

BCA

First and Second Semester Syllabus (SEP)

2024

Chamarajanagara University

Programme Objectives

1. To Provide them theoretical as well as practical knowledge on the effective usage of computers in solving a specific problem
2. To provide a foundation of computing principles for effectively using/managing information systems and enterprise software
3. To produce outstanding IT professionals who can apply the theoretical knowledge into practice in the real world
4. To helps students analyse the requirements for software development
5. To develop among students the programming techniques and the problem solving skills through programming
6. To prepare students for pursuing higher studies in computer science and related subjects

Curriculum

Program: BCA

Subject: Computer Applications

Semester	Course No.	Theory/ Practical	Credits	L-T-P	No. of Hours	Paper Title	Marks	
							SEE	CIE
I	CAM11T	Theory	03	3-0-0	03	Digital Computer Organization	80	20
	CAM11P	Practical	02	0-0-2	04	Office Automation, Digital Electronics and HTML	40	10
	CAM12T	Theory	03	3-0-0	03	Problem Solving using C++	80	20
	CAM12P	Practical	02	0-0-2	04	C++ Programming	40	10
	CAM13T	Theory	03	3-0-0	03	Mathematical and Statistical Computing	80	20
	CAM13P	Practical	02	0-0-2	04	Mathematical and Statistical Computing using R	40	10
II	CAM21T	Theory	03	3-0-0	03	Data Structures	80	20
	CAM21P	Practical	02	0-0-2	04	Data Structures using C++	40	10
	CAM22T	Theory	03	3-0-0	03	Object Oriented Programming with Java	80	20
	CAM22P	Practical	02	0-0-2	04	Programming with Java	40	10
	CAM23T	Theory	03	3-0-0	03	Operating Systems	80	20
	CAM23P	Practical	02	0-0-2	04	Shell Programming	40	10

Program Credit Structure : BCA (Bachelor of Computer Application) I and II Semester

Sem	Course-ID	Credit	L:T:P	No of Hours	Course Title	C1 (CIE)	C2 (CIE)	C3 (SEE)
1	DSC-1.1T (Theory)	3	3:0:0	03	Computer Fundamentals and Digital Electronics	10	10	80
	DSC-1.1P (Practical)	2	0:0:2	04	Office Automation and HTML	05	05	40
	DSC-1.2T (Theory)	3	3:0:0	03	Problem Solving and C++ Programming	10	10	80
	DSC-1.2P (Practical)	2	0:0:2	04	Programming Using C++	05	05	40
	DSC-1.3T (Theory)	3	3:0:0	03	Mathematical and Statistical Computing	10	10	80
	DSC-1.3P (Practical)	2	0:0:2	04	Computational Mathematics and Statistics	05	05	40
	Language-1	3	3:0:0	03	Language-1	10	10	80
	Language-2	3	3:0:0	03	Language-2	10	10	80
	IC/EVS	2	2:0:0	02	IC/EVS	05	05	40
2	DSC-2.1T (Theory)	3	3:0:0	03	Data Structures and Applications	10	10	80
	DSC-2.1P (Practical)	2	0:0:2	04	Data Structures using C++	05	05	40
	DSC-2.2T (Theory)	3	3:0:0	03	Operating System and UNIX	10	10	80
	DSC-2.2P (Practical)	2	0:0:2	04	UNIX Shell Programming	05	05	40
	DSC-2.3T (Theory)	3	3:0:0	03	Object Oriented Programming using Java	10	10	80
	DSC-2.3P (Practical)	2	0:0:2	04	Java Programming	05	05	40
	Language-1	3	3:0:0	03	Language-1	10	10	80
	Language-2	3	3:0:0	03	Language-2	10	10	80
	IC/EVS	2	2:0:0	02	IC/EVS	05	05	40

Semester: I

Course Code: CAM11T	Course Title: Digital Computer Organization
Course Credits: 03 (3-0-0)	Hours/Week: 03
Total Contact Hours: 44	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Understand the digital computer system including classification of computers, anatomy of computer, input/output devices and memory organization of computer.
2. Illustrate the types of Software, Computer languages and Translator programs.
3. Apply Boolean algebra to simplify logical expressions and solve problems using Karnaugh maps and other minimization techniques.
4. Design and analyze combinational and sequential logic circuits, including adders, subtractors, flip-flops, encoders, decoders, multiplexers, and counters.
5. Perform conversions between decimal, binary, octal, and hexadecimal number systems and carry out arithmetic operations in binary.

