-type: none"> • Objective: Combine multiple concepts to create a complete animated 2D scene. • Example: A car moving on a road with a rising sun and trees. <p>Tool: Python with Matplotlib or Pygame</p>	
10 & 11	Reference and Text Books: <ol style="list-style-type: none"> 1. Python Graphics: A Reference for Creating 2D and 3D Images, Bernard Korites, Apress, 2nd Edition 2023. 2. Computer Graphics from Scratch: A programmer's Introduction to 3D Rendering, Gabriel Gambetta, no starch press, 2021 3. 2D Computer Graphics: Modern C++ and Standard Library, Hakan Blomqvist, 2023 	
12	Internal Continuous Assessment: 40%	Semester End Examination: 60%
13	Continuous Evaluation through: Students are expected to attend each practical and submit the written practical of the previous session. Performing Practical and writeup submission will be continuous internal evaluation. 2.5 marks can be awarded for each practical performance and writeup submission totaling to 50 marks and can be converted to 20 marks.	30 marks practical exam of 2 hours duration
14	Format of Question Paper: Duration 2 hours. Certified copy of Journal is compulsory to appear for the practical examination Practical Slip: Q1. From Module 1 13 marks Q2. From Module 2 12marks Q3. Journal and Viva 05 marks	

Syllabus

B. Sc. (Information Technology)

(Sem.- IV)

Title of Paper Mobile Programming

Sr.No	Heading	Particulars
1	Description the course : Including but Not limited to:	This course introduces the fundamentals of Flutter and Dart for building cross-platform mobile applications. Students will learn to create responsive user interfaces, manage app state, handle user inputs, and implement navigation and animations. The course also covers integrating APIs, working with databases, and deploying functional mobile apps for Android and iOS.
2	Vertical :	Skill Enhancement Course
3	Type :	Practical
4	Credits :	2 credits (30 Hours of Practical work in a semester)
5	Hours Allotted :	30
6	Marks Allotted:	50 Marks
7	Course Objectives(CO): CO1: Understand the fundamentals of Flutter and Dart programming for mobile app development. CO2: Learn how to set up the Flutter SDK and development environment. CO3: Develop skills to create basic Flutter applications using widgets like Text, Row, and Column. CO4: Explore the use of StatelessWidget and StatefulWidget for managing app states. CO5: Master the implementation of responsive UIs using MediaQuery and layouts. CO6: Gain knowledge of form creation, input handling, and validation in Flutter apps. CO7: Learn to navigate between screens and implement app navigation features like drawers. CO8: Understand how to use Flutter animations, including AnimatedContainer and FadeTransition. CO9: Explore database integration with APIs using packages like http and FutureBuilder. CO10: Build apps with themes, user interactions (e.g., taps and long presses), and custom styling.	

8	Course Outcomes (OC): OC1: Demonstrate the ability to configure Flutter and build a functional development environment. OC2: Create and run basic Flutter apps with appropriate UI components. OC3: Develop responsive and adaptive UIs for multiple screen sizes. OC4: Implement interactive features like counters, sliders, and switches in Flutter apps. OC5: Design and validate user input forms using TextFormField. OC6: Develop navigation flows between screens and integrate drawers for better usability. OC7: Create animations for smooth transitions and enhanced user experiences. OC8: Build applications that fetch and display data from public APIs asynchronously. OC9: Apply effective state management strategies to handle app states efficiently. OC10: Demonstrate the ability to debug, test, and optimize Flutter apps for deployment.	
9	Module 1	30 Hrs
	<ol style="list-style-type: none">1. Install Flutter SDK on your computer and run the flutter doctor command to check your setup.2. Create a "Hello, World!" Flutter application and run it on an emulator.3. Modify the app's title and primary color in the MaterialApp widget.4. Create a StatelessWidget that displays a greeting message.5. Write a Dart program to calculate the sum of two numbers entered by the user.6. Implement a Dart program that uses if-else statements to determine if a number is odd or even.7. Demonstrate the use of a switch-case statement in Dart.8. Write a program to print a multiplication table using a for loop.9. Create a Flutter app with a Text widget that displays your name.10. Build an app with a Column widget to arrange multiple Text widgets vertically.11. Use a Row widget to arrange three buttons horizontally.12. Create a Flutter app using Scaffold with an AppBar, Body, and a FloatingActionButton.13. Create a simple counter app using StatefulWidget to increment and display a number.14. Implement a TextField widget to accept user input and display it using a Text widget.15. Design a Flutter app with a Container widget and customize its padding, margin, and color. <p>Use the Stack widget to overlay a Text widget on an Image.</p>	
	Module 2	30 Hrs
	<ol style="list-style-type: none">1. Build a responsive UI using MediaQuery to adapt to different screen sizes.2. Create a Flutter form with TextFormField widgets to accept a username and password.3. Implement form validation to ensure the fields are not empty.4. Add navigation between two screens in Flutter using the Navigator class.	

