Anekant Education Society's
Tuljaram Chaturchand College of Arts, Science and Commerce, Baramati
Autonomous

B.Sc.(Computer Science) Academic Year 2019-2020

- F.Y.B.Sc. (Computer Science) Credit Structure -

Subject	Semester I				Semester II				Total
	Paper		Practical Paper		Paper		Practical Paper		
	I	II	III	IV	I	II	III	IV	
Computer Science	2	2	2	Grade	2	2	2	Grade	12
Mathematics	2	2	2	---	2	2	2	---	12
Electronics	2	2	2	---	2	2	2	---	12
Statistics	2	2	2	---	2	2	2	---	12
Physical Education	---	---	---	---	---	---	---	---	2

Total Number of Credits = 48(Core) + 2 (Physical Education) = 50

1 Theory Credit = 18 Lectures

- S.Y.B.Sc. (Computer Science) Credit Structure -

Subject	Semester I				Semester II				Total
	Paper		Practical Paper		Paper		Practical Paper		
	I	II	III	IV	I	II	III	IV	
Computer Science	3	3	2	Grade	3	3	2	Grade	16
Mathematics	3	3	2	---	3	3	2	---	16
Electronics	3	3	2	---	3	3	2	---	16
English	3	---	---	---	3	---	---	---	6
Evs.	---	---	---	---	---	---	---	---	2
Certificate Courses	2				2				4

Total Number of Credits = 54 (Core) + 2 (EVS) + 4 (Certificate Courses) = 60

1 Theory Credit = 16 Lectures

- T.Y.B.Sc. (Computer Science) Credit Structure -

Semester V

Subject	Theory						Practical			Total
	I	II	III	IV	V	VI	Lab Course I	Lab Course II	Lab Course III	
Computer Science	3	3	3	3	3	3	2	2	2	24
Certificate Courses	2									2

- T.Y.B.Sc. (Computer Science) Credit Structure –

Semester VI

Subject	Theory						Practical			Total
	I	II	III	IV	V	VI	Lab Course I	Lab Course II	Lab Course III	
Computer Science	3	3	3	3	3	3	2	2	2	24

Total Number of Credits = 48 (Core) + 2 (Certificate Courses) = 50

1 Theory Credit = 16 Lectures

B.Sc. (Computer Science): Total credits = 50 + 60 + 50 = 160

Class :F.Y.B.Sc. (Computer Science)			
Semester I		**Semester II**	
CSCO 1101	Basic Programming using C	CSCO 1201	Advanced Programming using C
CSCO1102	DBMS-I	CSCO 1202	DBMS-II
CSCO1103	Lab Course I : Basics of C	CSCO 1203	Lab Course I : Advanced C
CSCO1104	Lab Course II : DBMS I	CSCO1204	Lab Course II : DBMS II
Physical Education			

Class :S.Y.B.Sc. (Computer Science)			
Semester III		**Semester IV**	
CSCO 2301	Data Structures using C	CSCO2401	Object Oriented Concepts using Java
CSCO2302	Introduction to Web Technology	CSCO2402	Software Engineering
CSCO2303	Lab Course I : Based On CSCO2301	CSCO2403	Lab Course I: Based On 2401
CSCO2304	Lab Course II: based On CSCO2302	CSCO2404	Lab Course II : Based On CSCO2402 with Mini Project
Certificate Course I		Certificate Course II	
Environment Science (EVS) An Educational Trip conduct in IV semester			

Class :T.Y.B.Sc. (Computer Science)			
Semester V		**Semester VI**	
CSCO3501	System Programming &Operating System	CSCO3601	Advanced Operating System
CSCO 3502	Theoretical Computer Science	CSCO3602	Compiler Construction
CSCO3503	Computer Networks - I	CSCO3603	Computer Networks - II
CSCO3504	Web Development – I	CSCO3604	Web Development–II
CSCO3505	Advanced Programming in Java	CSCO3605	Advanced Java Technologies – Frameworks
CSCO3506	Object Oriented Software Engineering	CSCO3606	Software Metrics & Project Management
CSCO3507	Lab Course I: Based on CSCO3501	CSCO3607	Lab Course I: Based on CSCO3601
CSCO3508	Lab Course II: Based on CSCO3505	CSCO3608	Lab Course II: Based on CSCO3605 & Mini Project using JAVA
CSCO3509	Lab Course III: Based on CSCO3504	CSCO3609	Lab Course III: Based on CSCO3604 & Mini Project using PHP.
Certificate Course III		An Educational Trip conduct in this semester.	

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

Semester-I
&
Semester-II

Credit Structure & Syllabus

(Academic Year 2019-2020, Autonomous)

Course Structure for F. Y. B. Sc. (Computer Science)
Subject: Computer Science

Sem	Paper Code	Title of Paper	No. of Credits	Exam	Marks
I	CSCO1101	Basic Programming using C	2	I / E	60 + 40
	CSCO 1102	DBMS – I	2	I / E	60 + 40
	CSCO1103	Lab Course – I Basics on C	2	I / E	60 + 40
	CSCO1104	Lab Course – II Based on DBMS I	Grade	I/E	60 +40
II	CSCO 1201	Advanced Programming using C	2	I / E	60 + 40
	CSCO 1202	DBMS – II	2	I / E	60 + 40
	CSCO 1203	Lab Course – I Advanced C Prog.	2	I / E	60 + 40
	CSCO1204	Lab Course– II DBMS II (PL/PgSql)	Grade	I/E	60 + 40
		Physical Education	2	----	----

SYLLABUS (CBCS) FOR F. Y. B. Sc. (Computer Science)
(w.e.f from June, 2019)
Academic Year 2019-2020

Class : F.Y. B. Sc.(Computer Science) (Semester- I)

Subject : Computer Science   Paper Code : CSCO1101

Title of Paper: Basic Programming Using C   Paper : I

Credit: 2   No. of lectures: 36

Learning Objectives: Students successfully completing this course will be able:

1. To understand and design algorithm for problem solving
2. To develop Problem Solving abilities using computers
3. To develop skills for writing programs using 'C'

Learning Outcome: Problem solving and programming capability.

Chapter	**Topic Contents**	**No. of Lectures**
Unit – I	**Problem-Solving Using Computer** 1.1 Problem Solving 1.2 Algorithms & Flowcharts (More Problems covered) 1.3. Programming Languages Machine language Assembly language High level languages	8
Unit – II	**Introduction to C** 2.1 History 2.2 Structure of a C program 2.3 Application Areas 2.4 C Program development life cycle 2.5 Sample programs	2
Unit – III	**C Tokens** 3.1 Keywords 3.2 Identifiers 3.3 Variables 3.4 Constants – character, integer, float, string, escape sequences 3.5 Data types – built-in and user defined 3.6 Operators and Expressions Operator types (arithmetic, relational, logical,assignment, bitwise, conditional , other operators), precedence and associatively rules.	5
Unit – IV	**Control Structures** 4.1 Decision making structures If, if-else, switch 4.2 Loop Control structures While, do-while, for 4.3 Nested structures 4.4 break, continue and goto	8

Unit – V	**Functions in C** 5.1 What is a function 5.2 Advantages of Functions 5.3 Standard library functions 5.4 User defined functions :Declaration, definition, function call, parameter passing (by value), return keyword 5.5 Scope of variables, storage classes 5.6 Recursion	8
Unit – VI	**Arrays** 6.1 Array declaration, initialization 6.2 Types – one, two and multidimensional 6.3 Passing arrays to functions	5

References:

1. Yashavant Kanetkar : Let Us C 7th Edition, PBP Publications
2. E Balaguruswamy : Programming in ANSI C 4th Edition, Tata Mc-Graw Hill Publishing Co.Ltd.-New Delhi
3. Brian W. Kernighan and Dennis M. Ritchie : The C Programming Language 2nd Edition, Prentice Hall Publication
4. The Complete Reference to C, Herbert Schildt
5. Problem Solving with C, Harrow
6. Yeshwant Kanitkar :Graphics using C- BPB Publication.

F.Y. B. Sc. (Computer Science) (Semester- I) (wef. 2019-20)

Subject : Computer Science **Paper Code**: **CSCO1102**
Title of Paper: DBMS-I **Paper** : II
Credit: 2 **No. of lectures**: 36

Learning objective: Students successfully completing this course will be able to:

- Understand design and implementation of a database system.
- Study the physical, logical database designs and database modeling.
- Understanding and development for essential DBMS concepts.
- Understand creations, manipulation and querying of data in databases.

Learning Outcomes:

- Master the basics of database concepts and database management system
- Model an application's data requirements using conceptual modeling tools like ER model, relational model.
- Write SQL commands to create tables, insert, update, delete and querying data.

