SEMESTER II

Course Title	**Java Programming**							
Course Type	Hard Core	***Total Hours***	48	Hours / Week		04	***Credits***	03
							(L : T : P)	(2 : 1 : 0)
Course Code		***Evaluation***	***Internal***	C1 + C2 = 15 + 15			30 Marks	100 Marks
			External	C3	***Duration***	03 Hours	70 Marks	

COURSE PREREQUISITE: C++

COURSE OBJECTIVE: This course aims to enable learners:

1. To understand the basic concepts and fundamentals of platform independent object oriented language.
2. To develop skills to write programs using inheritance, packages and interfaces.
3. To acquire skills in writing programs using exception handling techniques and multithreading.
4. To understand the concept of handling strings and events.
5. To develop skills to create java applications using applets, swings and networking concepts.

COURSE OUTCOME (CO): At the end of the course, the student will be able to

COURSE OUTCOMES	
CO1	Explain the basics of OOPs concepts.
CO2	Identify classes, objects, members of a class and use packages and interfaces appropriately.
CO3	Demonstrate for Java program formulate thread, synchronization and exception handling concepts.
CO4	Use the concept of string, event handling, simple data structures like arrays and members of classes of Java API in application development.
CO5	Design and develop Java based UI and applications using applets, swing components.

Course Content

Unit No	Content	Hours
1	**UNIT-I-Java Basics** History of Java, Java buzzwords, data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, using objects as parameters, Returning objects, recursion, nested and inner classes, exploring string class	10
2	**UNIT-II-Inheritance, Packages and Interfaces** **Inheritance** - Inheritance Basics, Using 'super', Creating Multilevel hierarchy, Method Overriding, Dynamic method dispatch, Using Abstract Classes, Using final with Inheritance. **Packages and Interfaces** - Packages, Access Protection, Importing packages, Interfaces, Default interface methods, Use static methods in an interface. **Exploring java.io-** The I/O classes and interfaces, The Byte streams, The Character streams	8
3	**UNIT III- Exception Handling and Multithreaded Programming** Exception Handling Fundamentals, Exception Types, Uncaught Exception, Using try and catch, Multiple catch clauses, Nested try Statements, throw, throws, finally, Java Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exception, Three Recently Added Exception Features. The Java Thread Model, The Main Thread, Creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread priorities, Synchronization, Inter Thread communication, Suspending, Resuming, and Stopping Threads, Obtaining A Thread's State, Using Multithreading. **Enumerations, Type Wrappers, Autoboxing, Annotations**	10
4	**UNIT-IV-String and Event Handling** String fundamentals, String Constructors, The Length () method, character extraction, String comparison, searching strings, Modifying a string, changing the case of the characters within the string, String buffer and String builder. Delegation event model, Event classes, Event Listener Interfaces, handling mouse and keyboard events, Adapter classes. The AWT class hierarchy, user interface components- labels, button, check box, checkbox groups, choices, lists, scroll bars, textfield, textarea, menu bars and menus, dialog boxes. layout manager – layout manager types – border, grid, flow, card and grid bag.	10
5	**UNIT-V- Applets, Swings and Networking with Java.Net** The Applet Class, Types of Applets, Applet Basics, Applet Architecture, An Applet Skeleton, Simple Applet Display Methods, Requesting Repaint, The HTML APPLET tag . Swings- Introduction, limitations of AWT, two key swing features, MVC architecture, components and containers, a simple swing application, exploring swing- ImageIcon and JLabel, JTextField, buttons – The JButton class, Check boxes, Radio buttons, Combo boxes, JTabbedPane, JScrollPane, Trees, and JTable. Exploring collection framework, Collection overview, Collection classes and interfaces, Array class.	10

Reference Books.

1. Herbert Schildt. Java - The Complete Reference, Ninth Edition. Oracle Press, McGraw Hill Education (India) Edition 2014.
2. Cay S. Horstmann, Gary Cornell. Core Java, Core Java Volume-1 – Fundamentals, 9th edition, Pearson Education, 2014.
3. Timothy Budd, “Understanding Object-oriented programming with Java”, Updated Edition, Pearson Education, 2000.

