Savitribai Phule Pune University

(Formerly University of Pune)

Three Year B.Sc. Degree Program in Computer Science

(Faculty of Science & Technology)

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

Choice Based Credit System Syllabus

To be implemented from Academic Year 2021 – 2022

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

Semester V (Total credits=22)

Course type	Paper Code	Paper title	Credits		Evaluation		
			T	P	CA	UA	TOTAL
DSEC - I	CS-351	Operating Systems - I	2		15	35	50
	CS-352	Computer Networks - II	2		15	35	50
	CS-357	Practical course based on CS 551		2	15	35	50
DSEC - II	CS-353	Web Technologies – I	2		15	35	50
	CS-354	Foundations of Data Science	2		15	35	50
	CS-358	Practical course based on CS 553		2	15	35	50
DSEC - III	CS-355	Object Oriented Programming - I (Core Java)	2		15	35	50
	CS-356	Theoretical Computer Science and Compiler Construction - I	2		15	35	50
	CS-359	Practical Course based on CS 555		2	15	35	50
SECC - I	CS-3510	Python Programming	1	1	15	35	50
SECC - II	CS-3511	Block Chain / Cloud Technology	1	1	15	35	50

Semester VI (Total credits=22)

Course type	Paper Code	Paper title	Credits		Evaluation		
			T	P	CA	UA	TOTAL
DSEC - I	CS-361	Operating Systems - II	2		15	35	50
	CS-362	Software Testing	2		15	35	50
	CS-367	Practical course based on CS 561		2	15	35	50
DSEC - II	CS-363	Web Technologies – II	2		15	35	50
	CS-364	Data Analytics	2		15	35	50
	CS-368	Practical course based on CS 563		2	15	35	50
DSEC - III	CS-365	Object Oriented Programming - II (Advanced Java)	2		15	35	50
	CS-366	Theoretical Computer Science and Compiler Construction - II	2		15	35	50
	CS-369	Practical Course based on CS 564 and CS 565		2	15	35	50
SECC - III	CS-3610	Mobile Application Development / Software Testing Tools	1	1	15	35	50
SECC - IV	CS-3611	Project	1	1	15	35	50

SavitribaiPhule Pune University
T.Y.B.Sc. (Computer Science) - Sem – V
Course Type: DSEC – I Course Code : CS-351
Course Title : Operating System – I

Teaching Scheme: 2.30 HRs/ week	No. of Credits: 2	Examination Scheme: IE : 15 marks UE: 35 marks

Prerequisites

Data structures like stack, queue, linked list, tree, graph, hashing, file structures, any structured programming language

Course Objectives:
1. To understand the concept of operation systemand its principle
2. To study the various functions and services provided by operating system
3. To understand the notion of process and threads

Course Outcomes: After completion of this course students will be able to understand the concept of
1. Processes and Thread Scheduling by operating system
2. Synchronization in process and threads by operating system
3. Memory management by operating system using with the help of various schemes

Course Contents

Chapter 1	**Introduction to Operating Systems**	**6 lectures**

- Operating Systems Overview- system Overview and Functions of operating systems
- What does an OS do?
- Operating system Operations
- Operating system structure
- Protection and security
- Computing Environments- Traditional, mobile , distributed, Client/server, peer to peer computing
- Open source operating System
- Booting
- Operating System services,
- System calls Types of System calls and their working.

Chapter 2	**Processes and Threads**	**6 lectures**

- Process Concept – The processes, Process states, Process control block.
- Process Scheduling – Scheduling queues, Schedulers, context switch
- Operations on Process – Process creation with program using fork(), Process termination
- Thread Scheduling- Threads, benefits, Multithreading Models, Thread Libraries

Chapter 3	**Process Scheduling**	**6 lectures**

- Basic Concept – CPU-I/O burst cycle, Scheduling Criteria ,CPU scheduler, Preemptive scheduling, Dispatcher
- Scheduling Algorithms – FCFS, SJF, Priority scheduling, Round-robin scheduling, Multiple queue scheduling, Multilevel feedback queue scheduling

Chapter 4	**Synchronization**	**5 lectures**

- Background
- Critical Section Problem

- Semaphores: Usage, Implementation
- Classic Problems of Synchronization – The bounded buffer problem, The reader writer problem, The dining philosopher problem

Chapter 5	**Memory Management**	**13 lectures**

- Background – Basic hardware, Address binding, Logical versus physical address space, Dynamic loading, Dynamic linking and shared libraries
- Swapping
- Contiguous Memory Allocation – Memory mapping and protection, Memory allocation, Fragmentation
- Paging – Basic Method, Hardware support, Protection, Shared Pages
- Segmentation – Basic concept, Hardware
- Virtual Memory Management – Background, Demand paging, Performance of demand paging, Page replacement – FIFO, Optimal, LRU

Reference Books:

1. Operating System Concepts, Avi Silberschatz, Peter Galvin, Greg Gagne, Student Edition, Wiley Asia
2. Operating Systems: Internals and Design Principles, William Stallings, Prentice Hall of India.
3. Distributed Operating Systems Concepts and Design, Pradeep K. Sinha, PHI
4. Advanced Concepts in Operating Systems, M Singhal and NG Shivaratri, Tata McGraw Hill Inc, 2001 (Text Book)
5. Distributed Operating Systems, Maarten van Steen, A S Tanenbaum. Third edition. Pearson Education Asia, 2001

SavitribaiPhule Pune University
T.Y.B.Sc. (Computer Science) Sem - V
Course Code: DSEC- I Course Code : CS-352
Course Title :Computer Networks-II

Teaching Scheme	No. of Credits	Examination Scheme
02:30 **Hrs** / week	2	IE :15 marks UE: 35 marks

Prerequisites:

Prerequisites: Basic knowledge of Networking and ISO/OSI model

Course Objectives

- To understand different protocols of application layer.
- To understand concepts of multimedia.
- Explore the different methods used for Network/INTERNET security.

Course Outcomes

On completion of the course, student will be able to–

- Student will understand the different protocols of Application layer.
- Developed understanding of technical aspect of Multimedia Systems
- Develop various Multimedia Systems applicable in real time.
- Identify information security goals.
- Understand, compare and apply different encryption techniques.
- Come to know about INTERNET security.

Course Contents

Chapter 1	**Application Layer**	**10 Lect**

1.1. 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

1.2. Remotelogging-Telnet:TimesharingEnvironment,Logging,NVT Character set,Embedding Options,mode of operation

1.3. 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

1.4. File Transfer
FTP-Communication over data control connection,File type,data structure,Transmission mode,anonymous FTP

1.5. WWW-Architecture,Client,Server,URL,Cookies

1.6. HTTP-HTTP transaction,messages

Chapter 2	**Multimedia**	**08 Lect**

2.1. Digitizing audio and video,Audio and Video compression

2.2. Streaming Stored audio/video

- First approach
- Second approach
- Third approach
- Fourth approach

2.3. Streaming live audio/video
Real time interactive audio/video- Characteristics, Time relationship, timestamp, Playback buffer, ordering multicasting, translation

2.4. RTP-Packet format

2.5. RTCP-Message types

2.6. Voice over IP-SIP,SIP session
H.323-Architecture,Protocols

Chapter 3	**Cryptography and Network Security**	**09 Lect**

3.1. Cryptography,plain text and cipher text,cipher key,categories of cryptography- Symmetric key,asymmetric key,three types of keys comparison

3.2. Symmetric key cryptography

- Traditional ciphers – substitution cipher,shift cipher,Transposition cipher
- Simple Modern ciphers-XOR,Rotation cipher,s-box,p-box
- Modern round ciphers-DES,AES,IDEA
- Mode of operation-ECB,CBC,CFB,OFB

3.3. Asymmetric key cryptography-RSA, Diffie Hellman, Man in the middle attack

3.4. 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
- Digital signature
- Entity Authentication-Passwords,Fixed passwords challenge-response

Chapter 4	**Security in the Internet**	**09 Lect**

4.1. IPSecurity(IPSec)

- Two modes
- Two security protocols
- Services provided by IPSec
- Security association
- Internet key exchange
- Virtual private network

4.2. SSL/TLS

- SSL services
- Security parameters
- Sessions and connections
- Four protocols
- Transport layer security

4.3. PGP

- Security parameters
- Services
- PGP algorithms
- Key rings
- PGP certificates

4.4. Firewalls

- Packet filter firewall
- Proxy firewall

Reference Books:

R1. Data communications and networking by Behrouz Forouzan 4th/5th edition, McGraw Hill Pvt Ltd.

R2. Computer Networks by Andrew S Tanenbaum, 4th/5th edition, Pearson Education

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) Sem V
Course Type:DSEC – II Course Code: CS-353
Course Title : Web Technologies I

Teaching Scheme	No. of Credits	Examination Scheme
02.30 **Hrs** / week	2	IE : 15 marks UE: 35 marks

Prerequisites

HTML basics for form designing

Course Objectives

- To Design dynamic and interactive Web pages.
- To Learn Core-PHP, Server Side Scripting Language
- To Learn PHP-Database handling

Course Outcomes

On completion of the course, student will be able to–

- Understand how to develop dynamic and interactive Web Page

Course Contents

Chapter 1	**Introduction of Web , HTML and HTTP basics**	**4 Lect**

1.7. HTML: Creating Forms, creating Tables, Managing home page, HTML5 Semantic and Form Elements.
1.8. CSS: Three ways to use CSS, Box Model, Navigation Bar
1.9. Introduction to Web server and Web browser
1.10. HTTP basics

Chapter 2	**PHP basics**	**4 Lect**

2.7. Introduction to PHP
2.8. What does PHP do?
2.9. Lexical structure
2.10. Language basics

Chapter 3	Function and String	**7 Lect**

3.1 Defining and calling a function
3.2 Default parameters
3.3 Variable parameters, Missing parameters
3.4 Variable function, Anonymous function
3.5 Types of strings in PHP
3.6 Printing functions
3.7 Encoding and escaping
3.8 Comparing strings
3.9 Manipulating and searching strings
3.10 Regular expressions

Chapter 4	Arrays	**5 Lect**

4.1 Indexed Vs Associative arrays
4.2 Identifying elements of an array
4.3 Storing data in arrays
4.4 Multidimensional arrays

4.4Extracting multiple values 4.5 Converting between arrays and variables 4.6 Traversing arrays 4.7 Sorting 4.8 Action on entire array		
Chapter 5	Introduction to Object Oriented Programming	**6 Lect**
5.1 Classes 5.2 Objects 5.3 Introspection 5.4 Serialization 5.5 Inheritance 5.6 Interfaces 5.7Encapsulation		
Chapter 6	Files and directories	**4 Lect**
6.1 Working with files and directories 6.2 Opening and Closing, Getting information about file, Read/write to file, Splitting name and path from file, Rename and delete files 6.3 Reading and writing characters in file 6.4 Reading entire file 6.5 Random access to file data 6.6 Getting information on file 6.7 Ownership and permissions		
Chapter 7	Databases (PHP-PostgreSQL)	**6 Lect**
7.1 Using PHP to access a database 7.2 Relational databases and SQL 7.3 PEAR DB basics 7.4 Advanced database techniques		
Reference Books:		
1. HTML & CSS: The Complete Reference, Fifth Edition Author: Thomas A. Powell First published: 01 Jan 2010. 2. Programming PHP By Rasmus Lerdorf and Kevin Tatroe, O'Reilly publication 3. Beginning PHP 5 , Wrox publication 4. PHP web sevices, Wrox publication 5. Mastering PHP , BPB Publication 6. PHP cookbook, O'Reilly publication 7. PHP for Beginners, SPD publication 8. Programming the World Wide Web , Robert W Sebesta(3rd Edition) 9. HTML 5 Black Book : Covers Css3, Javascript, XML, XHTML, Ajax, PHP And Jquery by Kogent Learning Solutions Inc, Published November 2011 by Dreamtech Press 10. www.php.net.in 11. www.W3schools.com 12. www.wrox.com		

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) Sem V
Course Type:DSEC – II Course Code: CS-354
Course Title : Fondation of Data Science

Teaching Scheme	No. of Credits	Examination Scheme
02:30 Hrs / week	2	IE : 15 marks UE: 35 marks

Prerequisites

- Problem solving using computers
- Basic mathematics and statistics
- Knowledge of Databases

Course Objectives

- Provide students with knowledge and skills for data-intensive problem solving and scientific discovery
- Be prepared with a varied range of expertise in different aspects of data science such as data collection, visualization, processing and modeling of large data sets.
- Acquire good understanding of both the theory and application of applied statistics mathematics and computer science based existing data science models to analyse huge data sets originating from diversified application areas.
- Be able to create models using the knowledge acquired from the program to solve future challenges and real-world problems requiring large scale data analysis.
- Be better trained professionals to cater the growing demand for data scientists in industry.