Units	Title & Content	No. Of lecture
Unit I	**1. Introduction to File organization & DBMS** 1.1 Introduction 1.2 Types of file organization 1.3 File system Vs DBMS 1.4 Data models 1.5 Levels of abstraction 1.6 Data independence 1.7 Structure of DBMS 1.8 Users of DBMS 1.9 Advantages of DBMS	**04**
Unit II	**2.Conceptual Design (E-R model)** 2.1 Overview of DB design 2.2 ER data model (entities , attributes, entity sets, relations,relationship sets) 2.3 Additional constraints (Key constraints, Mapping constraints), 2.4 Conceptual design using ER modelling 2.4 Case studies	**10**
Unit III	**3. Relational data model** 3.1 Structure of Relational Databases (concepts of a table, a row, a relation, a Tuple and a key in a relational database) 3.2 Conversion of ER to Relational model 3.3. Integrity constraints (primary key, referential integrity, unique constraint, Null constraint, Check constraint)	**04**
Unit IV	**4. Relational algebra** 4.1 Preliminaries 4.2 Relational algebra (selection, projection set operations, renaming, joins, division) 4.3 Problems.	**04**
Unit V	**5. Introduction to SQL** 5.1 Introduction 5.2 Basic structure 5.3 Set operations 5.4 Aggregate functions 5.5 Null values 5.6 PL/PgSqL: Data types, Language structure	**08**

Unit VI	**6. Operations with SQL** 6.1 Nested Subqueries 6.2 Modifications to Database 6.3 DDL and DML commands with examples 6.4 SQL mechanisms for joining relations (inner joins, outer joins and their types) 6.5 Examples on SQL (case studies)	**06**

References

1. Shamkant B. Navathe, Ramez Elmasri, Database Systems, ,ISBN:9780132144988, PEARSON HIGHER EDUCATION
2. Richard Stones, Neil Matthew, Beginning Databases with PostgreSQL: From Novice to Professional, ISBN:9781590594780, Apress
3. Korry Douglas,PostgreSQL, ISBN:9780672327568, Sams
4. ,JohnWorsley, Joshua Drake , Practical PostgreSQL(BCD),ISBN:9788173663925 Shroff/O'reilly
5. Joshua D. Drake, John C Worsley , Practical Postgresql , **(O’Reilly publications)**
6. Bipin C Desai , “An introduction to Database systems” , Galgotia Publications
7. Henrey Korth, Sudarshan, Silberschatz “Database System Concepts” (4th Ed), McGraw Hill,.

F.Y. B. Sc.(Computer Science) **Semester I (wef. 2019-20)**

Subject	: Computer Science	Paper Code	: CSCO1103
Title of Paper: Lab Course – I (Basic C)		Paper	: III (Lab Course-I)
Credit: 2		No. of Practical	: 10 /Semester

Learning Objectives: Students successfully completing this course will be able to:

1. Design and implement a 'C' programs for different problems
2. Understand appropriate use of language structure.

Learning Outcome: Problem solving and programming capability.

Semester I (Credits – 02) No. of Practicals – 10)	
	Title of Experiment/ Practical
1	Assignment to demonstrate use of data types, simple operators & expressions.
2	Assignment to demonstrate decision making statements (if and if-else, nested structures)
3	Assignment to demonstrate decision making statements (switch - case)
4	Assignment to demonstrate use of simple loops
5	Assignment to demonstrate use of nested loops
6	Assignment to demonstrate menu driven programs.
7	Assignment to demonstrate writing C programs in modular way (use of user defined functions)
8	Assignment to demonstrate recursive functions.
9	Assignment to demonstrate use of arrays (1-d arrays) and functions
10	Assignment to demonstrate use of arrays (1-d arrays) and functions

F.Y. B. Sc.(Computer Science) **Semester I (wef. 2019-20)**

Subject	: Computer Science	Paper Code	: CSCO1104 **(Grade)**
Title of Paper	: Lab Course – II (**DBMS I**)	Paper	: IV(Lab Course-II)
Credit	: **Grade**	No. of Practical	: 10 /Semester

Learning Objectives: Students successfully completing this course will be able to:

- Define & manipulate the database Concepts.
- Understand SQL with DDL and DML Commands.

Learning Outcome: To know the DBMS Concepts and to operate Database Software.

Semester I No. of Practicals – 10	
	Title of Experiment/ Practical
1	Create simple tables , with only the primary key Constraint
2	Create more than one table with integrity constraint
3	Create more than one table, with referential integrity constraint.
4	Drop a table from database, Alter the table.
5	Insert/Update/Delete statements.
6	Query for the tables using simple form of Select Statement
7	Query solving for tableoperations(Aggregate function)
8	Nested Query solving for tableoperations(Union, Intersect, Except)
9	Nested Query solving for tableoperations(Set membership, Cardinality, Comparison)
10	To Small Case Studies.

Class : F.Y. B. Sc. (Computer Science) (Semester- II)
Subject : Computer Science Paper Code: CSCO1201
Title of Paper: Advanced Programming using C Paper : I
Credit: 2 No. of lectures: 36

Learning Objectives: Students successfully completing this course will be able to:

1. Understand and design Programs through advanced C Concepts
2. Design graphics Programming.

Learning Outcome: To develop advanced as well as Graphics programming capability.

Units	**Topics Contents**	**No. of Lectures**
Unit – I	**Pointers** 1.1 Pointer declaration, initialization 1.2 Dereferencing pointers 1.3 Pointer arithmetic 1.4 Pointer to pointer 1.5 Arrays and pointers 1.6 Functions and pointers – passing pointers to functions, function returning pointers 1.7 Dynamic memory allocation	8
Unit – II	**Strings** 2.1 Declaration and initialization, format specifiers 2.2 Standard library functions 2.3 Strings and pointers 2.4 Array of strings 2.5 Command Line Arguments	6
Unit – III	**Structures and Unions** 3.1 Creating structures 3.2 Accessing structure members (dot Operator) 3.3 Structure initialization 3.4 Typedef 3.5 Array of structures 3.6 Passing structures to functions 3.7 Nested structures 3.8 Pointers and structures 3.9 Self referential structure 3.10 Unions 3.11 Difference between structures and unions	10
Unit – IV	**File Handling** 4.1 Streams 4.2 Types of Files 4.3 Operations on files 4.4 Random access to files	6

Unit – V	**C Preprocessor** 4.1 Format of Preprocessor directive 4.2 File Inclusion directive 4.3 Macro substitution, nested macro, augmented macro	2
Unit – VI	**Graphics programming using C** 6.1 Graphics driver and mode 6.2 Drawing simple graphical objects – line, circle, rectangle etc. 6.3 Outputting text, curves & Polygons	4

References:

1. Yashavant Kanetkar : Let Us C 7th Edition, PBP Publications
2. E Balaguruswamy : Programming in ANSI C 4th Edition, Tata Mc-Graw Hill Publishing Co. Ltd.-New Delhi
3. Brian W. Kernighan and Dennis M. Ritchie : The C Programming Language 2nd Edition, Prentice Hall Publication
4. Herbert Schildt,The Complete Reference to C,
5. Harrow , Problem Solving with C

Class : F.Y. B. Sc.(Computer Science) (Semester- II)
Subject : Computer Science Paper Code : CSCO1202
Title of Paper: DBMS-II Paper : II
Credit: 2 No. of lectures: 36

Prerequisites: Knowledge of DBMS

Learning Objectives:-Students successfully completing this course will be able to:

- Understand fundamental concepts of RDBMS (PL/PgSQL)
- Understand data security and its importance
- Understand client server architecture

Learning Outcomes:

- Develop the database design by normalization.
- Knowing functional dependencies and design of the relational database.
- Design concept of Transaction and Query processing.

Unit	Title & Content	No. Of lecture
Unit I	1. **Relational Database Design** 1.1 Preliminaries 1.2 Normalization (1NF,2NF,3NF,BCNF,4NF, 5 NF) 1.3 Controlling the program flow, conditional statements, loops 1.4 Handling errors and exceptions, Cursors 1.5 Views, Stored Functions, Stored Procedures, Triggers	**12**
Unit II	2 **Transaction Concepts and concurrency control** 2.1 Transaction, properties of transaction, state of the transaction. 2.2 Executing transactions concurrently associated problem in concurrent execution. 2.3 Schedules, types of schedules,Serializability, precedence graph for Serializability. 2.4 Ensuring Serializability by locks, different lock modes, 2PL and its variations. 2.5Basic timestamp method for concurrency, Thomas Write Rule. 2.6Locks with multiple granularity, dynamic database concurrency (Phantom Problem). 2.7 Timestamps versus locking. 2.8 Deadlock handling methods 2.8.1 Detection and Recovery (Wait for graph). 2.8.2 Prevention algorithms (Wound-wait, Wait-die)	**10**
Unit III	3 **Database Integrity and Security Concepts** 3.1 Domain constraints 3.2 Referential Integrity 3.3 Introduction to database security concepts 3.4 Methods for database security 3.4.1Discretionary access control method 3.4.2Mandatory access control and role base access control for multilevel security. 3.5 Use of views in security enforcement. 3.6 Overview of encryption technique for security. 3.7 Statistical database security.	**06**
Unit IV	4 **Crash Recovery** 4.1 Failure classification 4.2 Recovery concepts 4.3 Log base recovery techniques (Deferred and Immediate update) 4.4 Checkpoints	**04**

	4.5 Recovery with concurrent transactions (Rollback, checkpoints, commit) 4.6 Database backup and recovery from catastrophic failure.	
Unit V	5. **Client-Server Technology** 5.1 Describe client-server computing. 5.2 Evolution of Client - Server information systems. 5.3 Client – Server Architecture benefits. 5.4 Client Server Architecture - Components, Principles, Client Components, Communication middleware components, Database middleware components, Client Server Databases	**04**

References:-

1. Elmasri and Navathe,Fundamentals of Database Systems (4th Ed)
2. Henrey Korth, Sudarshan, Silberschatz,Database System Concepts (4th Ed)
3. Practical PostgreSQL O’REILLY
4. Richard Stones , Neil Matthew,Beginning Databases with PostgreSQL, From Novice to Professional, 2ndEdition, Apress

Class	: F.Y. B. Sc.(Computer Science) **Semester II**		
Subject	: Computer Science	Paper Code	: CSCO1203
Title of Paper:	Lab Course –I (Advanced C)	Paper	: III (Lab Course-I)
Credit: 2		No. of Practicals	: 10 /Semester

Learning Objectives: Students successfully completing this course will be able to:

1. Design and implement a 'C' programs for different problems
2. Understand use of appropriate Graphics Functions.

Learning Outcome: Problem solving and programming capability and develop Advanced as well as Graphics programming capability.