SEMESTER II

<table>
<tr><td><b><i>Course Title</i></b></td><td colspan="8"><b>Advanced Database Management System</b></td></tr>
<tr><td rowspan="2"><b><i>Course Type</i></b></td><td rowspan="2">Hard Core</td><td rowspan="2"><b><i>Total Hours</i></b></td><td rowspan="2">48</td><td rowspan="2" colspan="2">Hours / Week</td><td rowspan="2">04</td><td><b><i>Credits</i></b></td><td>03</td></tr>
<tr><td><b><i>(L : T : P)</i></b></td><td>(3 : 0 : 0)</td></tr>
<tr><td rowspan="2"><b><i>Course Code</i></b></td><td rowspan="2"></td><td rowspan="2"><b><i>Evaluation</i></b></td><td><b><i>Internal</i></b></td><td colspan="3">C1 + C2 = 15 + 15</td><td>30 Marks</td><td rowspan="2">100 Marks</td></tr>
<tr><td><b><i>External</i></b></td><td>C3</td><td><b><i>Duration</i></b></td><td>03 Hours</td><td>70 Marks</td></tr>
</table>

COURSE PREREQUISITE

1. Basic knowledge of Database Management Systems (DBMS)
2. Familiarity with SQL and relational database concepts
3. Understanding of basic programming concepts, preferably in C++ or Java
4. Exposure to NoSQL and Cloud Databases (Recommended)

COURSE OBJECTIVE: This course aims to enable learners:

1. To deepen understanding of database architecture and models.
2. To master advanced SQL and query optimization techniques.
3. To apply concepts of database indexing and optimization.
4. To understand and implement transaction management and concurrency control.
5. To explore distributed, NoSQL, and specialized databases.
6. To develop the skills to design and implement complex database systems that support real-world applications.

COURSE OUTCOME (CO): At the end of the course, the student will be able to

COURSE OUTCOMES	
CO1	Demonstrate the fundamentals of data models and conceptualize and depict a database system. Make use of ER diagrams in developing ER Models.
CO2	Summarize SQL and relational database design.
CO3	Illustrate transaction processing, concurrency control techniques, and recovery.
CO4	Infer the database design in real-world entities.
CO5	Explore and evaluate modern databases, including NoSQL and cloud-based systems, and their applications in various domains.

Unit No	Course Content	Hours
1	**UNIT I - Introduction to DBMS Concepts & Architectures** Database and Need for DBMS, Characteristics of DBMS, Database 3-tier schema (ANSI/SPARC) and system architecture of DBMS, Views of data - Schemas and instances, Data Independence. Centralized systems, Client-Server systems, Transaction servers, Data servers, Cloud-based servers. Indexing and Hashing - Basic concepts of indexing, ordered index, B+ tree index (overview only), Hashing concepts, types of hashing.	10
2	**UNIT II - Data Modelling and Relational Database Design** Data Modelling using ER Diagram: Representation of Entities, Attributes, Relationships, and their Types, Cardinality, Generalization, Specialization. Relational Data Model: Structure of Relational Database Model, Types of Keys, Referential Integrity Constraints, Codd's Rules. Database Design: E-R to Relational Mapping, Informal Design Guidelines for Relation Schemas, Functional Dependencies, Normal Forms based on Primary Keys - General Definitions of 1st, 2nd, and 3rd Normal Forms.	10
3	**UNIT III - Advanced SQL and Query Optimization** Overview of SQL Query Language, SQL Data Definition, Basic Structure of SQL Queries, SQL Data Types and Schemas. Basic Operations: Joins, Subqueries, Transactions, Views, Aggregate Functions. Integrity Constraints, Basic Query Optimization Techniques, Indexes in SQL.	8
4	**UNIT IV - Transaction Management and Concurrency Control** Concept of Transaction, ACID Properties, States of Transaction, Concurrency Control Problems, Lock-based Protocols, Deadlock Handling. Crash Recovery and Backup: Log-based Recovery, Checkpointing, Basic Backup Strategies.	10
5	**UNIT V - Modern Databases (NoSQL and Cloud Databases)** Introduction to NoSQL Databases: Types of NoSQL Databases (Key-Value, Document, Column-Family, Graph), Basic Use Cases. Cloud Databases: Introduction to Cloud Database Concepts, Benefits and Challenges, Overview of Popular Cloud DBMS (Amazon RDS, Google Cloud SQL, Azure SQL Database).	10

Reference Books

1. Henry F Korth, Abraham Silberschatz and S. Sudharshan, "Database System Concepts", Sixth Edition, McGraw Hill, 2011..
2. R. Elmasri, S.B. Navathe, "Fundamentals of Database Systems", Fifth Edition, Pearson Education/Addison Wesley, 2007.
3. Thomas Cannolly and Carolyn Begg, "Database Systems, A Practical Approach to Design, Implementation and Management", Third Edition, Pearson Education, 2007.