Course Outcomes

On completion of the course, student will be able to–

- Perform Exploratory Data Analysis
- Obtain, clean/process, and transform data.
- Detect and diagnose common data issues, such as missing values, special values, outliers, inconsistencies, and localization.
- Demonstrate proficiency with statistical analysis of data.
- Present results using data visualization techniques.
- Apply concepts of data analysis, data collection, modeling, and inference
- Prepare data for use with a variety of statistical methods and models and recognize how the quality of the data and the means of data collection may affect conclusions.

Course Contents

Chapter 1	**Introduction to Data Science**	**6 Lect**

1.11. The Art of Data Science, The 5 V's: Volume, Velocity, Variety, Veracity, Value
1.12. Why Data Science?
1.13. Evolution of Data Science
1.14. Applications of Data Science
1.15. The Data Science Lifecycle
1.16. Data Scientist's Toolbox

Chapter 2	**Statistical and Mathematical Foundations**	**10 Lect**

2.1.Exploring Probability and Inferential Statistics
2.2.Basic Statistical descriptions of data
  2.2.1. Measuring the Central Tendency: Mean, Median, and Mode
  2.2.2. Measuring the Dispersion of Data: Range, Quartiles, Variance,

2.2.3. Standard Deviation, and Interquartile Range

2.3.Quantifying Correlation

2.4.Multiple hypothesis testing, Parameter Estimation methods,

2.5.Confidence intervals, Bayesian statistics and Probability Distributions (Uniform, Normal, Poisson, Binomial, Bernoulli distribution).

2.6.Introducing Regression Methods

2.6.1 Linear Regression

2.6.2: Polynomial Regression

2.6.3: Logistic Regression

2.7.Measuring Data Similarity and Dissimilarity, Proximity measures

2.8.Detecting Outliers

Chapter 3	**Data Collection and Preprocessing**	**14 Lect**

3.1. Properties of Data

Structured, semi-structured, Unstructured Data, graph based, streaming data, Quantitative and Categorical Data, Big Data and Little Data

3.2.Types of data

Integers, Floats, Text Data, Text Files, Dense Numerical Arrays, Compressed or Archived Data, SV Files, JSON Files, XML Files, HTML Files , Tar Files, GZip Files, Zip Files, Image Files: Rasterized, Vectorized, and/or Compressed

3.3.Data Objects and Attribute Types: What Is an Attribute?, Nominal , Binary, Ordinal Attributes, Numeric Attributes, Discrete versus Continuous Attributes

3.4.Data sources - Open Data, Social Media Data, Multimodal Data, standard datasets

3.5.Collecting data

Hunting, scraping, logging, Combining datasets, aggregation and grouping

3.6.Data munging

3.6.1. Data Quality: Why Preprocess the Data?

3.6.2. Cleaning Data - Missing Values, Noisy Data (Duplicate Entries, Multiple Entries for a Single Entity, Missing Entries, NULLs, Huge Outliers, Out-of-Date Data, Artificial Entries, Irregular Spacings, Formatting Issues - Irregular between Different Tables/Columns, Extra Whitespace, Irregular Capitalization, Inconsistent Delimiters, Irregular NULL Format, Invalid Characters, Incompatible Datetimes)

3.6.3. Data Transformation by Normalization, Label and One Hot Encoding

Chapter 4	**Data Visualization**	**6 Lect**

4.5. Exploratory Data Analysis, purpose, types

4.6. Visualization workflow

4.7. Data Representation: chart types: categorical, hierarchical, relational, temporal & spatial;

4.8. 2-D charts: bar charts, Clustered bar charts, dot plots, connected dot plots, pictograms, proportional shape charts, bubble charts, radar charts, polar charts, Range chart, Box-and-whisker plots, univariate scatter plots, histograms word cloud, pie chart, waffle chart, stacked bar chart, back-to-back bar chart, treemap.

Reference Books:

1) The Data Science Design Manual, Steven S. Skiena, Springer, 2017
2) Introducing data science: big data, machine learning, and more, using Python tools, Cielen D., Meysman A. D., & Ali M., Manning Publications Co., 2016
3) Python Data Science Essentials, Alberto Boschetti, Luca Massaron, Second Edition, 2016 Packt Publishing
4) The Data Science Handbook, Field Cady, John Wiley & Sons, Inc, 2017

5) Python Data Science Handbook - Essential Tools for Working with Data, Jake VanderPlas, O'Reilly, 2017
6) Andy Kirk, Data Visualization A Handbook for Data Driven Design, Sage Publications, 2016
7) The Data Warehouse Etl Toolkit: Practical Techniques For Extracting, Cleaning, Conforming, And Delivering Data; by Ralph Kimball; Publisher: WILEY INDIA, Year – 2004
8) A Hands-On Introduction to Data Science CHIRAG SHAH University of Washington Cambridge University Press
9) Practical Statistics for Data Scientists: 50 Essential Concepts, Peter Bruce,Shroff/O'Reilly; First edition, 2017

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – V
Course Type: DSEC – III Course Code : CS-355
Course Title : Object Oriented Programming – I (Core Java)

Teaching Scheme	No. of Credits	Examination Scheme
02:30 Hrs / week	2	IE : 15 marks UE: 35 marks

Prerequisites

- Knowledge of C Programming language

Course Objectives

- To learn Object Oriented Programming language
- To study various java programming concept like Interface, File and Exception Handling etc.
- To design User Interface using Swing and AWT

Course Outcomes

On completion of the course, student will be able to–

- Understand the concept of classes, objects and packages.
- To develop GUI based application.

Course Contents

Chapter 1	An Introduction to Java	3 Lect

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

Chapter 2	An Overview of Java	3 Lect

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)

Chapter 3	Objects and Classes	6 Lect

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 , Anonymous Classes
3.8 Creating, Accessing and using Packages
3.9 Creating jar file and manifest file
3.10 Wrapper Classes
3.11Garbage Collection (finalize() Method)
3.12 Date and time processing

Chapter 4	**Inheritance and Interface**	**6 Lect**
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 and Functional Interface 4.8 Object Cloning		
Chapter 5	**Exception Handling**	**3 Lect**
5.1 Dealing Errors, Exception class, Checked and Unchecked exception 5.2 Catching exception, Multiple catch block, Nested try block and exception handling 5.3 Creating user defined exception 5.4 Assertions		
Chapter 6	**Strings , Streams and Files**	**6 Lect**
6.1 String class and String Buffer Class , StringTokenizer class 6.2 Formatting string data using format() method , toString method 6.3 Stream classes Byte Stream classes Character Stream Classes 6.4 Using the File class , Creation of files 6.5 Reading/Writing characters and bytes 6.6 Handling primitive data types 6.7 Random Access files		
Chapter 7	**User Interface Components with AWT and Swing**	**6 Lect**
7.1 What is AWT ? What is Swing? Difference between AWT and Swing. 7.2 The MVC Architecture and Swing 7.3 Layout Manager and Layouts, The JComponent class 7.4 Components – JButton, JLabel, JText, JTextArea, JCheckBox and JRadioButton, JList, JComboBox, JMenu and JPopupMenu Class, JMenuItem and JCheckBoxMenuItem, JRadioButtonMenuItem , JScrollBar 7.5 Dialogs (Message, confirmation, input), JFileChooser, JColorChooser 7.6 Event Handling: Event sources, Listeners 7.7 Mouse and Keyboard Event Handling 7.8 Adapters 7.9 Anonymous inner class		
Chapter 8	**Applet**	**3 Lect**
8.1 Applet Life Cycle , appletviewer tool 8.2 Applet HTML Tags 8.3 Passing parameters to Applet 8.4 repaint() and update() method		
Reference Books:		
R1. Complete reference Java by Herbert Schildt(5th edition) R2. Java 2 programming black books, Steven Horlzner R3. Programming with Java , A primer ,Forth edition , By E. Balagurusamy R4. Core Java Volume-I-Fundamentals, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Pres		

Savitribai Phule Pune University

T.Y.B.Sc. (Computer Science) Sem - V

Course Type: DSEC- III **Course Code: CS-356**

Paper Title: Theoretical Computer Science and Compiler Construction I

Teaching Scheme	No. of Credits	Examination Scheme
02:30 **Hrs** / week	2	IE : 15 marks UE: 35 marks

Prerequisites

- Mathematical Preliminaries Sets (Subset, Set Operations), Relations (Properties of Relations, Closure of Relations) and Functions
- Discrete Mathematics- Graphs, Trees, Logic and Proof Techniques

Course Objectives

- To understand the Finite Automata, Pushdown Automata and Turing Machine.
- To understand the Regular Language, Context Free Language, Context Sensitive Language and Unrestricted Language.
- To understand the relation between Automaton and Language

Course Outcomes

On completion of the course, student will be able to–

- Understand the use of automata during language design.
- Relate various automata and Languages.

