



Progressive Education Society's
Modern College of Arts, Science & Commerce, (Autonomous) Ganeshkhind,
Pune 411016



Program Code: BSC CS09

Bachelor of Science (Computer Science)
(Under Faculty of Science & Technology)

A.Y: 2025 – 2026

Opposite to University of Pune, Pashan Road 411016

e-Mail: moderncollege16@gmail.com

Phone: +917768050797 or +917768020797 Fax:(020) 25650931

Name of Program: Bachelor of Science (Computer Science)

Program Structure:

- The Program is a Four Year (Eight semesters) Full Time Degree Program.
- The Program shall be based on a credit system comprising 176 credits.

Medium of Instruction: English

Instructions for Teachers for Internal Evaluation for 20 Marks and 40 Marks:

External Examination:

Award of Class:

Letter Grade	Grade Point
O (outstanding)	10
A+ (Excellent)	9
A (Very good)	8
B+ (Good)	7
B (Above average)	6
C (Average)	5
P (Pass)	4
F (Fail)	0
Ab (Absent)	0

Question Paper Pattern:CIE

Max. Marks: 20 (Credit:02, Duration: 50 Min.)					
Question No.	Question		No. of sub questions	Marks to each sub question	Total Question Marks
1	Multiple Choice Questions 5		5	1	5
2	Define any 5		6	1	5
3	Attempt any two of the following		3	2	4
4	Attempt any two of the following		3	3	6
Total Marks:					20

Max. Marks: 20 (Credit 4, Duration: 50Min.)					
Question No.	Question		No. of sub questions	Marks to each sub question	Total Question Marks
1	Multiple Choice Questions 5		5	1	5
2	Define any 5		6	1	5
3	Attempt any two of the following		3	2	4
4	Attempt any two of the following		3	3	6
Total Marks:					20

For 2 credits CIE 20 Marks for internal Examination and 20 Marks for CBCS activity (Open book test, Seminars, Online Test, Surprise Test, Preparation of Models, Group Discussions etc.) average of 40 marks will be considered.

Question Paper Pattern: ESE

Max. Marks: 60 (Credit:04, Duration: 3 Hrs.)					
Question No.	Question		No. of sub questions	Marks to each sub question	Total Question Marks
1	Attempt All		10	1	10
2	Attempt any 5		7	2	10
3	Attempt any 5		7	3	15
4	Attempt any 3		5	5	15
5	Attempt any 2		4	5	10
Total Marks:					60

Max. Marks: 30 (Credit 2, Duration: 2 Hrs.)					
Question No.	Question		No. of sub questions	Marks to each sub question	Total Question Marks
1	Attempt All		5	1	5
2	Attempt any 5		7	2	10
3	Attempt any 2		4	5	10
4	Attempt any 1		2	5	5
Total Marks:					30

	T.Y.BSc (Computer Science) Semester -V						
	Course Code	Course Title	Credits		Evaluation		
			T H	P R	CIE	ESE	Total
Major Mandatory (6T) =(2T+2T+2 T) 4P=2P+2P	COM35101	Object oriented Software engineering	2	-	20	30	50
	COM35102	Theoretical Computer Science	2	-	20	30	50
	COM35103	Python Programming	2	-	20	30	50
	COM35104	Practical on Python Programming	-	2	20	30	50
	COM35105	Practical on Advanced Java	-	2	20	30	50
	COM35106	Data Science	2	-	20	30	50
	COM35107	Advanced Java	2	-	20	30	50
Minor (4) 2T+2P	CELE 35201	Wireless Communication & Industrial IOT/ Mathematics	2	-	20	30	50
	CELE 35202	Practicals on Wireless Communication & IOT Applications/ Mathematics	-	2	20	30	50
Major Elective	COM35401	Data Science Practical using R	-	2	20	30	50
FP/CEP	COM35601	Field Project	-	2	20	30	50
		Total	12	10	220	330	550

