



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES

(AUTONOMOUS)

**Dr. Visweswaraiiah Road, (Bangalore-Tirupathi Bye-pass Road), Murukambattu,
Chittoor – 517127, Andhra Pradesh, India.**

B.Tech Course Structures and Syllabi Under R23 Regulations

(Applicable for 2023-2024 Regular Students & 2024-2025 Lateral Students)

**Department of
Computer Science and Engineering**



B.Tech- R23 - COURSE STRUCTURE AND SYLLABI

Semester III (Second Year)

S.No	Course Code	Course Title	Course Category	Scheme of Instructions Hours per Week				Scheme of Examination Maximum Marks			
				L	T	P	C	I	E	Total	
1	23MBA24xa/ 23MBA24xb/ 23MBA24xc	Managerial Economics and Financial Analysis / Business Environment / Organizational Behavior	MBA	2	0	0	2	30	70	100	
2	23BSC23x	Discrete Mathematics & Graph Theory	BSC	3	0	0	3	30	70	100	
3	23ECE23x	Digital Logic & Computer Organization	ECE	3	0	0	3	30	70	100	
4	23CSE231	Advanced Data Structures & Algorithm Analysis	PCC	3	0	0	3	30	70	100	
5	23CSE232	Object Oriented Programming Through Java	PCC	3	0	0	3	30	70	100	
6	23CSE233	Advanced Data Structures and Algorithm Analysis Lab	PCC	0	0	3	1.5	30	70	100	
7	23CSE234	Object Oriented Programming Through Java Lab	PCC	0	0	3	1.5	30	70	100	
8	23CSE235	Python Programming	SEC	0	1	2	2	30	70	100	
9	23MAC23x	Environmental Science	MAC	2	0	0	-	P	-	-	
Contact Hours per week				16	1	8	19	-	-	-	
Total Hours per week				25				-	-	-	
Total credits								19	-	-	-
Total Marks								240	560	800	

Semester IV (Second Year)

S.No	Course Code	Course Title	Course Category	Scheme of Instructions Hours per Week				Scheme of Examination Maximum Marks			
				L	T	P	C	I	E	Total	
1	23BSC24x	Probability & Statistics	BSC	3	0	0	3	30	70	100	
2	23HSM24X	Universal Human Values- Understanding Harmony & Human Ethical Conduct	HSMC	2	1	0	3	30	70	100	
3	23CSE241	Operating Systems	PCC	3	0	0	3	30	70	100	
4	23CSE242	Database Management Systems	PCC	3	0	0	3	30	70	100	
5	23CSE243	Software Engineering	PCC	3	0	0	3	30	70	100	
6	23CSE244	Operating Systems Lab	PCC	0	0	3	1.5	30	70	100	
7	23CSE245	Database Management Systems Lab	PCC	0	0	3	1.5	30	70	100	
8	23CSE246	Full Stack Development -I	SEC	0	1	2	2	30	70	100	
9	23MECxxx	Design Thinking & Innovation	BSH	1	0	2	2	30	70	100	
Contact Hours per week				15	2	10	22	-	-	-	
Total Hours per week				27				-	-	-	
Total credits								22	-	-	-
Total Marks								270	630	900	
Mandatory community service project internship of 08 weeks duration during summer vacation											



II B.TECH - III SEMESTER

23CSE231 ADVANCED DATA STRUCTURES & ALGORITHM ANALYSIS **L T P C**
(Common to CSE, CSM, CAI and CSD) **3 0 0 3**

PRE-REQUISITES: A course on Data Structures

COURSE EDUCATIONAL OBJECTIVES:

1. Provide knowledge about algorithm analysis using time complexity and space complexity, and to introduce the concept of AVL trees and B trees.
2. To understand the concepts of Heap trees, Basics of Graphs and Divide and Conquer algorithm design strategy.
3. To gain knowledge of Greedy and Dynamic programming approach to problem solving.
4. To Understand back tracking and branch and bound strategies for solving the complex problems
5. To Develop skills of deciding a problem is solvable or not using NP hard and NP complete class problem.

UNIT 1: (9)

Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations. AVL Trees – Creation, Insertion, Deletion operations and Applications B-Trees – Creation, Insertion, Deletion operations and Applications

UNIT 2: (9)

Heap Trees (Priority Queues) – Min and Max Heaps, Operations and Applications Graphs – Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications. Divide and Conquer: The General Method, Quick Sort, Merge Sort, Strassen’s matrix multiplication, Convex Hull

UNIT 3: (9)

Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths. Dynamic Programming: General Method, All pairs shortest paths, Single Source Shortest Paths– General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees, 0/1 Knapsack, String Editing, Travelling Salesperson problem

UNIT 4: (9)

Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, Graph Coloring, 0/1 Knapsack Problem
Branch and Bound: The General Method, 0/1 Knapsack Problem, Travelling Salesperson problem

UNIT 5: (9)

NP Hard and NP Complete Problems: Basic Concepts, Cook’s theorem NP Hard Graph Problems: Clique Decision Problem (CDP), Chromatic Number Decision Problem (CNDP), Traveling Salesperson Decision Problem (TSP) NP Hard Scheduling Problems: Scheduling Identical Processors, Job Shop Scheduling

Total Hours: 45



II B.TECH - III SEMESTER

23CSE232	OBJECT ORIENTED PROGRAMMING THROUGH JAVA	L T P C
	(Common to CSE, CSM, CAI, CSD)	3 0 0 3

PRE-REQUISITES: A course on Introduction to Programming

COURSE EDUCATIONAL OBJECTIVES:

1. Identify Java language components and how they work together in applications
2. Learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
3. learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
4. understand how to design applications with threads in Java
5. understand how to use Java APIs for program development

UNIT 1: **(9)**

Object Oriented Programming: Basic concepts, Principles,

Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.

Data Types, Variables, and Operators : Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final,

Introduction to Operators, Precedence and Associativity of Operators, Assignment Operator (=), Basic Arithmetic Operators, Increment (++) and Decrement (--) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.

Control Statements: Introduction, if Expression, Nested if Expressions, if-else Expressions, Ternary Operator (?:), Switch Statement, Iteration Statements, while Expression, do-while Loop, for Loop, Nested for Loop, For-Each for Loop, Break Statement, Continue Statement.