Course Contents

Chapter 1	**Finite Automaton**	**11 Lect**

1.1 Introduction: Symbol, Alphabet, String, Prefix & Suffix of Strings, Formal Language, Operations on Languages.

1.2 Deterministic finite Automaton – Definition, DFA as language recognizer, DFA as pattern recognizer.

1.3. Nondeterministic finite automaton – Definition and Examples.

1.4. NFA To DFA (Myhill Nerode Method)

1.5. NFA with ε- transitions Definition and Examples.

1.6. NFA with ε-Transitions to DFA & Examples

1.7. Finite automaton with output – Mealy and Moore machine, Definition and Examples.

1.8. Minimization of DFA, Algorithm & Problem using Table Method.

Chapter 2	**Regular Expressions and Languages**	**4 Lect**

<table>
<tr><td colspan="3">2.1 Regular Expressions (RE): Definition & Example
2.2 Regular Expressions Identities.
2.3 Regular language-Definition and Examples.
2.4 Conversion of RE to FA-Examples.
2.5 Pumping lemma for regular languages and applications.
2.6 Closure Properties of regular Languages</td></tr>
<tr><td><b>Chapter 3</b></td><td><b>Context-Free Grammars and Languages</b></td><td><b>9 Lect</b></td></tr>
<tr><td colspan="3">3.1 Grammar - Definition and Examples.
3.2 Derivation-Reduction - Definition and Examples.
3.3 Chomsky Hierarchy.
3.4 CFG: Definition & Examples. LMD, RMD, Parse Tree
3.5 Ambiguous Grammar: Concept & Examples.
3.6 Simplification of CFG: Removing Useless Symbols, Unit Production, ϵ-production and Nullable Symbol.
3.7 Normal Forms: Greibach Normal Form (GNF) and Chomsky Normal Form (CNF)
3.8 Regular Grammar: Definition.
3.8.1 Left linear and Right Linear Grammar-Definition and Example.
3.8.2 Equivalence of FA & Regular Grammar
3.8.3 Construction of regular grammar equivalent to a given DFA.
3.8.4 Construction of a FA from the given right linear grammar</td></tr>
<tr><td><b>Chapter 4</b></td><td><b>Push Down Automata</b></td><td><b>5 Lect</b></td></tr>
<tr><td colspan="3">4.1 Definition of PDA and examples.
4.2 Construction of PDA using empty stack and final State method: Examples using stack method.
4.3 Definition DPDA & NPDA, their correlation and Examples of NPDA
4.4 CFG (in GNF) to PDA: Method and examples</td></tr>
<tr><td><b>Chapter 5</b></td><td>Turing Machine</td><td><b>7 Lect</b></td></tr>
<tr><td colspan="3">5.1 The Turing Machine Model, Definition and Design of TM
5.2 Problems on language recognizers.
5.3 Language accepted by TM.
5.4 Types of Turing Machines (Multitrack TM, Two-way TM, Multitape TM, Non-deterministic TM)
5.5 Introduction to LBA (Basic Model) & CSG. (Without Problems)
5.6 Computing TM, Enumerating TM, Universal TM.
5.7 Recursive Languages
5.7.1. Recursive and Recursively enumerable Languages.
5.7.2. Difference between recursive and recursively enumerable language.
5.8 Turing Machine Limitations
5.9 Decision Problem, Undecidable Problem, Halting Problem of TM</td></tr>
</table>

Reference Books

R1. Introduction to Automata Theory, Languages and Computation, John E. Hopcraft, Rajeev Motwani, Jeffrey D. Ullman, Third Edition, Pearson Education Publication, 2008

R2. Introduction to Automata theory, Languages and computation By John E. Hopcroft and Jeffrey Ullman – Narosa Publishing House, 1995

R3. Theory of Computer Science Automata, Languages and Computation, K.L.P. Mishra, N. Chandrasekaran, Publication- Prentice Hall of India, 2008

R4. Introduction to Computer Theory Daniel I. A. Cohen – 2nd edition – John Wiley & Sons, 1996

R5. Introduction to Languages and The Theory of Computation John C. Martin The McGraw-Hill, Fourth Edition, 2011

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) Sem - V
Course Type: DSEC- I Course Code: CS 357
Course Title : Practical Course based on CS 351

Teaching Scheme: 4.20 Hrs/ week	No. of Credits: 2	Examination Scheme: IE : 15 marks UE: 35 marks

Course Objectives:
1. To understand the concept of process scheduling with the help of simulation.
2. To study the concept demand paging concepts in operating system.
3. To understand the working of operating system shell.

Course Outcomes: After completion of this course students will be able to understand the concept of
1. Installation of Linux operating system and administration.
2. Processes and Thread Scheduling by operating system
3. Memory management by operating system using with the help of various schemes

Guidelines:
1. Operating system platform – Linux
2. Programming language - C/C++/Java

List of Assignments:

- Installation of Linux Operating System & Linux Administration (1 Slot)
- Simulation of Operating System Shell and its working (2 Slots)
- Simulation of CPU Scheduling Algorithms – FCFS, SJF, Priority and Round Robin (4 Slots)
- Simulation of demand paging using memory page replacement algorithms – FIFO, LRU, OPT (5 Slots)

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) Sem – V
Course Type: DSEC - II **Course Code: CS 358**
Course Title : Practical Course based on CS 353 and CS 354

Teaching Scheme:	No. of Credits:	Examination Scheme:
4.20HRs/ week	2	IE : 15 marks UE: 35 marks

Course Objectives:

- To Design dynamic and interactive Web pages.
- To Learn Core-PHP, Server Side Scripting Language
- To Learn PHP- Database handling

Course Outcomes:

- Understand how to develop dynamic and interactive Web Page

Guidelines:

3. Operating Environment: HTML5.0, PHP 5.0 and above , Webserver
4. Submission
5. Assessment

List of Assignments:

Session 1 : Assignments based on HTML and HTML5.0
Session 2 : Assignments based CSS, Box Model, Navigation Bar
Session 3 : Assignments using Bootstrap
Session 4 : Assignments using Bootstrap
Session 5 : Assignments based Function and String
Session 6 : Assignments based Function and String
Session 7 : Assignments based Arrays
Session 8 : Assignments based Object Oriented Programming
Session 9 : Assignments based Object Oriented Programming
Session 10: Assignments based Files and directories
Session 11 : Assignments based Databases (PHP-PostgreSQL)
Session 12 : Assignments based Databases (PHP-PostgreSQL)

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) Sem – V
Course Type: DSEC - II **Course Code: CS 358**
Course Title : Practical Course based on CS 353 and CS 354

Teaching Scheme 4 hrs 20 mins / week Batch Size : 12	No. of Credits 2	Examination Scheme IE : 15 marks UE: 35 marks

Operating Environment:

- **Operating system:** Linux
- **Editor:**Any linux based editor like vi, gedit etc.
- **Compiler** : cc or gcc
- **Languages**: Python

Lab Book:

The lab book is to be used as a hands-on resource, reference and record of assignment submission and completion by the student. The lab book contains the set of assignments which the student must complete as a part of this course.

Programming Assignments:

Programs should be done individually by the student in their respective login. The codes should be uploaded on either the local server, Moodle, Github or any open source LMS. Print-outs of the programs and output may be taken but not mandatory for assessment.

Assessment:

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include-timely completion, performance, innovation, efficient codes and good programming practices.

Course Contents:

Suggested Assignments for Foundations of Data Science

Assignment 1: The Data Science environment

Getting introduced to Python and essential packages like NumPy, SciPy, pandas, scikit-learn,matplotlib, jupyter, beautiful-soup,

Assignment 2: Loading the dataset

Select a dataset from a list of publicly available datasets at UCI Machine Learning Repository and load it using Pandas. (Import different dataformat files like .CSV,.htm,.json etc. Briefly describe what the dataset is about and size of the dataset (e.g. number of tables, number of instances and attributes, etc.)

Assignment 3: Basic statistical operations

Select one attribute and discuss appropriate measures of the central tendency and dispersion for the attribute. Use a subset of the attribute values (of your own choice) from the dataset and compute the mean, median, mode, range, quartiles, and variance for the attribute.

Assignment 4: Data preprocessing

Apply data preprocessing techniques that are likely required for the dataset.
1)Partition them into appropriate number of bins by equal-frequency as well as equal-width partitioning.
2)Use smoothing by bin means to smooth the data based on the above partitioning,
3)Normalize the attribute based on min-max normalization and z-score normalization.
Comment on which method you would prefer to use for partitioning, smoothing, and normalization for the given attribute.

Assignment 5: Data Visualization with matplotlib

View the data using various 2-D, 3-D plots and charts, setting styles, saving the figures, customizing the legends, multiple subplots,

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) Sem – V
Course Type: DSEC - III Course Code: CS359
Course Title : Practical Course based on CS 355

Teaching Scheme 4 hrs 20 mins / week Batch Size : 12	No. of Credits 2	Examination Scheme IE : 15 marks UE: 35 marks

Course Objectives:

1. Covers the complete scope of the syllabus.
2. Bringing uniformity in the way course is conducted across different colleges.
3. Continuous assessment of the students.

Course Outcomes:

1. Use an integrated development environment to write, compile, run, and test simple object-oriented Java programs.
2. Read and make elementary modifications to Java programs that solve real-world problems.
3. Validate input in a Java program.

Guidelines:

6. Operating Environment :

- **Operating system:** Linux
- **Editor :** Anylinux based editor like vi, gedit and Use of IDE – Eclipse etc.
- **Compiler** : javac

7. Submission :

Each assignment will be assessed on a scale of 0 to 5 as indicated below.

Not done	0
Incomplete	1
Late Complete	2
Needs improvement	3
Complete	4
Well Done	5

8. Assessment :

Easy : All exercises are compulsory.
Medium : All exercises are compulsory.

List of Assignments :

Assignment 1 : Java Tools and IDE, Simple java programs [Slot – 2]

Introduction to the java environment
Use of java tools like java, javac, jdb and javadoc

Use of IDE – Eclipse (demo)
Defining simple classes and creating objects.

Assignment 2 : Array of Objects and Packages [Slot – 2]

Defining a class.
Creating an array of objects.
Creating a package. (Using package command)
Using packages. (Using import command)

Assignment 3 : Inheritance and Interfaces [Slot – 2]

To implement inheritance in java.
To define abstract classes.
To define and use interfaces and Functional Interface.
Use predefined interfaces like Cloneable.

Assignment 4 : Exception Handling [Slot – 2]

Demonstrate exception handling mechanism in java.
Defining user defined exception classes.
Use of try, catch, throw, throws and finally keywords.
Defining user defined exception classes.

Assignment 5 : I/O and File Handling [Slot – 2]

Performing Input / Output operations using console and files.
Use of Types of streams, Byte and Character stream classes, File class

Assignment 6 : GUI Designing, Event Handling and Applets [Slot – 2]

To demonstrate GUI creation using Swing package and Layout managers.
Understand the Event Handling mechanism in java.
Using Event classes, Event Listeners and Adapters.
Creating java applets which run in a web browser.

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC – I Course Code : CS-3510
Course Title: Python Programming (Theory)

Teaching Scheme	No. of Credits	Examination Scheme:
01:15 Hrs / week	1	(Theory + Practical) IE : 15 marks UE: 35 marks

Course Objectives

1. To introduce programming concepts using python
2. Student should be able to develop Programming logic using python
3. To develop basic concepts and terminology of python programming
4. To test and execute python programs

Course Outcomes

On completion of the course, student will be able to–

- Develop logic for problem solving
- Determine the methods to create and develop **Python programs** by utilizing the data
- structures like lists, dictionaries, tuples and sets.
- To be familiar about the basic constructs of programming such as data, operations, conditions, loops, functions etc.
- To write python programs and develop a small application project

Course Contents

Chapter 1	**An Introduction to Python**	**3 Lect**

1.1 Introduction to Python

The Python Programming Language, History, features, Applications,Installing Python, Running Simple Python program

1.2 Basics of Python

Standard data types - basic, none, Boolean (true & False), numbers, Variables, Constants,Python identifiers and reserved words, Lines and indentation, multi-line statements and Comments,Input/output with print and input ,functions Declaration, Operations on Data such as assignment, arithmetic, relational, logical and bitwise operations, dry run, Simple Input and output etc.