T.Y.BSc (Computer Science) Semester -VI							
Course Type	Course Code	Course Title	Credits		Evaluation		
			TH	PR	CIE	ESE	Total
Major Mandatory (6T) =(2T+2T+2T) 4P=2P+2P	COM36101	Software Testing	2	-	20	30	50
	COM36102	Compiler Construction	2	-	20	30	50
	COM36103	Information and Computer Network Security	2	-	20	30	50
	COM36104	Practical on Operating System	-	2	20	30	50
	COM36105	Practical on BlockChain Technology	-	2	20	30	50
Minor (4) 2T+2P	CELE 36201	Embedded Systems Design/ Mathematics	2	-	20	30	50
	CELE 36202	Practicals on Embedded System Applications/ Mathematics	-	2	20	30	50
Major DSE	COM36106	BlockChain Technology	2	-	20	30	50
	COM36107	Operating System	2	-	20	30	50
OJT	COM36601	On Job Training	-	4	40	60	100
		Total	12	10	220	330	550



*Fifth
Semester*

Course Title: - Object Oriented Software Engineering
Course Type: Major Mandatory Paper1(Theory)
Course Code: COM35101
Semester- V

Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
---	-----------------------------	-------------------------------	--

Prerequisites: Student should have basic knowledge of:

- Understanding of the Software Development Lifecycle
- Proficiency in UML (Unified Modeling Language)

The main objectives of this course are to:

- To understand software engineering discipline.
- To learn analysis and design principles for software project development.

Expected Course Outcomes:

- Understand Object-Oriented Concepts and System Development.
- Learn iterative methods like Booch, Jacobson, and Coad & Yourdon.
- Focus on defining classes, attributes, operations, and system architecture.
- Explore Object-Oriented Testing strategies and test case design.

UNIT	Contents	No of Lectures
1	Object Oriented Concepts and Principles Object Oriented System Development <ul style="list-style-type: none"> ● Function/Data Methods (With Visibility) ● Object Oriented Analysis ● Object Oriented Construction ● Identifying the Elements of an Object Model ● Identifying Classes and Objects ● Specifying the Attributes (With Visibility) ● Defining Operations 	8
2	Object Oriented Analysis <ul style="list-style-type: none"> ● Iterative Development and the Rational Unified Proc ● Inception ● Understanding Requirements ● Use Case Model From Inception to Elaboration ● Elaboration 	6
3	Object Oriented Design <ul style="list-style-type: none"> ● The Booch Method, The Coad and Yourdon Method and Jacobson Method and Raumbaugh Method ● The Generic Components of the OO Design Model 	6

	<ul style="list-style-type: none"> • The System Design Process • Partitioning the Analysis Model • Concurrency and SubSystem Allocation • Task Management Component • The Data Management Component • The Resource Management Component • Inter SubSystem Communication • Object Design Process <p>Expected Course Outcomes: for object oriented software engineering</p>	
4	Architectural Modeling <ul style="list-style-type: none"> • Components Diagram • Deployment Diagram • Collaboration Diagram 	6
5	Object Oriented Testing <ul style="list-style-type: none"> • Object Oriented Testing Strategies • Test Case Design for Object Oriented Software • InterClass Test Case Design 	4

Text Books

- 1.Object-Oriented Software Engineering, Nirali Prakashan ISBN: 978-93-54511-88-2
- 2.Object-Oriented Software Engineering, Vision Publication ISBN:978-93-94022-51-5

Reference Books

1. Object Oriented Software Engineering,Ivar Jacobson,Pearson Education INS
- 2.Object Oriented Analysis and Design,Bennett, Simon,McGraw Hill

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)

Online Courses:

1. <https://www.udemy.com/course/advanced-object-oriented-analysis-of-hard-problems/?couponCode=BFCPSALE24>
2. https://onlinecourses.nptel.ac.in/noc24_cs119