SEMESTER II

<table>
<tr><td><b><i>Course Title</i></b></td><td colspan="8"><b>Java Programming Laboratory</b></td></tr>
<tr><td rowspan="2"><b><i>Course Type</i></b></td><td rowspan="2">Hard Core</td><td rowspan="2"><b><i>Total Hours</i></b></td><td rowspan="2">24</td><td colspan="2" rowspan="2">Hours / Week</td><td rowspan="2">02</td><td><b><i>Credits</i></b></td><td>01</td></tr>
<tr><td><b><i>(L : T : P)</i></b></td><td>(0 : 0 : 1)</td></tr>
<tr><td rowspan="2"><b><i>Course Code</i></b></td><td rowspan="2"></td><td rowspan="2"><b><i>Evaluation</i></b></td><td><b><i>Internal</i></b></td><td colspan="3">C1 + C2 = 15 + 15</td><td>30 Marks</td><td rowspan="2">100 Marks</td></tr>
<tr><td><b><i>External</i></b></td><td>C3</td><td><b><i>Duration</i></b></td><td>03 Hours</td><td>70 Marks</td></tr>
</table>

COURSE PREREQUISITE: C++

COURSE OBJECTIVE: This course aims to enable learners:

1. To learn the concepts of data types, operators and control statements.
2. To develop skills to write programs using inheritance, exception handling and multithreading.
3. To acquire skills in writing programs using applets, swings, JDBC and MYSQL.

COURSE OUTCOME (CO): At the end of the course, the student will be able to

COURSE OUTCOMES	
CO1	Implement the programs to know the use of data types, operators and control statements.
CO2	Implement the OOPs Concept and write efficient Java programs using Inheritance, Exceptions and Multithreading.
CO3	Implement the concepts like Applet and Swings.

Course Content

1. Write a program to check if the entered number is ODD or EVEN.
2. Write a program to find the factorial of a given number using recursion.
3. Write a program to find the sum of the digits in a given number.
4. Write a program to swap two numbers without using a temporary variable.
5. Write a program that accepts a name and displays the name with a greeting message using Class.
6. Write a program to generate a salary for an employee using class, object, constructors, methods and access control. Different parameters to be considered are Emp_No, Emp_Name, Age, Basic, DA, HRA,PF, PT. Find Gross and Net salary.
7. Write a program to demonstrate Constructor Overloading and Method Overloading.
8. Implement Inner class and demonstrate its Access protection.
9. Write a program in Java for String handling which performs the following:
    a. Checks the capacity of StringBuffer objects.
    b. Reverses the contents of a string given on console and converts the resultant string in upper case.
    c. Reads a string from the console and appends it to the resultant string of (b).
10. Write a program to demonstrate Inheritance.
11. Write a program to implement Multiple inheritance using interfaces to calculate the area of a rectangle and triangle.
12. Write a JAVA program which has
    a. A Class called Account that creates an account with 500 Rs minimum balance, a deposit() method to deposit amount, a withdraw() method to withdraw amount and also throws Less Balance Exception if an account holder tries to withdraw money which makes the balance become less than 500 Rs.
    b. A Class called Less Balance Exception which returns the statement that says withdrawal amount (Rs) is not valid.
    c. A Class which creates 2 accounts, both account deposit money and one account tries to withdraw more money which generates a Less Balance Exception and takes appropriate action for the same.
13. Write a JAVA program using Synchronized Threads, which demonstrates the Producer Consumer concept.
14. Write a JAVA program to implement a Queue using user defined Exception Handling (also make use of throw, throws.).

15. Complete the following:
    a. Create a package named shape.
    b. Create some classes in the package representing some common shapes like Square, Triangle and Circle. Import and compile these classes in other programs.
16. Write a JAVA Program To Create an enumeration Day of Week with seven values SUNDAY through SATURDAY. Add a method called Workday( ) to the Day of Week class that returns true if the value on which it is called is MONDAY through FRIDAY. For example, the call Day Of Week SUNDAY is Workday ( ) returns false.
17. Write a JAVA program which has
    a. A Interface class for Stack Operations
    b. A Class that implements the Stack Interface and creates a fixed length Stack.
    c. A Class that implements the Stack Interface and creates a Dynamic length Stack.
    d. A Class that uses both the above Stacks through Interface reference and does the Stack operations that demonstrates the runtime binding.
18. Write a program to print a chessboard pattern.
19. Write a JAVA Program which uses File Input Stream / File Output Stream Classes.
20. Write a program to demonstrate utilities of Linked List Class.
21. Write a JAVA applet program, which handles keyboard events.
22. Write a JAVA Swing program, to design a form.

