Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/ IT) - All Programs

SEMESTER- III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS306M	DCC	Fundamentals of Data Structures	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

COURSE OBJECTIVES:

The student will have ability to:

1. To understand efficient storage mechanisms of data for easy access.
2. To design and implementation of various basic and advanced data structures.
3. To introduce various techniques for representation of the data in the real world.
4. To develop applications using data structures.
5. To understand the concept of protection and management of data.

COURSE OUTCOMES:

Upon completion of the subject, students will be able to:

1. Get a good understanding of applications of Data Structures.
2. Develop applications using data structures.
3. Handle operations like searching, insertion, deletion, traversing mechanism etc. on various data structures.
4. Decide the appropriate data type and data structure for a given problem.
5. Select the best algorithm to solve a problem by considering various problem characteristics, such as the data size, the type of operations, etc.

SYLLABUS

UNIT I — 10 HOURS

Introduction: Overview of Data structures, Types of data structures, Primitive and Non-Primitive data structures and Operations, Introduction to Algorithms & complexity notations. Characteristics of Array, One-Dimensional Array, Operation with Array, Two-Dimensional Arrays, Three or Multi-Dimensional Arrays, Sparse matrix, Drawbacks of linear arrays. Strings, Array of Structures, Pointer and one-dimensional Arrays, Pointers and Two-Dimensional Arrays, Pointers and Strings, Pointer and Structure.

UNIT II — 9 HOURS

Linked List: Linked List as an ADT, Linked List Vs. Arrays, Dynamic Memory Allocation & De-allocation for a Linked List, Types of Linked List: Circular & Doubly Linked List. Linked. **List operations:** All possible insertions and deletion operations on all types of Linked list Reverse a Single Linked List; Divide a singly linked list into two equal halves, Application of Linked List.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/ IT) - All Programs

SEMESTER- III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS306M	DCC	Fundamentals of Data Structures	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

UNIT III — 8 HOURS

Stack: The Stack as an ADT, Stack operation, Array Representation of Stack, Link Representation of Stack, Application of stack – Recursion, Polish Notation. Types of Recursions, problem based on Recursion: Tower of Hanoi. **The Queue**: The Queue as an ADT, Queue operation, Array Representation of Queue, Linked Representation of Queue, Types of Queue: Circular Queue & Dequeue, Introduction of Priority Queue, Application of Queues.

UNIT IV — 10 HOURS

Tree: Definitions and Concepts of Binary trees, Types of Binary Tree, Representation of Binary tree: Array & Linked List. General tree, forest, Expression Tree. Forest and general tree to binary tree conversion. Binary Search Tree Creation, Operations on Binary Search Trees: insertion, deletion & search an element, Traversals on Binary SEARCH TREE and algorithms. Height balanced Tree: AVL, B-Tree, 2-3 Tree, B+Tree: Creation, Insertion & Deletion. **Graph:** Definitions and Concepts Graph Representations: Adjacency MATRIX, Incidence matrix, Graph TRAVERSAL (DFS & BFS), Spanning Tree and Minimum Cost Spanning Tree: Prim's & Kruskal's Algorithm.

UNIT V — 8 HOURS

Sorting: Sorting Concept and types of Sorting, Stable & Unstable sorting. Concept of Insertion Sort, Selection sort, Bubble sort, Quick Sort, Merge Sort, Heap & Heap Sort, Shell Sort & Radix sort. Algorithms and performance of Insertion, selection, bubble, Quick sort & Merge sort.

TEXTBOOKS:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. Cambridge, MA, USA: MIT Press, 2009.
2. M. A. Weiss, *Data Structures and Algorithm Analysis in C++*, 4th ed. Upper Saddle River, NJ, USA: Pearson Education, 2014.

REFERENCE:

1. S. Sahni, *Data Structures, Algorithms, and Applications in C++*, 2nd ed. New Delhi, India: Universities Press, 2005.
2. M. T. Goodrich, R. Tamassia, and D. M. Mount, *Data Structures and Algorithms in C++*, 2nd ed. Hoboken, NJ, USA: Wiley, 2011.
3. R. K. Shukla, *Data Structures Using C & C++*. New Delhi, India: Wiley India, 2016.

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS306M	**DCC**	**Fundamentals of Data Structures**	60	20	20	30	20	3	0	2	**4**

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

4. E. Balagurusamy, *Data Structures Using C*. New Delhi, India: Tata McGraw-Hill, 2017.
5. P. S. Deshpande and O. G. Kakde, *C & Data Structures*. Boston, MA, USA: Charles River Media, 2015.
6. G. Pai, *Data Structures*. New Delhi, India: Tata McGraw-Hill, 2015.

