S.Y.B.Sc.(Computer Science)

Semester-III

&

Semester-IV

Credit Structure & Syllabus

(Academic Year 2020-2021, Autonomous)

Course Structure for S. Y. B. Sc. (Computer Science) Sem-III & IV
Subject: Computer Science

Sem	Paper Code	Title of Paper	No. of Credits	Exam	Marks
III	CSCO2301	Data Structure Using C	3	I / E	60 + 40
	CSCO2302	Introduction to Web Technology	3	I / E	60 + 40
	CSCO2303	Lab Course – I based On Data structure	2	I / E	60 + 40
	CSCO2304	Lab Course – II Based on Web Technology	Grade	I/E	60 +40
		Certificate Course I	2	---	----
IV	CSCO 2401	Object Oriented Concepts using Java	3	I / E	60 + 40
	CSCO2402	Software Engineering	3	I / E	60 + 40
	CSCO2403	Lab Course – I on CSCO2401	2	I / E	60 + 40
	CSCO2404	Lab Course– II CSCO2402 with S.E. Project.	Grade	I/E	60 + 40
		Certificate Course II	2	----	----
		Environment Science	2	----	----

SYLLABUS (CBCS) FOR S. Y. B. Sc. (Computer Science)
(w.e.f from June, 2020)
Academic Year 2020-2021
Class: S.Y. B. Sc.(Computer Science) (Semester- III)

Subject: Computer Science **Paper Code :** CSCO 2301
Title of Paper: Data Structure Using C **Paper:** I
Credit: 3 (4 Lectures/Week) **No. of lectures:** 48

Prerequisites:

- Basic knowledge of algorithms and problem solving.
- Knowledge of C Programming Language.

Objective:

1. To learn the systematic way of solving problem
2. To understand the different methods of organizing large amount of data
3. To efficiently implement the different data structures
4. To efficiently implement solutions for specific problems

Learning Outcomes: On completion of the course, student will be able to

1. Use well-organized data structures in solving various problems.
2. Differentiate the usage of various structures in problem solution.
3. Implementing algorithms to solve problems using appropriate data structures.

1. Introduction to data structures **[2]**

1.1 Concept
1.2 Data type, Data object, ADT
1.2.1 Data Type
1.2.2 Data Object
1.2.3 ADT -Definition, Operation, examples on rational number
1.3 Need of Data Structure
1.4 Types of Data Structure

2. Algorithm analysis **[3]**

2.1 Algorithm – definition, characteristics
2.2 Space complexity, time complexity
2.3 Asymptotic notation (Big O, Omega Ω, Theta Notation Θ)

3. Linear data structures **[6]**

3.1 Introduction to Arrays - array representation
3.2 Sorting algorithms with efficiency - Bubble sort, Insertion sort, Merge sort, Quick Sort
3.3 Searching techniques –Linear Search, Binary search

4. Linked List **[8]**

4.1 Introduction to Linked List
4.2 Implementation of Linked List – Static & Dynamic representation,
4.3 Types of Linked List
4.4 Operations on Linked List - create, display, insert, delete, reverse, search, sort, concatenate &merge
4.5 Applications of Linked List – polynomial manipulation
4.6 Generalized linked list – Concept and Representation

5. Stacks **[6]**

5.1 Introduction
5.2 Representation- Static & Dynamic
5.3 Operations – Create , Init , Push , Pop & Display
5.4 Application - infix to postfix, infix to prefix, Evaluation of Expression
5.5 Simulating recursion using stack

6. Queues **[4]**

6.1 Introduction
6.2 Representation - Static & Dynamic
6.3 Operations – Create , Init , Insert , Remove & Display
6.4 Circular queue, priority queue (with implementation)
6.5 Concept of doubly ended queue (Dequeue)

7. Trees **[12]**

7.1 Concept & Terminologies
7.2 Binary tree, binary search tree
7.3 Representation – Static and Dynamic
7.4 Operations on BST & Heap Tree – create, Insert, delete, traversals (preorder, inorder, postorder), counting leaf, non-leaf & total nodes , non recursive inorder traversal
7.5 Application - Heap sort
7.6 Height balanced tree- AVL trees- Rotations, AVL tree examples.