Chapter 2	**Control Statements**	**5 Lect**

2.1 **Sequence Control** – Precedence of operators, Type conversion

2.2 **Conditional Statements**: if, if-else, nested if-else,

2.3 **Looping**- for, while, nested loops, loop control statements (break, continue, pass)

a. **Strings**: declaration, manipulation, special operations, escape character, string formatting operator, Raw String, Unicode strings, Built-in String methods.

Chapter 3	**Lists, functions, tuples and dictionaries, Sets**	**9 Lect**

3.1 **Python Lists:** Concept, creating and accessing elements, updating & deleting lists, traversing a List, reverse Built-in List Operators, Concatenation, Repetition, In Operator, Built-in List functions and methods.

3.2 Functions: Definitions and Uses, Function Calls, Type Conversion Functions, Math Functions, Composition, Adding New Functions, Flow of Execution, Parameters and Arguments, Variables and Parameters, Stack Diagrams, Void Functions, Anonymous functions Importing with from, Return Values, Boolean Functions, More Recursion, Functional programming tools - filter(), map(), and reduce(),recursion, lambda forms.

3.3 Tuples and Dictionaries: Tuples, Accessing values in Tuples, Tuple Assignment, Tuples as return values, Variable-length argument tuples, and Basic tuples operations, Concatenation, Repetition, in Operator, Iteration, Built-in tuple functions, indexing, slicing and matrices. Creating a Dictionary, Accessing Values in a dictionary, Updating Dictionary, Deleting Elements from Dictionary, Properties of Dictionary keys, Operations in Dictionary, Built-In Dictionary Functions, Built-in Dictionary Methods.

3.4 Sets- Definition, transaction of set(Adding, Union, intersection), working with sets

Chapter 4	**Modules ,Working with files, Exception handling**	**4 Lect**

4.1 Modules: Importing module, Creating & exploring modules, Math module, Random module, Time module

4.2 Packages: Importing package, creating package,examples

4.3 Working with files: Creating files and Operations on files (open, close, read, write), File object attributes, file positions, Listing Files in a Directory, Testing File Types, Removing files and directories, copying and renaming files, splitting pathnames, creating and moving directories

4.4 Regular Expression- Concept of regular expression, various types of regular expressions, using match function.

4.5 Exception Handling: Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions.

Chapter 5	**Classes and objects**	**3 Lect**

5.1 Overview of OOP (Object Oriented Programming), Class Definition, Creating Objects, Instances as Arguments, Instances as return values, Built-in Class Attributes,

5.2 Inheritance, Data Encapsulation, Data Hiding (double underscore prefix).

Reference Books:

1. An Introduction to Computer Science using Python 3 by Jason Montojo, Jennifer Campbell, Paul Gries, The pragmatic bookshelf-2013
2. James Payne, "Beginning Python: Using Python and Python 3.1,Wrox Publication
3. Introduction to Computer Science Using Python- Charles Dierbach, Wiley Publication Learning with Python ", Green Tea Press, 2002
4. Introduction to Problem Solving with Python by E balguruswamy,TMH publication-2016
5. Beginning Programming with Python for Dummies Paperback – 2015 by John Paul Mueller
6. Object-oriented Programming in Python, Michael H. Goldwasser, David Letscher, Pearson Prentice Hall-2008

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC – I **Course Code : CS-3510**
Course Title: Python Lab Course

Teaching Scheme 2.10 hrs / week Batch Size : 12	No. of Credits 1	Examination Scheme: (Theory + Practical) IE : 15 marks UE: 35 marks

Course Objectives:

1. To understand why Python is a useful scripting language for developers.
2. To learn how to design and program Python applications.

Course Outcomes:

At the end of the course, the student will be able

1. To develop adequate skills in programming like write, test and debug programs.
2. To explain basic principles of python programming language
3. To understand, use and implement list, tuples, set, dictionary and functions
4. To understand, use and implement of file handling, exception handling, regular expression and object oriented concepts

Lab Book:

The lab book is to be used as a hands-on resource, reference and record of assignment submission and completion by the student. The lab book contains the set of assignments which the student must complete as a part of this course.

Programming Assignments:

Programs should be done individually by the student in their respective login. The codes should be uploaded on either the local server, Moodle, Github or any open source LMS. Print-outs of the programs and output may be taken but not mandatory for assessment.

Assessment:

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will be assigned grade/marks based on parameters with appropriate Weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include-timely completion, performance, innovation, efficient codes and good programming practices.

Assignment 1 - Python Basics

a) Basic Installation and Simple Programs

1) Python Interpreter installation and Demonstration
2) Anaconda (Jupyter Notebook) Installation and Demonstration
3) Framework Demonstration**(Flask, Bottle, CherryPy, Dash, Django, Falcon)**
4) Python Program to Print Hello world!
5) Python Program to Calculate the Area of a Triangle
6) Python Program to Swap Two Variables
7) Python Program to Generate a Random Number

b) Control Statements and Loops

1) Write a Python Program to Check if a Number is Positive, Negative or Zero
2) Write a Python Program to Check Leap Year
3) Write a Python Program to Print all Prime Numbers in an Interval
4) Write a Python Program to Print the Fibonacci sequence
5) Write a Python Program to Check Armstrong Number
6) Write a Python Program to Find the Sum of Natural Numbers
7) Write a Python Program to Find the Factorial of a Number

Assignment 2 – Arrays, Strings, and Functions

Arrays

1) Write a Python program to create an array of 5 integers and display the array items. Access individual element through indexes.
2) Write a Python program to append a new item to the end of the array.
3) Write a Python program to append items from a specified list.
4) Write a Python program to insert a new item before the second element in an existing array.
5) Write a Python program to reverse the order of the items in the array.
6) Write a Python program to get the number of occurrences of a specified element in an array.
7) Write a Python program to remove the first occurrence of a specified element from an array.

Strings

1) Write a python program to check whether the string is Symmetrical or Palindrome
2) Write a python program to Reverse words in a given String
3) Write a python program to remove i'th character from string in different ways
4) Write a python program Words Frequency in String Shorthands
5) Write a python program Convert Snake case to Pascal case
6) Write a python program to print even length words in a string
7) Write a python program to accept the strings which contains all vowels

Functions

1) Write a Python function to find the Max of three numbers.
2) Write a Python program to reverse a string.
3) Write a Python function to calculate the factorial of a number (a non-negative integer). The function accepts the number as an argument.
4) Write a Python function that accepts a string and calculate the number of upper case letters and lower case letters.
5) Write a Python function that checks whether a passed string is palindrome or not.
6) Write a Python program to access a function inside a function.
7) Write a Python program to detect the number of local variables declared in a function.

Assignment 3 - List, Tuples, Sets, and Dictionary

List

1) Write a Python program to sum all the items in a list.
2) Write a Python program to multiplies all the items in a list.
3) Write a Python program to get the largest number from a list.

4) Write a Python program to get the smallest number from a list.
5) Write a Python program to count the number of strings where the string length is 2 or more and the first and last character are same from a given list of strings.
6) Write a Python program to get a list, sorted in increasing order by the last element in each tuple from a given list of non-empty tuples.
7) Write a Python program to remove duplicates from a list.

Tuples

1) Write a Python program to create a tuple.
2) Write a Python program to create a tuple with different data types.
3) Write a Python program to convert a tuple to a string.
4) Write a Python program to convert a list to a tuple.
5) Write a Python program to remove an item from a tuple.
6) Write a Python program to slice a tuple.
7) Write a Python program to reverse a tuple.

Sets

1) Write a Python program to create a set.
2) Write a Python program to iterate over sets.
3) Write a Python program to add and remove member(s) in a set.
4) Write a Python program to create an intersection of sets.
5) Write a Python program to create a union and difference of sets.
6) Write a Python program to create a symmetric difference.
7) Write a Python program to check if a set is a subset of another set.

Dictionary

1) Write a Python script to sort (ascending and descending) a dictionary by value.
2) Write a Python script to add a key to a dictionary.
3) Write a Python script to merge two Python dictionaries.
4) Write a Python program to remove duplicates from Dictionary.
5) Write a Python program to sum all the items in a dictionary.
6) Write a Python program to multiply all the items in a dictionary.
7) Write a Python program to remove a key from a dictionary.

Assignment 4 - File Handling and Date-Time

File Handling

1) Write a Python program to read an entire text file.
2) Write a Python program to read first or last n lines of a file.
3) Write a Python program to append text to a file and display the text.
4) Write a Python program to read a file line by line and store it into a list.
5) Write a Python program to read a file line by line store it into a variable.
6) Write a Python program to count the number of lines in a text file.
7) Write a Python program to copy the contents of a file to another file .

Date-Time

1) Write a python program to get Current Time
2) Get Current Date and Time using Python
3) Write a python to Find yesterday’s, today’s and tomorrow’s date
4) Write a python program to convert time from 12 hour to 24 hour format
5) Write a python program to find difference between current time and given time
6) Write a python Program to Create a Lap Timer

7) Find number of times every day occurs in a Year

Assignment 5 - Exception handling and Regular expression

Exception handling

1) Assertions in Python
2) The except Clause with No Exceptions
3) The except Clause with Multiple Exceptions
4) The try-finally Clause
5) Argument of an Exception
6) User-Defined Exceptions
7) Raising Exception

Common Examples of Exception:

1) Division by Zero
2) Accessing a file which does not exist.
3) Addition of two incompatible types
4) Trying to access a nonexistent index of a sequence
5) Removing the table from the disconnected database server.
6) ATM withdrawal of more than the available amount

Regular expression

1) Write a python program to Check if String Contain Only Defined Characters using Regex
2) Write a python program to find the most occurring number in a string using Regex
3) Write a python Regex to extract maximum numeric value from a string
4) Write a python to Check whether a string starts and ends with the same character or not
5) Write a python Program to check if a string starts with a substring using regex
6) Write a python Program to Check if an URL is valid or not using Regular Expression
7) Write a python Program to Parsing and Processing URL using Python – Regex

Assignment 6 - Classes and Objects (OOP)

Classes and Objects (OOP)

1) Write a Python program to demonstrate working of classes and objects.
2) Write a Python program to demonstrate class method & static method.
3) Write a Python program to demonstrate constructors.
4) Write a Python program to import built-in array module and display the namespace of the said module.
5) Write a Python program to demonstrate inheritance.
6) Write a Python program to demonstrate aggregation/compositions
7) Write a Python function student_data () which will print the id of a student (student_id). If the user passes an argument student_name or student_class the function will print the student name and class.
8) Write a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle.
9) Write a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle.
10) Write a Python class named Student with two attributes student_id, student_name. Add a new attribute student_class. Create a function to display the entire attribute and their values in Student class.