Course Contents

Unit-1	
<p>Fundamentals of Computers: Introduction to Computers - Computer Definition, Characteristics Of Computers, Evolution And History Of Computer, Types Of Computer, Basic Organization Of A Digital Computer.</p> <p>Input / Output Organization: Peripheral Devices, Input – Output Interface.</p> <p>Memory Organization: Computers Memory System Overview- Characteristics And Types of Memory System.</p> <p>Types Of Software: System Software, Application Software and Utility Software;</p> <p>Computer Languages: Machine Level, Assembly Level & High-Level Languages,</p> <p>Language Translators: Assembler, Interpreter and Compiler.</p>	11 Hours
Unit-2	
<p>Number Systems: Introduction, Decimal, Binary, Octal and Hexadecimal. Inter-Conversions, Addition, Subtraction, Multiplication and Division In Binary Number System. 1's and 2's Complement Method in Binary Number System. Subtraction Using 1's and 2's Compliment, Weighted Number System, Binary Coded Decimal (BCD), Addition of BCD Numbers. Non-Weighted Number System, Applications, Excess-3, Gray Code Conversions, Gray and Binary Codes.</p>	11 Hours

Unit-3	
Boolean Algebra: Basic Laws, Demorgan's Theorem, Duality Theorem, Sum of Product Method And Products Of Sum Method. Karnaugh Map (Upto 4 Variables, Don't Care Condition).	11 Hours
Fundamentals Of Gates: Basic Gates, Derived Gates and Universal Gates (Design).	
Unit-4	
Combinational And Sequential Logic Circuits: Half Adder, Full Adder, Half - Subtractor and Full-Subtractor.	11 Hours
Flip-Flops: SR, JK, Master-Slave JK, T Flip-Flops, Decoders - 3 To 8 Lines, Encoders- Octal To Binary.	
Multiplexer: 4 To 1 Line, Counters-3 Bits Binary Ripple Counter, 3 Shift Registers- Serial-In-Parallel-Out, Parallel-In-Serial-Out.	

Reference:

1. Computer Fundamentals, V Rajaraman.
2. Computer System Architecture (3rd edition) Morris Mano PHI.
3. Computer Organization – by V.Carl Hamacher, Z.G.Vranesic, and S.G.Zaky, 3rd Edition. McGraw Hill,
4. Computer Organization & Design, (3rd Edition) by – D.A.Patterson & J.L.Hennessy – Morgan Kaufmann Publishers (Elseviers)

Course Code: CAM11P	Course Title: Office Automation and HTML
Course Credits: 02 (0-0-2)	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Confidently work on Office Automation software such as word processor, spreadsheet and power point.
2. Understand the Web Programming basics and create simple web pages using HTML.

Laboratory Program List

Required Explanations should be explained during lab period only.

Part A:

1. Demonstrate working concepts of Logical Gates (NOT, AND, OR, NAND, NOR Gates) using simulator software.
2. Demonstrate working principals and concepts of Universal Gates (NAND & NOR Gates) using simulator software.
3. Demonstrate working principals and concepts of Half Adder and Full Adder using simulator software.
4. Demonstrate working principals and concepts of Half Subtractor and Full Subtractor using simulator software.
5. Using spread sheet, with suitable example, write steps and create worksheet called "Employee" and calculate the following using formulas
 - i. Enter Employee Code, Name and Basic Salary.
 - ii. Calculate DA (20% of Basic Salary).
 - iii. Calculate HRA (10% of Basic Salary).
 - iv. Calculate CCA (8.5% of Basic Salary).
 - v. Calculate Total Salary (Basic Salary + DA + HRA + CCA)
 - vi. Calculate Deductions (10% of Total Salary).
 - vii. Calculate Net Salary (Total Salary – Deductions).
6. Using spread sheet draw X-Y Line Chart and Bar Charts based on the following worksheet data and write the steps

ITEM MONTHLY SALES
(in Thousands)

Cotton 2,750
Wool 3,100
Yarn 2,975
Jute 2,100

Fiber 3,010

7. Using spread sheet write the steps and execute the following:
Roll,No,StudName,Marks1, Marks2, Mark3 ,Total Percentage, Result
 - i. Create appropriate records
 - ii. Calculate total and marks using formula.
 - iii. Update result column using IF function.
(Result: Distinction, First Class, Second Class, Pass, Fail).
8. Using Power Point with suitable examples write steps and execute the following:
 - i. Create presentation slides with Titles, Sub Titles and Charts choosing different slide layouts.
 - ii. Use Design templates for background.
 - iii. Format the slide design.
9. Using PowerPoint , Create the presentation for
 - i) "Components of PC" using organization chart.
 - ii. Use different views such as slide view, slide sorter view and slide show view.
10. Using PowerPoint , Create the presentation to demonstrate
 - i) Insert Images, Shapes.
 - ii) Charts for tabulated data
 - iii) Animations of Slides

Part B:

1. Design a page having suitable background colour and text colour with title "My First Web Page" using all the attributes of the Font tag.
2. Write HTML code to design a page containing some text in a paragraph by giving suitable heading style
3. Write a HTML program for the demonstration of Lists.
 - a. Unordered List
 - b. Ordered List
4. Write a HTML program for demonstrating Hyperlinks.
 - a. Navigation from one page to another.
 - b. Navigation within the page.
5. Write a HTML program for time-table using tables.
6. Write a HTML program to develop a static Registration Form.
7. Write a HTML program to develop a static Login Page.
8. Write a HTML code to create a web page with pink color background and display moving message in red color.
9. Write a HTML program to develop a static Web Page for Shopping Cart.
10. Write a HTML program to develop simple calculator.

Evaluation Scheme for Lab Examination [Marks: 40]

- **Writing:** One program from both Part A and Part B (10 Mark each): $10 \times 2 = 20$
- **Execution:** Any one of the written Program: 10 Marks
- **Record:** 05 Marks
- **Viva:** 05 Marks

Course Code: CAM12T	Course Title: Problem Solving using C++
Course Credits: 03 (3-0-0)	Hours/Week: 03
Total Contact Hours: 44	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Understand the fundamental concepts and benefits of Object-Oriented Programming (OOP) and how it differs from Procedure-Oriented Programming paradigms.
2. Interpret and apply C++ syntax and structure, including input-output statements, keywords, identifiers, constants, variables, data types, operators, expressions and file handling to create basic programs and solve problems.
3. Describe the control structures, functions, and different parameter passing methods and write programs to solve problems.
4. Demonstrate the concepts of classes and objects, access specifiers, constructors, destructors, and OOP features like polymorphism, inheritance with the help of programs.

Course Contents

Unit-1	
Introduction to Programming: Program development life cycle, Introduction to Procedure Oriented Programming and Object-Oriented Programming (OOP) paradigms, basic concepts of OOP, benefits and applications of OOP. Introduction to C++: Overview of C++, Structure of C++ Program, Input-Output statements, Keywords, Identifiers, Constants, Variables, Data types, Operators, Types of operators, Expressions, Precedence of Operators, Type Conversion, Storage classes.	11 Hours
Unit-2	
Control statements: Selection And Iteration Statements, Loop Control Statements. Modular Programming: Functions and Its Types, Recursion, Functions with Default Arguments, Inline Functions, Function Overloading, Call by Value and Reference, Math Library Functions.	11 Hours
Unit-3	
Derived Data Types: Arrays, Array Types, Strings, String Manipulation Functions, Pointers, Pointer Arithmetic. Managing Console, I/O Operations: C++ Stream, C++ Stream Classes,	11 Hours

<p>Unformatted I/O Operations, Formatted Console I/O Operations, Managing Output with Manipulators.</p> <p>User Defined Data Type: Class Definition, Instance Variables, Member Methods, Accessing Members, Access specifiers, this pointer, Friend Function, Constructors, Types of Constructors, Destructor.</p>	
Unit-4	
<p>Polymorphism: Operator Overloading, Rules for Operator Overloading, Overloading Unary and Binary Operators.</p> <p>Inheritance: Inheritance, Types of Inheritance, Virtual Functions and Abstract Classes.</p> <p>File Handling: Introduction To Files and File Handling, File Opening Modes, Classes For File Stream Operations, File I/O Operations (Opening, Reading, Writing, Append And Closing).</p>	11 Hours

Reference Books:

1. Object-Oriented Programming With C++, By M. T. Somashekara, D. S. Guru, H. S. Nagendraswamy, K. S. Manjunatha, PHI Learning Pvt. Ltd.
2. Object-Oriented Programming With C++, By E Balaguruswamy, Tata McGraw-Hill Publicationm Company Ltd.
3. The C++ Programming Language, By Stroustrup, Bjarne, Addison Wesley.
4. How To Solve It by Computer - R G Dromey, Prentice-Hall International.

Course Code: CAM12P	Course Title: C++ Programming
Course Credits: 02 (0-0-2)	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Demonstrate fundamental C++ programming concepts by writing programs for simple problems.
2. Utilize features of C++, such as recursion, function overloading, and friend functions, to enhance the functionality and efficiency of programs.
3. Design and construct classes and objects in C++ to model real-world entities, demonstrate inheritance, operator overloading, constructors, and file handling operations.

