# 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 |

**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**<br>        1.1. Types of program – System program and Application program. 1.2. Difference between system programming and application programming.<br>        1.3. Elements of Programming environment - Editor, Preprocessor, Assembler, Compiler, Interpreter, Linker and Loader, Debugger, Device drivers, Operating System.<br>         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**<br>        2.1 What Operating Systems Do – User View, System View, Defining OS<br>        2.2 Computer System Architecture – Single processor system, Multiprocessor systems, Clustered Systems<br>        2.3 Operating System Operations – Dual mode operation, Timer<br>        2.4 Process Management<br>        2.5 Memory Management<br>        2.6 Storage Management – File system management, Mass storage management, Cashing, I/O systems<br>        2.7 Protection and Security<br>        2.8 Distributed Systems<br>        2.9 Special Purpose System – Real time embedded systems, Multimedia systems, Handheld systems,<br>        2.10 Computer Environment – Traditional computing, Client server computing, Peer to peer Computing | 06 |

| 3. | **System Structure**<br>3.1 Operating System Services<br>3.2 User Operating-System Interface – Command interpreter, GUI<br>3.3 System Calls<br>3.4 Types of System Calls – Process control, File management, Device management, Information maintenance, Communication, Protection | 02 |
|---|---|---|
| 4. | **Process Management**<br>4.1 Process Concept – The process, Process states, Process control block.<br>4.2 Process Scheduling – Scheduling queues, Schedulers, context switch<br>4.3 Operations on Process – Process creation with program using fork(), Process termination<br>4.4 Inter-process Communication – Shared memory system, Message passing systems. | 05 |
| 5. | **Process Scheduling**<br>5.1 Basic Concept – CPU-I/O burst cycle, CPU scheduler, Preemptive scheduling, Dispatcher<br>5.2 Scheduling Criteria<br>5.3 Scheduling Algorithms – FCFS, SJF, Priority scheduling, Round-robin scheduling, Multiple queue scheduling, Multilevel feedback queue scheduling.<br>5.4 Multithreaded Programming<br>5.5 Multithreading Models 6.6 Thread Scheduling | 10 |
| 6. | **Multithreaded Programming**<br>6.1 Overview<br>6.2 Multithreading Model<br>6.3 Thread Libraries P-Tread, Java Thread<br>6.4 Thread Life Cycle | 06 |
| 7. | **Process Synchronization**<br>7.1 Background<br>7.2 Critical Section Problem<br>7.3 Semaphores: Usage, Implementation<br>7.4 Classic Problems of Synchronization – The bounded buffer problem, The reader writer problem, The dining philosopher problem | 04 |
| 8. | **Deadlocks**<br>8.1 System model<br>8.2 Deadlock Characterization – Necessary conditions, Resource allocation graph<br>8.3 Deadlock Prevention<br>8.4 Deadlock Avoidance - Safe state, Resource allocation graph algorithm, Banker's Algorithm<br>8.5 Deadlock Detection<br>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**<br>  2.1 Deterministic finite Automaton – Definition, DFA as<br>      Language recognizer, DFA as a pattern recognizer.<br>2.2 Nondeterministic finite automaton – Definition and Ex..<br>2.3 NFA TO DFA<br>2.4 NFA with ε- transitions Definition and Examples.<br>2.5 NFA with ε-Transitions to DFA & Examples<br>2.6 Finite automaton with output – Mealy and Moore machine,<br>      Definition and Examples.<br>2.7 Minimization of DFA, Algorithm & Problem using Table<br>Method. | 15 |
| Unit –II | **Regular Languages**<br>3.1 Regular language-Definition and Examples.<br>3.2 Conversion of RE To FA-Examples.<br>3.3 Pumping lemma for regular languages and applications.<br>3.4 Closure properties of regular Languages<br>      (Union, Concatenation, Complement, Intersection and<br>       Kleene closure) | 5 |
| Unit – III | **Context Free Grammar and Languages**<br>4.1 Grammar - Definition and Examples.<br>4.2 Derivation-Reduction - Definition and Examples.<br>4.3 Chomsky Hierarchy.<br>4.4 CFG : Definition & Examples. LMD, RMD, ,Parse Tree<br>4.5 Ambiguous Grammar : Concept & Examples.<br>4.6 Simplification of CFG :<br>4.6.1 Removing Useless Symbols,<br>4.6.2 Removing unit productions<br>4.6.3 Removing ε productions & Nullable symbols<br>4.7 Normal Forms :<br>4.7.1 Chomsky Normal Form (CNF) Method & Problem<br>4.7.2 Greibach Normal form (GNF) Method & Problem<br>4.8 Regular Grammar : Definition.<br>4.8.1 Left linear and Right Linear Grammar-Definition and<br>Example. | 12 |