Semester II (Credits – 02) No. of Practicals – 10)	
	Title of Experiment/ Practical
1	Assignment to demonstrate use of pointers.
2	Assignment to demonstrate concept of strings (string & pointers)
3	Assignment to demonstrate array of strings.
4	Assignment to demonstrate use of bitwise operators.
5	Assignment to demonstrate structures and unions.
6	Assignment to demonstrate structures (using array and functions).
7	Assignment to demonstrate command line arguments and preprocessor directives.
8	Assignment to demonstrate file handling (text files & binary files)
9	Assignment to demonstrate graphics programming.
10	C Programming – Case study (Menu Driven Application Base) in Groups

Class	: F.Y. B. Sc.(Computer Science) **Semester II**		
Subject	: Computer Science	Paper Code	: CSCO1204 (Grade)
Title of Paper:	Lab Course – II (**DBMS II)**	Paper	: IV (Lab Course-II)
Credit	: Grade	No. of Practicals:	10 /Semester

Learning Objectives: Students successfully completing this course will be able to:

- Understand SQL with DDL and DML Commands.
- Understand RDBMS concepts.

Learning Outcome: To know the RDBMS Concepts and to operate Database Software.

Semester II No. of Practicals – 10	
	Title of Experiment/ Practical
1	Simple Queries
2	Nested Queries
3	Queries using aggregate functions
4	Queries using Views
5	Cursors
6	Exception Handling
7	Stored Function
8	Triggers
9	Case Study(1)
10	Case Studies (2)

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 Software Technologies 1.6 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 2.7 Browser Portability	12
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.	9
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 4.5 Introduction to CSS 3.0 4.6 Alpha Color Space, Opacity	16
Unit 5	**JavaScript** 5.1 Introduction to JavaScript 5.2 JavaScript Basics – Data Types, Control Structure 5.3 JavaScript Functions 5.4 Working with events 5.5 JS Popup boxes 5.6 JavaScript Objects	3
Unit 6	**Emerging Trends in Web Technologies** 6.1 Introduction to –	4

	6.1.1) CMS-Wordpress/Drupal/Joomla 6.1.2) jQuery 6.1.3) AngularJS 6.1.4) Bootstrap	

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 (1st level, 2nd 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.

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

Semester- V
&
Semester-VI

Credit Structure & Syllabus

(Academic Year 2021-2022, Autonomous)

Course Structure for T. Y. B. Sc. (Computer Science) Sem-V & VI
Subject: Computer Science

Sem	Paper Code	Title of Paper	No. of Credits	Exam	Marks
V	CSCO3501	System Programming &Operating System	3	I / E	60 + 40
	CSCO 3502	Theoretical Computer Science	3	I / E	60 + 40
	CSCO3503	Computer Networks - I	3	I / E	60 + 40
	CSCO3504	Web Development – I	3	I / E	60 + 40
	CSCO3505	Advanced Programming in Java	3	I / E	60 + 40
	CSCO3506	Object Oriented Software Engineering	3	I / E	60 + 40
	CSCO3507	Lab Course I: Based on CSCO3501	2	I / E	60 + 40
	CSCO3508	Lab Course II: Based on CSCO3505	2	I / E	60 + 40
	CSCO3509	Lab Course III: Based on CSCO3504	2	I/E	60 +40
		Certificate Course - III	2	---	----
VI	CSCO3601	Advanced Operating System	3	I / E	60 + 40
	CSCO3602	Compiler Construction	3	I / E	60 + 40
	CSCO3603	Computer Networks - II	3	I / E	60 + 40
	CSCO3604	Web Development–II	3	I / E	60 + 40
	CSCO3605	Advanced Java Technologies – Frameworks	3	I / E	60 + 40
	CSCO3606	Software Metrics & Project Management	3	I / E	60 + 40
	CSCO3607	Lab Course I: Based on CSCO3601	2	I / E	60 + 40
	CSCO3608	Lab Course II: Based on CSCO3605 & Mini Project using JAVA	2	I / E	60 + 40
	CSCO3609	Lab Course III: Based on CSCO3604 & Mini Project using PHP.	2	I/E	60 +40

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- V)
(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-V)	**Paper Code:** CSCO3501
Title of Paper: System Programming &Operating System	**Paper:** I
Credit: 3 (4 Lectures/Week)	**No. of lectures:** 48

Aim: To understand the design and implementation issues of System programs that play an important role in program development. And also to understand the design and implementation issues of Operating System.

Objectives:

- To understand the design structure of Assembler and macro processor for an hypothetical simulated computer.
- To understand the working of linkers and loaders.
- To understand Complexity of Operating system as a software. .
- To understand design issues related to process management and various related algorithms
- To understand design issues related to memory management and various related algorithms
- To understand design issues related to file management and various related algorithms.

Learning Outcome:

After the completion of this course student to should understand the basic structure of Operating System

Unit No.	Chapter name with Topics	No. of Lectures Required
1.	**Introduction to System Programming** 1.1. Types of program – System program and Application program. 1.2. Difference between system programming and application programming. 1.3. Elements of Programming environment - Editor, Preprocessor, Assembler, Compiler, Interpreter, Linker and Loader, Debugger, Device drivers, Operating System. 1.4. Simulation of simple computer smac0 (hypothetical computer) - Memory, Registers, Condition Codes, Instruction format, Instruction Set, smac0 programs.	08
2.	**Operating System as System Software** 2.1 What Operating Systems Do – User View, System View, Defining OS 2.2 Computer System Architecture – Single processor system, Multiprocessor systems, Clustered Systems 2.3 Operating System Operations – Dual mode operation, Timer 2.4 Process Management 2.5 Memory Management 2.6 Storage Management – File system management, Mass storage management, Cashing, I/O systems 2.7 Protection and Security 2.8 Distributed Systems 2.9 Special Purpose System – Real time embedded systems, Multimedia systems, Handheld systems, 2.10 Computer Environment – Traditional computing, Client server computing, Peer to peer Computing	06

3.	**System Structure** 3.1 Operating System Services 3.2 User Operating-System Interface – Command interpreter, GUI 3.3 System Calls 3.4 Types of System Calls – Process control, File management, Device management, Information maintenance, Communication, Protection	**02**
4.	**Process Management** 4.1 Process Concept – The process, Process states, Process control block. 4.2 Process Scheduling – Scheduling queues, Schedulers, context switch 4.3 Operations on Process – Process creation with program using fork(), Process termination 4.4 Inter-process Communication – Shared memory system, Message passing systems.	**05**
5.	**Process Scheduling** 5.1 Basic Concept – CPU-I/O burst cycle, CPU scheduler, Preemptive scheduling, Dispatcher 5.2 Scheduling Criteria 5.3 Scheduling Algorithms – FCFS, SJF, Priority scheduling, Round-robin scheduling, Multiple queue scheduling, Multilevel feedback queue scheduling. 5.4 Multithreaded Programming 5.5 Multithreading Models 6.6 Thread Scheduling	**10**
6.	**Multithreaded Programming** 6.1 Overview 6.2 Multithreading Model 6.3 Thread Libraries P-Tread, Java Thread 6.4 Thread Life Cycle	**06**
7.	**Process Synchronization** 7.1 Background 7.2 Critical Section Problem 7.3 Semaphores: Usage, Implementation 7.4 Classic Problems of Synchronization – The bounded buffer problem, The reader writer problem, The dining philosopher problem	**04**
8.	**Deadlocks** 8.1 System model 8.2 Deadlock Characterization – Necessary conditions, Resource allocation graph 8.3 Deadlock Prevention 8.4 Deadlock Avoidance - Safe state, Resource allocation graph algorithm, Banker’s Algorithm 8.5 Deadlock Detection 8.6 Recovery from Deadlock – Process termination, Resource preemption	**08**

Reference Books:

1. Siberchatz, Galvin, Gagne Operating System Concepts (8th Edition).

2. Pabitra Pal ChoudharyOperating Systems : Principles and Design (PHI Learning Private Limited)

SYLLABUS (CBCS) FOR T. Y. B. Sc. (Computer Science) Sem-V
(w.e.f June, 2021) Academic Year 2021-2022

Class	**:** T.Y. B. Sc.(Computer Science) (Semester- V)	**Paper Code**	**:** CSCO3502
Subject :	Theoretical Computer Science	**Paper** :	II
Credit	**:** 3 (4 Lectures/week)	**No. of lectures :**	48

Prerequisite:

- Sets, Operations on sets, Finite & infinite sets Formal Language
- Relation, Equivalence Relation,(reflexive, transitive and symmetric closures)

Learning Objectives: Students successfully completing this course will be able:

- To have an understanding of finite state and pushdown automata.
- To have a knowledge of regular languages and context free languages.
- To know the relation between regular language, context free language and corresponding recognizers.
- To study the Turing machine and classes of problems.