<p align="center"> Course Title: - Theoretical Computer Science Course Type: Major Mandatory Paper2(Theory) Course Code: COM35102 Semester- V </p>			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: Student should have basic knowledge of: <ul style="list-style-type: none"> Mathematical Preliminaries Sets (Subset, Set Operations), Relations (Properties of Relations, Closure of Relations) and Functions Discrete Mathematics- Graphs, Trees, Logic and Proof Techniques 			
The main objectives of this course are to: <ul style="list-style-type: none"> To understand 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 			
UNIT	Contents	No of Lectures	
1	Finite Automaton <ul style="list-style-type: none"> Introduction: Symbol, Alphabet, String, Prefix & Suffix of Strings, Formal Language, Operations on Languages Deterministic finite Automaton – Definition, DFA as language recognizer, DFA as pattern recognizer. Nondeterministic finite automaton – Definition and Examples. NFA To DFA (Myhill Nerode Method) NFA with ϵ- transitions Definition and Examples. NFA with ϵ-Transitions to DFA & Examples Finite automaton with output – Mealy and Moore machine, Definition and Examples. Minimization of DFA, Algorithm & Problem using Table Method. 	8	
2	Regular Expressions and Languages <ul style="list-style-type: none"> Regular Expressions (RE): Definition & Example Regular Expressions Identities. Regular language-Definition and Examples. Conversion of RE to FA-Examples. Pumping lemma for regular languages and applications. 	4	

	<ul style="list-style-type: none"> ● Closure Properties of regular Languages 	
3	Context-Free Grammars and Languages <ul style="list-style-type: none"> ● Grammar - Definition and Examples. ● Derivation-Reduction - Definition and Examples. ● Chomsky Hierarchy. ● CFG: Definition & Examples. LMD, RMD, Parse Tree ● Ambiguous Grammar: Concept & Examples. ● Simplification of CFG: Removing Useless Symbols, Unit Production, ϵ-production and Nullable Symbol. ● Normal Forms: Greibach Normal Form (GNF) and Chomsky Normal Form (CNF) ● Regular Grammar: Definition. ● Left linear and Right Linear Grammar-Definition and Example. ● Equivalence of FA & Regular Grammar ● Construction of regular grammar equivalent to a given DFA. ● Construction of a FA from the given right linear grammar 	8
4	Push Down Automata <ul style="list-style-type: none"> ● Definition of PDA and examples. ● Construction of PDA using empty stack and final State method: Examples using stack ● Definition DPDA & NPDA, their correlation and Examples of NPDA ● CFG (in GNF) to PDA: Method and examples 	5
5	Turing Machine <ul style="list-style-type: none"> ● The Turing Machine Model, Definition and Design of TM ● Problems on language recognizers. ● Language accepted by TM. ● Types of Turing Machines (Multitrack TM, Two-way TM, Multitape TM, Non-deterministic TM) ● Introduction to LBA (Basic Model) & CSG. (Without Problems) 	5

Text Books
1.Theoretical computer Science - Nirali Prakashan 2.Theoretical computer Science -Vision Publication

Reference Books

1. Introduction to Automata Theory, Languages and Computation, John E. Hopcraft, Rajeev Motwani, Jeffrey D. Ullman, Third Edition, Pearson Education Publication, 2008
2. Introduction to Automata theory, Languages and computation By John E. Hopcroft and Jeffrey Ullman – Narosa Publishing House, 1995
3. Theory of Computer Science Automata, Languages and Computation, K.L.P. Mishra, N. Chandrasekaran, Publication- Prentice Hall of India, 2008
4. Introduction to Computer Theory Daniel I. A. Cohen – 2nd edition – John Wiley & Sons, 1996

<p align="center">Course Title: - Python Programming Course Type: Major Mandatory Paper 2 (Theory) Course Code: COM35103 Semester- V</p>			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: Student should have basic knowledge of: <ul style="list-style-type: none"> ● Familiarity with simple programming ideas like variables and loops ● Basic understanding of math concepts like arithmetic and sets. 			
The main objectives of this course are to: <ul style="list-style-type: none"> ● Build a strong foundation in Python programming and its basic concepts. ● Gain the ability to manipulate data using Python's lists, dictionaries, sets, and tuples. ● Explore real-world applications, including regular expressions and functional programming tools. 			
Expected Course Outcomes: <ul style="list-style-type: none"> ● Learn Python basics, data types, and control structures. ● Master lists, tuples, dictionaries, sets, and string manipulation. ● Develop functions, handle files, and work with modules. ● Implement exception handling, recursion, and regular expressions. 			
UNIT	Contents	No of Lectures	
1	An Introduction to Python <ul style="list-style-type: none"> ● The Python Programming Language, History, features, Applications, Installing Python, Running Simple Python program ● 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. 	6	
2	Control Statement. <ul style="list-style-type: none"> ● Sequence Control – Precedence of operators, Type conversion. ● Conditional Statements: if, if-else, nested if-else, ● Looping- for, while, nested loops, loop control statements (break, continue, pass) ● Strings: declaration, manipulation, special operations, escape 	7	