Laboratory Program List

Part -A

1. Write a C++ Program to swap 2 numbers with and without using temporary variable.
2. Write a C++ program to read radius of a circle and to find area and circumference.
3. Write a C++ Program to convert the Fahrenheit to Celsius and vice-versa.
4. Write a C++ program to find the roots of the quadratic Equation
5. Write a C++ program to demonstrate functions of simple calculator using Switch statement
6. Write a C++ program to read a number, find the sum of the digits, reverse the number and check it for palindrome
7. Write a C++ Program to generate Fibonacci series.
8. Write a C++ Program to compute sum of principle diagonal, lower triangular and upper triangular elements of a matrix.
9. Write a C++ Program to reverse a given string without using built-in function.
10. Write a C++ Program to demonstrate the usage of any five Math.h library functions.

Part-B

1. Write a C++ Program to demonstrate call by value and call by reference.
2. Write a C++ Program to generate factorial of a given number using recursion.
3. Write a C++ Program to create a Class for representing student details with appropriate member functions to accept and display the details.
4. Write a C++ Program to demonstrate function overloading.
5. Write a C++ Program to demonstrate friend function.
6. Write a C++ Program for single inheritance.

7. Write a C++ Program to demonstrate multilevel inheritance.
8. Write a C++ Program to demonstrate operator overloading.
9. Write a C++ Program to demonstrate the usage of default and parameterized constructors.
10. Write a C++ Program to read and display the contents of a text file.

Evaluation Scheme for Lab Examination [Marks: 40]

- **Writing:** One program from both Part A and Part B (10 Mark each): $10 \times 2 = 20$
- **Execution:** Any one of the written Program: 10 Marks
- **Record:** 05 Marks
- **Viva:** 05 Marks

Course Code: CAM13T	Course Title: Mathematical and Statistical Computing
Course Credits: 03 (3-0-0)	Hours/Week: 03
Total Contact Hours: 44	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Construct, evaluate, and apply logical statements and truth tables, understand the principles of set theory, perform various set operations, and effectively use Venn diagrams for solving complex problems.
2. Understand Cartesian products, relations, and their properties, including equivalence relations and partitions. They will also gain skills in function composition, inverse functions, and representing relations through matrices and directed graphs.
3. Organize and interpret data using statistical methods, calculate measures of central tendency and dispersion, analyze correlation between variables, and perform linear regression analysis.

Course Contents

Unit-1	
Mathematical Logic Introduction: Statements Connectives - Negation, Conjunction, Disjunction- Statement Formulas and Truth Tables- Conditional and Bi Conditional Statements- Tautology, Contradiction. Set Theory: Sets And Subsets, Set Operations and The Laws of Set Theory, Counting and Venn Diagrams.	11 Hours
Unit-2	
Cartesian Products and Relations, Properties of Relations. Computer Recognition: Relation Matrices and Directed Graphs, Equivalence Relations and Partitions. Functions: One-to-One, Onto Functions, Function Composition and Inverse Functions.	11 Hours
Unit-3	
Statistical methods: Introduction, Definitions, Classifications, Frequency Distribution, Mean - Arithmetic Mean for Grouped and Ungrouped Data. Median: Meaning, Calculations of Median for Ungrouped.	11 Hours

Mode: Meaning, Calculations of Mode for Discrete Series and Continuous Series.	
Unit-4	
Standard Deviation: Meaning, Standard Deviation for Actual Mean Method, Assumed Mean Method and Step Deviation Method Using Discrete Series And Continuous Series.	11 Hours
Coefficient of Variation: Meaning and Problems.	
Correlation: Meaning, Types, Rank Correlations and Problems.	
Simple Linear Regression: Meaning, Properties of Regression Coefficients.	

Reference Books:

1. Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics", 5th Edition, Pearson Education, 2004.
2. Kenneth H. Rosen, "Discrete Mathematics and its Applications", 6th Edition, McGraw Hill, 2007.
3. Jayant Ganguly, "A Treatise on Discrete Mathematical Structures", Sanguine Pearson, 2010.
4. D.S. Malik and M.K. Sen, "Discrete Mathematical Structures: Theory and Applications", Thomson, 2004.
5. Thomas Koshy, "Discrete Mathematics with Applications", Elsevier, 2005, Reprint 2008.
6. Fundamentals of Mathematical Statistics by Gupta and Kapoor (Sultan Chand).
7. Mathematical Statistics by John Freund (Prentice Hall India Pvt. Ltd.)

Course Code: CAM13P	Course Title: Mathematical and Statistical Computing using R
Course Credits: 02 (0-0-2)	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Develop practical skills in implementing set operations and function operations using R programming.
2. Implement logic gates using R and perform comprehensive statistical analysis including calculations of central tendency and conduct linear regression analysis.
3. Compute Cartesian products and analyze relations for properties such as reflexivity, symmetry, and transitivity through R scripts.