Learning Outcome: Knowledge of automata, formal language theory and computability

Units	Topic Contents	No. of Lectures
Unit -I	**Finite Automata** 2.1 Deterministic finite Automaton – Definition, DFA as Language recognizer, DFA as a pattern recognizer. 2.2 Nondeterministic finite automaton – Definition and Ex.. 2.3 NFA TO DFA 2.4 NFA with ε- transitions Definition and Examples. 2.5 NFA with ε-Transitions to DFA & Examples 2.6 Finite automaton with output – Mealy and Moore machine, Definition and Examples. 2.7 Minimization of DFA, Algorithm & Problem using Table Method.	**15**
Unit –II	**Regular Languages** 3.1 Regular language-Definition and Examples. 3.2 Conversion of RE To FA-Examples. 3.3 Pumping lemma for regular languages and applications. 3.4 Closure properties of regular Languages (Union, Concatenation, Complement, Intersection and Kleene closure)	**5**
Unit – III	**Context Free Grammar and Languages** 4.1 Grammar - Definition and Examples. 4.2 Derivation-Reduction - Definition and Examples. 4.3 Chomsky Hierarchy. 4.4 CFG : Definition & Examples. LMD, RMD, ,Parse Tree 4.5 Ambiguous Grammar : Concept & Examples. 4.6 Simplification of CFG : 4.6.1 Removing Useless Symbols, 4.6.2 Removing unit productions 4.6.3 Removing ϵ productions & Nullable symbols 4.7 Normal Forms : 4.7.1 Chomsky Normal Form (CNF) Method & Problem 4.7.2 Greibach Normal form (GNF) Method & Problem 4.8 Regular Grammar : Definition. 4.8.1 Left linear and Right Linear Grammar-Definition and Example.	**12**

	4.8.2 Equivalence of FA & Regular Grammar 4.8.2.1 Construction of regular grammar equivalent to a given DFA 4.8.2.2 Construction of a FA from the given right linear grammar 4.9 Closure Properties of CFL's(Union, concatenation and Kleen closure) Method and examples	
Unit- IV	**Push Down Automaton** 5.1 Definition of PDA and examples 5.2 Construction of PDA using empty stack and final State method : Examples using stack method 5.3 Definition DPDA & NPDA, their correlation and Examples of NPDA 5.4 CFG (in GNF) to PDA : Method and examples	**6**
Unit – V	**Turing Machine** 6.1 The Turing Machine Model and Definition of TM 6.2 Design of Turing Machines 6.3 Problems on language recognizers. 6.4 Language accepted by TM 6.5 Recursive Languages 6.5.1. Recursive and Recursively enumerable Languages. 6.5.2. Difference between recursive and recursively enumerable language. 6.6 Turing Machine Limitations 6.7 Decision Problem, Undecidable Problem, Halting Problem of TM	**10**

References :-

1. Introduction to Automata theory, Languages and computation By John E. Hopcroft and Jeffrey Ullman – Narosa Publishing House.
2. Introduction to Automata theory, Languages and computation By John Hopcroft, Rajeev Motwani and Jeffrey Ullman –Third edition Pearson Education
3. Introduction to Computer Theory Daniel I. A. Cohen – 2nd edition – John Wiley & Sons
4. Theory of Computer Science (Automata, Language & Computation) K. L. P. Mishra & N. Chandrasekaran, PHI Second Edition
5. Introduction to Languages and The Theory of Computation John C. Martin TMH, Second Edition

SYLLABUS (CBCS) FOR T.Y.B.Sc. (Computer Science) (SEM-V)
(w.e.f. A.Y.-2021-2022)

Class: T.Y.B.Sc. (Computer Science)(Semester-V) **Paper Code:** COMP3503

Title of paper: Computer Network-I **Paper:** III

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

Pre-requisites: Basics knowledge of computer

Objectives: This course will prepare students in Basic networking concepts.

1. Understand different types of networks, various topologies and application of networks.

2. Understand types of addresses, data communication.

3. Understand the concept of networking models, protocols, functionality of each layer.

4. Learn basic networking hardware and tools.

5. Understand wired and wireless networks, its types, functionality of layer.

Learning Outcomes : Equip with knowledge and learn the skills necessary to support for their Career in Network Security

Units No.	Title & Contents	No. of Lectures
I	**Introduction to Computer Network** Computer Networks- Goals, applications Network Hardware's – Broadcast and point to point. Topology – Star, Bus, Mesh, Ring etc. Network Types : LAN, MAN, WAN, Wireless Network, internetwork Data Communication – Definition, Components, data representation, Data flow. , Protocols and Standards Defacto, Dejure standard Network Software- Protocol Hierarchies, Design issues of the layer, Connection and connectionless services,	08
II	**Network Models** Reference Model – OSI Reference Model, TCP/IP Reference Model, Comparison of OSI & TCP/IP Model, Addressing – Physical, Logical and Port addresses	04
III	**Transmission Media** Guided Media – Twisted pair cable, Coaxial Cable, Fiber optic cable Unguided Media – Radio Waves, Micro wave Transmission, Infrared, Light wave Transmission	04
IV	**Lower layers : Physical and Data link layers** Communication at the physical layer, Data and signals. Transmission Impairment, Data rate limits, Performance Transmission Modes. Switching – Circuit, Message and Packet Switching. Design issues of Data Link Layer, Services – Framing, Error control, Flow Control, Congestion Control, Link layer addressing. Data link Protocols – simplex, stop and wait and stop and wait Automatic Repeat Request (ARQ). Sliding Window Protocols – One-bit sliding window protocol, Pipeline technique, Go back N and Selective Repeat Automatic Repeat Request with comparison. DLL Protocols – HDLC, PPP Physical and Data link layer devices – Repeater, Hubs, Bridge	18

V	**The Medium Access Sub layer** Introduction., Random Access Protocols – ALOHA – Pure & Slotted CSMA – 1 Persistent, P-persistent and non-persistent CSMA/CD, CSMA/CA. ,Controlled Access – Reservation, Polling and Token Passing, Channelization – FDMA, TDMA, CDMA	07
VI	**Wired and Wireless LAN** IEEE Standards, changes in the standard – bridged Ethernet, switched Ethernet, full duplex Ethernet. Fast Ethernet, Gigabit Ethernet, Ten-Gigabit Ethernet: Goals, MAC Sublayer, Topology and Implementation. Backbone Network – Bus backbone, Star backbone, Remote LANs Virtual LANs: Membership, configuration, communication, Advantages. Wireless LAN - IEEE 802.11 Architecture – BSS, ESS, Station Types, Bluetooth Architecture – Piconet, Scatternet	07

Reference Books:

1) Computer Networks by Andrew Tanenbaum, Pearson Education.[4th Edition]

2) Data Communication and Networking by Behrouz Forouzan,
TATA McGraw Hill. [4th/5th Ed.]

3) Networking All In One Dummies Wiley Publication.[5th Edition]

SYLLABUS (CBCS) FOR T.Y.B.Sc. (Computer Science) (Semester-V)

(w.e.f. from Academic Year 2021-2022)

Class : T.Y.B.Sc. (Computer Science) (Sem-V)	**Paper Code :** CSCO3504
Title of Paper : Web Development-I	**Paper :** IV
Credits : 03 (4 Lectures/Week)	**No. of lectures :** 48

Prerequisite : Know HTML Programming

Objectives :

- To design dynamic, interactive web pages.
- To learn the server side scripting language.
- To learn database connectivity with PHP

Outcome : On completion of the course, student will be able to understand how to develop dynamic and interactive web pages.