Reference Books

1. Herbert Schildt. Java - The Complete Reference, Ninth Edition. Oracle Press, McGraw Hill Education (India) Edition-2014.
2. Jim Keogh, J2EE The Complete Reference, Tata McGraw Hill Education Private Limited-2011.
3. Timothy Budd, “Understanding Object-oriented programming with Java”, Updated Edition, Pearson Education, 2000.
4. Java 8 Programming Black Book, Dreamtech Press-2012

SEMESTER II

Course Title	**Advanced Database Management Systems Laboratory**							
Course Type	Hard Core	***Total Hours***	24	Hours / Week		02	***Credits***	01
							(L : T : P)	(0 : 0 : 1)
Course Code		***Evaluation***	***Internal***	C1 + C2 = 15 + 15			30 Marks	100 Marks
			External	C3	***Duration***	03 Hours	70 Marks	

COURSE PREREQUISITE: NIL

COURSE OBJECTIVE: This course aims to enable learners:

1. Design database schema for a given application.
2. Working on database systems, designing of databases, creating relational databases and analysis of table design.
3. Acquire skills in using SQL commands for data definition and data manipulation.

COURSE OUTCOME (CO)

COURSE OUTCOMES	
CO1	Implementation of data models, modeling notations, SQL queries.
CO2	Design and implement a database schema for a given problem.
CO3	Design and build a simple database system considering the different concepts of RDBMS.

Course Content

Pgm No.	Content	Hours
1	**Introduction Lab** Basic Concepts of DBMS will be covered Like: ● Types of DBMS ● DBMS Languages ● SQL-Structured Query Language ● Basic Instructions given with respect to Lab ● Installation Procedure given to students with respect to MySQL Software ● CREATE, USE, INSERT, DESC and SELECT commands will be explained..	3
2	**Database Number -1 : Banking Database** Consider the following relations: **BRANCH** (branch name: varchar , City : Varchar , Asset : Varchar) **ACCOUNT** (account Number : Varchar, Branch name : Varchar, balance: float) **CUSTOMER** (customer name : Varchar, Street : Varchar , City : Varchar) **DEPOSITOR** (customer name : varchar, account number : varchar) **LOAN** (loan number : varchar, branch name : varchar, amount : integer) **BORROWER** (customer name : varchar, loan number : varchar) ● The tables will be created using CREATE, DESC and tuples will be inserted using INSERT commands and displayed using SELECT Commands. ● Primary Key and Foreign Key relationships will be explained through a demo.	3
3	**Database Number -1 : Banking Database (Continuation) Queries :** ● List the loan number from loan having amount 10000 with a specific branch name. ● List the loan number with an amount between 1000 and 10000. ● List the cname with substring. ● List the number of tuples in the customer. ● List customer name, loan num and amount with specific branch name. ● Various Aggregate functions will be used to retrieve the data from the above constructed databases. ● Various Comparison operators will be used to retrieve the data from the above constructed databases ● Various Logical operators will be used to retrieve the data from the above constructed databases	3

4	**Database Number -2 : Accessories Database** Consider the following relations: **PRODUCT** (maker: varchar, model : varchar, type: varchar) **PC** (model : varchar , speed : varchar, ram : varchar, hdd: varchar , removable disk : int , price : int) **LAPTOP** (model : varchar , speed : varchar, ram : varchar, hdd: varchar , screen: varchar, price : int) **PRINTER** (model: varchar, color : blob , type : varchar, price : int) ● The tables will be created using CREATE, DESC and tuples will be inserted using INSERT commands and displayed using SELECT Commands. ● Primary Key and Foreign Key relationships will be explained through a demo. ● Find the model,speed,RAM,HDD for all pc's, whose price is under 35000. ● Rename the speed column to mhz and HDD column to GB in pc. ● Find all manufacturers of laptops. ● Find all the tuples in the printer for color. ● Various Aggregate functions will be used to retrieve the data from the above constructed databases. ● Various Comparison operators will be used to retrieve the data from the above constructed databases ● Various Logical operators will be used to retrieve the data from the above constructed databases.	3
5	**Database Number -3 : Order Processing Database** Consider the following relations: **CUSTOMER** (Cust id : varchar, cust name : varchar, city : varchar) **ORDER** (Order num : varchar, order date : date, cust id : varchar , order amount : int) **ITEM** (item id : varchar , unit price : int) **ORDERITEM** (order num: varchar, item id: varchar, quantity : int) **WAREHOUSE** (warehouse id : varchar, city: varchar) **SHIPMENT** (order num: varchar, warehouse id : varchar, shipdate : date) ● The tables will be created using CREATE, DESC and tuples will be inserted using INSERT commands and displayed using SELECT Commands. ● Primary Key and Foreign Key relationships will be explained through a demo. ● Produce a list in customer name, number of orders, average order amount where the middle column is the total number of orders by the customers and the last column is average order amount for the third column. ● List the order number for orders that were shipped from all the warehouses that the company asked in a specific city. ● Various Aggregate functions will be used to retrieve the data from the above constructed databases. ● Various Comparison operators will be used to retrieve the data from the above constructed databases ● Various Logical operators will be used to retrieve the data from the above constructed databases.	3