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

**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**<br>Computer Networks- Goals, applications<br>Network Hardware's – Broadcast and point to point.<br>Topology – Star, Bus, Mesh, Ring etc.<br>Network Types : LAN, MAN, WAN, Wireless Network, internetwork<br>Data Communication – Definition, Components, data representation, Data flow. , Protocols and Standards Defacto, Dejure standard<br>Network Software- Protocol Hierarchies, Design issues of the layer, Connection and connectionless services, | 08 |
| II | **Network Models**<br>Reference Model – OSI Reference Model, TCP/IP Reference Model, Comparison of OSI & TCP/IP Model,<br>Addressing – Physical, Logical and Port addresses | 04 |
| III | **Transmission Media**<br>Guided Media – Twisted pair cable, Coaxial Cable, Fiber optic cable<br>Unguided Media – Radio Waves, Micro wave Transmission, Infrared, Light wave Transmission | 04 |
| IV | **Lower layers : Physical and Data link layers**<br>Communication at the physical layer, Data and signals.<br>Transmission Impairment, Data rate limits, Performance<br>Transmission Modes.<br>Switching – Circuit, Message and Packet Switching.<br>Design issues of Data Link Layer, Services – Framing, Error control, Flow Control, Congestion Control, Link layer addressing.<br>Data link Protocols – simplex, stop and wait and stop and wait Automatic Repeat Request (ARQ).<br>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<br>Physical and Data link layer devices – Repeater, Hubs, Bridge | 18 |

| | | |
|---|---|---|
| V | **The Medium Access Sub layer**<br><br>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**<br><br>IEEE Standards, changes in the standard – bridged Ethernet, switched Ethernet, full duplex Ethernet.<br>Fast Ethernet, Gigabit Ethernet, Ten-Gigabit Ethernet: Goals, MAC Sublayer, Topology and Implementation.<br>Backbone Network – Bus backbone, Star backbone, Remote LANs<br>Virtual LANs: Membership, configuration, communication, Advantages.<br>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]

**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**<br>1.1 HTTP basics, Web Server, Web Browser<br>1.2 Introduction to PHP(Why PHP?)<br>1.3 What does PHP do?<br>1.4 Lexical structure<br>1.5 Language basics | 04 |
| 2. | **Function and String**<br>2.1 Defining and calling a function<br>2.2 Default parameters<br>2.3 Variable parameters, Missing parameters<br>2.4 Variable function, Anonymous function<br>2.5 Types of strings in PHP<br>2.6 Printing functions<br>2.7 Encoding and escaping<br>2.8 Comparing strings<br>2.9 Manipulating and searching strings<br>2.10 Regular Expressions | 08 |
| 3. | **Arrays**<br>3.1 Indexed Vs Associative arrays<br>3.2 Identifying elements of an array<br>3.3 Storing data in arrays<br>3.4 Multidimensional arrays<br>3.5 Extracting multiple values<br>3.6 Converting between arrays and variables<br>3.7 Traversing arrays<br>3.8 Sorting<br>3.9 Action on entire arrays<br>3.10 Using arrays | 06 |
| 4. | **Introduction to Object Oriented Programming**<br>4.1 Classes and Objects<br>4.2 Inheritance<br>4.3 Interfaces<br>4.4 Encapsulation<br>4.5 Traits<br>4.6 Autoloading classes | 16 |