UNIT 2: **(9)**

Classes and Objects: Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

Methods: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static.

UNIT 3: **(9)**

Arrays: Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three-dimensional Arrays, Arrays as Vectors.



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

Inheritance: Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance.

Interfaces: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

UNIT 4: (9)

Packages and Java Library: Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, Temporal Adjusters Class, Temporal Adjusters Class.

Exception Handling: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions.

Java I/O and File: Java I/O API, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java(Text Book 2)

UNIT 5: (9)

String Handling in Java: Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer.

Multithreaded Programming: Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread Priority-Synchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

Java Database Connectivity: Introduction, JDBC Architecture, Installing MySQL and MySQL Connector/J, JDBC Environment Setup, Establishing JDBC Database Connections, ResultSet Interface

Java FX GUI: Java FX Scene Builder, Java FX App Window Structure, displaying text and image, event handling, laying out nodes in scene graph, mouse events (Text Book 3)

Total Hours: 45

COURSE OUTCOMES:

On successful completion of the course the student will be		POs related to Cos
CO1	Analyse problems, design solutions using OOP principles, and implement them efficiently in Java. (L4)	PO1, PO2, PO5
CO2	Design and implement classes to model real-world entities, with a focus on attributes, behaviours, and relationships between objects (L4)	PO1, PO4,PO5
CO3	Demonstrate an understanding of inheritance hierarchies and polymorphic behaviour, including method overriding and dynamic method dispatch. (L3)	PO1,PO3,PO4, PO5



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

CO4	Apply Competence in handling exceptions and errors to write robust and fault-tolerant code. (L3)	PO1,PO4, PO5
CO5	Perform file input/output operations, including reading from and writing to files using Java I/O classes, graphical user interface (GUI) programming using JavaFX.(L3)	PO1, PO2, PO4, PO5

TEXT BOOKS:

- 1) JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
- 2) Joy with JAVA, Fundamentals of Object Oriented Programming, DebasisSamanta, MonalisaSarma, Cambridge, 2023.
- 3) JAVA for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

REFERENCES:

- 1) The complete Reference Java, 11thedition, Herbert Schildt,TMH
- 2) Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson

REFERENCE WEBSITE:

- 1) <https://nptel.ac.in/courses/106/105/106105191/>
- 2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview

CO-PO MAPPING:

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	2	-	-	-	-	-	-	-
CO2	2	-	-	3	3	-	-	-	-	-	-	-
CO3	3	-	3	3	3	-	-	-	-	-	-	-
CO4	3	-	-	3	3	-	-	-	-	-	-	-
CO5	2	3	-	3	3	-	-	-	-	-	-	-
CO*	2.6	3	3	3	2.8	-	-	-	-	-	-	-



23CSE233	II B.TECH - III SEMESTER ADVANCED DATA STRUCTURES & ALGORITHM ANALYSIS LAB	L T P C
	(Common to CSE, CSM and CAI)	0 0 0 1.5

PRE-REQUISITES: A course on Data Structures

COURSE EDUCATIONAL OBJECTIVES:

- Acquire practical skills in constructing and managing Data structures
- Apply the popular algorithm design methods in problem-solving scenarios

List of Demo/ Experiments (Only for Skill Enhancement, Not for Exams) (15)

Experiments covering the Topics:

- Operations on AVL trees, B-Trees, Heap Trees
- Graph Traversals
- Sorting techniques
- Finding Biconnected components in a graph
- Shortest path algorithms using greedy Method
- 0/1 Knapsack Problem using Dynamic Programming and Backtracking
- Travelling Salesperson problem using Branch and Bound
- N-Queens Problem using Backtracking
- Job Sequencing using Branch and Bound

Sample Programs:

1. Construct an AVL tree for a given set of elements which are stored in a file. And implement insert and delete operation on the constructed tree. Write contents of tree into a new file using in-order.
2. Construct B-Tree an order of 5 with a set of 100 random elements stored in array. Implement searching, insertion and deletion operations.
3. Construct Min and Max Heap using arrays, delete any element and display the content of the Heap.
4. Implement BFT and DFT for given graph, when graph is represented by
 - a) Adjacency Matrix
 - b) Adjacency Lists
5. Write a program for finding the biconnected components in a given graph.
6. Implement Quick sort and Merge sort and observe the execution time for various input sizes (Average, Worst and Best cases).
7. Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists.
8. Implement Job Sequencing with deadlines using Greedy strategy.
9. Write a program to solve 0/1 Knapsack problem Using Dynamic Programming.
10. Implement N-Queens Problem Using Backtracking.
11. Use Backtracking strategy to solve 0/1 Knapsack problem.
12. Implement Travelling Sales Person problem using Branch and Bound approach.



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

COURSE OUTCOMES:

On successful completion of the course the student will be		POs related to COs
CO1	Acquire the Knowledge on structure and model of the sorting techniques.	PO1
CO2	Analyze the Time and space complexity.	PO2
CO3	Design solutions for user requirements using software functionality.	PO3
CO4	Investigate on different dynamic programming in developing TSP	PO4
CO5	Develop the the N-queens problem using backtracking algorithm	PO5, PO6
CO6	Follow the ethical principles in implementing the programs	PO8
CO7	Do experiments effectively as an individual and as a team member in a group.	PO9
CO8	Communicate verbally and in written form, the understanding about the experiments.	PO10
CO9	Continue updating their skill related to object oriented concepts and implementing programs in future.	PO12

REFERENCE BOOKS:

1. Fundamentals of Data Structures in C++, Horowitz Ellis, SahniSartaj, Mehta, Dinesh, 2ndEdition, Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, 2ndEdition, University Press
3. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
4. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill

REFERENCE WEBSITE:

1. <http://cse01-iiith.vlabs.ac.in/>
2. <http://peterindia.net/Algorithms.html>

CO-PO MAPPING:

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	-	-	-	-	-	-	-	-
CO2	-	3	-	-	-	-	-	-	-	-	-	-
CO3	-	-	3	-	-	-	-	-	-	-	-	-
CO4	-	-	-	3	-	-	-	-	-	-	-	-
CO5	-	-	-	-	3	1	-	-	-	-	-	-
CO6	-	-	-	-	-	-	-	3	-	-	-	-
CO7	-	-	-	-	-	-	-	-	3	-	-	-
CO8	-	-	-	-	-	-	-	-	-	3	-	-
CO9	-	-	-	-	-	-	-	-	-	-	-	3
CO*	3	3	3	3	3	1	-	3	3	3	-	3



II B.TECH - III SEMESTER

23CE234 OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB L T P C
(Common to CSE, CSM, CAI, CSD) 0 0 3 1.5

PRE-REQUISITES: A course on Introduction to Programming

COURSE EDUCATIONAL OBJECTIVES:

1. Practice object-oriented programming in the Java programming language
2. Implement Classes, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User defined Exception handling mechanism
3. Illustrate inheritance, Exception handling mechanism, JDBC connectivity
4. Construct Threads, Event Handling, implement packages, Java FX GUI

EXPERIMENTS COVERING THE TOPICS:

- Object Oriented Programming fundamentals- data types, control structures
- Classes, methods, objects, Inheritance, polymorphism,
- Exception handling, Threads, Packages, Interfaces
- Files, I/O streams, JavaFX GUI

LIST OF EXPERIMENTS:

EXERCISE – 1:

- a) Write a JAVA program to display default value of all primitive data type of JAVA
- b) Write a java program that display the roots of a quadratic equation $ax^2+bx=0$. Calculate the discriminate D and basing on value of D, describe the nature of root.

EXERCISE - 2

- a) Write a JAVA program to search for an element in a given list of elements using binary search mechanism.
- b) Write a JAVA program to sort for an element in a given list of elements using bubble sort
- c) Write a JAVA program using String Buffer to delete, remove character.

EXERCISE - 3

- a) Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside main method.
- b) Write a JAVA program implement method overloading.
- c) Write a JAVA program to implement constructor.
- d) Write a JAVA program to implement constructor overloading.

EXERCISE - 4

- a) Write a JAVA program to implement Single Inheritance
- b) Write a JAVA program to implement multilevel Inheritance
- c) Write a JAVA program for abstract class to find areas of different shapes

EXERCISE - 5

- a) Write a JAVA program give example for "super" keyword.
- b) Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?
- c) Write a JAVA program that implements Runtime polymorphism

EXERCISE - 6

- a) Write a JAVA program that describes exception handling mechanism
- b) Write a JAVA program Illustrating Multiple catch clauses
- c) Write a JAVA program for creation of Java Built-in Exceptions
- d) Write a JAVA program for creation of User Defined Exception

EXERCISE - 7

- a) Write a JAVA program that creates threads by extending Thread class. First thread display "Good Morning "every 1 sec, the second thread displays "Hello "every 2 seconds and



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

the third display "Welcome" every 3 seconds,(Repeat the same by implementing Runnable)

- b) Write a program illustrating **is Alive** and **join ()**
- c) Write a Program illustrating Daemon Threads.
- d) Write a JAVA program Producer Consumer Problem

EXERCISE – 8

- a) Write a JAVA program that import and use the user defined packages
- b) Without writing any code, build a GUI that display text in label and image in an ImageView (use JavaFX)
- c) Build a Tip Calculator app using several JavaFX components and learn how to respond to user interactions with the GUI

COURSE OUTCOMES:

On successful completion of the course the student will be		POs related to COs
CO1	Acquire the Knowledge on structure and model of the Java programming language.	PO1
CO2	Analyze the complex examples using java programming language.	PO2
CO3	Design solutions for user requirements using software functionality.	PO3
CO4	Investigate on event handling concepts in developing the object oriented programming	PO4
CO5	Develop the data base connectivity to the Java application	PO5
CO6	Follow the ethical principles in implementing the programs	PO8
CO7	Do experiments effectively as an individual and as a team member in a group.	PO9
CO8	Communicate verbally and in written form, the understanding about the experiments.	PO10
CO9	Continue updating their skill related to object oriented concepts and implementing programs in future.	PO12

REFERENCE BOOKS:

1. P. J. Deitel, H. M. Deitel, "Java for Programmers", Pearson Education, PHI, 4th Edition, 2007.
2. P. Radha Krishna, "Object Oriented Programming through Java", Universities Press, 2nd Edition, 2007
3. Bruce Eckel, "Thinking in Java", Pearson Education, 4th Edition, 2006.
4. Sachin Malhotra, Saurabh Chaudhary, "Programming in Java", Oxford University Press, 5th Edition, 2010.

REFERENCE WEBSITE:

- <https://java-iitd.vlabs.ac.in/>
- <http://peterindia.net/JavaFiles.html>

CO-PO MAPPING:

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	-	-	-	-	-	-	-	-
CO2	-	3	-	-	-	-	-	-	-	-	-	-
CO3	-	-	3	-	-	-	-	-	-	-	-	-
CO4	-	-	-	3	-	-	-	-	-	-	-	-
CO5	-	-	-	-	3	-	-	-	-	-	-	-
CO6	-	-	-	-	-	-	-	3	-	-	-	-
CO7	-	-	-	-	-	-	-	-	3	-	-	-
CO8	-	-	-	-	-	-	-	-	-	3	-	-
CO9	-	-	-	-	-	-	-	-	-	-	-	3
CO*	3	3	3	3	3	-	-	3	3	3	-	3



II B.TECH - III SEMESTER

23CSE235

PYTHON PROGRAMMING
(SKILL ENHANCEMENT COURSE)
(Common to ALL)

L T P C

0 1 2 2

PRE-REQUISITES: A course on Introduction to Programming

COURSE EDUCATIONAL OBJECTIVES:

1. Introduce core programming concepts of Python programming language.
2. Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries
3. Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these

UNIT-I:

History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook.

Parts of Python Programming Language: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, the type () Function and Is Operator, Dynamic and Strongly Typed Language.

Control Flow Statements: if statement, if-else statement, if...elif...else, Nested if statement, while Loop, for Loop, continue and break Statements, Catching Exceptions Using try and except Statement.