LIST OF PRACTICALS

1. To develop a program to find an average of an array using AVG function.
2. To implement a program that can insert, delete and edit an element in array.
3. To implement an algorithm for insert and delete operations of circular queue and implement the same using array.
4. Write a menu driven program to implement the push, pop and display option of the stack with the help of static memory allocation.
5. Write a menu driven program to implement the push, pop and display option of the stack with the help of dynamic memory allocation.
6. Write a menu driven program to implementing the various operations on a linear queue with the help of static memory allocation.
7. Write a menu driven program to implementing the various operations on a linear queue with the help of dynamic memory allocation.
8. Write a menu driven program to implement various operations on a linear linked list.
9. Write a menu driven program to implement various operations on a circular linked list
10. Write a program for implementation of Bubble sort
11. Write a program for Insertion sort
12. Write a program for Merge Sort
13. Write a program to implement Heap sort
14. Write a program to implement Quick sort
15. Write a program to Construct a Binary Search Tree and perform deletion, inorder traversal on it
16. Write a program to develop an algorithm for binary tree operations and implement the same.
17. Write a program to design an algorithm for sequential search, implement and test it.
18. Write a program to develop an algorithm for binary search and perform the same.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/ IT) - All Programs

SEMESTER- III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS307M	DCC	Principles of Operating Systems	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

COURSE OBJECTIVES:

The student will have ability to:

1. Learn the fundamentals of Operating Systems.
2. Study the mechanisms of Operating System to handle processes and threads and their communication.
3. Gain knowledge of process management concepts that includes architecture, Mutual exclusion algorithms, deadlock detection and recovery algorithms.
4. Learn the mechanisms involved in memory management in Operating System.
5. Understand the components and management aspects of disc scheduling

COURSE OUTCOMES:

Upon completion of the subject, students will be able to:

1. Describe the structure of Operating System in detail.
2. Design and Implement Process management Techniques in Operating System.
3. Calculate CPU Scheduling criteria.
4. Understand The Memory Management of Operating System.
5. Elaborate Disc Scheduling.

SYLLABUS

UNIT I — 10 HOURS

Introduction to Operating System: Introduction and Need of operating system, Layered Architecture/Logical Structure of Operating system, Type of OS (Multiprogramming, Timesharing, Real-Time, Networked, Distributed, Clustered, Handheld), Operating system as Resource Manager and Virtual Machine, System Calls/Monitor Calls, Firmware- BIOS, Boot Strap Loader. Threads- processes versus threads, threading, concepts, models, kernel & user level threads, thread usage, benefits.

UNIT II — 9 HOURS

Process Management: Process Model, Creation, Termination, States & Transitions, Context Switching, Process Control Block, CPU and I/O bound, CPU scheduler- short, medium, long-term, dispatcher, Scheduling: - preemptive and non-preemptive, Static and Dynamic Priority. Criteria/Goals/Performance Metrics, scheduling algorithms- FCFS, SJFS, shortest remaining time, round robin, Priority scheduling, multilevel queue scheduling, multilevel feedback queue scheduling.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/ IT) - All Programs

SEMESTER- III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS307M	DCC	Principles of Operating Systems	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

UNIT III — 8 HOURS

Inter process Communication: Introduction to Message Passing, Race Condition, Critical Section Problem, Peterson's Solution, Semaphore, Classical Problems of Synchronization Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem. **Deadlock-** System model, Resource types, Deadlock Problem, Deadlock Characterization, Methods for Deadlock Handling, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock Detection, Recovery from Deadlock.

UNIT IV — 9 HOURS

Memory Management: concepts, functions, logical and physical address space, address binding, degree of multiprogramming, swapping, static & dynamic loading- creating a load module, loading, static & dynamic linking, memory allocation schemes- first fit, next fit, best fit, worst fit and quick fit.

Virtual Memory- concept, virtual address space, paging scheme, pure segmentation and segmentation with paging scheme hardware support and implementation details, memory fragmentation, demand paging, working set model, page fault frequency, thrashing, page replacement algorithms- optimal, FIFO, LRU, Bleady's anomaly; TLB (translation look aside buffer).

UNIT V — 9 HOURS

File Management: Concepts, Naming, Attributes, Operations, Types, Structure, File Organization & Access (Sequential, Direct ,Index Sequential) Methods, Memory Mapped Files, Directory Structures One Level, Two Level, Hierarchical/Tree, Acyclic Graph, General Graph, File System Mounting, File Sharing, Path Name, Directory Operations, Overview Of File System in Linux & Windows.

Input/output Subsystems- Concepts, Functions/Goals, Input/ Output devices- Block and Character, Spooling, Disk Structure & Operation, Disk Attachment, Disk Storage Capacity, Disk Scheduling Algorithm- FCFS, SSTF, Scan Scheduling, C-Scan Schedule.

TEXTBOOKS:

1. Silberschatz, *Operating System Concepts*, 10th ed. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2018.
2. S. Tanenbaum, *Modern Operating Systems*, 4th ed. Harlow, U.K.: Pearson Education, 2014.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/ IT) - All Programs

SEMESTER- III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS307M	DCC	Principles of Operating Systems	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

REFERENCE:

1. A. S. Godbole, Operating Systems, 3rd ed. New Delhi, India: Tata McGraw-Hill Education, 2017.
2. W. Stallings, Operating Systems: Internals and Design Principles, 8th ed. Harlow, U.K.: Pearson Education, 2014.
3. V. Shukla, Operating Systems, 3rd ed. Indore, India: Kataria & Sons, 2013.
4. M. Singhal and N. G. Shivaratri, Advanced Concepts in Operating Systems. New Delhi, India: Tata McGraw-Hill Education, 2017.