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC – II Course Code : CS-3511
Course Title: Cloud Computing (Theory)

Teaching Scheme 01:15 Hrs / week	No. of Credits 1	Examination Scheme: (Theory + Practical) IE : 15 marks UE: 35 marks

Prerequisite:

Knowledge of Operating System, Fundamentals of Computer Networks.
Good Understanding of Object Oriented Programming Concepts

Course Objectives

- To understand the principles and paradigm of Cloud Computing
- To appreciate the role of Virtualization Technologies

Course Outcomes

On completion of the course, student will be able to–

- design and deploy Cloud Infrastructure

Course Contents

Chapter 1	**Introduction to Cloud Computing**	**4 Lect**

1.1 Introduction
1.2 Definition and essential characteristics of cloud computing
1.3 History and Evolution of Cloud Computing,
1.4 Evolution of cloud computing,
1.5 Emerging technologies supported by cloud computing.
1.6 Advantages and disadvantages of cloud computing

Chapter 2	**Cloud Computing Models**	**6 Lect**

2.1 Types of service (IaaS, PaaS, SaaS)
2.1 Deployment models of cloud computing.(Public, Private, and Hybrid.)

Chapter 3	**Components of Cloud Computing**	**4 Lect**

3.1 Cloud computing architecture: virtualization virtual machines, bare metal servers
3.2 Types of cloud storage
3.3 To build a secure cloud

Chapter 4	Cloud Security, and Career Opportunities	**4 Lect**

4.1 Cloud Security
4.2 Encryption
4.3 Careers and Opportunities

Reference Books:

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC – II Course Code : CS-3511
Course Title: Cloud Computing Lab Course

Teaching Scheme 2.10 hrs / week Batch Size : 12	No. of Credits 1	Examination Scheme: (Theory + Practical) IE : 15 marks UE: 35 marks

Assignment 1 - Working and Implementation of Infrastructure as a service.

Assignment 2 – Working and Implementation of Software as a service.

Assignment 3 - Working and Implementation of Platform as a services.

Assignment 4 - Practical Implementation of Storage as a Service.

Assignment 5 - Working of Google drive to make spreadsheet and notes.

Assignment 6 - Case studies on open source development tools for cloud computing (any one)

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC – II Course Code : CS-3511
Course Title: Block Chain Technology (Theory)

Teaching Scheme	No. of Credits	Examination Scheme:
01:15 Hrs / week	1	(Theory + Practical) IE : 15 marks UE: 35 marks

Prerequisite:
Understanding of Object Oriented Programming Concepts
Knowledge of Javascript or Python

Course Objectives

1. To understand the principles and paradigm of Block Chain.

Course Outcomes

On completion of the course, student will be able to–

- Design simple blockchain applications.

Course Contents

Chapter 1	Introduction to Blockchain	4 Lect

1.1 Digital Trust
1.2 Asset
1.3 Transaction
1.4 Distributed Ledger Technology
1.5 Types of network
1.6 Components of blockchain or DLT
1.7 Ledger
  1.7.1 Blocks
  1.7.2 Blockchain

Chapter 2	PKI (Public Key Infrastructure) and Cryptography	6 Lect

2.1 PKI (Public Key Infrastructure) and Cryptography
  2.1.1 Private Key
  2.1.2 Public Key
  2.1.3 Hashing
  2.1.4 Digital Signature
2.2 Security
  2.2.1 DDos
2.3 Cryptocurrency

Chapter 3	How Blockchain Works	4 Lect

3.1 How Blockchain works
3.2 Structure of Blockchain
3.3 Block
3.4 Hash
3.5 Blockchain
3.6 Lifecycle of Blockchain
3.7 Actors of Blockchain

3.8 Blockchain Developer 3.9 Blockchain Operator 3.10 Blockchain Regulator 3.11 Blockchain user 3.12 Building a small blockchain application		
Chapter 4	**Applications of block chain**	**4 Lect**
4.1 Introduction to Bitcoin 4.2 Introduction to Ethereum 4.3 Introduction To Hyperledger Fabric V1.1		
Reference Books:		
• Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Princeton University Press (July 19, 2016). • Antonopoulos, Mastering Bitcoin: Unlocking Digital Cryptocurrencies • Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System • DR. Gavin Wood, “ETHEREUM: A Secure Decentralized Transaction Ledger,”Yellow paper.2014. • Nicola Atzei, Massimo Bartoletti, and TizianaCimoli, A survey of attacks on Ethereum smart contracts		

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC – II Course Code : CS-3511
Course Title: Block Chain Technology Lab Course

Teaching Scheme 2.10 hrs / week Batch Size : 12	No. of Credits 1	Examination Scheme: (Theory + Practical) IE : 15 marks UE: 35 marks

Assignment 1 - Write a blockchain application in JavaScript for the creation of Transaction block for the account holder.

Assignment 2 – Write a blockchain application in JavaScript to calculate hash code for the transaction.

Assignment 3 - Write a JavaScript code for the implementation of block chain technology.(At least two block).

Assignment 4 - Write a blockchain application in JavaScript to transfer cryptocurrency from one account to another account.

Assignment 5 - Write a blockchain application in JavaScript for the simple transaction.

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: DSEC - IV **Course Code: CS -361**
Course Title : Operating Systems-II

Teaching Scheme: 2.30 HRs/ week	No. of Credits: 2	Examination Scheme: IE : 15 marks UE: 35 marks

Prerequisites

Concepts of Operating System, Processes and Threads Scheduling, Synchronization

Course Objectives:

1. **To understand the issue of Deadlocks in Process management.**
2. **To understand the concept of File system management & disk scheduling**
3. **To study the concept of distributed operating systems**

Course Outcomes: After completion of this course students will be able to understand the concept of

1. Management of deadlocks and File System by operating system
2. Scheduling storage or disk for processes
3. Distributed Operating System and its architecture

Course Contents

Chapter 1	**Process Deadlocks**	**7 lectures**

- System model
- Deadlock Characterization – Necessary conditions, Resource allocation graph
- Deadlock Methods- Prevention and Deadlock Avoidance - Safe state, Resource allocation graph algorithm, Banker’s Algorithm
- Deadlock Detection
- Recovery from Deadlock – Process termination, Resource preemption

Chapter 2	**File system Management**	**5 lectures**

- File concept , File attributes, File operations
- Access Methods – Sequential, Direct, Other access methods
- Directory overview, Single level directory, Two level directory, Tree structure directory, Acyclic graph directory, General graph directory
- Allocation Methods – Contiguous allocation, Linked allocation, Indexed allocation
- Free Space Management – Bit vector, Linked list, Grouping, Counting, Space maps

Chapter 3	**Disk scheduling**	**5 lectures**

- Overview, Disk Structure
- Disk Scheduling, FCFS Scheduling, SSTF Scheduling, Scan Scheduling-Scan Scheduling, Look Scheduling , Disk Management

Chapter 4	**Introduction to Distributed operating systems & Architecture**	**12 lectures**

- What is a distributed system, Design goals
- Types of distributed systems
- Architectural styles : Layered architectures , Object-based architectures, Resource-centered architectures, Event-based architectures
- Middleware organization: Wrappers, Interceptors.
- System architecture. Centralized, Decentralized

- Example architectures : Network file system(NFS), Web-based distributed systems, real-time and mobile systems

Chapter 5	**Communication in distributed systems**	**7 lectures**

- Foundations, RPC, Message-Oriented communication, Multicast communication.

Reference Books:

6. Operating System Concepts, Avi Silberschatz, Peter Galvin, Greg Gagne, Student Edition, Wiley Asia
7. Operating Systems: Internals and Design Principles, William Stallings, Prentice Hall of India.
8. Distributed Operating Systems Concepts and Design, Pradeep K. Sinha, PHI
9. Advanced Concepts in Operating Systems, M Singhal and NG Shivaratri, Tata McGraw Hill Inc, 2001 (Text Book)
10. Distributed Operating Systems, Maarten van Steen, A S Tanenbaum. Third edition. Pearson Education Asia, 2001

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: DSEC - IV Course Code: CS -362
Course Title : Software Testing

Teaching Scheme: 2.30HRs/ week	No. of Credits: 2	Examination Scheme: IE : 15 marks UE: 35 marks

Prerequisites:

- Basic knowledge of algorithms, problem solving, expected inputs/outputs
- Knowledge of C and java Programming Language, compilation, debugging.

Course Objectives:

1. To provide the knowledge of software testing techniques
2. To understand how testing methods can be used as an effective tools in quality assurance of software.
3. To provide skills to design test case plan for testing software.
4.To provide knowledge of latest testing methods

Course Outcomes:

1. To understand various software testing methods and strategies.
2. To understand a variety of software metrics, and identify defects and managing those defects for improvement in quality for given software.
3. To design test cases and test plans, review reports of testing for qualitative software.
4. To understand latest testing methods used in the software industries.

Course Contents

Chapter 1	**Introduction to Software Testing**	**Book 1, 5**	**5 lectures**

Basics of Software Testing – faults, errors and failures
Testing objectives
Principles of testing
Testing and debugging
Testing metrics and measurements
Verification and Validation
Testing Life Cycle

Chapter 2	**Software Testing Strategies & Techniques**	**Book 1, 2, 5**	**10 lectures**

Testability - Characteristics lead to testable software
Test characteristics
Test Case Design
White Box Testing - Basis path testing, Control Structure Testing.
Black Box Testing - Boundary Value Analysis, Equivalence partitioning.
Differences between BBT & WBT

Chapter 3	**Levels of Testing**	**Book 1, 5**	**10 lectures**

A Strategic Approach to Software Testing
Test strategies for conventional Software
-Unit testing
- Integration testing – Top-Down, Bottom-up integration
- System Testing – Acceptance, performance, regression, Load/Stress testing, Security testing, Internationalization testing.
Alpha, Beta Testing
Usability and accessibility testing

Configuration, compatibility testing			
Chapter 4	**Testing Web Applications**	**Book 1**	**6 lectures**
Dimension of Quality, Error within a WebApp Environment Testing Strategy for WebApp Test Planning The Testing Process –an overview			
Chapter 5	**Agile Testing**	**Book 4**	**5 lectures**
Agile Testing, Difference between Traditional and Agile testing, Agile principles and values, Agile Testing Quadrants, Automated Tests.			
Reference Books:			
1. Software Engineering – A Practitioners Approach, Roger S. Pressman, 7thEdition, Tata McGraw Hill, 20 2. Effective Methods of Software Testing, William E Perry, 3rd Edition, Wiley Publishing Inc 3. Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing, Rex Black, Microsoft Press, 1999 4. Agile Testing: A Practical Guide for Testers and Agile Teams, Lisa Crispin and Janet Gregory, 1st Edition, Addison-Wesley Professional, 2008 5. Software Testing Principles and Practices By Srinivasan Desikan, Gopalaswamy Ramesh, Pearson			

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science)- Sem - VI
Course Type:DSEC – V Course Code: CS-363
Course Title : Web Technologies II

Teaching Scheme	No. of Credits	Examination Scheme
02.30 **Hrs** / week	2	IE : 15 marks UE: 35 marks

Prerequisites

- HTML5,CSS.
- Core PHP
- Bootstrap framework utility

Course Objectives

- To Learn different technologies used at client Side Scripting Language
- To Learn XML and XML parsers.
- To One PHP framework for effective design of web application.
- To Learn Java Script to program the behavior of web pages.
- To Learn AJAX to make our application more dynamic.
- Framework has some utility features that make easy to write API in more efficient way than Core PHP

Course Outcomes

On completion of the course, student will be able to–

- Build dynamic website.
- Using MVC based framework easy to design and handling the errors in dynamic website.