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

**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**<br>1.1 What is AWT ? What is Swing? Difference between AWT and Swing.<br>1.2 The MVC Architecture and Swing<br>1.3 Layout Manager and Layouts, The JComponent class<br>1.4 Components – JButton, JLabel, JText, JTextArea, JCheckBox and JRadioButton, JList, JComboBox, JMenu and JPopupMenu Class, JMenuItem and JCheckBoxMenuItem, JRadioButtonMenuItem ,JScrollBar<br>1.5 Dialogs (Message, confirmation, input), JFileChooser, JColorChooser<br>1.6 Event Handling: Event sources, Listeners<br>1.7 Mouse and Keyboard Event Handling<br>1.8 Adapters<br>1.9 Anonymous inner class | 10 |
| 2. | **Database Programming**<br>2.1 The design of jdbc, jdbc configuration<br>2.2 Types of drivers<br>2.3 Executing sql statements, query execution<br>2.4 Scrollable and updatable result sets<br>2.5 Metadata – DatabaseMetadata, ResultSetMetadata<br>2.6 Transactions – commit(), rollback(), SavePoint | 10 |
| 3. | **Servlet**<br>3.1 Introduction to Servlet and Hierarchy of Servlet<br>3.2 Life cycle of servlet<br>3.3 Tomcat configuration (Note: Only for Lab Demonstration)<br>3.4 Handing get and post request (HTTP)<br>3.5 Handling a data from HTML to servlet<br>3.6 Retriving a data from database to servlet<br>3.7 Session tracking – User Authorization, URL rewriting, Hidden form fields, Cookies and HttpSession | 12 |

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

**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**<br>1.1   Introduction, Object, Classes and Instance, Polymorphism, Inheritance<br>1. 2   Object Oriented System Development- Introduction, Function / Data Methods (With Visibility), Object Oriented Analysis, Object Oriented Construction<br>1.2   Identifying the Elements of an Object Model Aggregations,<br>1.3   Identifying Classes and Objects, Identity, Dynamic binding, Persistence, Meta classes<br>1.5   Specifying the Attributes (With Visibility)<br>1.6   Defining Operations<br>1.7   Finalizing the Object Definition | **04** |
| **Unit 2** | **Introduction to UML and Object Oriented Methodology**<br>2.1   Concept of UML<br>2.2   Advantages of UML<br>2.3   Object oriented Methods (The Booch Method, The Coad and Yourdon Method, Jacobson Method and Raumbaugh Method) | **06** |
| **Unit 3** | **Basic Structural Modeling**<br>3.1   Classes<br>3.2   Relationship<br>3.3   Common Mechanism<br>3.4   Class Diagram (Minimum three examples should be covered) | **05** |
| **Unit 4** | **Advanced Structural Modeling**<br>4.1   Advanced Classes<br>4.2   Advanced Relationship<br>4.3   Interface<br>4.4   Types and Roles<br>4.5   Packages<br>4.6   Object Diagram (Minimum three examples should be covered) | **05** |
| **Unit 5** | **Basic Behavioral Modeling**<br>5.1   Interactions<br>5.2   Use Cases and Use Case Diagram with stereo types (Minimum three examples should be covered)<br>5.3   Interaction Diagram (Minimum two examples should be covered)<br>5.4   Sequence Diagram (Minimum two examples should be covered)<br>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**<br>6.1    Iterative Development and the Rational Unified Process<br>6.2    Inception<br>6.3    Understanding Requirements<br>6.4    Use Case Model from Inception to Elaboration<br>6.5    Elaboration | **06** |
| **Unit 7** | **Object Oriented Design**<br>7.1    The Generic Components of the OO Design Model<br>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<br>7.3    Design process and benchmarking, Designing classes, Messages, Information hiding , Class hierarchy , Relationships , Databases , Object relational systems ,Designing interface objects<br>7.4    Object Design Process, Object oriented system development life cycle. | **05** |
| **Unit 8** | **Architectural modeling**<br>8.1    Component<br>8.2    Components Diagram (Minimum two examples should be covered)<br>8.3    Deployment Diagram (Minimum two examples should be covered)<br>8.4    Collaboration Diagram (Minimum two examples should be covered) | **06** |
| **Unit 9** | **Object Oriented Testing**<br>9.1    Object Oriented Testing Strategies<br>9.2    Test Case Design for Object Oriented Software<br>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