Chapter No.	Chapter name with Topics	No. of Lectures Required
1.	**Introduction to PHP** 1.1 HTTP basics, Web Server, Web Browser 1.2 Introduction to PHP(Why PHP?) 1.3 What does PHP do? 1.4 Lexical structure 1.5 Language basics	**04**
2.	**Function and String** 2.1 Defining and calling a function 2.2 Default parameters 2.3 Variable parameters, Missing parameters 2.4 Variable function, Anonymous function 2.5 Types of strings in PHP 2.6 Printing functions 2.7 Encoding and escaping 2.8 Comparing strings 2.9 Manipulating and searching strings 2.10 Regular Expressions	**08**
3.	**Arrays** 3.1 Indexed Vs Associative arrays 3.2 Identifying elements of an array 3.3 Storing data in arrays 3.4 Multidimensional arrays 3.5 Extracting multiple values 3.6 Converting between arrays and variables 3.7 Traversing arrays 3.8 Sorting 3.9 Action on entire arrays 3.10 Using arrays	**06**
4.	**Introduction to Object Oriented Programming** 4.1 Classes and Objects 4.2 Inheritance 4.3 Interfaces 4.4 Encapsulation 4.5 Traits	**16**

	4.6 Autoloading classes 4.7 Exception handling 4.8 Predefined exceptions 4.9 Namespaces in OOP in PHP 4.10 Predefined PHP classes and interfaces	
5.	**Databases (PHP-PostgreSQL)** 5.1 Introduction to PDO 5.2 Installing PDO 5.3 Predefined constants 5.4 Supported databases 5.5 The PDO class 5.6 PDO class methods 5.7 Security using PDO 5.8 PDOStatement class 5.9 Create, Read, Update and Delete (CRUD) operations	**14**

References :

1. Kevin Tatroe, Peter MacIntyre (2020), Programming PHP : Creating Dynamic Web Pages(4th ed.). O'Reilly.

Web References :

1. https://www.php.net/manual/en/manual.php
2. https://www.php-fig.org/
3. https://phptherightway.com
4. https://w3schools.com

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- V)
(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-V) **Paper Code:** CSCO3505

Title of Paper: Advanced Java Programming **Paper:** V

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

Aim: Advanced Java is everything that goes beyond Core Java – most importantly the APIs defined in Java Enterprise Edition, includes Swing, Database Servlet programming, Web Services, the Persistence API, etc. It is a Web & Enterprise application development platform which basically follows client & server architecture.

Objectives:

- To learn Swing and Database programming using Java
- To study web development concept using Servlet and JSP
- To learn socket programming concept

Learning Outcome:

After the completion of this course student is capable to develop standalone computer application as well as web-based application.

Unit No.	Chapter name with Topics	No. of Lectures Required
1.	**User Interface Components with AWT and Swing** 1.1 What is AWT ? What is Swing? Difference between AWT and Swing. 1.2 The MVC Architecture and Swing 1.3 Layout Manager and Layouts, The JComponent class 1.4 Components – JButton, JLabel, JText, JTextArea, JCheckBox and JRadioButton, JList, JComboBox, JMenu and JPopupMenu Class, JMenuItem and JCheckBoxMenuItem, JRadioButtonMenuItem ,JScrollBar 1.5 Dialogs (Message, confirmation, input), JFileChooser, JColorChooser 1.6 Event Handling: Event sources, Listeners 1.7 Mouse and Keyboard Event Handling 1.8 Adapters 1.9 Anonymous inner class	**10**
2.	**Database Programming** 2.1The design of jdbc, jdbc configuration 2.2 Types of drivers 2.3 Executing sql statements, query execution 2.4 Scrollable and updatable result sets 2.5 Metadata – DatabaseMetadata, ResultSetMetadata 2.6 Transactions – commit(), rollback(), SavePoint	**10**
3.	**Servlet** 3.1 Introduction to Servlet and Hierarchy of Servlet 3.2 Life cycle of servlet 3.3 Tomcat configuration (Note: Only for Lab Demonstration) 3.4 Handing get and post request (HTTP) 3.5 Handling a data from HTML to servlet 3.6 Retriving a data from database to servlet 3.7 Session tracking – User Authorization, URL rewriting, Hidden form fields, Cookies and HttpSession	**12**

4.	**JSP** 4.1 Simple first JSP program 4.2 Life cycle of **JSP** 4.2 Implicit Objects 4.3 Scripting elements – Declarations, Expressions, Scriptlets, Comments 4.4 JSP Directives – Page Directive, include directive 4.5 Mixing Scriptlets and HTML 4.6 Example of forwarding contents from database to servlet, servlet to JSP and displaying it using JSP scriptlet tag	**10**
5.	**Networking** 5.1 Networking basics – Protocol, Addressing, DNS, URL, Socket, Port 5.2 The java.net package – InetAddress, URL, URLConnection class 5.3 SocketServer and Socket class 5.4 Creating a Socket to a remote host on a port (creating TCP client and server) 5.5 Simple Socket Program Example	**6**

Reference Books:

1. Complete reference Java by Herbert Schildt
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
5. Core Java Volume-II-Advanced Features, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- V)
(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-V) **Paper Code:** CSCO3506

Title of Paper: Object Oriented Software Engineering **Paper:** VI

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

Prerequisites : Knowledge of Classical Software Engineering

Aim : To Understand Object Oriented Modeling techniques and their applicability.

Objectives:

- Understanding Object Orientation in Software engineering concepts and importance
- Understand the Unified Modeling Language concepts, importance and its components
- Understand Structural, Behavioral, Dynamic modeling techniques and diagrams.
- Understand Object Oriented analysis, design, testing concepts and its techniques

	Title and Contents	**No. of Lectures**
Unit 1	**Object Oriented Concepts and Principles** 1.1 Introduction, Object, Classes and Instance, Polymorphism, Inheritance 1. 2 Object Oriented System Development- Introduction, Function / Data Methods (With Visibility), Object Oriented Analysis, Object Oriented Construction 1.2 Identifying the Elements of an Object Model Aggregations, 1.3 Identifying Classes and Objects, Identity, Dynamic binding, Persistence, Meta classes 1.5 Specifying the Attributes (With Visibility) 1.6 Defining Operations 1.7 Finalizing the Object Definition	**04**
Unit 2	**Introduction to UML and Object Oriented Methodology** 2.1 Concept of UML 2.2 Advantages of UML 2.3 Object oriented Methods (The Booch Method, The Coad and Yourdon Method, Jacobson Method and Raumbaugh Method)	**06**
Unit 3	**Basic Structural Modeling** 3.1 Classes 3.2 Relationship 3.3 Common Mechanism 3.4 Class Diagram (Minimum three examples should be covered)	**05**
Unit 4	**Advanced Structural Modeling** 4.1 Advanced Classes 4.2 Advanced Relationship 4.3 Interface 4.4 Types and Roles 4.5 Packages 4.6 Object Diagram (Minimum three examples should be covered)	**05**
Unit 5	**Basic Behavioral Modeling** 5.1 Interactions 5.2 Use Cases and Use Case Diagram with stereo types (Minimum three examples should be covered) 5.3 Interaction Diagram (Minimum two examples should be covered) 5.4 Sequence Diagram (Minimum two examples should be covered) 5.6 Activity Diagram (Minimum two examples should be covered)	**06**

	5.6 State Chart Diagram (Minimum two examples should be covered)	
Unit 6	**Object Oriented Analysis** 6.1 Iterative Development and the Rational Unified Process 6.2 Inception 6.3 Understanding Requirements 6.4 Use Case Model from Inception to Elaboration 6.5 Elaboration	**06**
Unit 7	**Object Oriented Design** 7.1 The Generic Components of the OO Design Model 7.2 The System Design Process - Partitioning the Analysis Model, Concurrency and Sub System Allocation, Task Management Component, The Data Management Component, The Resource Management Component, Inter Sub System Communication 7.3 Design process and benchmarking, Designing classes, Messages, Information hiding , Class hierarchy , Relationships , Databases , Object relational systems ,Designing interface objects 7.4 Object Design Process, Object oriented system development life cycle.	**05**
Unit 8	**Architectural modeling** 8.1 Component 8.2 Components Diagram (Minimum two examples should be covered) 8.3 Deployment Diagram (Minimum two examples should be covered) 8.4 Collaboration Diagram (Minimum two examples should be covered)	**06**
Unit 9	**Object Oriented Testing** 9.1 Object Oriented Testing Strategies 9.2 Test Case Design for Object Oriented Software 9.3 Inter Class Test Case Design(Use of any freeware designing tool)	**05**

OUTCOMES:

- Understand the activities during the software application development by using Object oriented Design.
- Learn the preparing of documentation allocation for the projects.
- Design and develop the software project development using Object oriented modeling techniques

References

1. Ivar Jacobson, “Object Oriented Software Engineering”, Pearson Education INC
2. Craig Larman, “Applying UML and Patterns” Pearson Education INC
3. Bennett, Simon, “Object Oriented Analysis and Design” McGraw Hill
4. Ali Bahrami, “Object Oriented System Development”, McGraw Hill International Edition, 2008
5. Brahma Dathan, Sarnath Ramnath, “Object-Oriented Analysis, Design and Implementation”, Universities Press, 2010
6. Bernd Bruegge, Allen H. Dutoit, Object Oriented Software Engineering using UML,Patterns and Java, Pearson 2004
7. Craig Larman, Applying UML and Patterns – An Introduction to Object-Oriented Analysis and Design and Iterative Development” , 3rd Edition, Pearson Education, 2005
8. Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modeling Language User Guide”, Addison Wesley Long man, 1999
9. Martin Fowler, “UML Distilled A Brief Guide to Standard Object Modeling Language”, 3rd Edition, Addison Wesley, 2003
10. Russ Miles, Kim Hamilton, “Learning UML 2.0”, O’Reilly, 2008

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- V)

(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-V) **Paper Code:** CSCO3507

Title of Paper: Lab Course-I on CSCO3101 **Paper:** VII **(**Lab Course-I)