LIST OF PRACTICALS

1. Implement and update the BIOS settings of your PC.
2. Suppose there are five printers connected to a system. Each printing process requires a different amount of time to complete. The CPU allocates a fixed time slice to each process. After the time slice expires, the next process enters the CPU. If a process is not completed within its time slice, it is re-entered into the queue based on First-Come, First-Served (FCFS) scheduling in a rotational manner. Apply scheduling to this scenario.
3. Implement Non-Preemptive Priority CPU Scheduling.
4. Implement Non-Preemptive Shortest Job First (SJF) CPU Scheduling.
5. Suppose there are five different resources connected to a system (e.g., three printers and two scanners), each requiring different times to complete their tasks. Identify which scheduling algorithm provides the best CPU performance.
6. Implement scheduling where the CPU executes the process that arrives first (FCFS Scheduling).
7. Implement Round-Robin CPU Scheduling.
8. Write a program to implement Semaphore.
9. Consider a situation where five faculty members are seated around a table with four ballpoint pens placed on it. At any time, only one pen can be picked up by one faculty member for writing. What happens if all attempt to pick up a pen simultaneously? Find the solution.
10. Consider a dentist's clinic with one dentist and one treatment chair, along with *n* waiting chairs for patients:
    - If no patient is present, the dentist sleeps in the chair.
    - When a patient arrives, they wake up the dentist.
    - If multiple patients arrive while the dentist is treating one, the remaining patients wait if chairs are available; otherwise, they leave if no chairs are free. Find the solution for this scenario.
11. Write a program to implement Memory Management Algorithms such as *First Fit*, *Best Fit*, and *Worst Fit*.
12. Demonstrate Virtual Memory Techniques such as *LRU* (Least Recently Used) and *FIFO* (First-In, First-Out).

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/ IT) - All Programs

SEMESTER- III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS307M	DCC	Principles of Operating Systems	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

13. Implement Shortest Seek Time First (SSTF) Disk Scheduling Algorithm.
14. Implement SCAN Disk Scheduling Algorithm.
15. Implement Circular SCAN (C-SCAN) Disk Scheduling Algorithm.
16. Implement LOOK Disk Scheduling Algorithm.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech CSE with Specialization in AI Powered Mobile Application -Apple Authorized Training Center

SEMESTER-III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTMA310	DSE	**Fundamentals of Dart Programming**	**60**	**20**	**20**	**30**	**20**	2	0	2	3

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

COURSE OBJECTIVES:

The student will have ability to:

1. To introduce the fundamentals of the Dart programming language, including its syntax, data types, and basic constructs.
2. To develop problem-solving skills using control structures, functions, and modular programming in Dart.
3. To provide knowledge of data handling techniques using collections such as lists, sets, and maps along with exception handling.
4. To impart understanding of object-oriented programming concepts such as classes, inheritance, polymorphism, and encapsulation in Dart.
5. To enable learners to build efficient applications using advanced features like asynchronous programming, streams, generics, and file handling

COURSE OUTCOMES:

Upon completion of the subject, students will be able to:

1. Describe and explain the fundamental concepts of the Dart programming language, including syntax, data types, and control structures.
2. Apply Dart programming constructs to develop simple programs using functions, loops, and collections.
3. Analyze and implement object-oriented programming concepts such as classes, inheritance, and polymorphism in Dart.
4. Evaluate different programming approaches and handle exceptions effectively in Dart applications.
5. Design and develop efficient Dart applications using advanced features like asynchronous programming, streams, and generics.

SYLLABUS

UNIT I — 6 HOURS

Introduction to Dart & Basics: Overview of Dart language, Features and advantages of Dart, Installing Dart SDK and setting up environment, Structure of a Dart program, Writing and running first Dart program, Variables and data types, Numbers (int, double), Strings, Booleans, Dynamic type, Operators, Arithmetic operators, Relational operators, Logical operators, Assignment operators, Input and Output in Dart.

UNIT II — 6 HOURS

Control Flow & Functions: Conditional statements, if, if-else, nested if, switch-case, Loops, for loop, while loop, do-while loop, break and continue, Functions in Dart, Function declaration and definition, Optional parameters (positional, named, default), Arrow functions (=>), Recursion, Scope and lifetime of variables

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech CSE with Specialization in AI Powered Mobile Application -Apple Authorized Training Center

SEMESTER-III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTMA310	DSE	**Fundamentals of Dart Programming**	**60**	**20**	**20**	**30**	**20**	2	0	2	3

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

UNIT III — 6 HOURS

Collections & Exception Handling: Lists (fixed-length and growable), Sets, Maps, Iteration techniques, for-in loop, forEach(), Collection operations and methods, Exception handling, try, catch, finally, throw keyword, Custom exceptions

UNIT IV — 6 HOURS

Object-Oriented Programming in Dart: Classes and Objects, Constructors, Default constructors, Parameterized constructors, Instance variables and methods, Encapsulation (getters and setters), Inheritance, Method overriding, Abstract classes and interfaces, Polymorphism, Static members.

UNIT V — 6 HOURS

Advanced Dart Concepts: Null safety in Dart, Asynchronous programming, Future, async and await, Streams and stream handling, Generics, Libraries and packages, File handling (basic concepts), Introduction to command-line Dart applications.