SAMPLE EXPERIMENTS:

1. Write a program to find the largest element among three Numbers.
2. Write a Program to display all prime numbers within an interval
3. Write a program to swap two numbers without using a temporary variable.
4. Demonstrate the following Operators in Python with suitable examples.
 - i) Arithmetic Operators
 - ii) Relational Operators
 - iii) Assignment Operators
 - iv) Logical Operators
 - v) Bit wise Operators
 - vi) Ternary Operator
 - vii) Membership Operators
 - viii) Identity Operators
5. Write a program to add and multiply complex numbers
6. Write a program to print multiplication table of a given number.

UNIT-II:

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.

SAMPLE EXPERIMENTS:

7. Write a program to define a function with multiple return values.
8. Write a program to define a function using default arguments.
9. Write a program to find the length of the string without using any library functions.
10. Write a program to check if the substring is present in a given string or not.
11. Write a program to perform the given operations on a list:
 - i. addition
 - ii. insertion
 - iii. slicing
12. Write a program to perform any 5 built-in functions by taking any list.

UNIT-III:

Dictionaries: Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

Tuples and Sets: Creating Tuples, Basic Tuple Operations, tuple() Function, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozenset.

Sample Experiments:

13. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
14. Write a program to count the number of vowels in a string (No control flow allowed).
15. Write a program to check if a given key exists in a dictionary or not.
16. Write a program to add a new key-value pair to an existing dictionary.
17. Write a program to sum all the items in a given dictionary.

UNIT-IV:

Files: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules.

Object-Oriented Programming: Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.

SAMPLE EXPERIMENTS:

18. Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper-case words from source must be lowered.
19. Python program to print each line of a file in reverse order.
20. Python program to compute the number of characters, words and lines in a file.
21. Write a program to create, display, append, insert and reverse the order of the items in the array.
22. Write a program to add, transpose and multiply two matrices.
23. Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.

UNIT-V:

Introduction to Data Science: Functional Programming, JSON and XML in Python, NumPy with Python, Pandas.

SAMPLE EXPERIMENTS:

24. Python program to check whether a JSON string contains complex object or not.
25. Python Program to demonstrate NumPy arrays creation using array () function.
26. Python program to demonstrate use of ndim, shape, size, dtype.
27. Python program to demonstrate basic slicing, integer and Boolean indexing.
28. Python program to find min, max, sum, cumulative sum of array
29. Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows:
 - a) Apply head () function to the pandas data frame
 - b) Perform various data selection operations on Data Frame
30. Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

COURSE OUTCOMES:

On successful completion of this course the students should be able to:		POs related to COs
CO1	Learn various problem solving approaches and ability to identify an appropriate approach to solve the problem	PO1, PO2, PO3, PO5
CO2	Implement conditionals and loops to design the python programming	PO1, PO2, PO3, PO5
CO3	Implement lists, set, tuples and dictionaries to develop python program.	PO1, PO2, PO3, PO5
CO4	Able to modulate the given problem using structural approach of programming	PO1, PO2, PO3, PO5
CO5	Build Python Programs using packages to solve real-time problems.	PO1, PO2, PO3, PO4, PO5
CO6	Follow the ethical principles in implementing the programs	PO8
CO7	Do experiments effectively as an individual and as a team member in a group.	PO9
CO8	Communicate verbally and in written form, the understanding about the experiments.	PO10
CO9	Continue updating their skill related to lists, tuples and dictionaries implementing programs in future.	PO12

REFERENCE BOOKS:

1. Gowrishankar S, Veena A., Introduction to Python Programming, CRC Press.
2. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2nd Edition, Pearson, 2024
3. Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

REFERENCE WEBSITE:

1. <https://www.coursera.org/learn/python-for-applied-data-science-ai>
2. <https://www.coursera.org/learn/python?specialization=python#syllabus>

CO-PO MAPPING:

CO-PO	PO 1	PO 2	PO 3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	-	2	-	-	-	-	-	-	-
CO2	3	3	3	-	2	-	-	-	-	-	-	-
CO3	3	3	3	-	2	-	-	-	-	-	-	-
CO4	3	3	3	-	2	-	-	-	-	-	-	-
CO5	3	3	3	3	2	-	-	-	-	-	-	-
CO6	-	-	-	-	-	-	-	3	-	-	-	-
CO7	-	-	-	-	-	-	-	-	3	-	-	-
CO8	-	-	-	-	-	-	-	-	-	3	-	-
CO9	-	-	-	-	-	-	-	-	-	-	-	3
CO*	3	3	3	3	2	-	-	3	3	3	-	3



II B.TECH. - IV SEMESTER

23CSE241

OPERATING SYSTEMS

L T P C
3 0 0 3

PRE-REQUISITES: A course on Introduction to Programming

COURSE EDUCATIONAL OBJECTIVES:

1. Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection.
2. Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
3. Illustrate different conditions for deadlock and their possible solutions.

UNIT –1: OPERATING SYSTEMS OVERVIEW & SYSTEM STRUCTURES (9)

Operating Systems Overview: Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems.

System Structures: Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging.

UNIT –2: PROCESSES, THREADS AND CONCURRENCY & CPU SCHEDULING (9)

Processes: Process Concept, Process scheduling, Operations on processes, Inter-process communication.

Threads and Concurrency: Multithreading models, Thread libraries, Threading issues.

CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling.

UNIT III: SYNCHRONIZATION TOOLS & DEADLOCKS (9)

Synchronization Tools: The Critical Section Problem, Peterson’s Solution, Mutex Locks, Semaphores, Monitors, Classic problems of Synchronization.

Deadlocks: system Model, Deadlock characterization, Methods for handling Deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from Deadlock.

UNIT –IV: MEMORY-MANAGEMENT STRATEGIES & STORAGE MANAGEMENT (9)

Memory-Management Strategies: Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping.

Virtual Memory Management: Introduction, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing.

Storage Management: Overview of Mass Storage Structure, HDD Scheduling.

UNIT V: FILE SYSTEM& PROTECTION (9)

File System: File System Interface, File concept, Access methods, Directory Structure, File system Implementation, File-system structure, File-system Operations, Directory implementation, Allocation method, Free space management, File-System Internals: File System Mounting, Partitions and Mounting, File Sharing.