Laboratory Program List

Part A

1. Write a R Program to implement operations of Set (Union, Intersection, Difference, Subset).
2. Write a R program to implement inverse function.
3. Write a R Program to implement one-to-one function.
4. Write a R Program to implement Cartesian Product of Two sets.
5. Write a R Program to check whether the given relation is Reflexive.
6. Write a R Program to check whether the given relation is Transitive.
7. Write a R Program to implement logic gates (NOT, AND, OR, XOR).
8. Write a R Program to check whether the given relation is Symmetric.

Part B

1. Write a R Program to Calculate central tendency (mean, median, mode).
2. Write a R Program to Calculate standard deviation and variance for discrete & continuous series.
3. Write a R Program to Calculate coefficient of variance for discrete & continuous series.
4. Write a R Program to Calculate simple Linear Algebra Operations.
5. Write a R Program to Calculate arithmetic mean for grouped and ungrouped data.
6. Write a R Program to Calculate cumulative sums, and products, minima, maxima.
7. Write a R Program to Calculate frequency distribution for discrete & continuous series.
8. Write a R Program to Calculate Simple Linear Regression.

Evaluation Scheme for Lab Examination [Marks: 40]

- **Writing:** One program from both Part A and Part B (10 Mark each): $10 \times 2 = 20$
- **Execution:** Any one of the written Program: 10 Marks
- **Record:** 05 Marks
- **Viva:** 05 Marks

Semester II

Course Code: CAM21T	Course Title: Data Structures
Course Credits: 03 (3-0-0)	Hours/Week: 03
Total Contact Hours: 44	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Understand the basics of Data Structures.
2. Identify the appropriate data structures and algorithms for solving real world problems.
3. Understand the practical applications of Tree and Graph.
4. Understand the fundamentals of sorting and searching algorithms.

Course Contents:

Unit-1	
<p>Introduction: Data Structure Definition, Basic Terminology and Concepts, Importance of Data Structures in Programming. Classification of Data Structures. Primitive Data Structures, Non-Primitive Data Structures.</p> <p>Stack: Definition, Memory Representation, Algorithms for Stack Operations (Push, Pop), Applications of Stack.</p>	11 Hours
Unit-2	
<p>Queue: Definition, Memory Representation, Linear Queue, Circular Queue, Enqueue, Dequeue. Applications of Queue.</p> <p>Linked Lists: Definition, Types.</p> <p>Singly Linked List: Implementation, Insertion [At the Beginning], Deletion [At the End].</p> <p>Doubly Linked List: Memory Representation of Singly Linked List and Doubly Linked Lists. Applications of Linked List.</p>	11 Hours
Unit-3	
<p>Tree: Definition, Memory Representation Using Array and Linked List.</p> <p>Binary Tree: Definition, Traversal Algorithms [Pre-Order, In-Order, Post-Order], Construction of Tree from In-Order and Pre-Order, In-Order and Post-Order.</p> <p>Binary Search Trees: Insertion of a Node, Deletion of A Node.</p> <p>Advanced Tree Structures AVL And B-Trees: Definition and Applications.</p>	11 Hours

Unit-4	
<p>Graph: Definition, Memory Representation of Graph. Adjacency Matrix, Adjacency List. Graph Traversal Algorithms: Breadth-First Search (BFS), Depth-First Search (DFS).</p> <p>Sorting Techniques: Bubble Sort, Selection Sort [Algorithm, Time & Space Complexity].</p> <p>Searching Techniques: Linear And Binary Search Sort [Algorithm, Time & Space Complexity].</p> <p>Heap: Heap Operations and Applications.</p>	<p>11 Hours</p>

Reference Books:

1. Data Structures Through C++ (4th Edition) Yashvant Kanetkar.
2. Data Structures and Algorithm Analysis in C++" by Mark Allen Weiss.
3. Data structure and Algorithms using C++ by Sachi Nandan Mohanty, Pabitra Kumar Tripathy.
4. Data Structures and Algorithms in C++, Second Edition by Adam Drozdek.

Course Code: CAM21P	Course Title: Data Structures using C++
Course Credits: 02 (0-0-2)	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Implement data structures using C++.
2. Demonstrate searching and sorting techniques using C++.
3. Demonstrate advanced programming skills through C++ programming language.

Laboratory Program List

Part A:

1. Write a C++ Program to find GCD of two numbers.
2. Write a C++ Program to implement Tower of Hanoi.
3. Write a C++ Program to print Fibonacci series.
4. Write a C++ Program to find largest and smallest element in an array.
5. Write a C++ Program to perform stack operations.
6. Write a C++ Program to perform Linear queue operations
7. Write a C++ Program to insert a node at the beginning of a singly linked list.
8. Write a C++ Program to delete a node at the end of a singly linked list.