**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 |

**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**<br>Set A -<br>Assignment 1<br>Assignment 2<br>Set B –<br>Assignment 1<br>Assignment 2 |
| 2. | **Database Programming**<br>Set A -<br>    Assignment 1<br>    Assignment 2<br>Set B –<br>    Assignment 1<br>    Assignment 2 |
| 3. | **Servlet**<br>Set A -<br>    Assignment 1<br>    Assignment 2<br>Set B –<br>    Assignment 1<br>    Assignment 2 |
| 4. | **JSP**<br>Set A -<br>    Assignment 1<br>    Assignment 2<br>Set B –<br>    Assignment 1<br>    Assignment 2 |
| 5. | **Networking**<br>Set A -<br>    Assignment 1<br>    Assignment 2<br>Set B –<br>    Assignment 1<br>    Assignment 2 |

**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 |

**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** <br> 1.1.Background – Basic hardware, Address binding, Logical versus physical address space, Dynamic loading, Dynamic linking and shared libraries , Overlays <br> 1.2 Swapping <br> 1.3 Contiguous Memory Allocation – Memory mapping and protection, Memory allocation, Fragmentation MFT MVT <br> 1.4 Paging – Basic Method, Hardware support, Protection, Shared Pages <br> 1.5 Segmentation – Basic concept, Hardware <br> 1.6 Virtual Memory Management – Background, Demand paging, Performance of demand paging, Page replacement – FIFO, OPT, LRU, MFU,LFU, MRU Second chance page replacement <br> 1.7 Thrashing- Locality Model, Working Set Model, Prepaging, I/O Interlock, | 10 |
| 2. | **File System** <br> 2.1 Introduction <br> 2.2 File Concepts- Tape based system, Disk based , File Attribute, Operations on file <br> 2.3 Access Methods- Sequential access, Direct Access, Indexed Access <br> 2.4 Directory Structure and Implementation- Single level directory, Two level directory, Tree structure directory, Acyclic Graph Directory. <br> 2.5 File Structure <br> 2.6 File System Mounting and File Sharing <br> 2.7 Allocation Methods- Contiguous Allocation, Linked Allocation, Indexed Allocation <br> 2.8 Free Space Management – Bitmap or Bit Vector, Linked List, Grouping, Counting | 08 |
| 3. | **I/O Systems** <br> 3.1 Introduction <br> 3.2 I/O Hardware <br> 3.3 Polling, | 08 |

| | | |
|---|---|---|
| | 3.4 Interrupt<br> 3.5 Direct Memory Access<br> 3.6 Application I/O Interface<br>3.7 Kernel I/O Subsystem- I/O Scheduling, Buffering, Caching , Spooling and Device Reservation, Error Handling,<br> 3.8 Disk Scheduling- First Come First Served (FCFS), Shortest Seek Time First (SSTF), Scan, C-Scan. | |
| **4.** | **Introduction to Distributed Operating System & Architecture**<br>4.1 Distributed system design goals<br> 4.2 Types of Distributed System<br> 4.3 Architectural Styles- Layered Architecture, Object based architecture, Resource centered architectures, Event based Architecture<br>4.4 System Architecture- Centralized, Decentralized | 08 |
| **5.** | **Unix Kernel and File Management**<br>5.1 System Structure, User Perspective, Architecture of Unix Operating System<br>5.2 Buffer cache: Header, Buffer Pool, Retrieving, Reading and Writing Buffer<br>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**<br>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.<br>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.