Protection: Goals of protection, Principles of protection, Protection Rings, Domain of protection, Access matrix.

Total Hours: 45



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

COURSE OUTCOMES:

On successful completion of the course the student will be		POs related to COs
CO1	Describe the basics of the operating systems, mechanisms of OS to handle processes, threads, and their communication. (L1)	PO1, PO2
CO2	Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection. (L2)	PO1, PO3, PO4
CO3	Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system. (L3)	PO1,PO2, PO3,PO4,PO5
CO4	Illustrate different conditions for deadlock and their possible solutions. (L2)	PO1, PO2, PO4
CO5	Analyze the memory management and its allocation policies. (L4)	PO1, PO4

TEXT BOOKS:

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018.
2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson , 2016

REFERENCE BOOKS:

1. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
2. Operating Systems: A Concept Based Approach, D.M Dhamdhere, 3rd Edition, McGraw- Hill, 2013

REFERENCE WEBSITE:

1. <https://nptel.ac.in/courses/106/106/106106144/>
2. <http://peterindia.net/OperatingSystems.html>

CO-PO MAPPING:

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	-	-	-
CO2	2	-	3	3	-	-	-	-	-	-	-	-
CO3	2	2	3	2	3	-	-	-	-	-	-	-
CO4	2	2	-	3	-	-	-	-	-	-	-	-
CO5	3	-	-	2	-	-	-	-	-	-	-	-
CO*	2.4	2	3	2.5	3	-	-	-	-	-	-	-



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

II B.Tech. - IV Semester

23CSE242

DATABASE MANGEMENT SYSTEM
(Common to CSE, CSM, CAI and CSD)

L T P C
3 0 0 3

PRE-REQUISITES: A course on Introduction to Programming

COURSE EDUCATIONAL OBJECTIVES:

1. Discuss the basic Database concepts and the applications, data models and ER Model.
2. Understand the Relational database design principles
3. Master the basics of SQL and construct queries using SQL.
4. Understand the Normalization process in Database Management System.
5. Familiar with the basic issues of transaction processing and concurrency control.

UNIT -1: INTRODUCTION TO DATABASE MANAGEMENT SYSTEM AND ENTITY RELATIONSHIP MODEL (9)

Database system - Characteristics (Database Vs File System) - Database Users - Advantages of Database systems - Database applications - Brief introduction of different Data Models - Concepts of Schema - Instance and data independence - Three tier schema architecture for data independence - Database system structure environment - Centralized and Client Server architecture for the database - Introduction to Entity Relationship Model - Representation of entities - Attributes - Entity set - Relationship - Relationship set - Constraints - Sub classes - super class - Inheritance- Specialization - Generalization using ER Diagrams.

UNIT -2: RELATIONAL MODEL (9)

Introduction to Relational model - Concepts of domain - Attribute - Tuple - Relation importance of null values - Constraints (Domain, Key constraints, integrity constraints) and their importance - Relational Algebra, Relational Calculus - BASIC SQL: Simple Database schema - Data Base Language - types- Table definitions (create, alter), different DML operations (insert, delete, update).

UNIT -3: INTRODUCTION TO STRUCTURED QUERY LANGUAGE (9)

Basic SQL querying (select and project) using where clause arithmetic & logical operations - SQL functions(Date and Time, Numeric, String conversion) - Creating tables with relationship, Implementation of key and integrity constraints - Nested queries, sub queries, grouping, aggregation, ordering - Implementation of different types of Joins, view (updatable and non-updatable) - Relational set operations.

UNIT -4: NORMALIZATION (9)

Purpose of Normalization and schema refinement - Concept of functional dependency - normal forms based on functional dependency - Lossless join and dependency preserving decomposition (1NF, 2NF and 3 NF), concept of surrogate key - Boyce-Codd normal form(BCNF) - MVD - Fourth normal form(4NF) - Fifth Normal Form (5NF).

UNIT -5: TRANSACTION CONCEPT AND INDEXING CONCEPTS (9)

Transaction State - ACID properties - Concurrent Executions - Serializability - Recoverability, Implementation of Isolation - Testing for Serializability - Lock based - Time stamp based optimistic - Concurrency protocols - Deadlocks - Failure Classification - Storage, Recovery and



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

Atomicity - Recovery algorithm - Introduction to Indexing Techniques - B+ Trees, operations on B+Trees - Hash Based Indexing

Total Hours: 45

COURSE OUTCOMES:

On successful completion of the course, students will be able to		Pos
CO1	Demonstrate knowledge on Data models and Database Languages and Design Entity Relationship model for a database	PO1, PO3
CO2	Analyze the relational database theory, and be able to write relational algebra and relational calculus expressions for queries	PO1, PO2
CO3	Analyze and evaluate the database using SQL DML/DDDL	PO1, PO2, PO3, PO5
CO4	Analyze databases using normal forms to provide solutions for real time applications	PO1, PO2
CO5	Understand the properties of transactions in a database system, analyze serializability and indexing techniques.	PO1, PO3, PO4

TEXT BOOKS:

1. Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4)
2. Database System Concepts, 5th edition, Silberschatz, Korth, Sudarsan, TMH (For Chapter 1 and Chapter 5)

REFERENCE BOOKS:

1. Introduction to Database Systems, 8th edition, C J Date, Pearson.
2. Database Management System, 6th edition, Ramez Elmasri, Shamkant B. Navathe, Pearson
3. Database Principles Fundamentals of Design Implementation and Management, Carlos Coronel, Steven Morris, Peter Robb, Cengage Learning.