Course Contents

Chapter 1	**Introduction to** Web Techniques	**6 Lect**

1.1 Variables
1.2 Server information
1.3 Processing forms
1.4 Setting response headers
1.5 Maintaining state
1.6 PHP error handling

Chapter 2	Handling email with php	**3 Lect**

2.1 Email background
2.2 Internet mail protocol
2.3 Structure of an email message
2.4 Sending email with php
2.5 Email id validation and verification

Chapter 3	XML	**6 Lect**

3.1What is XML?
3.2 XML document Structure
3.3 PHP and XML
3.4 XML parser
3.5 The document object model
3.6 The simple XML extension
3.7 Changing a value with simple XML

Chapter 4	WEB DESIGNING TECHNOLOGIES(JavaScript)	**9 Lect**

4.1 Overview of JavaScript
4.2 Object Orientation and JavaScript Basic Syntax(JS datatypes, JS variables)
4.4 Primitives, Operations and Expressions
4.5 Screen Output and keyboard input(Verification and Validation)

4.6 JS Control statements and JS Functions 4.7 JavaScript HTML DOM Events(onmouseup, onmousedown, onclick, onload, onmouseover, onmouseout). 4.8 JS Strings and JS String methods 4.9 JS popup boxes(alert, confirm, prompt). 4.10 Jquery library , Including jquery library in page 4.11 Jquery selector , DOM manipulation using jquery		
Chapter 5	AJAX	**5 Lect**
5.1 Introduction of AJAX 5.2 AJAX web application model 5.3 AJAX –PHP framework 5.4 Performing AJAX validation 5.5 Handling XML data using php and AJAX 5.6 Connecting database using php and AJAX		
Chapter 6	PHP framework CodeIgniter	**7 Lect**
6.1 CodeIgniter - Overview, Installing CodeIgnite 6.2 Application Architecture 6.3 MVC Framework , Basic concept of CodeIgniter, Libraries 6.4 Working with databases 6.5 Load external JS and CSS page & redirecting from controller , Adding JS and CSS , Page redirection. 6.6 Loading dynamic data on page & session management, cookies management		
Reference Books:		
1. Programming PHP By RasmusLerdorf and Kevin Tatroe O'Reilly publication 2. Beginning PHP 5, Wrox publication 3. PHP web services , Wrox publication 4. AJAX Black Book Kogent solution 5. Mastering PHP BPB Publication 6. PHP cookbook O'Reilly publication 7. Professional Codeigniter , Wrox Publication, Author: Thomas Myer 8.Codeihniter 2 CookBook, PACKT Publication , Author : Rob Foster 9. JQuery CookBook, O'reilly Publication. 10. PHP for Beginners, SPD publication 11. Programming the World Wide Web , Robert W Sebesta(3rd Edition) 12. www.php.net.in 13. www.W3schools.com 14. www.wrox.com 15. https://www.tutorialspoint.com/codeigniter/index.htm 16. http://codeigniter.com/docs		

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) Sem - VI
Course Type:DSEC – V Course Code: CS-364
Course Title : Data Analytics

Teaching Scheme	No. of Credits	Examination Scheme
02 **Hrs** / week	2	IE : 15 marks UE: 35 marks

Prerequisites

- Basic of mathematics and statistics
- Basic programming Knowledge of python
- Knowledge of databases

Course Objectives

- Deploy the Data Analytics Lifecycle to address data analytics projects.
- Develop in depth understanding of the key technologies in data analytics.
- Apply appropriate analytic techniques and tools to analyze data, create models, and identify insights that can lead to actionable results.

Course Outcomes

On completion of the course, student will be able to–

- Use appropriate models of analysis, assess the quality of input, and derive insight from results.
- Demonstrate knowledge of statistical data analysis techniques utilized in business decision making.
- Apply modeling and data analysis techniques to the solution of real world business problems
- Analyze data, choose relevant models and algorithms for respective applications.
- Compare and evaluate different data mining techniques like classification, prediction, clustering and association rule mining

Course Contents

Chapter 1	**Introduction to Data Analytics**	**6 Lect**

1.1 What is Analytics?
1.2 Data analysis vs Data analytics
1.3 Applications in Retail, E-commerce, Finance, Sports, Others - healthcare, education, telecom etc.
1.4 Diagnostic Analytics - Correlations
1.5 Predictive Analytics
1.6 Prescriptive Analytics
1.7 Exploratory Analysis
1.8 Mechanistic Analysis - Regression

Chapter 2	**Mathematical Models**	**6 Lect**

2.1. Philosophies of Modeling
Occam's Razor
Bias-Variance Trade-Offs
2.2. Types of models – linear and non linear, flat and hierarchical
2.3.Evaluating Models
Evaluating Classifiers, Class imbalance
AUC, ROC curves
Evaluating Multiclass Systems

Evaluating Value Prediction Models		
Chapter 3	**Mining Frequent Patterns, Associations, and Correlations**	**8 Lect**
3.1 What kind of patterns can be mined: Class/Concept Description: Characterization and Discrimination, Mining Frequent Patterns, Associations, and Correlations, Classification and Regression for Predictive Analysis, Cluster Analysis, Outlier Analysis 3.2 Mining frequent patterns - Market Basket Analysis. 3.3 Frequent Itemsets, Closed Itemsets, and Association Rules 3.4 Frequent Itemset Mining Methods 3.5 Apriori Algorithm 3.6 Generating Association Rules from Frequent Itemsets		
Chapter 4	**Text Analytics**	**8 Lect**
4.1. Tokenization 4.2.Bag of words 4.3.Stemming and lemmatization 4.4.TF-IDF, stop words and n-grams, synonyms and parts of speech tagging 4.5.Sentiment Analysis 4.6.Introduction to NLP		
Chapter 5	**Machine Learning Overview**	**8 Lect**
5.1.Introduction to Machine Learning, deep learning, Artificial intelligence 5.2.Applications for machine learning in data science 5.3.The modeling process: Engineering features and selecting a model, Training your model, Validating a model, Predicting new observations 5.4.Types of machine learning - Supervised learning, Unsupervised learning, Semi-supervised learning, ensemble techniques 5.5.Regression models - Linear regression 5.6.Introduction to classification and clustering.		
Reference Books:		
1) Data Mining Concepts and Techniques, Third Edition, Jiawei Han, Micheline Kamber, Jian Pei, Morgan Kaufmann, 2012. 2) Introduction to Data mining, Pang-NING TAN, Michael SteinBach, Vipin Kumar, Pearson 3) The Data Science Design Manual, Steven S. Skiena, Springer, 2017 4) Introducing data science: big data, machine learning, and more, using Python tools, Cielen D., Meysman A. D., & Ali M., Manning Publications Co., 2016 5) Python Data Science Essentials, Alberto Boschetti, Luca Massaron, Second Edition, 2016 Packt Publishing 6) The Data Science Handbook, Field Cady, John Wiley & Sons, Inc, 2017 7) Python Data Science Handbook - Essential Tools for Working with Data, Jake VanderPlas, O'Reilly, 2017 8) A Hands-On Introduction to Data Science CHIRAG SHAH University of Washington Cambridge University Press 9) David Dietrich, Barry Hiller, "Data Science & Big Data Analytics", EMC education services, Wiley publications, 2012		

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: DSEC – VI Course Code : CS-365
Paper Title : Object Oriented Programming – II (Advanced Java)

Teaching Scheme	No. of Credits	Examination Scheme
02: 30 Hrs / week	2	IE : 15 marks UE: 35 marks

Prerequisites

- Knowledge of Core Java (CS – 355)

Course Objectives

- To learn database programming using Java
- To study web development concept using Servlet and JSP
- To develop a game application using multithreading
- To learn socket programming concept

Course Outcomes

On completion of the course, student will be able to–

- To access open database through Java programs using Java Data Base Connectivity (JDBC) and develop the application.
- To understand and Create dynamic web pages, using Servlets and JSP.
- Work with basics of framework to develop secure web applications.

Course Contents

Chapter 1	**Collections**	**4 Lect**

1.1 Introduction to the Collection framework
1.2 List – ArrayList, LinkedList and Vector,Stack,Queue
1.3 Set - HashSet, TreeSet, and LinkedHashSet
1.4 Map – HashMap, LinkedHashMap, Hashtable and TreeMap
1.5 Interfaces such as Comparator, Iterator, ListIterator, Enumeration

Chapter 2	**Database Programming**	**8 Lect**

2.1 The 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 (Database : PostgreSQL)

Chapter 3	**Servlet**	**8 Lect**

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

Chapter 4	**JSP**	**6 Lect**

4.1 Simple first JSP program
4.2 Life cycle of JSP
4.3 Implicit Objects
4.4 Scripting elements - Declarations, Expressions, Scriplets, Comments

4.5 JSP Directives - Page Directive, include directive 4.6 Mixing Scriplets and HTML 4.7 JSP Actions - jsp:forward , jsp:include, jsp:useBean, jsp:setProperty and jsp:getProperty 4.8 Custom Tags 4.9 Example of forwarding contents from database to servlet , servlet to JSP and displaying it using JSP scriplet tag		
Chapter 5	**Multithreading**	**4 Lect**
5.1 What are threads? 5.2 Life cycle of thread 5.3 Running and starting thread using Thread class 5.4 Thread priorities 5.5 Running multiple threads 5.6 The Runnable interface 5.7 Synchronization and interthread communication		
Chapter 6	**Networking**	**3 Lect**
6.1 Networking basics – protocol, Addressing, DNS, URL, Socket, Port 6.2 The java.net package – InetAddress, URL, URLConnection class 6.3 SocketServer and Socket class 6.4 Creating a Socket to a remote host on a port (creating TCP client and server) 6.5 Simple Socket Program Example		
Chapter 7	**Spring**	**3 Lect**
7.1 Introduction of Spring framework 7.2 Spring Modules / Architecture 7.3 Spring Applications 7.4 Spring MVC 7.5 Spring MVC Forms, Validation		
Reference Books:		
R1. Complete reference Java by Herbert Schildt(5th edition) R2. Java 2 programming black books, Steven Horlzner R3. Programming with Java , A primer ,Forth edition , By E. Balagurusamy R4. Core Java Volume-I-Fundamentals, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press R5. Core Java Volume-II-Advanced Features, Eighth Edition, Cay S. Horstmann, Gary Cornell, Prentice Hall, Sun Microsystems Press R6. Getting started with Spring Framework: covers Spring 5 by J Sharma and Ashish Sarin R7. Spring 4 for Developing Enterprise Applications: An End-to-End Approach by Henry H. Liu		

Savitribai Phule Pune University

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

Course Type: DSEC- VI **Course Code: CS-366**

Paper Title: Theoretical Computer Science and Compiler Construction II

Teaching Scheme	No. of Credits	Examination Scheme
02 **Hrs** / week	2	IE: 15 marks UE: 35 marks

Prerequisites

- Knowledge of Automata Theory and Languages.

Course Objectives

- 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 and design code generation and optimization techniques.

Course Outcomes

On completion of the course, student will be able to–

- Understand the process of scanning and parsing of source code.
- Learn the conversion code written in source language to machine language.
- Understand tools like LEX and YACC.

Course Contents

Chapter 1	**Introduction**	**3 Lect**

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.