Part B:

1. Write a C++ Program to construct a binary search tree
2. Write a C++ Program for Binary Tree traversal.
3. Write a C++ Program to implement DFS
4. Write a C++ Program to implement BFS
5. Write a C++ Program to Sort an Array (Selection Sort)
6. Write a C++ Program to Sort an Array (Bubble Sort)
7. Write a C++ Program to perform Linear Search of an Element in an Array.
8. Write a C++ Program to perform Binary Search of an Element in an Array.

Evaluation Scheme for Lab Examination [Marks: 40]

- **Writing:** One program from both Part A and Part B (10 Mark each): $10 \times 2 = 20$
- **Execution:** Any one of the written Program Output: 10 Marks
- **Record:** 05 Marks
- **Viva:** 05 Marks

Course Code: CAM22T	Course Title: Object Oriented Programming with Java
Course Credits: 03 (3-0-0)	Hours/Week: 03
Total Contact Hours: 44	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Understand the Java programming fundamentals.
2. Describe with examples of basic Java OOP concepts.
3. Understand the Java Interfaces and Packages.
4. Deliberate the Details of Multithreading, Exception Handling & File Handling
5. Design GUI applications using tools like AWT.

Course Contents:

Unit-1	
<p>Fundamentals of Object-oriented Programming: Object-oriented Paradigm, Object and Classes, data Abstraction and encapsulation, Inheritance, Polymorphism, Dynamic Binding, Message Passing, Benefits of OOPS and Application of OOPS</p> <p>Introduction to JAVA: JAVA History, Features of Java, Difference between C++ and JAVA, JAVA Environment- JDK, API, Overview of JAVA Language, JAVA Program Structure, Java Tokens, JVM, Implementing a java Program, Command Line Arguments,</p> <p>Java Programming Fundamental: Constants, Variables , Data types, Type Casting, Operators and Expressions, evaluation of Expressions, Operator Precedence and associativity, Mathematical Functions.</p> <p>Control Statements: Selection statement, Looping Statements, jumping Statements</p>	11 Hours
Unit-2	
<p>Classes and Objects: Defining a Classes and objects, Constructors, Method Overloading, Overloading Constructor, static Members, Recursion, introducing access Control, this keyword, Garbage Collection, the finalize Method.</p> <p>Inheritance: Define Inheritance , types of inheritance Using Super, Method Overring , Dynamic Method Dispatch, abstract Methods and Classes, final-classes, variable and Methods visibility Control</p> <p>Arrays and Strings: One dimensional Arrays, creating an array, Two dimensional Arrays, variable Size arrays</p>	11 Hours

<p>Strings:- Define String, String Methods, Sting Buffer classes, Wrapper Classes, Autoboxing and Unboxing</p> <p>Interface: Defining Interfaces, Extending Interfacing, Implementing Interfaces, Nested Interface, Various form of Interface Implementation, Accessing Interface Variables.</p>	
Unit-3	
<p>Packages:- Java API Packages, Using System Package, Naming Conventions</p> <p>Creating package, accessing a package, using a package, adding a class to a Package , hiding classes, static import</p> <p>Exception Handling: Exception types, Exception, Using try catch and multiple catch, Nested try, throw, throws and finally, Creating User defined Exceptions.</p> <p>Multi-Threading: Introduction, Creating Threads-Implementing Runnable, extending thread class, stopping and blocking a thread, Life Cycle of threads, Thread Priority, Synchronization</p>	11 Hours
Unit-4	
<p>File Handling: Byte Stream, Character Stream, File IO Basics, File Operations, creating file, reading file, Writing File.</p> <p>Applet Programming: Introduction, Types Applet, How Applet differ from Application, Applet Life cycle, Creating an Executable Applet, Design Web Page, Applet tag, Running the applet, passing parameters to applets, the graphics class, line and rectangle, circle and ellipses .drawing arcs drawing polygons, line graph, using control loops in applets.</p> <p>Event and GUI programming: Event handling in java, Event types Event Classes , Mouse and key events, GUI Basics, Panels, Frame, window, Layout Managers: Flow Layout, Border Layout, Grid Layout, GUI components like Buttons, Check Boxes, Radio Buttons, Labels, Text Fields, Text Areas, Combo Boxes, Lists, Scroll Bars, Sliders, Windows, Menus, Dialog Box,</p>	11 Hours

Reference Books:

1. Ivan Bayross, Web Enabled Commercial Application Development Using Html, Dhtml,javascript, Perl Cgi , BPB Publications, 2009.
2. Cay Horstmann, BIG Java, Wiley Publication , 3rd Edition., 2009
3. Herbert Schildt , Java 7, The Complete Reference, , 8th Edition, 2009.
4. E Balagurusamy , Programming with JAVA, TMH, 2007
5. D.S. Guru, M.T. Somashekara, & K.S. Manjunatha, Object Oriented Programming with Java, PHI Learning, 2017.