REFERENCE WEBSITE:

1. <https://www.w3schools.in/sql/database-concepts>
2. <https://www.javatpoint.com/dbms-tutorial>
3. <https://www.geeksforgeeks.org/introduction-of-dbms-database-management-system-set-1/>
4. <https://nptel.ac.in/courses/106/105/106105175/>
5. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01275806667282022456_shared/overview

CO-PO MAPPING:

CO\PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO.1	3	-	3	-	-	-	-	-	-	-	-	-
CO.2	3	3	-	-	-	-	-	-	-	-	-	-
CO.3	3	3	3	-	3	-	-	-	-	-	-	-
CO.4	3	3	-	-	-	-	-	-	-	-	-	-
CO.5	3	-	2	3	-	-	-	-	-	-	-	-
CO*	3	3	2.6	3	3	-	-	-	-	-	-	-



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

II B.Tech. - IV Semester

23CSE243

SOFTWARE ENGINEERING

L T P C
3 0 0 3

PRE-REQUISITES: A course on Introduction to Programming

COURSE EDUCATIONAL OBJECTIVES:

1. To provide knowledge of basic software engineering methods, practices and their appropriate application.
2. To understand software requirements and the SRS documents.
3. To design components based on different software architectural styles and to Study the notations of Unified Modeling Language to identify, analyze, and model structural and behavioral concepts of the system.
4. To understanding of approaches to verification and validation including static analysis, and Reviews.
5. To describe software measurement and software risks.

UNIT –1: INTRODUCTION

(9)

Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.

Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

UNIT –2: SOFTWARE PROJECT MANAGEMENT & REQUIREMENTS ANALYSIS AND SPECIFICATION

(9)

Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.

Requirements Analysis And Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT III: SOFTWARE DESIGN & AGILITY & FUNCTION-ORIENTED SOFTWARE DESIGN & USER INTERFACE DESIGN

(9)

Software Design: Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling. approaches to software design.

Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)

Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.

User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT –IV: CODING AND TESTING & SOFTWARE RELIABILITY AND QUALITY MANAGEMENT

(9)

Coding And Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing, Testing object- oriented programs, Smoke testing, and Some general issues associated with testing.

Software Reliability And Quality Management: Software reliability. Statistical testing,



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

Software quality, Software quality management system, ISO 9000. SEI Capability maturity model. Few other important quality standards, and Six Sigma.

UNIT V: COMPUTER-AIDED SOFTWARE ENGINEERING & SOFTWARE MAINTENANCE & SOFTWARE REUSE (9)

Computer-Aided Software Engineering (Case): CASE and its scope, CASE environment, CASE support in the software life cycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.

Software Maintenance: Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.

Software Reuse: reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.

Total Hours: 45

COURSE OUTCOMES:

On successful completion of the course the student will be		POs related to COs
CO1	Identify the key activities in managing a software project and can compare different process models.	PO1, PO2
CO2	Identify software requirements and design SRS document by analyzing the data flows.	PO1, PO3, PO4
CO3	Design class based components and conduct component level design based on architectural styles and patterns. Represent classes, responsibilities and states using UML notation and model structural concepts of the system. Model behavioral concepts of the system and analyze and document the requirements through use case driven approach	PO1,PO2, PO3,PO4,PO5
CO4	Identify various types of testing and development metrics for various phases of software development.	PO1, PO2, PO4
CO5	Identify the software risks and analyze the quality assurance activities, Represent classes, responsibilities and states using UML notation and model structural concepts of the system	PO1, PO4

TEXT BOOKS:

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition, PHI.
2. Software Engineering A practitioner's Approach, Roger S. Pressman, 9th Edition, Mc- Graw Hill International Edition.

REFERENCE BOOKS:

1. Software Engineering, Ian Sommerville, 10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

E-RESOURCES:

- 1) <https://nptel.ac.in/courses/106/105/106105182/>
- 2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012605895063871_48827_shared/overview
- 3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003_904735_shared/overview



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

CO-PO MAPPING:

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	-	-	-	-	-	-	-
CO2	2	-	3	3	-	-	-	-	-	-	-	-
CO3	2	2	3	2	3	-	-	-	-	-	-	-
CO4	2	2	-	3	-	-	-	-	-	-	-	-
CO5	3	-	-	2	-	-	-	-	-	-	-	-
CO*	2.4	2	3	2.5	3	-	-	-	-	-	-	-



II B.TECH - IV SEMESTER

23CSE244

OPERATING SYSTEMS LAB

L T P C
3 0 0 3

PRE-REQUISITES: A course on Introduction to Programming

COURSE EDUCATIONAL OBJECTIVES:

1. Provide insights into system calls, file systems, semaphores,
2. Develop and debug CPU Scheduling algorithms, page replacement algorithms, thread implementation
3. Implement Bankers Algorithms to Avoid the Dead Lock

EXPERIMENTS COVERING THE TOPICS:

- UNIX fundamentals, commands & system calls
- CPU Scheduling algorithms, thread processing
- IPC, semaphores, monitors, deadlocks
- Page replacement algorithms, file allocation strategies
- Memory allocation strategies

SAMPLE EXPERIMENTS:

1. Practicing of Basic UNIX Commands.
2. Write programs using the following UNIX operating system calls
fork, exec, getpid, exit, wait, close, stat, opendir and readdir
3. Simulate UNIX commands like cp, ls, grep, etc.,
4. Simulate the following CPU scheduling algorithms
 - a) FCFS
 - b) SJF
 - c) Priority
 - d) Round Robin
5. Control the number of ports opened by the operating system with
 - a) Semaphore
 - b) Monitors.
6. Write a program to illustrate concurrent execution of threads using pthreads library.
7. Write a program to solve producer-consumer problem using Semaphores.
8. Implement the following memory allocation methods for fixed partition
 - a) First fit
 - b) Worst fit
 - c) Best fit
9. Simulate the following page replacement algorithms
 - a) FIFO
 - b) LRU
 - c) LFU
10. Simulate Paging Technique of memory management.
11. Implement Bankers Algorithm for Dead Lock avoidance and prevention
12. Simulate the following file allocation strategies
 - a) Sequential
 - b) Indexed
 - c) Linked



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

COURSE OUTCOMES:

On successful completion of the course the student will be		POs related to COs
CO1	Trace different CPU Scheduling algorithms (L2).	PO1, PO2
CO2	Implement Bankers Algorithms to Avoid the Dead Lock (L3).	PO1, PO3, PO4
CO3	Evaluate Page replacement algorithms (L5).	PO1,PO2, PO3,PO4,PO5
CO4	Illustrate the file organization techniques (L4).	PO1, PO2, PO4
CO5	Illustrate Inter process Communication and concurrent execution of threads (L4)	PO1, PO4

REFERENCES:

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 10th Edition, Wiley, 2018.
2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson, 2016
3. Operating Systems -Internals and Design Principles, Stallings W, 9th edition, Pearson, 2018
4. Operating Systems: A Concept Based Approach, D.M Dhamdhere, 3rd Edition, McGraw-Hill, 2013

REFERENCE WEBSITE:

1. <https://www.cse.iitb.ac.in/~mythili/os/>
2. <http://peterindia.net/OperatingSystems.html>

CO-PO MAPPING:

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	-	-	-	-	-	-	-	-	-	-	-
CO2	-	3	-	-	-	-	-	-	-	-	-	-
CO3	-	-	3	-	-	-	-	-	-	-	-	-
CO4	-	-	-	3	-	-	-	-	-	-	-	-
CO5	-	-	-	-	3	-	-	-	-	-	-	-
CO6	-	-	-	-	-	-	-	3	-	-	-	-
CO7	-	-	-	-	-	-	-	-	3	-	-	-
CO8	-	-	-	-	-	-	-	-	-	3	-	-
CO9	-	-	-	-	-	-	-	-	-	-	-	3
CO*	3	3	3	3	3	-	-	3	3	3	-	3



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

II B.Tech. - IV Semester

23CSE245

DATABASE MANAGEMENT SYSTEMS LAB

L T P C

(Common to CSE, CSM, CAI, CSD)

3 0 0 3

PRE-REQUISITES: A course on Introduction to Programming

COURSE EDUCATIONAL OBJECTIVES:

This Course will enable students to

1. Populate and query a database using SQL DDL/DML Commands
2. Declare and enforce integrity constraints on a database
3. Writing Queries using advanced concepts of SQL
4. Programming PL/SQL including procedures, functions, cursors and triggers

EXPERIMENTS COVERING THE TOPICS:

1. DDL, DML, DCL commands
2. Queries, nested queries, built-in functions,
3. PL/SQL programming- control structures
4. Procedures, Functions, Cursors, Triggers,
5. Database connectivity- ODBC/JDBC

EXPERIMENTS :

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
5. i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE1APPLICATION ERROR.
8. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
10. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers
12. Create a table and perform the search operation on table using indexing and non1indexing techniques.
13. Write a Java program that connects to a database using JDBC
14. Write a Java program to connect to a database using JDBC and insert values into it
15. Write a Java program to connect to a database using JDBC and delete values from it

REFERENCE BOOKS:

1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007
3. Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007
4. Ramez Elmasri, Shamkant, B. Navathe, " Database Systems, ", Pearson Education, 6th Edition, 2013.
5. Database Principles Fundamentals of Design Implementation and Management, 10th edition, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning, 2022

REFERENCE WEBSITE:

1. <https://www.scoopworld.in>
2. <https://vlabs.iitb.ac.in/vlabs-dev/labs/dblab/index.php>

COURSE OUTCOMES:

On successful completion of the course the student will be		POs
CO1	Demonstrate practical knowledge on creation and alteration of tables, insertion and Querying of data.	PO1
CO2	Analyze the database schemas for the different types of database	PO2
CO3	Design the databases using SQL DML/DDL Commands	PO3
CO4	Design the complex PL/SQL programs for different problems	PO4
CO5	Use the procedure, function, trigger and cursor concepts in PL/SQL	PO5
CO6	Follow the ethical principles in implementing the programs	PO8
CO7	Do experiments effectively as an individual and as a team member in a group.	PO9
CO8	Communicate verbally and in written form, the understanding about the experiments.	PO10
CO9	Continue updating their skill related to SQL Commands and Queries and implementing programs in future.	PO12



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

CO PO MAPPING:

CO-PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
C01	3	-	-	-	-	-	-	-	-	-	-	-
C02	-	3	-	-	-	-	-	-	-	-	-	-
C03	-	-	3	-	-	-	-	-	-	-	-	-
C04	-	-	-	3	-	-	-	-	-	-	-	-
C05	-	-	-	-	3	-	-	-	-	-	-	-
C06	-	-	-	-	-	-	-	3	-	-	-	-
C07	-	-	-	-	-	-	-	-	3	-	-	-
C08	-	-	-	-	-	-	-	-	-	3	-	-
C09	-	-	-	-	-	-	-	-	-	-	-	3
CO*	3	3	3	3	3	-	-	3	3	3	-	3



II B.Tech. - IV Semester

23CSE246

FULL STACK DEVELOPMENT – 1
(Skill Enhancement Course)
(Common to CSE, CSM and CAI)

L	T	P	C
0	1	2	2

PRE-REQUISITES: A course on Introduction to Programming

COURSE EDUCATIONAL OBJECTIVES:

- CO1: Design Websites.
- CO2: Apply Styling to web pages.
- CO3: Make Web pages interactive.
- CO4: Design Forms for applications.
- CO5: Choose Control Structure based on the logic to be implemented.
- CO6: Understand HTML tags, Attributes and CSS properties

TRADE FOR EXERCISES:

1. LISTS, LINKS AND IMAGES

- a. Write a HTML program, to explain the working of lists.
Note: It should have an ordered list, unordered list, nested lists and ordered list in an unordered list and definition lists.
- b. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes.
- c. Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles.
- d. Write a HTML program, in such a way that, rather than placing large images on a page, the preferred technique is to use thumbnails by setting the height and width parameters to something like to 100*100 pixels. Each thumbnail image is also a link to a full sized version of the image. Create an image gallery using this technique

2. HTML TABLES, FORMS AND FRAMES

- Write a HTML program, to explain the working of tables. (use tags: <table>, <tr>, <th>, <td> and attributes: border, rowspan, colspan)
- Write a HTML program, to explain the working of tables by preparing a timetable. (Note: Use <caption> tag to set the caption to the table & also use cell spacing, cell padding, border, rowspan, colspan etc.).
- Write a HTML program, to explain the working of forms by designing Registration form. (Note: Include text field, password field, number field, date of birth field, checkboxes, radio buttons, list boxes using <select>&<option> tags, <text area> and two buttons ie: submit and reset. Use tables to provide a better view).
- Write a HTML program, to explain the working of frames, such that page is to be divided into 3 parts on either direction. (Note: first frame image, second frame paragraph, third frame hyperlink. And also make sure of using "no frame" attribute such that frames to be fixed).