8. Graph **[7]**

8.1 Concept & terminologies
8.2 Graph Representation – Adjacency matrix, adjacency list, inverse Adjacency list, adjacency multi list, orthogonal list
8.3 Traversals – BFS and DFS
8.4 Spanning Tree
8.5 Applications – AOV network – topological sort, AOE network – critical path

References:

1. Fundamentals of Data Structures ---- By Horowitz Sahani (Galgotia)
2. Data Structures using C and C++ ----By Yedidyah Langsam, Aaron M. Tenenbaum, Moshe J. Augenstein
3. Introduction to Data Structures using C---By Ashok Kamthane
4. Data Structures using C --- Bandopadhyay & Dey (Pearson)
5. Data Structures using C -- By Srivastav

Class: S.Y. B. Sc.(Computer Science) (Semester- III)

Subject: Computer Science **Paper Code :** CSCO 2302

Title of Paper: Introduction to Web Technology **Paper:** II

Credit: 3 (4 Lectures/Week) **No. of lectures:** 48

OBJECTIVES: Students successfully completing this course will be able:

1. To understand different Web technologies.
2. To keep pace with the rapidly changing landscape of web application development.
3. To Design dynamic and interactive web pages.

Title and Contents		No. of Lectures
Unit 1	**Basics of Web Design** 1.1 History of the Internet 1.2 World Wide Web Consortium (W3C) 1.3 Personal, Distributed and Client/Server Computing 1.4 Key Software Trend: Object Technology 1.5 Browser Portability 1.6 Software Technologies 1.7 Client Server Architecture	04
Unit 2	**Introduction to HTML5** 2.1 Difference between HTML & HTML5 2.2 HTML Document and Basic Structure 2.3 Working with HTML Text, Heading, Paragraph, formatting 2.4 HTML Color, Link, Image 2.5 HTML Lists, Tables and Frames 2.6 HTML Forms Block, Layout	16
Unit 3	**Specific Elements of HTML5** 3.1 Header & Footer 3.2 Navigation Section 3.3 Article & Aside 3.4 The Meter Element 3.5 Working with Hyperlinks and Multimedia 3.6 Working with Forms and controls.	8
Unit 4	**The Basic of CSS** 4.1 Into. Concepts of CSS & Creating of CSS, Using Inline CSS,CSS Color 4.2 Using Internal CSS, Using ID's and Classes, Creating External CSS. 4.3 Linking to External CSS, Inefficient Selectors and Efficient Selectors. 4.4 HTML Elements State, the CSS Box Model, Fonts	12
Unit 5	**Main CSS3.0 Specific Properties** 5.1 Introduction to CSS 3.0 5.2 Alpha Color Space, Opacity 5.3 Box Shadow & Border Radius	6
Unit 6	**Emerging Trends in Web Technologies** 6.1 Introduction to – 6.1.1) CMS-Wordpress/Drupal/Joomla 6.1.2) jQuery 6.1.3) AngularJS 6.1.4) Bootstrap	2

Note: Regular Practical assignment on HTML5 are 50% and one mini Project.

References:

1) Html & CSS: The Complete Reference, Fifth Edition by Thomas A. Powell and published by McGraw Hill.
2) HTML 5 in simple steps by Kogent Learning Solutions Inc., Publisher Dreamtech Press
3) Head First HTML with CSS & XHTML Book by Elisabeth Freeman and Eric Freeman.
4) The Essential Guide to CSS and HTML Web Design Book by Craig Grannell.
5) Beginning XML by Joe Fawcett, Liam R.E. Quin & Danny Ayers Published by John Wiley & Sons, Inc.

Class: S.Y. B. Sc.(Computer Science) (Semester- III)

Subject: Computer Science **Paper Code :** CSCO 2303

Title of Paper: Lab Course I based On Data structure **Paper:** III (Lab Course-I)

Credit: 2 (3 Hour Practical /batch/Week) **No. of Practical's:** 12

Prerequisites:

- Basic knowledge of algorithms and problem solving.
- Knowledge of C Programming Language.

- Assignment 1 – Sorting Algorithms
  - Bubble Sort
  - Insertion Sort
  - Quick Sort
  - Merger Sort

- Assignment 2 - Recursive Sorting Algorithms
  - Quick sort,
  - Merge Sort

- Assignment 3 - Searching Method
  - Linear search,
  - Binary search

- Assignment 4 - Stack
  - Static Stack Implementation
  - Dynamic Stack Implementation

- Assignment 5 Queue
  - Static and Dynamic Implementation
  - Linear Queue,
  - Circular queue

- Assignment 6 Linked List –
  - Dynamic Implementation of Singly Linked List
  - Dynamic Implementation of Doubly Linked List
  - Dynamic Implementation of Circular Linked List.