	<ul style="list-style-type: none"> ○ character, string formatting operator, Raw String, Unicode ○ strings, Built-in String methods. 	
3	List, tuples and dictionaries, Sets. <ul style="list-style-type: none"> ● 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. ● Tuples: 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. ● Dictionaries: 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. ● Sets: Definition, transaction of set(Adding, Union, intersection), working with sets. 	6
4	Functions <ul style="list-style-type: none"> ● Functions: Definitions and Uses, Function Calls, Type Conversion Functions, Math Functions, 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. 	5
5	Modules, working with Files, Exception Handling. <ul style="list-style-type: none"> ● Modules: Importing module, Creating & exploring modules, Math module, Random module, Time module. ● Packages: Importing package, creating package, examples. ● 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. ● Regular Expression- Concept of regular expression, various types of regular expressions, using match function. ● Exception Handling: Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions. 	6
6	Python for Data Analysis	

	<ul style="list-style-type: none"> ● Numpy: Introduction to Numpy. ● Creating arrays, using arrays and scalars ● Universal array Function ● Pandas: What is pandas? Where it is used? ● Matplotlib: Python for Data Visualization ● Introduction to Matplotlib
--	---

Text Books
<ul style="list-style-type: none"> ● Python Programming, Nirali Publication - ISBN: 978-93-54511-95-0 ● Python Programming, Vision Publication- ISBN: 978-93-90646-58-6

Reference Books
<ol style="list-style-type: none"> 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.

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)
Online Courses: <ul style="list-style-type: none"> ● Python Programming for Beginners to Intermediate: www.udemy.com/course/python-programming-for-beginners-to-intermediate/ ● Complete Python 3 Programming Bootcamp: https://www.udemy.com/course/complete-python-programming-course-beginner-to-advanced/?couponCode=BFCPSALE24 ● Python for Data Science (SWAYAM):

Course Title: Practical on Python Programming Course Type: Major Mandatory Paper3(Practical) Course Code: COM35103 Semester- V			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Practicals: 10	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: Student should have basic knowledge of: <ul style="list-style-type: none"> ● Procedure and object oriented programming languages ● Database management and relational management system concepts 			
The main objectives of this course are to: <ul style="list-style-type: none"> ● To understand the fundamental concepts of databases. ● To understand user requirements and frame it in a data model. ● To understand creations, manipulation and querying of data in databases. 			
Expected Course Outcomes: <ul style="list-style-type: none"> ● Learn to write simple Python programs and use Python IDEs. ● Work with strings, functions, and perform operations on lists, tuples, sets, and dictionaries. ● Handle files and manage date-time data. ● Implement exception handling and use regular expressions for text processing. 			
UNIT	Contents		No of Sessions
1	Python basics and IDE, Simple Python Program.		2
2	Strings and Functions.		2
3	List, Tuple, Sets and Dictionary.		2
4	File Handling and Date-Time.		2
5	Exception Handling and Regular Expression.		2