3. HTML 5 AND CASCADING STYLE SHEETS, TYPES OF CSS

- a. Write a HTML program, that makes use of <article>, <aside>, <figure>, <figcaption>, <footer>, <header>, <main>, <nav>, <section>, <div>, tags.
- b. Write a HTML program, to embed audio and video into HTML web page.
- c. Write a program to apply different types (or levels of styles or style specification formats) - inline, internal, external styles to HTML elements. (identify selector, property and value).



4. SELECTOR FORMS

- a. Write a program to apply different types of selector forms
 - Simple selector (element, id, class, group, universal)
 - Combinator selector (descendant, child, adjacent sibling, general sibling)
 - Pseudo-class selector
 - Pseudo-element selector
 - Attribute selector

5. CSS WITH COLOR, BACKGROUND, FONT, TEXT AND CSS BOX MODEL

- a. Write a program to demonstrate the various ways you can reference a color in CSS.
- b. Write a CSS rule that places a background image halfway down the page, tilting it horizontally. The image should remain in place when the user scrolls up or down.
- c. Write a program using the following terms related to CSS font and text:
 - i. font-size
 - ii. font-weight
 - iii. font-style
 - iv. text-decoration
 - v. text-transformation
 - vi. text-alignment
- d. Write a program, to explain the importance of CSS Box model using
 - i. Content
 - ii. Border
 - iii. Margin
 - iv. Padding

6. APPLYING JAVASCRIPT - INTERNAL AND EXTERNAL, I/O, TYPE CONVERSION

- a. Write a program to embed internal and external JavaScript in a web page.
- b. Write a program to explain the different ways for displaying output.
- c. Write a program to explain the different ways for taking input.
- d. Create a webpage which uses prompt dialogue box to ask a voter for his name and age. Display the information in table format along with either the voter can vote or not

7. JAVASCRIPT PRE-DEFINED AND USER-DEFINED OBJECTS

- a. Write a program using document object properties and methods.
- b. Write a program using window object properties and methods.
- c. Write a program using array object properties and methods.
- d. Write a program using math object properties and methods.
- e. Write a program using string object properties and methods.
- f. Write a program using regex object properties and methods.
- g. Write a program using date object properties and methods.
- h. Write a program to explain user-defined object by using properties, methods, accessors, constructors and display.

8. JAVASCRIPT CONDITIONAL STATEMENTS AND LOOPS

- a. Write a program which asks the user to enter three integers, obtains the numbers from the user and outputs HTML text that displays the larger number followed by the words "LARGER NUMBER" in an information message dialog. If the numbers are equal, output HTML text as "EQUAL NUMBERS".
- b. Write a program to display week days using switch case.
- c. Write a program to print 1 to 10 numbers using for, while and do-while loops.
- d. Write a program to print data in object using for-in, for-each and for-of loops
- e. Develop a program to determine whether a given number is an 'ARMSTRONG NUMBER' or not. [Eg: 153 is an Armstrong number, since sum of the cube of the digits is equal to the number i.e., $1^3 + 5^3 + 3^3 = 153$]



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

- f. Write a program to display the denomination of the amount deposited in the bank in terms of 100's, 50's, 20's, 10's, 5's, 2's & 1's. (Eg: If deposited amount is Rs.163, the output should be 1-100's, 1-50's, 1- 10's, 1-2's & 1-1's)

9. JAVASCRIPT FUNCTIONS AND EVENTS

- a. Design a appropriate function should be called to display
- Factorial of that number
 - Fibonacci series up to that number
 - Prime numbers up to that number
 - Is it palindrome or not
- b. Design a HTML having a text box and four buttons named Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate function should be called to display
11. Factorial of that number
 12. Fibonacci series up to that number
 13. Prime numbers up to that number
 14. Is it palindrome or not
- c. Write a program to validate the following fields in a registration page
- i. Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
 - ii. Mobile (only numbers and length 10 digits)
 - iii. E-mail (should contain format like xxxxxxx@xxxxxx.xxx)

TEXT BOOKS:

1. Programming the World Wide Web, 7th Edition, Robert W Sebesta, Pearson, 2013.
2. Web Programming with HTML5, CSS and JavaScript, John Dean, Jones & Bartlett Learning, 2019 (Chapters 1-11).
3. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasanth Subramanian, 2nd edition, Apress, O'Reilly.

REFERENCE BOOKS:

-Nil-

REFERENCE WEBSITE:

1. <https://www.w3schools.com/html>
2. <https://www.w3schools.com/css>
3. <https://www.w3schools.com/js/>
4. <https://www.w3schools.com/nodejs>
5. <https://www.w3schools.com/typescript>



SREENIVASA INSTITUTE OF TECHNOLOGY AND MANAGEMENT STUDIES - Chittoor
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
(Accredited by NBA)

COURSE OUTCOMES:

On successful completion of the course the student will be		POs related to COs
C01	Demonstrate knowledge on web page design elements	PO1
C02	Analyze user requirements to develop web applications.	PO2
C03	Design client-server applications using web technologies.	PO3
C04	Manually Test the functionality of the web application	PO4
C05	select appropriate design tools and procedure to implement web applications	PO5
C06	Follow ethical principles in designing, and implementing various Technologies.	PO8
C07	Do experiments effectively as an individual and as a member in a group.	PO9
C08	Communicate verbally and in written form, the understandings about the experiments	PO10
C09	Continue updating their skill related to various web technologies for implementation of various web applications during their life time	PO12

CO PO MAPPING:

CO\PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
C01	3	-	-	-	-	-	-	-	-	-	-	-
C02	-	3	-	-	-	-	-	-	-	-	-	-
C03	-	-	3	-	-	-	-	-	-	-	-	-
C04	-	-	-	3	-	-	-	-	-	-	-	-
C05	-	-	-	-	3	-	-	-	-	-	-	-
C06	-	-	-	-	-	-	-	3	-	-	-	-
C07	-	-	-	-	-	-	-	-	3	-	-	-
C08	-	-	-	-	-	-	-	-	-	3	-	-
C09	-	-	-	-	-	-	-	-	-	-	-	3
CO	3	3	3	3	3	-	-	3	3	3	-	3