- Assignment 7 Tree –
  - Binary Search Tree Traversal: Create, add, delete, and display nodes.

- Assignment 8 Graph –
  - Adjacency matrix to adjacency list conversion, in degree, out degree

Class: S.Y. B. Sc.(Computer Science) (Semester- III)

Subject: Computer Science **Paper Code :** CSCO 2304

Title of Paper: Lab Course II based On Web Technology **Paper:** IV (Grade)

Credit: 2 (3 Hour Practical/Week/batch) **No. of Practical:** 13

OBJECTIVES: Students successfully completing this course will be able:

1. To understand different Web technologies.
2. To keep pace with the rapidly changing landscape of web application development.
3. To Design dynamic and interactive web pages.

Learning Outcome:

1. Understand different Web technologies.
2. Keep pace with the rapidly changing landscape of web application development.
3. Design dynamic and interactive web pages.

Assignment on Web Technology Using (HTML5, CSS & Java Script)	
Sr. No.	**Assignment Name**
1	Be acquainted with elements, Tags and basic structure of HTML files.
2	Develop the concept of basic and advanced text formatting.
3	Practice the use of multimedia components in HTML documents.
4	Designing of webpage-Document Layout, Working with List, Working with Tables.
5	Practice Hyper linking, Designing of webpage-Working with Frames, Forms and Controls.
6	Prepare creating style sheet, CSS properties, Background, Text, Font and styling etc.
7	Working with List, HTML elements box, Positioning and Block properties in CSS.
8	Designing with cascading style sheet-Internal and External style sheet.
9	Working with CSS 3.0,Alpha Color Space, Opacity
10	Practice the use JavaScript Basics Programs Data Types, Control Structure
11	Develop the concept of basic and advance using JavaScript Functions
12	Practice the use JavaScript events (onClick, onMousemove and onMouseover events of button object)
13	Designing event driven JavaScript program and use Popup boxes
14	Working with JavaScript Objects

Class: S.Y. B. Sc.(Computer Science) (Semester- IV)

W.e.f. A.Y.- 2020-2021

Subject: Computer Science **Paper Code :** CSCO 2401

Title of Paper: Object Oriented Concepts using Java **Paper:** I

Credit: 3 (4 lectures /Week) **No. of Lectures:** 48

Prerequisites:

- Knowledge of C Programming Language.

Objective:

- To learn Object Oriented Programming language
- To handle abnormal termination of a program using exception handling
- To handle complex problems using object oriented concepts
- To design programs using multithreading

Learning Outcomes: On completion of the course, student will be able to

- Understand Object Oriented Concepts
- Handle different type of Exceptions in program.

1. An Introduction to Java [4]

1.1 A Short History of Java
1.2 Features or buzzwords of Java
1.3 Comparison of Java and C++
1.4 Java Environment
1.5 Simple java program
1.6 Java Tools – jdb, javap, javadoc
1.7 Java IDE – Eclipse/NetBeans (Note: Only for Lab Demonstration)

2. An Overview of Java [4]

2.1 Types of Comments
2.2 Data Types
2.3 Final Variable
2.4 Declaring 1D, 2D array
2.5 Accepting input using Command line argument
2.6 Accepting input from console (Using BufferedReader class)

3. Objects and Classes [8]

3.1 Defining Your Own Classes
3.2 Access Specifiers (public, protected, private, default)
3.3 Array of Objects
3.4 Constructor, Overloading Constructors and use of ‘this’ Keyword
3.5 static block, static Fields and methods
3.6 Predefined class – Object class methods (equals(), toString(), hashcode(), getClass())
3.7 Inner class 3.8 Creating, Accessing and using Packages
3.9 Creating jar file and manifest file
3.10 Wrapper Classes
3.11 Garbage Collection (finalize() Method)
3.12 Date and time processing

4. Inheritance and Interface [7]

4.1 Inheritance Basics (extends Keyword) and Types of Inheritance
4.2 Superclass, Subclass and use of Super Keyword
4.3 Method Overriding and runtime polymorphism
4.4 Use of final keyword related to method and class
4.5 Use of abstract class and abstract methods
4.6 Defining and Implementing Interfaces
4.7 Runtime polymorphism using interface
4.7 Object Cloning