TEXTBOOKS:

1. K. Choi, *The Art of Dart: Master the Dart Programming Language*. 2023.
2. Success KPK, Dart Programming: Journey from Beginner to Advanced. 2024
3. R. Tech, *Mastering Dart & Flutter 2025*.

REFERENCE:

1. Dart Apprentice: Fundamentals. Kodeco, 2022. (Widely used modern foundation text)
2. Dart Apprentice: Beyond the Basics. Kodeco, 2022. (Advanced concepts including OOP & concurrency)
3. Data Structures & Algorithms in Dart. 2023. (For advanced problem-solving and generics concepts)

LIST OF PRACTICALS

1. Write a Dart program to demonstrate basic syntax, variables, and data types.
2. Develop a program to perform arithmetic and logical operations using operators.
3. Write a Dart program to take user input and display formatted output.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech CSE with Specialization in AI Powered Mobile Application -Apple Authorized Training Center

SEMESTER-III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTMA310	DSE	**Fundamentals of Dart Programming**	**60**	**20**	**20**	**30**	**20**	2	0	2	3

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

4. Implement a program to demonstrate conditional statements (if-else, switch-case).
5. Write a Dart program to print patterns using loops (for, while, do-while).
6. Develop a program to implement functions with optional and named parameters.
7. Write a program to find factorial of a number using recursion.
8. Write a Dart program to perform operations on Lists (add, remove, search elements).
9. Implement a program using Sets and Maps for storing and retrieving data.
10 Develop a program demonstrating exception handling using try-catch-finally.
11. Create a custom exception and write a program to handle user-defined exceptions.
12. Write a Dart program to demonstrate class, object, and constructors.
13. Implement a program to demonstrate inheritance and method overriding.
14. Develop a program to demonstrate abstract classes and polymorphism.
15. Write a Dart program to demonstrate asynchronous programming using Future and async/await.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech CSE/IT - All Programs

SEMESTER-III (2025-2029)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTDSE322M	DSE	Principle of Programming Language	60	20	20	30	20	2	0	2	3

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

COURSE OBJECTIVES:

The student will have ability to:

1. Improve the background for choosing appropriate programming languages for certain classes of programming problems.
2. Understand principles to program in an imperative (or procedural), an object-oriented, a functional, and a logical programming language.
3. Understand the significance of implementation of a programming language in a compiler or interpreter
4. Increase the ability to learn new programming languages
5. Increase the capacity to express programming concepts and choose among alternative ways to express things.

COURSE OUTCOMES:

Upon completion of the subject, students will be able to:

1. Gain insight and develop understanding of the underlying principles and concepts of programming languages.
2. Competent with analyzing programming language design issues related to data types, expressions and control structures.
3. Describe the concept of sub-programming with the help of Functions. Also develop understanding with the parameter passing techniques and concept of function overloading.
4. Analyze various memory management techniques as well as apply various concepts of object-oriented programming.
5. Develop understanding with the exception handling concepts and gain knowledge of logical and functional programming.

SYLLABUS

UNIT I — 8 HOURS

Preliminary Concepts: Reasons for Studying, Concepts of Programming Languages, Programming Domains, Language Evaluation Criteria, Influences on Language Design, Language Categories, Programming Paradigms – Imperative, Object Oriented, Functional Programming, Logic Programming.

UNIT II — 8 HOURS

Data Types: Introduction, Primitive, Character, User Defined, Record, Union, Pointer and Reference Types, Design and Implementation Uses Related to these Types. Names, Variable, Concept of Binding

UNIT III — 8 HOURS

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTDSE322M	DSE	Principle of Programming Language	60	20	20	30	20	2	0	2	3

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

Expressions and Statements: Arithmetic Relational and Boolean Expressions, Short Circuit Evaluation Mixed Mode Assignment, Assignment Statements, Control Structures

UNIT IV — 8 HOURS

Subprograms and Blocks: Fundamentals of Sub-Programs, Scope and Lifetime of Variable, Static and Dynamic Scope, Design Issues of Subprograms and Operations, Local Referencing Environments, Parameter Passing Methods, Overloaded Sub-Programs, Generic Sub-Programs.

UNIT V — 8 HOURS

Abstract Data Types: Abstractions and Encapsulation, Introductions to Data Abstraction, Static and Stack Based Storage Management. Heap Based Storage Management. Garbage Collection. Object Oriented Programming in Smalltalk, C++, Java, C#, Php, Perl.

TEXTBOOKS:

1. R. W. Sebesta, *Concepts of Programming Languages*, 12th ed. Harlow, U.K.: Pearson Education, 2023.
2. D. A. Watt and W. Findlay, *Programming Language Design Concepts*. Hoboken, NJ, USA: Wiley, 2004.