Credit: 3 (3Hr practical/week/batch) **No. of Practical: 14**

Assignment No.	Name of Assignment	No of Practical Sessions Required
1.	Simulator	02
2.	Assembler	02
3.	Shell Program to implement System Calls	02
4.	Process Scheduling	04
5.	Memory Management	04

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- V)

(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-V) **Paper Code:** CSCO3508

Title of Paper: Lab. Course – II : Advanced Java Prog. **Paper:** VIII Lab Course - II

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

LAB WORKBOOK (Proposed)

Chapter No.	Chapter name with Topics
1.	**User Interface Components with AWT and Swing** Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2
2.	**Database Programming** Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2
3.	**Servlet** Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2
4.	**JSP** Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2
5.	**Networking** Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2

SYLLABUS (CBCS) FOR T.Y.B.Sc. (Computer Science) (Semester-V)

(w.e.f. from Academic Year 2021-2022)

Class : T.Y.B.Sc. (Computer Science) (Sem-V) | **Paper Code :** CSCO3509
Title of Paper : Lab. Course – III : Based on CSCO3504 | **Paper :** IX (Lab. Course – III)
Credits : 02 (3 Hr. Practical/Week/batch) | **No. of Practicals :** 14

Prerequisite : HTML

Objectives :

- To design dynamic, interactive web pages.
- To learn the server side scripting language.
- To learn database connectivity with PHP

Outcome :

On completion of the course, student will be able to understand how to develop dynamic and interactive web pages.

Assignment No.	**Title**
1.	Assignment on basic programs using control structures
2.	Assignment on functions
3.	Assignment on functions
4.	Assignment on strings
5.	Assignment on strings
6.	Assignment on arrays
7.	Assignment on arrays
8.	Assignment on arrays
9.	Assignment on Object Oriented Programming
10.	Assignment on Object Oriented Programming
11.	Assignment on Object Oriented Programming
12.	Assignment on Databases
13.	Assignment on Databases
14.	Assignment on Databases

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- VI)
(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-VI)	**Paper Code:** CSCO3601
Title of Paper: Advanced Operating System Concepts	**Paper:** I
Credit: 3 (4 Lectures/Week)	**No. of lectures:** 48

Aim: To understand the design and implementation issues of of Operating System.

Objectives:

- To understand design issues related to memory management and various related algorithms
- To understand design issues related to file management and various related algorithms.
- To understand the structures of different types of Operating System

Learning Outcome:

After the completion of this course student to should understand the basic structure of Operating System

Umt No.	Chapter name with Topics	No. of Lectures Required
1.	**Memory Management** 1.1.Background – Basic hardware, Address binding, Logical versus physical address space, Dynamic loading, Dynamic linking and shared libraries , Overlays 1.2 Swapping 1.3 Contiguous Memory Allocation – Memory mapping and protection, Memory allocation, Fragmentation MFT MVT 1.4 Paging – Basic Method, Hardware support, Protection, Shared Pages 1.5 Segmentation – Basic concept, Hardware 1.6 Virtual Memory Management – Background, Demand paging, Performance of demand paging, Page replacement – FIFO, OPT, LRU, MFU,LFU, MRU Second chance page replacement 1.7 Thrashing- Locality Model, Working Set Model, Prepaging, I/O Interlock,	**10**
2.	**File System** 2.1 Introduction 2.2 File Concepts- Tape based system, Disk based , File Attribute, Operations on file 2.3 Access Methods- Sequential access, Direct Access, Indexed Access 2.4 Directory Structure and Implementation- Single level directory, Two level directory, Tree structure directory, Acyclic Graph Directory. 2.5 File Structure 2.6 File System Mounting and File Sharing 2.7 Allocation Methods- Contiguous Allocation, Linked Allocation, Indexed Allocation 2.8 Free Space Management – Bitmap or Bit Vector, Linked List, Grouping, Counting	**08**
3.	**I/O Systems** 3.1 Introduction 3.2 I/O Hardware 3.3 Polling,	**08**

	3.4 Interrupt 3.5 Direct Memory Access 3.6 Application I/O Interface 3.7 Kernel I/O Subsystem- I/O Scheduling, Buffering, Caching , Spooling and Device Reservation, Error Handling, 3.8 Disk Scheduling- First Come First Served (FCFS), Shortest Seek Time First (SSTF), Scan, C-Scan.	
4.	**Introduction to Distributed Operating System & Architecture** 4.1 Distributed system design goals 4.2 Types of Distributed System 4.3 Architectural Styles- Layered Architecture, Object based architecture, Resource centered architectures, Event based Architecture 4.4 System Architecture- Centralized, Decentralized	08
5.	**Unix Kernel and File Management** 5.1 System Structure, User Perspective, Architecture of Unix Operating System 5.2 Buffer cache: Header, Buffer Pool, Retrieving, Reading and Writing Buffer 5.3 File Representation: inodes: Structure of file Directories, Path conversion to inode, superblock, inode assignment, allocation of disk blocks	**08**
6.	**Real Time Operating Systems and Mobile OS** 6.1 Characteristics of Real Time operating Systems, Classification of Real Time Operating Systems, Scheduling in RTOS: Clock driven: cyclic, Event driven: EDF and rate Monotonic scheduling. 6.2 Mobile OS: Architecture, Android OS, iOS, Virtual OS, Cloud OS and their design issues	**08**

Reference Books

1. Siberchatz, Galvin, Gagne Operating System Concepts - (8th Edition).

2. Pabitra Pal ChoudharyOperating Systems : Principles and Design – (PHI Learning Private Limited)

3. Maurice J. Bach.The Design of the UNIX Operating System, PHI

4. Mahajan and Seema Shah,Distributed Computing 2ndEditionOXford.

5. MukeshSinghal, Niranjan G ShivaratAdvanced Concepts in Operating Systems

6. Rajkamal, Pedition Mobile Computing Oxford.

7. Jane W.S. Liu, Real Time Operating SystemPearson.

SYLLABUS (CBCS) FOR T. Y. B. Sc. (Computer Science)(Sem-VI)

(w.e.f June, 2021) Academic Year 2021-2022

Class **:** T.Y. B. Sc.(Computer Science) (Sem.- VI) **Paper Code** **:** CSCO3602
Subject : Compiler Construction **Paper** : II
Credit **:** 3 **No. of lectures :** 48

Prerequisite:

- Theoretical Computer Science

Learning Objectives: Students successfully completing this course will be able:

- To understand design issues of a lexical analyzer and use of LEX tool
- To understand design issues of a parser and use of YACC tool
- To understand issues related to memory allocation
- To understand and design code generation schemes

Learning Outcome: Understand the various phases of a compiler and to develop skills in designing a compiler.

Units	Topic Contents	No. of Lectures
Unit – I	**Introduction** 1.1 Definition of Compiler, Aspects of compilation. 1.2 The structure of Compiler. 1.3 Phases of Compiler – Lexical Analysis,Syntax Analysis, Semantic Analysis, Intermediate Code generation, code optimization, code generation. 1.4 Error Handling 1.5 Introduction to one pass & Multipass compilers, cross compiler, Bootstrapping.	**5**
Unit – II	**Lexical Analysis(Scanner)** 2.1 Review of Finite automata as a lexical analyzer, 2.2 Applications of Regular Expressions and Finite Automata (lexical analyzer, searching using RE), Input buffering, Recognition of tokens 2.3 LEX: A Lexical analyzer generator (Simple Lex Program)	**5**
Unit –III	**Syntax Analysis(Parser)** 3.1 Definition , Types of Parsers 3.2 Top-Down Parser – 3.2.1Top-Down Parsing with Backtracking: Method & Problems 3.2.2 Drawbacks of Top-Down parsing with backtracking, 3.2.3 Elimination of Left Recursion(direct & indirect) 3.2.4 Need for Left Factoring & examples 3.3 Recursive Descent Parsing : Definition 3.3.1 Implementation of Recursive Descent Parser Using Recursive Procedures 3.4 Predictive [LL(1)]Parser(Definition, Model) 3.4.1 Implementation of Predictive Parser[LL(1)] 3.4.2 FIRST & FOLLOW 3.4.3 Construction of LL(1) Parsing Table 3.4.4 Parsing of a String using LL(1) Table 3.5 Bottom-Up Parsers 3.6 Operator Precedence Parser -Basic Concepts 3.6.1 Operator Precedence Relations form Associativity & Precedence 3.6.2 Operator Precedence Grammar	**20**