Chapter 2	**Lexical Analysis (Scanner)**	**3 Lect**

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)

Chapter 3	**Syntax Analysis (Parser)**	**19 Lect**

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, 3.2.3Elimination of Left Recursion (direct & indirect) 3.2.4Need for Left Factoring & examples
3.3 Recursive Descent Parsing: Definition
3.3.1Implementation of Recursive Descent Parser Using Recursive Procedures
3.4 Predictive [LL (1)] Parser (Definition, Model) 3.4.1Implementation of Predictive Parser [LL (1)]
3.4.2 FIRST & FOLLOW
3.4.3 Construction of LL (1) Parsing Table 3.4.4Parsing of a String using LL (1) Table.
3.5 Bottom-Up Parsers
3.6 Operator Precedence Parser -Basic Concepts
3.6.1Operator Precedence Relations form Associativity & Precedence

3.6.2 Operator Precedence Grammar 3.6.3 Algorithm for LEADING & TRAILING (with ex.) 3.6.4Algorithm for Operator Precedence Parsing (with ex.) 3.6.5Precedence 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: Model, Types [SLR (1), Canonical LR, LALR] Method & examples. 3.9 YACC (from Book 3) –program sections, simple YACC program for expression evaluation		
Chapter 4	**Syntax Directed Definition**	**5 Lect**
4.1Syntax Directed Definitions (SDD) 4.1.1 Inherited & Synthesized Attributes 4.1.2 Evaluating an SDD at the nodes of a Parse Tree, Example 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		
Chapter 5	**Code Generation and Optimization**	**6 Lect**
5.1 Compilation of expression – 5.1.1 Concepts of operand descriptors and register descriptors with example. 5.1.2 Intermediate code for expressions – postfix notations, 5.1.3 Triples, Quadruples and Expression trees. 5.2 Code Optimization – Optimizing transformations – compile time evaluation, elimination of common sub expressions, dead code elimination, frequency reduction, strength reduction. 5.3 Three address code 5.3.1 DAG for Three address code 5.3.2 The Value-number method for constructing DAG's. 5.4 Definition of basic block, Basic blocks, and flow graphs 5.5 Directed acyclic graph (DAG) representation of basic block. 5.6 Issues in design of code generator.		
Reference Books		
R1. Compilers: Principles, Techniques, and Tools, Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, 2004 R2. Principles of Compiler Design By: Alfred V. Aho, Jeffrey D. Ullman, Narosa Publication House, 2002 R3. LEX & YACC, 2nd edition, O'reilly Publication, 2012		

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: DSEC- IV **Course Code: CS 367**
Course Title : Practical Course based on CS 361

Teaching Scheme:	No. of Credits:	Examination Scheme:
4.20 Hrs/ week	2	IE : 15 marks UE: 35 marks

Course Objectives:

4. **To implement Banker's algorithm for Deadlocks in Process management.**
5. **To simulate File system management**
6. **To study and implement various algorithms of disk scheduling**

Course Outcomes: After completion of this course students will be able to understand the concept of

1. Management of deadlocks by operating system
2. File System management
2. Disk space management and scheduling for processes

Guidelines:

9. Operating system platform – Linux
10. Programming language - C/C++/Java

List of Assignments:

- Simulation of Banker's algorithm of deadlock avoidance in processes of operating system (2 slots)
- Simulation of File Allocation methods and free space management in storage - Contiguous allocation, Linked allocation, Indexed allocation (5 slots)
- Simulation of Disk Scheduling algorithms – FCFS, SSTF, Scan, Look (2 slots)
- Implementation of RPC , Remote-method invocation (3 slots)
- Implementation of a Concurrent client server application (***)

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: DSEC - V Course Code: CS 368
Course Title : Practical Course based on CS 363 an CS 364

Teaching Scheme:	No. of Credits:	Examination Scheme:
4.20HRs/ week	2	IE : 15 marks UE: 35 marks

Course Objectives:

- To Learn different technologies used at client Side Scripting Language
- To Learn XML and XML parsers.
- To One PHP framework for effective design of web application.
- To Learn Java Script to program the behavior of web pages.
- To Learn AJAX to make our application more dynamic.
  Framework has some utility features that make easy to write API in more efficient way than Core PHP

Course Outcomes:

- Build dynamic website.
- Using MVC based framework easy to design and handling the errors in dynamic website.

Guidelines:

1. **Operating Environment :HTML, PHP5.0 and above, Codeigniter**
2. **Submission**
3. **Assessment**

List of Assignments based on Web Technology CS-363:

Session 1 : Assignments based on Self Processing Forms, Sticky Forms, File Upload.
Session 2 : Assignments based on COOKIES.
Session 3 : Assignment using SESSIONS.
Session 4 : Assignment using Assignments based on E-Mailing.
Session 5 : Assignments for creating xml documents, assignments based on SimpleXML.
Session 6 : Assignments based on DOMs.

Session 7 : Assignment using WEB DESIGNING TECHNOLOGIES(JavaScript)

* Assignment on types of dialogue boxes (alert,prompt,confirm)
* Assignment on event handling(onmouseup, onmousedown, onclick, onload, onmouseover, onmouseout).
* Assignment on Javascript function, display the string in different formatting styles(Bold, italic, underline, strikethrough, hypertext etc)

Session 8 : Assignment using WEB DESIGNING TECHNOLOGIES(JavaScript)
* Assignment on validate user input using javascript
* Assignment on Tag Selector using Jquery .
* Assignment on Including Jquery library in page
* Assignment on content management to manipulate DOM .

Session 9 : Assignment using Ajax
* Write Ajax program to read a textfile and print the contents of the file when the user clicks on the Print button.

* Write Ajax program to carry out validation for a username entered in textbox. If the textbox is blank, print ‘Enter username’. If the number of characters is less than three,print’ Username is too short’. If value entered is appropriate the print ‘Valid username’.

Session 10 : Assignment using Ajax

* Create employee table as follows EMP (eno, ename, designation, salary). Write Ajax program to select the employees name and print the selected employee’s details.

*Write Ajax program to print Movie details by selecting an Actor’s name. Create table MOVIE and ACTOR as follows with 1 : M cardinality MOVIE (mno, mname, release_yr) and ACTOR(ano, aname).

Session 11 : Assignment using PHP framework CodeIgniter

* installation of codeIgniter and get familiar with codeIgniter architecture
* Application configuration setting :
* Configure application properties
* Learn to auto load library, helpers
* build static pages in codeIgniter

Try associate view from controller Load Js and css in page

Notes: each method in controller class acts as endpoint from that method we can server view file you want to display.

Session 12 : Assignment using PHP framework CodeIgniter

* **database connection and display** Dynamic data on page

* **full curd(create,update,edit,delete) operation of any entity**

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: DSEC - V **Course Code: CS 368**
Course Title : Practical Course based on CS 363 an CS 364

Teaching Scheme	No. of Credits	Examination Scheme
4 hrs 20 mins / week Batch Size : 12	2	IE : 15 marks UE: 35 marks

Operating Environment:

- **Operating system:** Linux
- **Editor:**Any linux based editor like vi, gedit etc.
- **Compiler** : cc or gcc
- **Languages**: Python

Lab Book:

The lab book is to be used as a hands-on resource, reference and record of assignment submission and completion by the student. The lab book contains the set of assignments which the student must complete as a part of this course.

Programming Assignments:

Programs should be done individually by the student in their respective login. The codes should be uploaded on either the local server, Moodle, Github or any open source LMS. Print-outs of the programs and output may be taken but not mandatory for assessment.

Assessment:

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student. Each lab assignment assessment will be assigned grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include-timely completion, performance, innovation, efficient codes and good programming practices.

Course Contents:

Suggested Assignments based on Data Analytics CS- 364

Assignment 1: Frequent itemset and association rule mining

Load Transactional data set. Do the needful data preprocessing. Display the set of frequent 2-itemsets and 3-itemsets. Repeat the process for different min_sup value.

Assignment 2: Linear regression

For Given dataset predict the value of specific attribute.

Assignment 3: Text Analytics

Take text file as input. Create bag of words. Find frequent item sets. Display word cloud.

Assignment 4: Google Analytics

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: DSEC- VI **Course Code: CS369**
Course Title : Practical Course based on CS 365

Teaching Scheme	No. of Credits	Examination Scheme
4 hrs 20 mins / week	2	IE : 15 marks
Batch Size : 12		UE: 35 marks

Course Objectives:

4. Covers the complete scope of the syllabus.
5. Bringing uniformity in the way course is conducted across different colleges.
6. Continuous assessment of the students.
7. Advanced Java is designed to develop web based, network centric, Enterprise level applications

Course Outcomes:

4. To Learn database Programming using Java
2. Understand and Create dynamic web pages using Servlets and JSP.
3. Work with basics of framework to develop secure web applications

Guidelines:

Operating Environment :

- **Operating system:** Linux
- **Editor :** Any linux based editor like vi, gedit and Use of IDE – Eclipse etc.
- **Compiler** : javac
- **Database** : postgresql

Submission :

Each assignment will be assessed on a scale of 0 to 5 as indicated below.

Not done	0
Incomplete	1
Late Complete	2
Needs improvement	3
Complete	4
Well Done	5

Assessment:

Easy : All exercises are compulsory.
Medium : All exercises are compulsory.

List of Assignments :

Assignment 1 : Collections [Slot-2]

Study the Collection framework in java
To Implement various Interfaces and classes through algorithms
To Demonstrate Cursor Objects (Enumeration, Iterator, ListIterator, Comparator)

Assignment 2 : Database Programming [Slot-2]

To communicate with a database using java.
To execute queries on tables.
To obtain information about the database and tables
To understand various ways to manage Transaction (Commit, Rollback, Save-points)

Assignment 3 : Servlets [Slot-2]

To understand server-side programming.
Simple steps to create and execute servlets
How to pass parameters using doGet and doPost methods
Handling data from HTML to servlet
How to connect servlet to a database .
Use of various session tracking methods like Cookies.

Assignment 4 : Java Server Pages [Slot-2]

Concept of Servlets.
JSP life-cycle.
Use of JSP implicit objects
JSP Directives.
Use of Scripting Elements.
To understand Actions tags in JSP
Understanding flow of JSP custom tags

Assignment 5 : Multithreading [Slot-2]

To create and use threads in java.
To demonstrate multithreading using Thread Synchronization, Inter-thread Communication, Thread Priorities

Assignment 6 : Networking [Slot -1]

To understand Networking Terminology
Introduction to the java.net package -InetAddress class, URL class, URL Connection class etc.
Introduction to Socket Programming -To understand important methods of Socket and Server Socket class.
Client and Server side Programming

Assignment 7 : Spring Framework [Slot-1]

To create and understand the steps to develop Spring application

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC – III Course Code : CS-3610
Course Title : Mobile Application Development (Theory)

Teaching Scheme	No. of Credits	Examination Scheme
01:15 Hrs / week	1	IE : 15 marks UE: 35 marks

Prerequisites: Fundamental of Networking, Object Oriented Concepts-JAVA Programming.

Course Objectives

- Understand system requirements for mobile applications.
- Generate suitable design using specific mobile development frameworks.
- Generate mobile application design.
- Implement the design using specific mobile development frameworks.
- Deploy the mobile applications in marketplace for distribution.