<p align="center"> Course Title: Practical on Advanced Java Course Type: Major Mandatory Paper(Practical) Course Code: COM35105 Semester- V </p>			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Practicals: 10	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: Student should have basic knowledge of: <ul style="list-style-type: none"> Knowledge of Advanced Java To learn and understanding concepts of Database, Pages and applications. 			
The main objectives of this course are to: <ul style="list-style-type: none"> Covers the complete scope of the syllabus. Bringing uniformity in the way course is conducted across different colleges. Continuous assessment of the students. Advanced Java is designed to develop web based, network centric, Enterprise level applications 			
Expected Course outcomes: <ul style="list-style-type: none"> To Learn database Programming using Java Understand and Create dynamic web pages using Servlets and JSP. Work with basics of framework to develop secure web applications 			
UNIT	Contents	No of Lectures	
1	Collections <ul style="list-style-type: none"> Study the Collection framework in java. To Implement various Interfaces and classes through algorithms. To Demonstrate Cursor Objects (Enumeration, Iterator, ListIterator, Comparator) 	2	
2	Multithreading <ul style="list-style-type: none"> To create and use threads in java. To demonstrate multithreading using Thread Synchronization, Inter-thread . Communication, Thread Priorities. 	2	
3	Database Programming <ul style="list-style-type: none"> To communicate with a database using java. To execute queries on tables. To obtain information about the database and tables. 	2	
4	Servlet and JSP <ul style="list-style-type: none"> 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 . 	2	

	<ul style="list-style-type: none"> ● Use of various session tracking methods like Cookies. ● JSP life-cycle. ● Use of JSP implicit objects. ● JSP Directives. ● Use of Scripting Elements. ● To understand actiontags in JSP. ● Understanding flow of JSP custom tags. 	
5	Spring Framework <ul style="list-style-type: none"> ● To create and understand the steps to develop Spring application. 	2

Course Title: - Data Science Course Type: Elective (Theory) Course Code: COM35106 Semester- V			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites Student should have basic knowledge of: <ul style="list-style-type: none"> ● Problem solving using computers ● Basic mathematics and statistics ● Knowledge of Databases 			
The main objectives of this course are to: <ul style="list-style-type: none"> ● Perform Exploratory Data Analysis ● Obtain, Clean/process, and transform data. ● Demonstrate Proficiency with statistical analysis of data. ● Present result using data visualization techniques. 			
UNIT	Contents		No of Lectures
1	An Introduction to Data Science <ul style="list-style-type: none"> ● Introduction to Data Science ● The to data science, The 3V's: Volume, Velocity, Variety ● Why learn Data Science? ● Application of Data Science ● Types of Data:- Structured. Semi-structured, Unstructured Data, Problems with unstructured data ● Data Sources:- Open Data, Social Media Data. ● Data Formats:- Integers, Floats, Text Data, Text File, Dense Numerical Arrays, Compressed or Archived Data, CSV Files, JSON Files 		07
2	Data Preprocessing <ul style="list-style-type: none"> ● Data Objects and Attribute Types:- What Is an Attribute? Nominal, Binary, Discrete vs Continuous Attributes ● Data Quality: Why Preprocess the Data? ● Data munging/ wrangling operations: - Cleaning Data - Missing Values, Noisy Data. ● Data Transformation :- Rescaling, Normalizing, Binarizing, 		07

3	Data Visualization <ul style="list-style-type: none"> • Introduction to Exploratory Data Analysis Data visualization and visual encoding • Basic data visualization tools Histograms, Bar charts/graphs, Scatter plots, Line charts, Pie charts. • Data Visualization types 	06
4	Data Analysis <ul style="list-style-type: none"> • Regression:- Linear regression • Classification:- Basics of classification, decision tree • Association Rule mining:- Apriori algorithm, support, confidence 	06
5	Introduction to R <ul style="list-style-type: none"> • R Programming Basics • Programs using List & Vectors • Matrix, String and Factors 	04

Reference Books

1. Data Mining Concepts and Techniques, Third Edition, Jiawei Han, Micheline
2. Data Science Fundamentals and Practical Approaches, Gypsy Nandi, A Hands-On Introduction to Data Science, Chirag Shah, University of Washington Cambridge University Press

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)

Online Courses: https://onlinecourses.swayam2.ac.in/imb25_mg23/preview

- <https://www.geeksforgeeks.org/what-is-data-science/>
- <https://pll.harvard.edu/course/data-science-visualization/2024-10>

Course Title: - Advanced Java
Course Type: Major Mandatory Paper(Theory)
Course Code:COM35107
Semester- V

Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
--	------------------------------	----------------------------	--

Prerequisites Student should have basic knowledge of:

- Basic Understanding of Programming
- Knowledge of Core Java

The main objectives of this course are to:

- 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

Expected 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.
- Understand and Create dynamic web pages, using Servlets and JSP.
- Work with basics of framework to develop secure web applications.