REFERENCE:

1. K. C. Louden and K. A. Lambert, *Programming Languages: Principles and Practices*, 3rd ed. Boston, MA, USA: Cengage Learning, 2011.
2. M. Gabbrielli, S. Martini, and S. Giallorenzo, *Programming Languages: Principles and Paradigms*, 2nd ed. Cham, Switzerland: Springer, 2023.
3. P. Sestoft, *Programming Language Concepts*, 2nd ed. Cham, Switzerland: Springer, 2017.
4. A. B. Tucker and R. E. Noonan, *Programming Languages: Principles and Paradigms*, 2nd ed. New Delhi, India: Tata McGraw-Hill, 2006.
5. T. W. Pratt, M. V. Zelkowitz, and T. V. Gopal, *Programming Languages: Design and Implementation*, 4th ed. Upper Saddle River, NJ, USA: Pearson Education, 2000.

LIST OF PRACTICALS

1. Examine name and scope including static variables in C and call resolution in Java
2. Examine garbage and memory leak in C and develop a mechanism to avoid or detect memory leak
3. Examine assignment operation including assignment of arrays in Java, assignment of lists in Python, and assignment of structures in C
4. Examine goto statement including scope of goto in C, jump into or out of the block, and non-local goto
5. Examine callbacks including callbacks in C and interface and inner classes in Java

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech CSE/IT - All Programs

SEMESTER-III (2025-2029)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTDSE322M	DSE	Principle of Programming Language	60	20	20	30	20	2	0	2	3

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

6. Examine closure in Python and in C
7. Examine functions including variable number of arguments in C, Java, and Python
8. Examine functions including tail recursion, keyword parameters in Python, and stack smashing in C
9. Examine generics including lists (linked lists and array lists), sets (hash set, preset, link hash set), and map
10. Examine inheritance including override in Java, final in Java, multiple inheritance in Python, and downcasting in Java
11. Examine Java thread model, pthread, and Python including racing, synchronization, interthread communication, and thread local storage

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/IT) – All Programs

SEMESTER III (2025-2020)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTDSE323	DSE	Go Lang	60	20	20	30	20	2	0	2	3

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

COURSE OBJECTIVES:

The student will have ability:

1. To introduce the design philosophy of Go: Simplicity, Safety, and Concurrency.
2. To master Go's type system, focusing on composition over class-based inheritance.
3. To develop highly concurrent systems using Go routines and Channels.
4. To build production-grade web services and high-performance APIs.
5. To understand the integration of Go with modern DevOps and Cloud tools.

COURSE OUTCOMES:

Upon completion of the subject, students will be able to:

1. Apply Go syntax and memory management concepts to solve computational problems.
2. Design modular software using Structs, Methods, and Implicit Interfaces.
3. Implement concurrent algorithms using Go routines and synchronization primitives (Channels, Wait Groups).
4. Build and test RESTful web services using Go's standard library and middleware logic.
5. Evaluate and optimize Go code performance for cloud-native software development.

SYLLABUS

UNIT I — 6 HOURS

The Go Environment & Foundations: History and Motivation; Installing Go; The go tool chain; Creating the first module; Compilation vs. Interpretation. Primitive Types, Variables, and Zero-values; Constants and iota; Scope rules. Control Structures: if, switch, and the versatile for loop; Logical operators and Bitwise operations.

UNIT II — 6 HOURS

Data Structures & Functional Logic: Arrays vs. Slices: Under-the-hood mechanics of slice headers, dynamic growth, and append logic. Maps (Hash tables) and String manipulation; Pointers: Stack vs. Heap allocation and Garbage Collection basics. Functions: Multiple returns, Variadic parameters, Anonymous functions, and Closures.

UNIT III — 6 HOURS

struts' & The Interface System: Struts': Initialization and Nesting; Methods: Value vs. Pointer receivers; The equivalent in Go. Interfaces: The "Implicit" implementation philosophy; Interface values and the Empty interface (interface {}). Composition: Embedding structs and interfaces to achieve reusability without class-based inheritance.

UNIT IV — 6 HOURS

Mastering Concurrency: Go routines: The M: N scheduler; Lightweight vs. OS threads; sync. Wait Group for

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTDSE323	DSE	Go Lang	60	20	20	30	20	2	0	2	3

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

task synchronization. Channels: Directional channels, Buffered vs. Buffered; The select statement for non-blocking I/O. Advanced Patterns: Worker Pools, Fan-in/Fan-out, Murexes (sync. Mutex), and Context cancellation for API timeouts.

UNIT V — 6 HOURS

Web Services & Production Readiness (9 hours): Error Handling: The Go idiom (error interface, Defer, Panic, Recover); Writing idiomatic Go code. Web Development: net/http package; JSON Marshaling/ Unmarshalling; Middleware design and Routing. Tooling & Testing: Unit testing with go test; Benchmarking; Compilation and Cross-platform builds.

TEXTBOOKS:

1. A. A. A. Donovan and B. W. Kernighan, *The Go Programming Language*. Boston, MA, USA: Addison-Wesley, 2015.
2. W. Kennedy, B. Ketelsen, and E. St. Martin, *Go in Action*. Shelter Island, NY, USA: Manning Publications, 2016.

REFERENCE:

1. K. Bodner, *Learning Go: An Idiomatic Approach to Real-World Go Programming*. Sebastopol, CA, USA: O'Reilly Media, 2021.
2. M. Ryer, *Go Programming Blueprints: Build real-world, high-performance, and robust Go applications*, 2nd ed. Birmingham, UK: Packt Publishing, 2016.
3. T. Tsoukalos, *Mastering Go: Create high-performance concurrent applications and micro services with Go*, 3rd ed. Birmingham, UK: Packt Publishing, 2021.