	3.6.3 Algorithm for LEADING & TRAILING(with ex.) 3.6.4 Algorithm for Operator Precedence Parsing (with ex.) 3.6.5 Precedence Functions 3.7 Shift Reduce Parser 3.7.1 Reduction, Handle, Handle Pruning 3.7.2 Stack Implementation of Shift Reduce Parser (with examples) 3.8 LR Parser 3.8.1Model 3.8.2 Types [SLR(1), Canonical LR, LALR] Method & examples. 3.9 YACC (from Book 3) –program sections, simple YACC program for expression evaluation	
Unit -IV	**Syntax Directed Definition** 4.1Syntax Directed Definitions(SDD) 4.1.1 Inherited & Synthesized Attributes 4.1.2 Evaluating an SDD at the nodes of a Parse Tree, Ex. 4.2 Evaluation Orders for SDD's 4.2.1 Dependency Graph 4.2.2 Ordering the Evaluation of Attributes 4.2.3 S-Attributed Definition 4.2.4 L-Attributed Definition 4.3 Application of SDT 4.3.1 Construction of syntax trees, 4.3.2 The Structure of a Type 4. 4 Translation Schemes 4.4.1 Definition, Postfix Translation Scheme	**8**
Unit – V	**5. Memory Allocation & Code Optimization** 5.1 Memory allocation – static and dynamic memory allocation, 5.2 Memory allocation in block structure languages, Array allocation and access. 5.3 Compilation of expression – 5.3.1 Concepts of operand descriptors and register descriptors with example. 5.3.2 Intermediate code for expressions – postfix notations, 5.3.3 triples and quadruples, expression trees. 5.4 Code Optimization – Optimizing transformations – compile time evaluation, elimination of common sub expressions, dead code elimination, frequency reduction, strength reduction 5.5 Three address code 5.5.1. DAG for Three address code 5.5.2 The Value-number method for constructing DAG's. 5.6 Definition of basic block, Basic blocks And flow graphs 5.7 Directed acyclic graph (DAG) representation of basic block 5.8 Issues in design of code generator	**10**

References :-

1. Compilers: Principles, Techniques, and Tools ,Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman

2. Principles of Compiler Design By : Alfred V. Aho, Jeffrey D. Ullman (Narosa Publication House)

3. LEX & YACC (O'reilly Publication)

SYLLABUS (CBCS) FOR T.Y.B.Sc. (Computer Science) (SEM-VI)
Academic Year 2021-2022

Class: T.Y.B.Sc. (Computer Science)(Semester-VI) **Paper Code**: COMP3603

Title of paper: Computer Network - II **Paper**: III

Credit -3 **No. of Lectures**: 48

Pre-requisites: Basics of computer networks covered last semester.

Learning Objectives:

- To enable students to get sound understanding of additional Network concepts,
- Understand importance of network security and cryptography.
- To develop attitude and interest along with necessary knowledge and skills among the students to encourage them to do further academic studies / research in this area, after the completion of this Course.

Learning Outcomes: Learn the security concepts and techniques.

Unit No	Title & Contents	No. of Lectures
I	**The Network Layer** 1.1 Design Issues Store-and-forward packet switching, Services Provided to the Transport Layer, Implementation of Connectionless Service, Implementation of Connection Oriented Service, Comparison of Virtual Circuit and Datagram subnets 1.2 Logical Addressing IPV4 Addresses – Address Space, Notations, Classful Addressing, Subnetting, Supernetting, Classless Addressing, Network Address Translation(NAT), (Enough problems should be covered on Addressing), 1.3 IPV4 Protocol Datagram Format, Fragmentation, Checksum, Options. 1.4 Routing Properties of routing algorithm, Comparison of Adaptive and Non- Adaptive Routing Algorithms 1.5 Congestion Control – Definition, Factors of Congestion, Difference between congestion control and flow control, General Principles of Congestion Control, Congestion Prevention Policies 1.6 Network Layer Devices –Routers	10
II	**Address Mapping** Protocol(ARP)-Cache Memory, Packet Format, Encapsulation, Operation, Four Different Cases, Proxy ARP, RARP , BOOTP, DHCP – Static Address Allocation, Dynamic Address Allocation, Manual and automatic Configuration.	05
III	**The Transport Layer** 3.1 Process-to-Process Delivery Client Server Paradigm, Multiplexing and De-multiplexing, Connectionless Vs Connection-Oriented Service, Reliable Vs Unreliable 3.2 User Datagram Protocol(UDP) Datagram Format, Checksum, UDP operations, Use of UDP 3.3 Transmission Control Protocol (TCP) TCP Services – Process to-Process Communication, Stream Delivery Service, sending and Receiving Buffers, Segments, Full –Duplex Communication, Connection oriented service, Reliable service. 3.4 TCP Features –Numbering System, Byte Number, Sequence Number, Acknowledgement Number, Flow Control, Error Control, Congestion Control 3.5 TCP Segment – Format	07

IV	**The Application Layer** 4.1 Domain Name System • Name space-Flat name space, Hierarchical name space • Domain Name Space -Label ,Domain name, FQDN, PQDN • Distribution of Domain Name Space-Hierarchy of name servers, zone, Root server, Primary and secondary servers. • DNS in the Internet: Generic domains, Country domains, inverse domain • Resolution- Resolver, mapping names to address, mapping addresses to names, recursive resolution, iterative resolution, caching • DNS messages-Header 4.2. Remote logging-Telnet: Time sharing Environment, Logging, NVT Character set, Embedding Options, mode of operation 4.3. Electronic Mail- • Architecture-First scenario, second scenario, Third scenario, Fourth scenario • User agent-services of user agent, types of UA Format of e-mail • MIME-MIME header • Message transfer agent-SMTP • Message Access Agent: POP and IMAP 4.4. File Transfer - FTP-Communication over data control connection, File type, data structure, Transmission mode, anonymous FTP 4.5. WWW- Architecture, Client, Server, URL, Cookies 4.6. HTTP-HTTP transaction, messages 4.7 Devices – Gateways , Transport & Application gateways	10
V	**Multimedia** Digitizing Audio and Video , Streaming stored Audio / Video , Streaming Live Audio / Video , Real-Time Interactive Audio / Video RTP , RTCP	04
VI	**Cryptography and Network Security** 6.1 Introduction – Need of security, Security approaches, Security Principles, Types of attacks. 6.2. Cryptography concepts and Techniques - Plain text and cipher text, Encryption & Decryption, Categories of cryptography- Symmetric key, asymmetric key, comparison, Traditional ciphers Technique – substitution cipher, Transposition cipher. (problem should be covered) 6.3. Symmetric key cryptography- • Algorithm types and modes (ECB,CBC,CFB,OFB) • Symmetric key algorithm – DES, AES, IDEA 6.4. Asymmetric key cryptography- RSA 6.5. Security Services • Message confidentiality-With Symmetric key cryptography, with asymmetric key cryptography • Message integrity-Document and fingerprint, message and message digest • Message authentication-MAC, HMAC 6.6 Communication Security- Firewall, IP Security, Virtual Private Network (VPN)., Wireless Security, Web Security.	12

Reference Books:

1) Computer Networks by Andrew Tanenbaum, Pearson Education.[Latest Edition]
2) Data Communication and Networking by Behrouz Forouzan, TATA McGraw Hill.[4th/5th Ed.]
3) Networking All In One Dummies Wiley Publication.[5th Edition]
4) Cryptography and Network Security : Atul Kahate
5) Computer Network Security : Kizza, Springer
6) Network Security – Harrington, Elsevie

SYLLABUS (CBCS) FOR T.Y.B.Sc. (Computer Science) (Semester-VI) (w.e.f. from Academic Year 2021-2022)

Class : T.Y.B.Sc. (Computer Science) (Sem-VI) **Paper Code :** CSCO3604
Title of Paper : Web Development-II **Paper :** IV
Credits : 03 (4 Lectures/Week) **No. of lectures :** 48
Prerequisite : Know the Core PHP
Objectives :

- To learn the latest technologies used with PHP.
- To learn using JSON with PHP.
- To learn AJAX for applying dynamic changes to application.
- To learn package management.

Outcome : On completion of the course, student will be able to build dynamic website.

Chapter No.	**Chapter name with Topics**	**No. of Lectures Required**
1.	**Web Techniques** 1.1 Variables 1.2 Server information 1.3 Processing forms 1.4 Setting response headers 1.5 State management 1.6 JWT(JSON Web Tokens) 1.7 JWT Vs Sessions 1.8 SSL	**10**
2.	**JSON with PHP** 2.1 Introduction to JSON 2.2 JSON syntax 2.3 Datatypes in JSON 2.4 JSON Vs XML 2.5 Encoding JSON in PHP 2.6 Decoding JSON in PHP 2.7 Accessing the decoded values 2.8 Looping through the values	**08**
3.	**AJAX** 3.1 Introduction to AJAX 3.2 AJAX web application model 3.3 AJAX-PHP framework 3.4 Performing AJAX validation 3.5 Connecting database using php and AJAX	**10**
4.	**Intermediate JavaScript** 4.1 More Objects - Constructor Fuctions - Factory Functions - Constructor Method - new keyword 4.2 this - this in methods - window scope - this in functions declarations - this in events 4.3 Prototypes and ES6 Classes - Built-in constructor functions	**14**

	- Constructor property - Inheritance - this in classes 4.4 Asynchronous JS - Async / Await - Promises 4.5 Array Iteration Methods - forEach - Map - Filter - Reduce 4.6 String Manipulations - trim and split methods - toUpperCase and toLowerCase methods - includes method	
5.	**Collaborative Platform** 5.1 Package management 5.2 Frameworks like CodeIgniter, Symfony, Laravel - Installation - Features	**06**

References :

1. Kevin Tatroe, Peter MacIntyre (2020), Programming PHP : Creating Dynamic Web Pages(4th ed.). O’Reilly.

Web References :

1. https://www.php.net/manual/en/manual.php
2. https://www.php-fig.org/
3. https://phptherightway.com
4. https://w3schools.com

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- VI)
(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-VI) **Paper Code:** CSCO3605
Title of Paper: Advanced Java Technologies – Frameworks **Paper:** V
Credit: 3 (4 Lectures/Week) **No. of lectures:** 48

Aim: To understand the design and implementation of Java Frameworks.