Course Outcomes

Completion of the course, the students will be able to

- Describe the requirements for mobile applications.
- Explain the challenges in mobile application design and development.
- Develop design for mobile applications for specific requirements.
- Implement the design using Android.

Course Contents

Chapter 1	INTRODUCTION MOBILE APPLICATIONS	3 Lect

1.17. Introduction to mobile Application.
1.18. Market and business drivers for mobile applications.
1.19. Publishing and delivery of mobile applications.
1.20. Requirements gathering and validation for mobile applications.

Chapter 2	BASIC DESIGN	4 Lect

2.1 Design constraints for mobile applications, both hardware and software related.
2.2 Architecting mobile applications.
2.3 User interfaces for mobile applications.
2.4 Touch events and gestures.
2.5 Achieving quality constraints performance, usability, security, availability and modifiability.

Chapter 3	TECHNOLOGY I - ANDROID	9 Lect

3.1 Introduction to Android OS.
3.2 Android architecture.
3.3 Activities and views
3.4 Interacting with UI
3.5 Persisting data using SQLite.
3.6 Google Maps.
3.7 GPS and Wifi.

Chapter 4	TECHNOLOGY II - IOS	2 Lect

4.1 Introduction and features of iOS.
4.2 UI implementation.

Reference Books:

1. http://developer.android.com/develop/index.html
2. Jeff McWherter and Scott Gowell, "Professional Mobile Application Development", Wrox, 2012
3. Charlie Collins, Michael Galpin and Matthias Kappler, “Android in Practice”, DreamTech, 2012
4. James Dovey and Ash Furrow, “Beginning Objective C”, Apress, 2012
5. David Mark, Jack Nutting, Jeff LaMarche and Frederic Olsson, “Beginning iOS 6 Development: Exploring the iOS SDK”, Apress, 2013.
6. Beginning Android Application Development Wei-Meng Lee Wiley

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC – III Course Code : CS-3610
Course Title : Mobile Application Development (Practical)

Teaching Scheme	No. of Credits	Examination Scheme
02:10 Hrs / week	1	IE : 15 marks UE: 35 marks

Prerequisites: Fundamental of Networking.

Course Objectives

- Generate suitable design using specific mobile development frameworks
- Generate mobile application design
- Implement the design using specific mobile development frameworks

Course Outcomes

Completion of the course, Upon the students will be able to

- Describe the requirements for mobile applications.
- Explain the challenges in mobile application design and development.
- Develop design for mobile applications for specific requirements.
- Implement the design using Android.

Guidelines:
Operating environment: Linux

List of Assignments

1. To study Android Studio and Android studio installation.
2. Creating a new project and using emulator.
3. Create a simple "Hello World" application.
4. Create various UI controls like button, textview, edittext checkbox etc.
5. To understand Activity, Intent, Create sample application with login module.(Check username and password)
6. Design simple GUI application with activity and intents e.g. calculator.
7. Create an android app for database creation using SQLite Database.
8. "Guess Number": The app should pick a secret number (0 – 9) and let the user guess what number it is. User is only allowed to input number in the text field. If the guess number is too high or too low, the program should provide a hint. If the guess number is correct, the program should congratulate the user. (Hint: using randomize to generate the random number)

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC III **Course Code: CS 3610**
Course Title: Software Testing Tools

Teaching Scheme:	No. of Credits:	Examination Scheme:
1.15HRs/ week	1 (Theory)	(Theory + Practical) IE:15 marks UE: 35 marks

Prerequisites

- Basic knowledge of algorithms, problem solving, expected inputs/outputs
- Knowledge of C and Java Programming Language, compilation, debugging

Course Objectives:

1. To provide the knowledge of software testing methods and strategies.
2. To understand how testing methods can be used as an effective tool in quality assurance of software.
3. To provide skills to design test case plan for testing software.
4.To provide knowledge of latest testing tools

Course Outcomes:

1. To understand various software testing methods and strategies.
2. To understand a variety of software metrics and identify defects and managing those defects for improvement in quality for given software.
3. To design test cases and test plans, review reports of testing for qualitative software.
4. To understand latest testing tools used in the software industries.

Course Contents

Chapter 1	**Introduction to Test case design**	**Book 3**	**4 lectures**

1.1 How to identify errors, bugs in the given application.
1.2 Design entry and exit criteria for test case, design test cases in excel.
1.3 Describe feature of a testing method used.

Chapter 2	**Test cases for simple programs**	**Book 3**	**4 lectures**

2.1 Write simple programs make use of loops and control structures.
2.2 Write Test Cases for above programs.

Chapter 3	**Test cases and Test plan**	**Book 3**	**4 lectures**

3.1 Write Test Plan for given application with resources required.
3.2 Write Test case for given application.
3.3 Prepare Test report for test cases executed.

Chapter 4	**Defect Report**	**Book 3**	**3 lectures**

4.1 Defect Life Cycle
4.2 Classification of Defect
4.3 Write Defect Report

Chapter 5	**Testing Tools**	**Book 3**	**3 lectures**

5.1 How to make use of Automation Tools
5.2 Types of Testing Tools

Reference Books:

1. Software Engineering – A Practitioners Approach, Roger S. Pressman, 7thEdition, Tata McGraw Hill, 20
2. Effective Methods of Software Testing, William E Perry, 3rd Edition, Wiley Publishing Inc
3. Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing, Rex Black, Microsoft Press, 1999

4. Software Testing Principles and Practices by Srinivasan Desikan, Gopalaswamy Ramesh, Pearson.

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC III **Course Code: CS 3610**
Course Title: Software Testing Tools

Teaching Scheme:	No. of Credits:	Examination Scheme:
2.10HRs/ week	1 (Practical)	(Theory + Practical) IE: 15 marks UE: 35 marks

Prerequisites

- Basic knowledge of algorithms, problem solving, expected inputs/outputs
- Knowledge of C and Java Programming Language, compilation, debugging

Course Objectives:
1. To provide the knowledge of software testing methods and strategies.
2. To understand how testing methods can be used as an effective tool in quality assurance of software.
3. To provide skills to design test case plan for testing software.
4.To provide knowledge of latest testing tools

Course Outcomes:
1. To understand various software testing methods and strategies.
2. To understand a variety of software metrics and identify defects and managing those defects for improvement in quality for given software.
3. To design test cases and test plans, review reports of testing for qualitative software.
4. To understand latest testing tools used in the software industries.

Course Contents

Chapter 1	**Introduction to Test case design**	**Web Ref:2,3**	**4 Sessions**

1. Design test cases for following applications
   A. Simple Calculator Application
   B. Online Air Ticket Booking / Railway Reservation Form
   C. E-Commerce shopping portal's Login form (like Flipkart, Amazon)
   D. Web pages of any website / College / University website

Chapter 2	**Test cases for simple programs**	**Web Ref: 3,4**	**2 Sessions**

2. Write programs and Design test cases for following programming statements.
   A. For, While, Do... While Loops
   B. If …Else , Switch …Case

Chapter 3	**Test cases and Test plan**	**Web Ref:3**	**2 Sessions**

3. Define Test cases and Test Plan for simple applications like
   A. Mobile app like calculator
   B. Notepad desktop app

Chapter 4	**Defect Report**	**Web Ref:4,5**	**2 Sessions**

4. Prepare a defect report after executing Test cases for
   A. Withdraw Amount from ATM machine
   B. Login form

Chapter 5	**Testing Tools**	**Web Ref: 1**	**2 Sessions**

5. Design and run Test cases using automated testing Tools for
   A. Text Editor like word / wordpad

Note: Preparation of system specification, test plan, test cases, defect report, execution using Automation Tool, answers to the simple questions and timely submission of assignments carries the equal marks.

Open Source Automation TestingTools: Selenium, JMeter, QTP, Bugzilla etc can be used.

Reference Books:

1. Software Engineering – A Practitioners Approach, Roger S. Pressman, 7thEdition, Tata McGraw Hill, 20
2. Effective Methods of Software Testing, William E Perry, 3rd Edition, Wiley Publishing Inc
3. Managing the Testing Process: Practical Tools and Techniques for Managing Hardware and Software Testing, Rex Black, Microsoft Press, 1999
4. Software Testing Principles and Practices by Srinivasan Desikan, Gopalaswamy Ramesh, Pearson

Website References:

1. http://www.selenium.dev/
2. http://www.toolsqa.com
3. https://www.guru99.com/selenium-tutorial.html
4. https://www.tutorialspoint.com/selenium
5. https://www.softwaretestinghelp.com/

Savitribai Phule Pune University
T.Y.B.Sc. (Computer Science) - Sem – VI
Course Type: SECC- IV **Course Code: CS3611**
Course Title : Project

Teaching Scheme 4 hrs 20 mins / week Batch Size : 12	No. of Credits 2	Examination Scheme IE : 15 marks UE: 35 marks

Project

Total Credits: 1

Teaching Scheme:
2.10 Hrs./Week

Guidelines:

- Students should work in a team of maximum 2 students.
- Students can choose a project topic without any restriction on technology or domain.
- The student group will work independently throughout the project work including: problem identification, information searching, literature study, design and analysis, implementation, testing, and the final reporting.
- Project guide must conduct project presentations (minimum 4) to monitor the progress of the project groups.
- At the end of the project, the group should prepare a report which should conform to international academic standards. The report should follow the style in academic journals and books, with clear elements such as: abstract, background, aim, design and implementation, testing, conclusion and full references, Tables and figures should be numbered and referenced to in the report.
- The final project presentation with demonstration (UE) will be evaluated by the project guide (appointed by the college) and one external examiner (appointed by the University).

Recommended Documentation contents:

Abstract

Introduction

- motivation
- problem statement
- purpose/objective and goals
- literature survey
- project scope and limitations

System analysis

- Existing systems
- scope and limitations of existing systems
- project perspective, features
- stakeholders

- Requirement analysis - Functional requirements, performance requirements, security requirements etc.

System Design

- Design constraints
- System Model: DFD
- Data Model
- User interfaces

Implementation details

- Software/hardware specifications

Outputs and Reports Testing

- Test Plan, Black Box Testing or Data Validation Test Cases, White Box Testing or Functional Validation Test cases and results

Conclusion and Recommendations

Future Scope

Bibliography and References

Project Related Assignments

Total Credits: 1

Teaching Scheme:

- 1.15 Hrs./week

Guidelines:

- The project assignments are a compulsory part of the project course and should be carried out by each project group.
- Project assignments are to be given by the guide for continuous internal evaluation.
- The project assignments are to be allotted to each group separately by the project guide on the basis of the implementation technology. A suggested list of assignments is given below.
  1. Project Time management: plan (schedule table), Gantt chart, Roles and responsibilities, data collection, Implementation
  2. Simple assignments to evaluate choice of technology
  3. Assignments on UI elements in chosen technology
  4. Assignments on User interfaces in the project
  5. Assignments on event handling in chosen technology
  6. Assignments on Data handling in chosen technology

7. Online and offline connectivity
8. Report generation
9. Deployment considerations
10. Test cases

- Each student within the group must work actively and contribute to the assignments, project work and report writing.

Evaluation guidelines:

IA (15 marks)			**UE (35 marks)**		
First presentation	Second presentation	Assignments	Project Logic/ Presentation	Assignments and Project Documentation	Viva
05	05	05	15	15	05