LIST OF PRACTICALS

1. Workspace setup and implementing a basic CLI tool for mathematical series generation.
2. Implement a "Student Record Search" using Slices and Binary Search logic.
3. Build a "Company Directory" using Maps with functions to add, delete, and list entries.
4. Implement a "Geometry Library" using Interfaces to calculate area/perimeter for different shapes.
5. Simulation of an "Order Processing System" using multiple Goroutines and Wait Groups.
6. Create a "Concurrent Web Scraper" that fetches multiple URLs simultaneously via Channels.
7. Build a basic REST API for a **"Sajag"** Cyber-Alert system where users can submit alerts via JSON.
8. Writing Unit Tests and Benchmarking performance for the API developed in Lab 7.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/ IT) - All Programs

SEMESTER- III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS402N	DCC	Software Engineering and Project Management	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

COURSE OBJECTIVES:

The student will have ability to:

1. Get knowledge of basic software engineering methods and practices.
2. Define software requirements and requirement engineering.
3. Apply approaches for various designs and their principles.
4. Explore testing in various domains.
5. Development of significant teamwork and project-based experience.

COURSE OUTCOMES:

Upon completion of the subject, students will be able to:

1. Compare various software process models and identify where these models are applicable.
2. Define and analyze software project management, the framework and the dimensions of software project management.
3. Comprehend System modeling using UML.
4. Identify software testing strategies by using testing tools.
5. Analyze software risks and risk management strategies.

SYLLABUS

UNIT I — 10 HOURS

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths.

A Generic view of process: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI), process patterns, process assessment, personal and team process models.

Process models: The waterfall model, incremental process models, evolutionary process models,
the unified process. Agile development-Agile Process, Extreme Programming

UNIT II — 9 HOURS

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS402N	DCC	Software Engineering and Project Management	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

System models: Context models, behavioral models, data models, object models, structured methods.

UNIT III — 10 HOURS

Design Engineering: Design Process- Design concepts: Abstraction, Architecture, patterns, Separation of Concerns, Modularity, Information Hiding, Functional Independence, Refinement, Aspects, Refactoring, Object Oriented Design Concepts, Design Classes- Design Model: Data, Architectural, Interface, Component, Deployment Level Design Elements

Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, Use Case Diagrams, Class Diagrams, Interaction Diagrams, State chart Diagrams, Activity Diagrams, Package Diagrams, Component Diagrams, Deployment Diagrams

UNIT IV — 7 HOURS

Software Implementation: - Structured coding Techniques, Coding Styles, Standards and Guidelines, Documentation Guidelines-Modern Programming Language Features: Type Checking-User defined data types-Data Abstraction-Exception Handling, Concurrency Mechanism.

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, Object oriented software, Web Apps-validation testing, system testing, and the art of debugging.

UNIT V — 9 HOURS

Metrics for Process and Products: Software measurement, metrics for software quality. Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, ISO 9000 quality standards.

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM, RMMM plan.

Maintenance: Software Maintenance, Software Supportability, Reengineering, Business Process Reengineering, Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering, Economics of Reengineering

TEXTBOOKS:

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/ IT) - All Programs

SEMESTER- III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS402N	DCC	Software Engineering and Project Management	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

1. R. S. Pressman, *Software Engineering: A Practitioner's Approach*, 6th ed. New York, NY, USA: McGraw-Hill International Edition, 2005.
2. P. Jalote, *An Integrated Approach to Software Engineering*. New Delhi, India: Narosa Publishing House, 2005.

REFERENCE:

1. R. Mall, *Fundamentals of Software Engineering*, 2nd ed. New Delhi, India: PHI Learning, 2004.
2. I. Sommerville, *Software Engineering*, 7th ed. Harlow, U.K.: Pearson Education, 2004.
3. G. Booch, J. Rumbaugh, and I. Jacobson, *The Unified Modeling Language User Guide*. Upper Saddle River, NJ, USA: Pearson Education, 1999.
4. R. C. Martin, *Clean Code: A Handbook of Agile Software Craftsmanship*. Upper Saddle River, NJ, USA: Prentice Hall, 2025.
5. A. Hunt and D. Thomas, *The Pragmatic Programmer: Your Journey to Mastery*, 20th Anniversary ed. Boston, MA, USA: Addison-Wesley, 2019.
6. T. Winters, T. Manshreck, and H. Wright, *Software Engineering at Google: Lessons Learned from Programming Over Time*. Sebastopol, CA, USA: O'Reilly Media, 2020.