	5. Create a Drawer widget for app navigation with three menu options. 6. Display a list of items in a ListView widget. 7. Use the GridView widget to display a grid of images. 8. Add a GestureDetector to detect taps and display a message in the console. 9. Implement a LongPress event to change the color of a container. 10. Create a basic animation using the AnimatedContainer widget. 11. Implement a FadeTransition to animate the opacity of a widget. 12. Use a Slider widget to select a value between 0 and 100 and display the value. 13. Create a Switch widget to toggle between two themes (light and dark). 14. Use the http package to fetch and display data from a public API. Create a FutureBuilder widget to display data asynchronously.	
10 & 11	Reference and Text Books: <ol style="list-style-type: none"> 1. Mastering Flutter: A Beginner's Guide, by Sufyan bin Uzayr, CRC Press, 1st, 2023 2. Flutter for Beginners, by Alessandro Biessek, Packt Publishing, 1st edition (2019). 3. Flutter Cookbook, by Simone Alessandria, Packt Publishing, 2nd Edition, 2023 4. Beginning App Development with Flutter, by Rap Payne, Apress, 1st edition (2019). 5. Flutter Apprentice, by Michael Katz, Kevin David Moore, and Vincent Ngo, Kodeco, 1st edition (2021). 6. Flutter Complete Reference 2.0, by Alberto Miola, Independently published, 2nd edition (2023). 7. Flutter in Action, by Eric Windmill, Manning Publications, 1st edition (2020). 8. Programming Flutter, by Carmine Zaccagnino, O'Reilly Media, 1st edition (2020). 	
12	Internal Continuous Assessment: 40%	Semester End Examination: 60%
13	Continuous Evaluation through: Students are expected to attend each practical and submit the written practical of the previous session. Performing Practical and writeup submission will be continuous internal evaluation. 2.5 marks can be awarded for each practical performance and writeup submission totaling to 50 marks and can be converted to 20 marks.	30 marks practical exam of 2 hours duration
14	Format of Question Paper: Duration 2 hours. Certified copy of Journal is compulsory to appear for the practical examination Practical Slip: Q1. From Module 1 13 marks	

	Q2. From Module 2 12marks
	Q3. Journal and Viva 05 marks

QUESTION PAPER PATTERN (External and Internal)

Internal Continuous Assessment: 40%	Semester End Examination: 60%
Continuous Evaluation through: Class test of 1 of 15 marks Class test of 2 of 15 marks Average of the two: 15 marks Quizzes/ Presentations/ Assignments: 5 marks Total: 20 marks	Format of Question Paper: External Examination (30 Marks)– 1 hr duration
Format of Question Paper: (Semester End Examination: 30 Marks. Duration:1 hour) Q1: Attempt any two (out of four) from Module 1 (15 marks) Q2: Attempt any two (out of four) from Module 2 (15 marks) Or Q1: Attempt any three (out of five) from Module 1 (15 marks) Q2: Attempt any three (out of five) from Module 2 (15 marks)	

Practical Examination

Internal Continuous Assessment: 40%	Semester End Examination: 60%
Continuous Evaluation through: Students are expected to attend each practical and submit the written practical of the previous session. Performing Practical and writeup submission will be continuous internal evaluation. 2.5 marks can be awarded for each practical performance and writeup submission totaling to 50 marks and can be converted to 20 marks.	30 marks practical exam of 2 hours duration
Format of Question Paper: Duration 2 hours. Certified copy of Journal is compulsory to appear for the practical examination Practical Slip: Q1. From Module 1 13 marks Q2. From Module 2 12marks Q3. Journal and Viva 05 marks	

Letter Grades and Grade Points:

Semester GPA/ Programme CGPA Semester/ Programme	% of Marks	Alpha-Sign/ Letter Grade Result	Grading Point
9.00 - 10.00	90.0 - 100	O (Outstanding)	10
8.00 - < 9.00	80.0 - < 90.0	A+ (Excellent)	9
7.00 - < 8.00	70.0 - < 80.0	A (Very Good)	8
6.00 - < 7.00	60.0 - < 70.0	B+ (Good)	7
5.50 - < 6.00	55.0 - < 60.0	B (Above Average)	6
5.00 - < 5.50	50.0 - < 55.0	C (Average)	5
4.00 - < 5.00	40.0 - < 50.0	P (Pass)	4
Below 4.00	Below 40.0	F (Fail)	0
Ab (Absent)	-	Ab (Absent)	0

Sd/-
Sign of the BOS
Chairman
Dr. Srivaramangai R
BOS in Information
Technology

Sd/-
Sign of the
Offg. Associate Dean
Dr. Madhav R. Rajwade
Faculty of Science &
Technology

Sd/-
Sign of the Offg. Dean
Prof. Shivram S. Garje
Faculty of Science &
Technology