Course Code: CAM22P	Course Title: Programming with Java
Course Credits: 02 (0-0-2)	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03

Course Outcomes (COs):

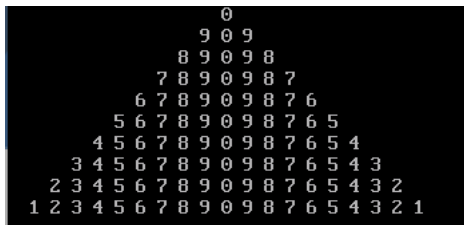
On successful completion of this course, students will be able to:

1. Implement simple programs using Java Fundamental concepts.
2. Identify classes, objects, members of class and the relationships among them needed for finding the solution to specific problems using Objected Oriented Programming concepts of Java.
3. Design & Develop simple GUI programs using AWT GUI tool.

Laboratory Program List

Part A:

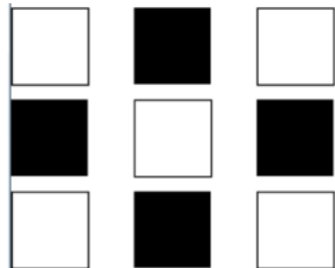
1. Write a JAVA Program to find whether the given number is Positive, Negative or Zero.
2. Write a JAVA program to check whether given Date is valid or not
3. Write a JAVA program to Reverse a Given Number and to check whether it is Palindrome or not
4. Write a JAVA Program to Generate the Following Pattern.



5. Write a JAVA program to Illustrate ATM transactions
6. Write a JAVA Programs to read numbers from users into array and Check whether that number is Amstrong or Not
7. Write a JAVA Program to find trace and norms of a give square Matrix
8. Write a JAVA Program to Generate the Employee Salary Slip Using Class and Object
9. Write a java Program to check whether Entered Character is a Vowel or Consonant Using constructor Concept
10. Write a java program that computes the area of a circle, rectangle and a Cylinder using Method overloading

Part-B Programs

1. Write a JAVA Program to demonstrate the concept of static to find the factorial of a number by recursive
2. Write a JAVA Program to Demonstrate 10 String Handling Functions
3. Write a JAVA program to find area of rectangle and triangle using interface.
4. Write a JAVA program to implement simple calculator using package.
5. Write a Java Program to Generate Student Marks Cards Using Inheritance
6. Write a program to implement multiple inheritance.
7. Write a Java Program to Calculate Bonus for Different Department Using Abstract Class
8. Write a program that reads two integer numbers for the variables a and b. If any other character except number (0-9) is entered then the error is caught by NumberFormatException object. After that ex.getMessage () prints the information about the error occurring causes
9. Write a java Program to Demonstrate multi-threading Implementing Runnable Interface
10. Write a java program to Generate Following Pattern using Applet



Evaluation Scheme for Lab Examination [Marks: 40]

- **Writing:** One program from both Part A and Part B (10 Mark each): $10 \times 2 = 20$
- **Execution:** Any one of the written Program Output: 10 Marks
- **Record:** 05 Marks
- **Viva:** 05 Marks

Course Code: CAM23T	Course Title: Operating Systems
Course Credits: 03 (3-0-0)	Hours/Week: 03
Total Contact Hours: 44	Formative Assessment Marks: 20
Exam Marks: 80	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Understand the fundamentals of the operating system.
2. Describe the concepts of process, process management, CPU Scheduling, process synchronization, Dead locks, memory management and Virtual Memory management.
3. Illustrate the file system and structure.
4. Understand the UNIX OS, Shell Programming, Conditional Control Structures in Shell Programming.

Course Contents:

Unit-1	
<p>Introduction: Definition, Computer System Components , User View , System View And System Goals , Batch Systems, Multi Programmed Systems, Time-Sharing Systems, Real-Time Systems, System Components, Operating System Services.</p> <p>Process: Process Concept, Process State Diagram Process Control Block, Process Scheduling- Scheduling Queues, Scheduler, Cooperating Process, Interprocess Communication.</p> <p>CPU Scheduling: Basic Concepts, Preemptive And Non-Preemptive Scheduling, Scheduling Criteria, Scheduling Algorithms-FCFS, Shortest Job First Priority Scheduling, Round Robin Scheduling.</p>	11 Hours
Unit-2	
<p>Process Synchronization: The Critical Section Problem, Solution For Critical Section Problem, Bakery Algorithm, Semaphores-Meaning, Types Of Semaphores, Synchronization Problems- Bounded Buffer Problem, Readers-Writers Problem .</p> <p>Deadlocks: Deadlock Characterization, Methods For Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery From Deadlock.</p>	11 Hours
Unit-3	
<p>Memory Management: Introduction, Logical Versus Physical Address Space, Dynamic Loading, Dynamic Linking, Swapping, Contiguous Allocation, Partitioned</p>	11 Hours