**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**<br>1.1 Definition of Compiler, Aspects of compilation.<br>1.2 The structure of Compiler.<br>1.3 Phases of Compiler – Lexical Analysis,Syntax Analysis, Semantic Analysis, Intermediate Code generation, code optimization, code generation.<br>1.4 Error Handling<br>1.5 Introduction to one pass & Multipass compilers, cross compiler, Bootstrapping. | **5** |
| Unit – II | **Lexical Analysis(Scanner)**<br>2.1 Review of Finite automata as a lexical analyzer,<br>2.2 Applications of Regular Expressions and Finite Automata (lexical analyzer, searching using RE), Input buffering, Recognition of  tokens<br>2.3 LEX: A Lexical analyzer generator (Simple Lex Program) | **5** |
| Unit –III | **Syntax Analysis(Parser)**<br>3.1 Definition , Types of Parsers<br>3.2 Top-Down Parser –<br>    3.2.1 Top-Down Parsing with Backtracking: Method & Problems<br>    3.2.2 Drawbacks of Top-Down parsing with backtracking,<br>    3.2.3 Elimination of Left Recursion(direct & indirect)<br>    3.2.4 Need for Left Factoring & examples<br>3.3 Recursive Descent Parsing : Definition<br>    3.3.1 Implementation of Recursive Descent Parser Using Recursive Procedures<br>3.4 Predictive [LL(1)]Parser(Definition, Model)<br>    3.4.1 Implementation of Predictive Parser[LL(1)]<br>    3.4.2  FIRST & FOLLOW<br>    3.4.3  Construction of LL(1) Parsing Table<br>    3.4.4 Parsing of a String using LL(1) Table<br>3.5 Bottom-Up Parsers<br>3.6 Operator Precedence Parser -Basic Concepts<br>    3.6.1 Operator Precedence Relations form Associativity & Precedence<br>    3.6.2 Operator Precedence Grammar | **20** |

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

**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**<br>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<br>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),<br>1.3 IPV4 Protocol Datagram Format, Fragmentation, Checksum, Options.<br>1.4 Routing Properties of routing algorithm, Comparison of Adaptive and Non- Adaptive Routing Algorithms<br>1.5 Congestion Control – Definition, Factors of Congestion, Difference between congestion control and flow control, General Principles of Congestion Control, Congestion Prevention Policies<br>1.6 Network Layer Devices –Routers | 10 |
| II | **Address Mapping**<br>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**<br>3.1 Process-to-Process Delivery Client Server Paradigm, Multiplexing and De-multiplexing, Connectionless Vs Connection-Oriented Service, Reliable Vs Unreliable<br>3.2 User Datagram Protocol(UDP) Datagram Format, Checksum, UDP operations, Use of UDP<br>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.<br>3.4 TCP Features –Numbering System, Byte Number, Sequence Number, Acknowledgement Number, Flow Control, Error Control, Congestion Control<br>3.5 TCP Segment – Format | 07 |

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

**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**<br>1.1 Variables<br>1.2 Server information<br>1.3 Processing forms<br>1.4 Setting response headers<br>1.5 State management<br>1.6 JWT(JSON Web Tokens)<br>1.7 JWT Vs Sessions<br>1.8 SSL | 10 |
| 2. | **JSON with PHP**<br>2.1 Introduction to JSON<br>2.2 JSON syntax<br>2.3 Datatypes in JSON<br>2.4 JSON Vs XML<br>2.5 Encoding JSON in PHP<br>2.6 Decoding JSON in PHP<br>2.7 Accessing the decoded values<br>2.8 Looping through the values | 08 |
| 3. | **AJAX**<br>3.1 Introduction to AJAX<br>3.2 AJAX web application model<br>3.3 AJAX-PHP framework<br>3.4 Performing AJAX validation<br>3.5 Connecting database using php and AJAX | 10 |
| 4. | **Intermediate JavaScript**<br>4.1 More Objects<br>- Constructor Fuctions<br>- Factory Functions<br>- Constructor Method<br>- new keyword<br>4.2 this<br>- this in methods<br>- window scope<br>- this in functions declarations<br>- this in events<br>4.3 Prototypes and ES6 Classes<br>- Built-in constructor functions | 14 |

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

**References** :

1. Kevin Tatroe, Peter MacIntyre (2020), Programming PHP : Creating Dynamic Web Pages(4$^{th}$ 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