6	**Database Number - 4 : Supplier Database** Consider the following relations: **SUPPLIER** (Supplier id : varchar, name : varchar , address : varchar) **PART** (part id : varchar, partname : varchar, color : tinyblob) **CATALOG** (supplier id : varchar, part id : varchar, cost : int) ● The tables will be created using CREATE, DESC and tuples will be inserted using INSERT commands and displayed using SELECT Commands. ● Database Number -4 : Supplier Database (Continuation) Queries ● Primary Key and Foreign Key relationships will be explained through a demo. ● Find the names of suppliers who supply only red parts. ● Find supplier id of suppliers who supply red and green parts. ● Find the supplier id of the supplier who supplies some red part OR whose address is 'Mysuru'. ● Find the supplier id of suppliers who supply some red and some green parts. ● Find the supplier id of suppliers who supply every part.	3
7	**Database Number -5 : University Database -** Consider the following relations: **CLASSROOM** (building varchar(30), roomno varchar(30) primary key, capacity int) **DEPARTMENT** (deptname varchar(30) primary key,building varchar(30),budget decimal(10,2) **COURSE**(courseid varchar(30) primary key, title varchar(30),deptname varchar(30), credits int) **INSTRUCTOR** (iid varchar(30) primary key, name varchar(30), deptname varchar(30),salary int) **STUDENT**(stdid varchar(30) primary key,name varchar(30),deptname varchar(30), totalcredit int) **TEACHES**(iid varchar(30),courseid varchar(30), secid varchar(30),sem varchar(30), year varchar(30)) **SECTION** (courseid varchar(30),secid varchar(30) primary key,sem varchar(30), year varchar(30),building varchar(30),roomno varchar(30),timeid varchar(30)) **TIMESLOT**(timeid varchar(30) primary key,day varchar(30), starttime time, endtime time); **PREREQUISITE** (courseid varchar(30),preid varchar(30)) **ADVISOR** (stdid varchar(30), iid varchar(30)) **TEXT**(iid varchar(30), courseid varchar(30),secid varchar(30),sem varchar(30), year varchar(30), grade varchar(30)) ● The tables will be created using CREATE, DESC and tuples will be inserted using INSERT commands and displayed using SELECT Commands. ● Primary Key and Foreign Key relationships will be explained through a demo.	4

8	**Database Number -7 : University Database (Queries) Continuation** ● Retrieve the names of the instructors and order them by the departmental name. ● Retrieve instructors with salaries between specified ranges: ○ Between 10,000 and 20,000 ○ Between 50,000 and 70,000 (specific to the MBA department) ● Find salary statistics for instructors: ○ Average salary for a particular department ○ Minimum salary for a department ○ Average salary grouped by department, where the average is greater than 50,000 ● Display instructor details: ○ Instructor teaching in a specific building, e.g., "Golden Jubilee Block"	5
9	**Database Number -7 : University Database (Queries) Continuation** ● Retrieve student details: ○ Students who have taken at least one MCA course ○ Students with the maximum CGPA ● Department and enrollment queries: ○ Total number of students in each department, section-wise ○ Enrollment of sections offered in a specific year (e.g., 2015 and odd 2019) ● Course details: ○ Delete courses that have never been offered ○ List all course titles of a specific instructor ○ Retrieve all course titles taught in specific semesters (odd/even 2018) ○ Create a view listing MCA course sections for even 2016	5
10	**Database Number -7 : University Database (Queries) Continuation** ● Instructor and department associations: ○ Department names for all instructors ○ Instructor names and details, including department and building name ○ Instructor names that include specific letters (e.g., 'ma') and the courses they teach ○ Instructor names in descending order of salary ○ Instructor names not matching specified names (e.g., "Charlie" and "Deepika") ● Credits and CGPA: ○ Total credits offered by each department ○ Total CGPA scored by students of each department ● Additional data manipulations: ○ Increase the salary of each instructor by 25,000 and list their names and IDs ● Room and course queries: ○ Courses belonging to the MBA department running in specific rooms during 2015	

Text Books

1. A. Silberschatz, Henry.F.Korth, S.Sudharshan, “Database System Concepts”, 7th Edition,2017

Reference Books

1. Raghu Ramakrishnan and J Gehrke,” Database Management Systems”, 3rd Edition, 2016.
2. C.J.Date, AKannan, S..Swamynathan ,”An Introduction to Database System”, 8th Edition,2009.
3. RamezElmasri, Shamkant.B.Navathe, “Database Systems”, 7th Edition,2016.