5. Exception Handling [4]

5.1 Dealing Errors

5.2 Exception class, Checked and Unchecked exception
5.3 Catching exception and exception handling
5.4 Creating user defined exception
5.5 Assertions

6. Strings, Streams and Files [7]

6.1 String class and StringBuffer Class
6.2 Formatting string data using format() method
6.2 Using the File class
6.3 Stream classes Byte Stream classes Character Stream Classes
6.4 Creation of files
6.5 Reading/Writing characters and bytes
6.6 Handling primitive data types
6.7 Random Access files

7. Collection [6]

7.1 Introduction to the Collection framework
7.2 List – Array List, Linked List and Vector, Stack, Queue
7.3 Set – Hash Set, Tree Set, and Linked Hash Set
7.4 Map – Hash Map, Linked Hash Map, Hash table and Tree Map
7.5 Interfaces such as Comparator, Iterator, List Iterator, Enumeration

8. Multithreading [8]

8.1 What are threads?
8.2 Life cycle of thread
8.3 Running and starting thread using Thread class
8.4 Thread priorities
8.5 Running multiple threads
8.6 The Run able interface
8.7 Synchronization and interthread communication

References:

1) Complete reference Java by Herbert Schildt (5th edition)
2) Java 2 programming black books, Steven Horlzner
3) Programming with Java , A primer ,Forth edition , By E. Balagurusamy
4) Core Java Volume-I-Fundamentals, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press

Class: S.Y. B. Sc.(Computer Science) (Semester- IV)

Subject: Computer Science **Paper Code:**CSCO2402

Title of Paper: Software Engineering **Paper:** II

Credit: 3 (4 Lectures/Week) **No. of lectures:** 48

Prerequisites:

- Basic knowledge of DBMS & RDBMS.
- Knowledge of C HTML5, CSS & JAVASCRIPT.

Objective:

- To teach basics of System Analysis and Design.
- To teach principles of Software Engineering.
- To teach various process models used in practice.
- To know about the system engineering and requirement engineering.
- To build analysis model.

Learning Outcomes: On completion of the course, student will be able to

- Students can adopt relevant methods and procedures.
- An ability to work in one or more significant application.
- Student Can collect, analyze, and evaluate end user requirement data.
- Using Soft. Eng. methods students are present and develop their own projects.

1. System Concepts [5] (R1: Chapter 1 & R3: Chapter 1)

1.1 System Definition

1.2 Characteristics of a System: Organization, Subsystem, Interaction, Interdependence, Integration, Central objective, Standards, Black box.

1.3 Elements of a system: Outputs, Inputs, Processor(s), Control, Feedback, Environment, Boundaries, Interface.

1.4 Types of Systems: Physical & Abstract Systems, Open & Closed Systems, Computer-based Systems (MIS : Management Information System & DSS : Decision Support System)

2. Software and Software Engineering [5] (R2 : Chapter 1)

2.1 The Nature of Software

2.1.1 Defining Software

2.1.2 Software Application Domains

2.1.3 Legacy Software

2.2 Software Engineering

2.3 The Software Process

2.4 Software Engineering Practice

2.4.1 The Essence of Practice

2.4.2 General Principles

2.5 Software Myths

3. System Development Life Cycle (SDLC) [8] (R3 : Chapter 1)

3.1 Introduction

3.2 Activities of SDLC

3.2.1 Preliminary Investigation (Request Clarification, Feasibility Study, Request Approval)

3.2.2 Determination of System Requirements

- **3.2.3** Design of System
- **3.2.4** Development of Software
- **3.2.5** System Testing (Unit Testing, Integration testing, System Testing)
- **3.2.6** System Implementation &Evaluation
- **3.2.7** System Maintenance

4. Process Models [6] (R2 : Chapter 2)

- **4.1** A Generic Process Model
- **4.2** Prescriptive Process Models
  - **4.2.1** The Waterfall Model
  - **4.2.2** Incremental Process Models
  - **4.2.3** Evolutionary Process Models
    - **4.2.3.1** Prototyping
    - **4.2.3.2** Spiral Model
  - **4.2.4** Concurrent Models