**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 Structs, Hibernate and Spring Boot to develop Web 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. | JSON | 6 |
| 2. | React | 6 |
| 3. | Spring Boot<br>3.1 Introduction to Spring<br>3.2 Dependency Injection<br>3.3 Spring AOP<br>3.4 Spring JDBC<br>3.5 Spring hibernate<br>3.6 Spring JDBC Transaction Management<br>3.7 Spring MVC<br>3.8 SOA and Web services | 12 |
| 4. | Hibernate<br>4.1 Hibernate Basics<br>4.2 Hibernate ORM Features<br>4.3 Hibernate project<br>4.4 Hibernate Annotations<br>4.5 Hibernate CRUD Operations<br>4.6 HQL<br>4.7 Native SQL query and Criterial Query<br>4.8 Caching in Hibernate<br>4.9 Transaction Management | 12 |
| 5. | Struts2 | 12 |

**Reference:**

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

**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**<br>1.1 Introduction to Software Project Management: Project phase and project life Cycle, Organizational structure.<br>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<br>1.3 Software Quality: defining software quality, ISO9126, External Standards.<br>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**<br>2.1 Project Integration Management<br>2.2 Project Scope Management<br>2.3 Project Time Management<br>2.4 Project Cost Management<br>2.5 Project Quality Management<br>2.6 Project Human Resource Management<br>2.6 Project Communications Management<br>2.7 Project Risk Management<br>2.8 Project Procurement Management<br>2.9 Project Stakeholder Management | **10** |
| **UNIT 3** | **SOFTWARE EVALUATION AND COSTING**<br>3.1 Project Evaluation: Strategic Assessment, Technical Assessment, cost-benefit analysis, flow forecasting, cost-benefit evaluation techniques, Risk Evaluation.<br>3.2 Selection of Appropriate Project approach: Choosing technologies, choice of process models, structured methods. | **04** |
| **UNIT 4** | **SOFTWARE ESTIMATION TECHNIQUES**<br>4.1 Software Effort Estimation: Problems with Over and under estimations, Basis of software Estimation, Software estimation techniques, expert Judgment, Estimating by analogy.<br>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**<br>5.1 Risk Management: Nature of Risk, Managing Risk, Risk Identification and Analysis, Reducing the Risk.<br>5.2 Resource Allocation: Scheduling resources, Critical Paths, Cost scheduling,<br>5.3 Monitoring and Control: Creating Framework, cost monitoring, prioritizing monitoring. | **06** |
| **UNIT 6** | **INTRODUCTION TO SOFTWARE PROJECT METRICS**<br>6.1 Introduction to Software Project Metrics, Types Of Software Project Metrics, Scope of Software Project Metrics ,Software metrics and Data collection,<br>6.2 Metrics Measurement: Measurement and prediction, Resource measurement, Productivity Measurement, Mapping measurement activities, Measurement tool, Role of Measures (Analyst, tools, Plans, Revision Plans,<br>6.3 Quality Measurement- Quality Standards (ISO,MC-Call, CMM, PSP/TSP)<br>6.4 Globalization issues in project management (Evaluation, Advantages, Dis-advantages )<br>6.5 Impact of the internet on project management (effect on management activities)<br>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

**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 |

**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<br>Set A -<br>    Assignment 1<br>    Assignment 2<br>Set B –<br>    Assignment 1<br>    Assignment 2 |
| **2.** | React<br>Set A -<br>    Assignment 1<br>    Assignment 2<br>Set B –<br>    Assignment 1<br>    Assignment 2 |
| **3.** | Spring Boot<br>Set A -<br>    Assignment 1<br>    Assignment 2<br>Set B –<br>    Assignment 1<br>    Assignment 2 |
| **4.** | Hibernate<br>Set A -<br>    Assignment 1<br>    Assignment 2<br>Set B –<br>    Assignment 1<br>    Assignment 2 |
| **5.** | Struts2<br>Set A -<br>    Assignment 1<br>    Assignment 2<br>Set B –<br>    Assignment 1<br>    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 |