<p>Memory Allocation, Paging, Virtual Memory Management-Segmentation, Segmentation With Paging.</p> <p>File System: File Concepts, File Attributes, File Operations, File Types, File Structure, Access Methods, Directorystructure, File-System Structure, Allocation Methods- Contiguous Allocation, Linked Allocation and Indexed Allocation, Free-Space Management.</p>	
Unit-4	
<p>Introduction to Unix System: The Unix Operating System, The UNIX architecture.</p> <p>Shell Programming: Vi editor, shell types, shell command line processing, shell script features, executing a shell script, system and user-defined variables, expr command, shell screen interface, read and echo statement, command substitution, escape sequence characters, shell script arguments, positional parameters, test command, file test, string test, numeric test.</p> <p>Conditional Control Structures: if statement, case statement Looping Control Structure-while, until, for, statements. Jumping Control Structures – break, continue, exit. Shell Programs covering the above concepts.</p>	<p>11 Hours</p>

Reference Books:

1. Operating System Concepts – 5th edition by Abraham Silberschartz and Peter Galvin, McGraw Hill,2000
2. Modern Operating Systems – Andrew S Tanenbaum, Prentice Hall
3. Operating Systems : Internals and Design Principles, William Stallings, Prentice Hall
4. Sumitabha Das: UNIX – Concepts and Applications, 4th Edition, Tata McGraw Hill, 2006.

Course Code: CAM23P	Course Title: Shell Programming
Course Credits: 02 (0-0-2)	Hours/Week: 04
Total Contact Hours: 60	Formative Assessment Marks: 10
Exam Marks: 40	Exam Duration: 03

Course Outcomes (COs):

On successful completion of this course, students will be able to:

1. Develop skill in shell scripting to perform simple operations and problems.
2. Perform file manipulation using shell scripts.
3. Understand and implement shell scripts for system information.

Laboratory Program List

Part A:

1. Write a shell script to swap 2 values.
2. Write a shell script to check if the given number is even or odd.
3. Write a shell script to find the largest of 3 numbers.
4. Write a shell script to perform arithmetic operations.
5. Write a shell script to find the sum of first 10 natural numbers.
6. Write a shell script to display multiplication table of a given number.
7. Write a shell script to find the length of a given string.
8. Write a shell script to find factorial of a given number.
9. Write a shell script which counts the numbers of lines and number of words present in a given file.
10. Write a shell script to display the Fibonacci series upto N number.

Part B:

1. Write a shell script to search for particular element from an array of elements.
2. Write a shell script to calculate the TA, HRA and DA of an employee.
3. Write a shell script that displays a list of all files in the current directory to which the user has read write and execute permissions.
4. Develop an interactive script that asks for a word and file name and then tells how many times that word occurred in the file.
5. Write a shell script to extract a sub string from a given string.
6. Perform the following operations
7. Concatenate 2 strings
8. Rename a file
9. Delete a file

10. Copy the file
11. Write a shell script to display the
 - a) Version of the shell
 - b) The user information
 - c) Login date and time
 - d) List of processes running on the system
 - e) User home directory
12. Write a C program to display PID of parent and PID of child process.
13. Write a shell script that takes two filename as arguments. It should check whether the contents of two files are same or not, if they are same then second file should be deleted.
14. Assume a file with the given information
15. First Name Middle Name Age
16. -----
17. Write a shell script to
 - a. Sort the first name in alphabetical order
 - b. Sort the age in terms of ascending order
 - c. Sort the age in terms of descending order
 - d. Sort the middle name in alphabetical order

Evaluation Scheme for Lab Examination [Marks: 40]

- **Writing:** One program from both Part A and Part B (10 Mark each): $10 \times 2 = 20$
- **Execution:** Any one of the written Program Output: 10 Marks
- **Record:** 05 Marks
- **Viva:** 05 Marks

CIE, SEE and QP Pattern for Theory Courses

Total Lecture hours per paper: 44

No. of Units 4 (11 Hours Each)

Internal Assessment C1 = 10 Marks, C2 = 10 Marks

Semester End Theory Exam C3 = 80 Marks

Question paper pattern

Instructions: Answer Part-A and Part-B:

Part-A

Answer any 10 out of 12 Questions (3 Questions drawn from each unit). Each question carries 2 Marks. (10 X 2 =20)

Q. No. 1 to Q. No. 12.

Part-B

Answer all the Questions. Each question carries 15 Marks. (4 X 15 =60)

(Each question with internal choice and with maximum of 3 sub questions)