SEMESTER II

<table>
<tr><td><b><i>Course Title</i></b></td><td colspan="8"><b>Analysis and Design of Algorithms</b></td></tr>
<tr><td rowspan="2"><b><i>Course Type</i></b></td><td rowspan="2">Soft Core</td><td rowspan="2"><b><i>Total Hours</i></b></td><td rowspan="2">52</td><td rowspan="2" colspan="2">Hours / Week</td><td rowspan="2">04</td><td><b><i>Credits</i></b></td><td>04</td></tr>
<tr><td><b><i>(L : T : P)</i></b></td><td>(2 : 1 : 1)</td></tr>
<tr><td rowspan="2"><b><i>Course Code</i></b></td><td rowspan="2"></td><td rowspan="2"><b><i>Evaluation</i></b></td><td><b><i>Internal</i></b></td><td colspan="3">C1 + C2 = 15 + 15</td><td>30 Marks</td><td rowspan="2">100 Marks</td></tr>
<tr><td><b><i>External</i></b></td><td>C3</td><td><b><i>Duration</i></b></td><td>03 Hours</td><td>70 Marks</td></tr>
</table>

COURSE PREREQUISITE

1. Basic understanding of programming concepts
2. Familiarity with basic mathematical concepts, such as logic and set theory.
3. Problem Solving and Analytical skills
4. Basic programming Skills

COURSE OBJECTIVE: This course aims to enable learners:

1. Describe computational solutions to well known problems like searching, sorting etc.
2. Estimate the computational complexity of different algorithms.
3. Devise an algorithm using appropriate design strategies for problem solving.
4. Gain the ability to analyze and apply data structures and Algorithms in real-world scenarios, focusing on efficiency and Optimization.

COURSE OUTCOME (CO)

COURSE OUTCOMES	
CO1	Describe computational solutions to well known problems like searching, sorting etc.
CO2	Implement abstract data types (ADT) such as arrays, strings, stacks, and queues to solve algorithmic problems.
CO3	Estimate the computational complexity of different algorithms.
CO4	Devise an algorithm using appropriate design strategies for problem solving.
CO5	Gain the ability to analyze and apply data structures and Algorithms in real-world scenarios, focusing on efficiency and Optimization.

Course Content

Unit No	Content	Hours
1	**UNIT I - Introduction and the fundamentals of the Analysis of Algorithm** Efficiency Notion of Algorithm, Fundamentals of Algorithmic Problem Solving, Important Problem Types, Fundamental data Structures. Analysis Framework, Asymptotic Notations and Basic efficiency classes, Mathematical analysis of Recursive and Non-recursive algorithms With Examples.	12
2	**UNIT -II- Brute Force and Divide and Conquer:** Selection Sort and Bubble Sort, Sequential Search and String Matching, Exhaustive Search, Merge- sort, Quick-sort, Binary Search, Binary tree Traversals and related properties.	10
3	**UNIT-III- Decrease-and-Conquer, Transform-and-Conquer** Insertion Sort, Depth First search and Breadth First Search, Topological sorting, Algorithms for Generating Combinatorial Objects. Presorting, Balanced Search Trees, Heaps and Heap sort, Problem Reduction.	10
4	**UNIT-IV - Space and Time Trade offs and Dynamic Programming** Sorting by Counting, Input Enhancement in String Matching, Computing a binomial coefficient, Warshall’s and Floyd’s Algorithms, The Knapsack Problem and Memory Functions.	10
5	**UNIT V - Limitations of Algorithm Power** Lower-Bound Arguments, Decision Trees, P, NP and NP-Complete Problems. Coping with Limitations of Algorithm Power: Backtracking: n-Queens problem, Hamiltonian Circuit Problem, Subset – Sum Problem. Branch-and- Bound: Assignment Problem, Knapsack Problem, Traveling Salesperson Problem.	10

Reference Books

1. Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2rd Edition, 2009. Pearson.
2. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 3rd Edition, PHI.
3. Design and Analysis of Algorithms , S. Sridhar, Oxford (Higher Education).