5. Requirements Engineering [8] (R2 : Chapter 5)

- **5.1** Introduction
- **5.2** Requirements Engineering Tasks
  - **5.2.1** Inception , Elicitation , Elaboration , Negotiation , Specification, Validation , Requirements Management
- **5.3** Initiating the Requirements Engineering Process
  - **5.3.1** Identifying the Stakeholders
  - **5.3.2** Recognizing Multiple Viewpoints
  - **5.3.3** Working toward Collaboration
- **5.4** Fact Finding Techniques**(R3 : Chapter 3)**
  - **5.4.1** Interview , Questionnaire , Record Review , Observation

6. Structured Analysis Development Strategy [10] (R3 : Chapter 4)

- **6.1** Structured Analysis
  - **6.1.1** Structured Analysis?
  - **6.1.2** Components of Structured Analysis
  - **6.1.3** Data Flow Analysis?
  - **6.1.4** Features & Tools of Data Flow Analysis
  - **6.1.5** Logical Data Flow Diagram (Logical DFD)
  - **6.1.6** Physical Data Flow Diagram
    - **6.1.6.1** Notations
    - **6.1.6.2** Drawing a Context Diagram
    - **6.1.6.3** Exploding a Context diagram into Greater detail (1^{st} level, 2^{nd} Level DFD etc…)
    - **6.1.6.4** Evaluating Data Flow Diagram for Correctness
  - **6.1.7** A Data Dictionary
    - **6.1.7.1** Concepts of Data Dictionary
    - **6.1.7.2** Importance of Dictionary
    - **6.1.7.3** Function of Data Dictionary

7. An Agile View of Process [6] (R2 : Chapter3)

- **7.1** Introduction Agility?
- **7.2** Introduction to Agile Process?
  - **7.2.1** The Politics of Agile Development
  - **7.2.2** Human Factors
  - **7.2.3** Agile Process Models
  - **7.2.4** Extreme Programming(XP)

7.2.5 Adaptive Software Development(ASD)
7.2.6 Dynamic Systems Development Method(DSDM)

Reference Books:

R1: System Analysis and Design (Second Edition) by Elias M. Awad, Galgotia Publications Pvt. Ltd.

R2: Software Engineering: A Practitioner's Approach (Seventh Edition) by Roger S.Pressman, McGraw Hill InternationalEdition.

R3: Analysis and Design of Information Systems (Second Edition) by James A. Senn, McGraw Hill International Editions.

Class: S.Y.B.Sc.(Computer Science) (Semester- IV)

Subject: Computer Science **Paper Code:** CSCO2403
Title of Paper: Lab course-I On Programming in Java **Paper:** III (Lab Course-I)
Credit: 2 (3 Hour Practical /Batch/Week) **No. of Practicals:** 12

Prerequisites:

- ✓ Basic knowledge of Object Oriented Programming.
- ✓ Knowledge of Java Programming Language.

Learning Outcomes: On completion of the course, student will be able to

- ➢ To write programs using Object Oriented Concepts
- ➢ Write Multithreaded Programs.
- ➢ Handle different type of Exceptions in program.

Sr. No	Assignment Title
1	**Java Tools**
2	**Classes , Methods , Objects**
3	**Array of Objects, Access Modifiers**
4	**Packages**
5	**Inheritance**
6	**Interfaces**
7	**Exception Handling and Assertions**
8	**I/O and String Handling**
9	**File Handling**
10	**Collection**
11	**Multithreading**
12	**Multithreading**

Class: S.Y.B.Sc.(Computer Science) (Semester- IV)

Subject: Computer Science **Paper Code:** CSCO2404

Title of Paper: Lab Course – II on Soft. Engineering using Mini Project

Paper: IV(Grade) (Lab Course-II)

Credit: Grade (3 Hour Practical/Week/batch) **No. of Practical:** 12

Prerequisites:

- Basic knowledge of DBMS & RDBMS.
- Knowledge of programming languages and scripting.

OBJECTIVES:

1. To understand the process of designing and implementing Software.
2. To understand the Software Engineering concept for building different application.

Learning Outcome:

1. Developing practical skill of designed and developed software.
2. Developing skill of analyzing static and dynamic view of software.
3. Developed practical experience of designed a mini project based on system analysis and design.

Sr. No.	Assignment Name
1.	Problem definition, Scope
2.	Feasibility Study
3.	Gathering Data Requirements and Functional Requirement
4.	ER Diagrams
5.	Designing the Normalization Database
6.	Designing the Queries related to Functional requirements
7.	User interface analysis
8.	Screen designed (by using HTML5)
9	Build a Software engineering project through all the above conceptual Ideas.