Objectives: To understand Hibernate, Spring Core, Spring MVC, Spring Boot to develop Enterprise Applications

Learning Outcome:

After the completion of this course student should understand Java Frameworks and capable to develop Web Development

Unit No.	Chapter name with Topics	No. of Lectures Required
1.	Hibernate ORM Overview Hibernate Overview Architecture Hibernate Configuration Hibernate Sessions Persistent Class O/R Mapping One-to-One Mappings One-to-Many Mappings Many-to-Many Mappings Hibernate using Annotation Hibernate Query Language (HQL) SQL Dialects in Hibernate Transaction Management	**14**
2.	Spring Core Introduction and Overview Applications of Spring Advantages of Spring Framework Dependency Injection (DI) Inversion of Control (IoC) Aspect Oriented Programming (AOP) Architecture / Modules Difference between constructor and setter injection Autowiring in Spring Bean Definition Bean Scopes Bean Life Cycle	**13**
3.	Spring MVC Introduction and Overview Multiple View page Multiple Controller Model Interface RequestParam Annotation Form Tag Library	**9**
4.	Spring Boot Introduction and Overview	**12**

	Spring Boot Versions Spring vs. Spring Boot vs. Spring MVC Spring Boot Architecture Spring Initializr Download and Install STS IDE Creating a Spring Boot Project Creating a Spring Boot Project Using STS Spring Boot Annotations Spring Boot Dependency Management Spring Boot Application Properties	

Reference:

1. https://spring.io/
2. https://hibernate.org/

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- VI)

(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-VI)	**Paper Code:** CSCO3606
Title of Paper: Software Metrics and Project Management	**Paper:** VI
Credit: 3 (4 Lectures/Week)	**No. of lectures:** 48

Prerequisites : Knowledge of Software Engineering

Aim : To Understand Software metrics and project management and their applicability.

Objectives:

- To know of how to do project planning for the software process.
- To learn the cost estimation techniques during the analysis of the project.
- To understand the quality concepts for ensuring the functionality of the software

Title and Contents		No. of Lectures
UNIT 1	**SOFTWARE PROJECT MANAGEMENT CONCEPTS** 1.1 Introduction to Software Project Management: Project phase and project life Cycle, Organizational structure. 1.2 An Overview of Project Planning: Select, Identifying Project scope and objectives, infrastructure, project products and Characteristics. Estimate efforts, Identify activity risks, and allocate resources- TQM, Six Sigma 1.3 Software Quality: defining software quality, ISO9126, External Standards. 1.4 Project Plan development and Execution, Change control, Configuration Management, Activity Planning, Schedule Development and Control	**08**
UNIT 2	**OVERVIEW OF PROJECT MANAGEMENT COMPONENTS** 2.1 Project Integration Management 2.2 Project Scope Management 2.3 Project Time Management 2.4 Project Cost Management 2.5 Project Quality Management 2.6 Project Human Resource Management 2.6 Project Communications Management 2.7 Project Risk Management 2.8 Project Procurement Management 2.9 Project Stakeholder Management	**10**
UNIT 3	**SOFTWARE EVALUATION AND COSTING** 3.1 Project Evaluation: Strategic Assessment, Technical Assessment, cost-benefit analysis, flow forecasting, cost-benefit evaluation techniques, Risk Evaluation. 3.2 Selection of Appropriate Project approach: Choosing technologies, choice of process models, structured methods.	**04**
UNIT 4	**SOFTWARE ESTIMATION TECHNIQUES** 4.1 Software Effort Estimation: Problems with Over and under estimations, Basis of software Estimation, Software estimation techniques, expert Judgment, Estimating by analogy. 4.2 Activity Planning: Project schedules, projects and activities, sequencing and scheduling Activities, networks planning	**10**

	models, Formulating a network model	
UNIT 5	**RISK MANAGEMENT** 5.1 Risk Management: Nature of Risk, Managing Risk, Risk Identification and Analysis, Reducing the Risk. 5.2 Resource Allocation: Scheduling resources, Critical Paths, Cost scheduling, 5.3 Monitoring and Control: Creating Framework, cost monitoring, prioritizing monitoring.	**06**
UNIT 6	**INTRODUCTION TO SOFTWARE PROJECT METRICS** 6.1 Introduction to Software Project Metrics, Types Of Software Project Metrics, Scope of Software Project Metrics ,Software metrics and Data collection, 6.2 Metrics Measurement: Measurement and prediction, Resource measurement, Productivity Measurement, Mapping measurement activities, Measurement tool, Role of Measures (Analyst, tools, Plans, Revision Plans, 6.3 Quality Measurement- Quality Standards (ISO,MC-Call, CMM, PSP/TSP) 6.4 Globalization issues in project management (Evaluation, Advantages, Dis-advantages) 6.5 Impact of the internet on project management (effect on management activities) 6.6 CASE Studies Software project Management	**10**

OUTCOMES:

- Understand the activities during the project scheduling of any software application.
- Learn the risk management activities and the resource allocation for the projects.
- Can apply the software estimation and recent quality standards for evaluation of the
- Software projects Acquire knowledge and skills needed for the construction of highly reliable software project
- Able to create reliable, replicable cost estimation that links to the requirements of project planning and managing.

REFERENCES:

1. Bob Hughes & Mike Cotterell, “Software Project Management”, Tata McGraw- Hill Publications, Fifth Edition 2012
2. Futrell , “Quality Software Project Management”, Pearson Education India, 2008
3. Gobalswamy Ramesh, “Managing Global Software Projects”, Tata McGraw Hill Publishing Company, 2003
4. Richard H.Thayer “Software Engineering Project Management”, IEEE Computer Society
5. S. A. Kelkar,” Software Project Management” PHI, New Delhi, Third Edition ,2013
6. Roger Pressman “ISE Software Engineering: A Practitioner's Approach” ISE HED IRWIN COMPUTER SCIENCE 9th Edition 2019
7. Kathy Schwalbe “Information Technology Project Management”Cengage Learning Canada Inc. 9th Edition
8. Norman Feton, Shari Lawrence pfleeger : Software Metrics : A rigorous and Practical Approch”PWS Publishing Company, 1997 2nd Edition
9. http://en.wikipedia.org/wiki/Comparison_of_project_management_softwar
10. http://www.ogc.gov.uk/methods_prince_2.asp

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- VI)

(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-VI)	**Paper Code:** CSCO3607
Title of Paper: Lab Course-I (on CSCO3202)	**Paper:** VII Lab Course - I
Credit: 3 (3 Hr. Practical / Week/batch)	**No of Practical :** 14

Assignment No.	Name of Assignment	No of Practical Sessions Required
1.	Deadlock	02
2.	File Management	04
3.	Disk Scheduling	04
4.	Real Time Scheduling	02
5.	Case Study on any one type of OS	02

SYLLABUS (CBCS) FOR T.Y.B. Sc. (Computer Science) (Semester- VI)
(w.e.f from Academic Year 2021-2022)

Class: T.Y.B.Sc. (Computer Science) (Sem-VI) **Paper Code:** CSCO3608

Title of Paper: Lab. Course – II : Advanced Java Tech–Frameworks And Mini Project using JAVA.

Paper: VIII Lab Course –II

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

LAB WORKBOOK

Chapter No.	Chapter name with Topics
1.	JSON Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2
2.	React Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2
3.	Spring Boot Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2
4.	Hibernate Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2
5.	Struts2 Set A - Assignment 1 Assignment 2 Set B – Assignment 1 Assignment 2

SYLLABUS (CBCS) FOR T.Y.B.Sc. (Computer Science) (Semester-VI)

(w.e.f. from Academic Year 2021-2022)

Class : T.Y.B.Sc. (Computer Science) (Sem-VI) **Paper Code :** CSCO3609
Title of Paper : Lab. Course – III : Based on CSCO3604 and Mini Project Using PHP
Paper : IX - Lab. Course – III
Credits : 02 (3 Hr. Practical/Week/batch) **No. of Practicals :** 14

Prerequisite : Core PHP

Objectives :

- To learn the latest technologies used with PHP.
- To learn using JSON with PHP.
- To learn AJAX for applying dynamic changes to application.
- To learn package management.

Outcome :

On completion of the course, student will be able to build dynamic website.

Assignment No.	**Title**
1.	Assignment on web techniques
2.	Assignment on web techniques
3.	Assignment on web techniques
4.	Assignment on web techniques
5.	Assignment on JSON with PHP
6.	Assignment on JSON with PHP
7.	Assignment on AJAX
8.	Assignment on AJAX
9.	Assignment on AJAX
10.	Assignment on Intermediate JavaScript
11.	Assignment on Intermediate JavaScript
12.	Assignment on Intermediate JavaScript
13.	Assignment on Intermediate JavaScript
14.	Assignment on Collaborative Platform