LIST OF PRACTICALS

1. Study and compare the SDLC models.
2. Prepare a SRS document in line with the IEEE recommended standards.
3. Study Requirement Engineering of project.
4. Study the UML drawing tools.
5. Draw the Entity relationship diagram of a project.
6. Draw the data flow diagrams at level 0 and level 1.
7. Draw use case diagram in UML.
8. Draw activity diagram in UML.
9. Draw class diagram in UML.
10. Draw the component diagram in UML.
11. Draw sequence diagram in UML.
12. Draw collaboration diagram in UML.
13. Use testing tools such as JUnit.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech (CSE/ IT) - All Programs

SEMESTER- III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTCS402N	DCC	Software Engineering and Project Management	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

Chairperson	Chairperson	Controller of Examination	Registrar
Board of Studies, Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore	Faculty of Studies, Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore	Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore	Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech CSE/IT, CSE (ES, MA, AI-IBM, DS, GenAI, MLCC, ICS, BDCE)

SEMESTER-III (2025-2029)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTIT301N	DCC	Computer Networks	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

COURSE OBJECTIVES:

The student will have ability to:

1. Understand the general overview of the concepts and fundamentals of computer networks.
2. Understand the various components required to build different networks.
3. Familiarize the students with the standard models for the layered approach to communication between machines in a network and the protocols of the various layers.

COURSE OUTCOMES:

Upon completion of the subject, students will be able to:

1. Understanding basic computer network technology.
2. Understand the functions of each layer in the OSI and TCP/IP reference model.
3. Obtain the skills of subnetting and routing mechanisms.
4. Familiarity with the essential protocols of computer networks, and how they can be applied in network design and implementation.

SYLLABUS

UNIT I — 9 HOURS

Introduction: Importance of Computer Networks, Classifications & Types. Layered Architecture: Protocol hierarchy, Interfaces and Services, Connection Oriented & Connection less Services, ISO- OSI Reference Model, TCP/IP model overview, comparison of TCP/IP and ISO-OSI reference model.

UNIT II — 9 HOURS

Data Link Layer &MAC Sublayer: Need, Services Provided, Design issues, Elementary data link protocols: simplex protocol, A simplex stop and wait protocol for an error-free channel, A simplex stop and wait protocol for noisy channel. Sliding Window protocols: A one-bit sliding window protocol, A protocol using Go-Back-N, A protocol using Selective Repeat, MAC Addressing, Binary Exponential Back-off (BEB) Algorithm, Distributed Random Access Schemes/Contention Schemes: for Data Services (ALOHA and Slotted- ALOHA), CSMA, CSMA/CA, CSMA/CD.

UNIT III — 9 HOURS

Network Layer: Need, Services Provided, Design Issues, Routing Algorithms and types of Routing Algorithm, IPv4, IPv6, Classful and classless Addressing, Subnetting, Supernetting.

UNIT IV — 10 HOURS

Transport Layer: Need, Design Issues, Multiplexing and Demultiplexing, transport layer services, UDP,UDP Header Format, Principles of reliable data transfer, TCP, Connection Management, TCP Flow Control, TCP

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTIT301N	DCC	Computer Networks	60	20	20	30	20	3	0	2	4

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

Congestion Control, TCP Header Format, TCP Timer Management, SCTP.

UNIT V — 8 HOURS

Session layer: Overview, Authentication, Session layer protocols.
Presentation layer: Overview, Data conversion, Encryption and Decryption, Presentation layer protocols (LPP, Telnet, X.25 packet Assembler/Disassembler).
Application Layer: Domain name system, SNMP, Electronic Mail; the World WEB, HTTP, FTP.

TEXTBOOKS:

1. Andrew S Tanenbaum, *Computer Networks*, 6thEd., Pearson Education, 2016.
2. Behrouz A. Forouzan, *TCP/IP-Protocol suite*, 4th Ed., McGraw-Hill, 2010.

REFERENCE:

1. William Stallings, *Data and Computer Communication*, 10th Ed., Pearson, 2014.
2. Comer, *Internet working with TCP/IP Volume one*, Addison-Wesley, 2015.
3. W. Richard Stevens, *TCP/IP Illustrated, Volume 1*, 2nd Ed., Addison-Wesley Professional Computing Series.

LIST OF PRACTICALS

1. Demonstrate Different Types of Network Equipment's.
2. Color coding standard of CAT 5, 6, 7 and crimping of cable in RJ-45.
3. LAN installations and Configurations.
4. Experiment with basic Network configuration commands.
5. Write a program for error detection and correction technique.
6. Write a program for framing.
7. Write a program for routing algorithm.
8. Socket Programming.
9. Study about different network simulators.
10. Establish and simulate peer to peer network using packet tracer.
11. Simulate LAN using hub and switch and discuss pros and cons of hub.
12. Router configuration using packet tracer.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech CSE (MA-Apple, AIML-Microsoft)

SEMESTER-III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTIT410	SEC	Java Full Stack – I	0	0	0	30	20	0	0	2	1

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

COURSE OBJECTIVES:

The student will have ability to:

1. Introducing the students to the Java programming environment and its execution model.
2. Build strong foundations in Object-Oriented Programming (OOP) for software modularity.
3. Enable students to handle runtime errors and manage concurrent tasks through multi-threading.
4. Provide hands-on expertise in the Java Collection Framework for efficient data manipulation.
5. Develop skills in establishing connectivity between Java applications and Relational Databases.