SEMESTER II

<table>
<tr><td><b><i>Course Title</i></b></td><td colspan="8"><b>Foundations of Cyber Security</b></td></tr>
<tr><td rowspan="2"><b><i>Course Type</i></b></td><td rowspan="2">Soft Core</td><td rowspan="2"><b><i>Total Hours</i></b></td><td rowspan="2">52</td><td rowspan="2" colspan="2">Hours / Week</td><td rowspan="2">04</td><td><b><i>Credits</i></b></td><td>04</td></tr>
<tr><td><b><i>(L : T : P)</i></b></td><td>(3 : 1 : 0)</td></tr>
<tr><td rowspan="2"><b><i>Course Code</i></b></td><td rowspan="2"></td><td rowspan="2"><b><i>Evaluation</i></b></td><td><b><i>Internal</i></b></td><td colspan="3">C1 + C2 = 15 + 15</td><td>30 Marks</td><td rowspan="2">100 Marks</td></tr>
<tr><td><b><i>External</i></b></td><td>C3</td><td><b><i>Duration</i></b></td><td>03 Hours</td><td>70 Marks</td></tr>
</table>

COURSE PREREQUISITE

1. Basic Understanding of Computer Hardware and Software
2. Familiarity with Operating Systems
3. Familiarity of Computer networks and various protocols with their applications

COURSE OBJECTIVE: This course aims to enable learners:

1. Understand the fundamentals of the cybersecurity domain and related issues.
2. Understand the interdisciplinary nature of cyber security domain
3. Knowledge about various Cyber laws and regulations
4. Illustrate Tools and Methods used on Cybercrime , Understanding about Phishing and Identity Theft.
5. Justify the need of computer forensics.

COURSE OUTCOME (CO): At the end of the course, the student will be able to

COURSE OUTCOMES	
CO1	Understand the fundamentals of the cyber security domain and related issues.
CO2	Describe Cyber offenses and Botnets
CO3	Knowledge about various Cyber laws and regulations
CO4	Illustrate Tools and Methods used on Cybercrime , Understanding about Phishing and Identity Theft.
CO5	Justify the need of computer forensics

Course Content

Unit No	Content	Hours
1	**UNIT-I- Introduction to Cybercrime:** Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Who are Cybercriminals? Classifications of Cyber Crimes, Cybercrime: The Legal Perspectives, Cybercrimes: Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes, Cybercrime Era: Survival Mantra for the Netizens. Cyber offenses: How Criminals Plan Them: How Criminals Plan the Attacks, Social Cyber stalking Cyber cafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector.	12
2	**UNIT-II- Cybercrime: Mobile and Wireless Devices:** Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era.	10
3	**UNIT-III- Tools and Methods Used in Cybercrime:** Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks. Phishing and Identity Theft: Introduction, Phishing, Identity Theft (ID Theft).	10
4	**UNIT-IV- Understanding Computer Forensics:** Introduction, Historical Background of Cyber forensics, Digital Forensics Science, The Need for Computer forensics , Cyber Forensics and Digital Evidence, Digital Forensics Life Cycle, Chain of Custody Concept, Network Forensics, Approaching a Computer Forensics Investigation, Understanding the Requirements, Relevance of the OSI 7 Layer Model to Computer Forensics, Computer Forensics from Compliance Perspective, Challenges in Computer Forensics, Special Tools and Techniques, Forensics Auditing, Anti forensics.	10

5	**UNIT-V- Cybercrime and Cyberterrorism:** Social, Political, Ethical and Psychological Dimensions: Introduction, Intellectual Property in the Cyberspace, The Ethical Dimension of Cybercrimes, The Psychology, Mindset and Skills of Hackers and Other Cybercriminals, Sociology of Cybercriminals, Information Warfare: Perception or an Eminent Reality? Cybercrime: Illustrations, Examples and Mini-Cases: Introduction, Real Life Examples, Example 1: Official Website of Maharashtra Government Hacked, Example 2: Email Spoofing instances, Example 3: Email Bombing involving a foreigner. Mini Cases: Mini Case 1: Indian Cyber Defamation case of a young couple, Mini Case 2: Internet Time Stealing, Mini Case 3: Mini Case of Indian case of online gambling. Illustrations of Financial Frauds in Cyber Domain, Digital Signature-Related Crime Scenarios.	10

Reference Books

1. Fundamentals of Cyber Security (Principles Theory & Practices), Bhushan Mayank, Publisher: BPB Publications, 2020.
2. Cyber security , Nina Godbole , Sunit Belapure , Wiley India Pvt Ltd , Reprint 2023
3. Cyber Security Fundamentals, Rajesh Kumar Gautam, BPB Publications, 2019.
4. Mowbray, Copyright © 2014 by John Wiley & Sons, Inc, ISBN: 91-118 - 84965 -1. Dreamtech Press. ISBN: 9789351194736, 2015 Martti Lehto, Pekka Neittaanmäki, "Cyber Security: Analytics, Technology and Automation edited" Springer International Publishing Switzerland, 2015.