COURSE OUTCOMES:

Upon completion of the subject, students will be able to:

1. Understand and implement Java code using fundamental constructs, arrays, and memory management (JVM/JRE).
2. Design and solve real-world problems using core OOP principles: Inheritance, Polymorphism, Abstraction, and Encapsulation.
3. Develop robust and thread-safe applications by implementing Exception Handling and Multi-threading.
4. Utilize the Java Collections Framework (List, Set, Map) and I/O streams for complex data storage and retrieval.
5. Create data-driven applications by integrating Java with SQL databases through the JDBC API.

SYLLABUS

UNIT I — 6 HOURS

Java Fundamentals and Environment: The Java Platform: History, Features, and Architecture (JVM, JRE, JDK). Basics: Identifiers, Data types, Variables, and Operators. Control Flow: Decision-making statements (if-else, switch-case) and Looping constructs (for, while, do-while). Arrays & Strings: Single and Multi-dimensional arrays; String class, String Builder, and String Buffer for efficient memory management.

UNIT II — 6 HOURS

Core Object-Oriented Programming: Foundations: Classes, Objects, Methods, and Constructors; The static and this keywords. Inheritance: IS - A and HAS-A relationships, Method Overriding, and the super keyword. Polymorphism: Method Overloading and Runtime Polymorphism. Abstraction: Abstract classes vs. Interfaces. Encapsulation: Access modifiers (Private, Public, Protected, Default) and Packages.

UNIT III — 6 HOURS

Reliability and Concurrency: Exception Handling: The Try-Catch-Finally block, Hierarchy of Exceptions, Checked vs. Unchecked exceptions, throw vs. throws, and creating Custom Exceptions. Multithreading: Introduction

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech CSE (MA-Apple, AIML-Microsoft)

SEMESTER-III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTIT410	SEC	Java Full Stack – I	0	0	0	30	20	0	0	2	1

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

to Threads, Lifecycle of a Thread, Thread class vs. Runnable interface. Synchronization: Understanding Race Conditions, Synchronized methods/blocks, and Inter-thread Communication (wait, notify, notify All).

UNIT IV — 6 HOURS

Data Structures and File I/O: Collections Framework: Overview of List (Array List, Linked List), Set (Hash Set, Tree Set), and Map (Hash Map, Tree Map) interfaces. Generics: Type-safe collections and Iterators. Java I/O: Byte Streams vs. Character Streams, Scanner and Buffered Reader classes. Serialization: Saving and restoring object states.

UNIT V — 6 HOURS

Database Persistence with JDBC: Database Basics: Introduction to Relational Databases, SQL Recap (DDL, DML, and DCL), Joins, and Sub queries. JDBC Architecture: Driver types and the JDBC API. Database Operations: Establishing connections, Statement vs. Prepared Statement, handling Result Set, and Transaction management (Commit/Rollback).

TEXTBOOKS

1. H. Schildt, *Java: The Complete Reference*, 12th ed. New York, NY, USA: McGraw-Hill, 2021.
2. J. Bloch, *Effective Java*, 3rd ed. Boston, MA, USA: Addison-Wesley, 2018.

REFERENCE BOOKS:

1. K. Sharan, *Beginning Java 17 Fundamentals*. Berkeley, CA, USA: Apress, 2022.
2. K. Sierra, B. Bates, and T. Gee, *Head First Java: A Brain-Friendly Guide*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2023.
3. Oracle, "The Java™ Tutorials," Oracle Documentation, 2024. [Online]. Available: https://docs.oracle.com/javase/tutorial/
4. C. Horstmann, *Core Java Volume I—Fundamentals*, 12th ed. Hoboken, NJ, USA: Pearson Education, 2022.

LIST OF PRACTICALS

1. Basic Logic: Implement a program to calculate the electricity bill based on slab rates using control statements.
2. Inheritance: Design a "Staff Management" system where different types of employees inherit from a base Staff class.
3. Exceptions: Build a "Withdrawal" system that throws an Insufficient Balance Exception if the amount exceeds the balance.
4. Multithreading: Create two threads—one to print even numbers and one for odd numbers in sequence.
5. Collections: Use a Hash Map to store and retrieve contact details (Name as key, Number as value).
6. File I/O: Write a program to read a text file and count the number of vowels and consonants.

Shri Vaishnav Vidyapeeth Vishwavidyalaya, Indore

Shri Vaishnav Institute of Information Technology

Choice Based Credit System (CBCS) in the light of NEP-2020

B. Tech CSE (MA-Apple, AIML-Microsoft)

SEMESTER-III (2025-29)

COURSE CODE	CATEGORY	COURSE NAME	TEACHING & EVALUATION SCHEME					L	T	P	CREDITS
			THEORY			PRACTICAL					
			END SEM University Exam	Two Term Exam	Teachers Assessment*	END SEM University Exam	Teachers Assessment*				
BTIT410	SEC	Java Full Stack – I	0	0	0	30	20	0	0	2	1

Legends: **L** - Lecture; **T** - Tutorial/Teacher Guided Student Activity; **P** – Practical; **C** - Credit; ***Teacher Assessment** shall be based following components: Quiz/Assignment/ Project/Participation in Class, given that no component shall exceed more than 10 marks.

7. JDBC Integration: Create a "Student Information System" that allows adding, updating, and viewing records in a MySQL database.