SEMESTER II

<table>
<tr><td><b><i>Course Title</i></b></td><td colspan="8"><b>Web Technologies</b></td></tr>
<tr><td rowspan="2"><b><i>Course Type</i></b></td><td rowspan="2">Soft Core</td><td rowspan="2"><b><i>Total Hours</i></b></td><td rowspan="2">52</td><td colspan="2" rowspan="2">Hours / Week</td><td rowspan="2">04</td><td><b><i>Credits</i></b></td><td>04</td></tr>
<tr><td><b><i>(L : T : P)</i></b></td><td>(3 : 1 : 0)</td></tr>
<tr><td rowspan="2"><b><i>Course Code</i></b></td><td rowspan="2"></td><td rowspan="2"><b><i>Evaluation</i></b></td><td><b><i>Internal</i></b></td><td colspan="3">C1 + C2 = 15 + 15</td><td>30 Marks</td><td rowspan="2">100 Marks</td></tr>
<tr><td><b><i>External</i></b></td><td>C3</td><td><b><i>Duration</i></b></td><td>03 Hours</td><td>70 Marks</td></tr>
</table>

COURSE PREREQUISITE: C++/Java/ Python Programming

COURSE OBJECTIVE: This course aims to enable learners:

1. To learn HTML tags and JavaScript Language programming concepts and techniques.
2. To develop the ability to logically plan and develop web pages.
3. To learn to write, test, and debug web pages using HTML and JavaScript.
4. To understand the concept of XAMPP server and node js.
5. To Implement basics of MEAN and FULL stack.

COURSE OUTCOME (CO): At the end of the course, the student will be able to

COURSE OUTCOMES	
CO1	Apply the concepts and usage of web based programming techniques.
CO2	Demonstrate the development of XHTML documents using JavaScript and CSS
CO3	Demonstrate applications of AngularJs and JQuery for the given problem.
CO4	Design and implement user interactive dynamic web based applications.
CO5	Implement modern web applications using MEAN and FULL STACK.

Course Content

Unit No	Content	Hours
1	**OVERVIEW OF WEB TECHNOLOGIES:** HTML5 and CSS: Web browsers, web servers, MIME, HTTP Introduction to XHTML5 tags, Basic syntax and structure, text markups, images, lists, tables, Media tags-audio and video, forms, frames. CSS: Introduction to CSS, Levels of CSS, Selectors, Font, color and Text Properties, BOX Model, Span and Div tags. Introduction to JavaScript, controls statements, Arrays and functions, pattern matching, Element Access, Event Handling.	10
2	**XML and AJAX:** XML-Documents and Vocabularies-Versions and Declaration - Namespaces JavaScript and XML: Ajax-DOM based XML processing Event-oriented Parsing: SAX-Transforming XML Documents Selecting XML Data: XPATH-Template based Transformations: XSLT-Displaying XML Documents in Browsers - Evolution of AJAX -Web applications with AJAX -AJAX Framework.	10
3	**CLIENT-SIDE SCRIPTING :** Introduction to JQuery, Syntax, selectors, events, JQuery HTML, JQuery Effects, JQuery CSS. Introduction to Angular JS, Directives, Expressions, Directives, Controllers, Filters, Services, Events, Forms, Validations, Examples. Bootstrap: Introduction to Bootstrap, First example, containers, Bootstrap elements: colors, tables, images, buttons, button groups, progress bars, Forms, utilities, Classes, alerts, custom forms, Grid System.	12
4	**SERVER-SIDE SCRIPTING:** Essentials of PHP Installation of Web Server, XAMPP Configurations-PHP Forms- GET and POST method - Regular Expressions-Cookies-Sessions- Usage of Include and require statements- File: read and write from the file-PHP Filters-PHP XML Parser-Introduction to Node.js-Node.js Modules and filesystem-Node.js Events.	10
5	**MySQL and MEAN STACK:** PHP with MySQL- Performing basic database operation (DML) (Insert, Delete, Update, Select)-Prepared Statement- Uploading Image or File to MySQL- Retrieve Image or File from MySQL- Uploading Multiple Files to MySQL-Introduction to MEAN and FULL Stack-Real time example for modern web applications using MEAN-MEAN vs Full Stack.	10

Reference Books.

1. Christopher Murphy, Richardclark, OliStudholme, DivysManian, “Beginning HTML5 and CSS3”, Apress Publication,2012.
2. Grant, Andrew, “Beginning AngularJS”, Apress Publication,2014.
3. Matt Doyle, “Beginning PHP 5.3”, Wiley Publisher,2010.
4. Chris Bates, “Web Programming Building Internet Applications”, Wiley India, 3rd Edition, 2007.
5. Glenn J. Myatt, Making Sense of Data, John Wiley & Sons, Pete Warden, Big Data Glossary, O‟Reilly,2016.