UNIT	Contents	No of Lectures
1	Collections <ul style="list-style-type: none">● Introduction to the Collection framework● List - ArrayList, LinkedList● Set - HashSet, TreeSet,● Map - HashMap and TreeMap● Interfaces such as Comparator, Iterator, ListIterator, Enumeration	5
2	Multithreading <ul style="list-style-type: none">● What are threads?● Life cycle of thread● Creating threads - Thread class , Runnable interface● Thread priorities● Running multiple threads● Synchronization and interthread communication	6
3	Database Programming <ul style="list-style-type: none">● The design of jdbc● Types of drivers● Executing sql statements, query execution● Scrollable and updatable Resultset	6

4	Servlet and JSP <ul style="list-style-type: none"> ● Introduction to Servlet and Hierarchy of Servlet ● Life cycle of servlet ● Handling get and post request (HTTP) ● Handling data from HTML to servlet ● Retrieving data from database to servlet ● Session tracking – User Authorization, URL rewriting, Hidden form fields, Cookies and HttpSession ● Introduction to JSP, Life cycle of JSP ● Implicit Objects ● Scripting elements - Declarations, Expressions, Scriptlets, Comments ● JSP Directives - Page Directive, include directive ● Mixing Scriptlets and HTML ● JSP Actions - jsp:forward , jsp:include, jsp:useBean, jsp:setProperty and ● jsp:getProperty 	10
5	Spring Framework <ul style="list-style-type: none"> ● Introduction of Spring framework ● Spring Modules / Architecture ● Spring Applications ● Spring MVC 	3

Text Books

- Object Oriented Programming using Java-II - ISBN · **9789354512612**
- Object oriented Programming using Java-II - ISBN: **9789354512612**

Reference Books

- R1. Complete reference Java by Herbert Schildt(5th edition)
- R2. Java 2 programming black books, Steven Horlznner
- 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

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)

Online Courses:

https://onlinecourses.nptel.ac.in/noc22_cs47/preview

<https://www.javatpoint.com/java-tutorial>

Course Title: - Practical on Data Science Using R Course Type: VSC(Practical) Course Code: COM35401 Semester- V			
Teaching Scheme: 6 Hours / Week	No. of Credits: 2	No. of Practical Sessions: 10	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: <ul style="list-style-type: none"> • Problem solving using computers • Basic mathematics and statistics • Knowledge of R 			
The main objectives of this course are to: <ul style="list-style-type: none"> • Perform Exploratory Data Analysis • Obtain, Clean/process, and transform data. • Demonstrate Proficiency with statistical analysis of data. • Present result using data visualization techniques. 			
UNIT	Contents	Sessions	
1	Write R Program create data frames,vectors,strings and bar-plot	1	
2	Data Pre-processing Write a R program to find all null values in a given data set and remove them.	1	
3	Write a R program to implement complete data pre-processing in a given dataset. (Missing value, encoding categorical value, Splitting the dataset into the training and test sets and feature scaling.	2	
4	Classification Write a R program build Decision Tree Classifier	2	
5	Association Rules Write a R Programme to read the dataset ("Iris.csv"). dataset and apply Apriori algorithm.	2	

6	<p>Write an R program to read the .csv file. Solve following:</p> <ul style="list-style-type: none">- To display the shape of the dataset.- To display the top rows of the dataset with their columns.- To display the number of rows randomly.- To display the number of columns and names of the columns.	2
---	--	---



*Sixth
Semester*

<p align="center">Course Title: - Software Testing Course Type: Major Mandatory Paper 1(Theory) Course Code: COM36101 Semester- V</p>			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: <ul style="list-style-type: none"> • Basic knowledge of test cases, planning and expected inputs/output • Knowledge of manual and automated testing tools 			
The main objectives of this course are to: <ul style="list-style-type: none"> • To provide the knowledge of software testing methods and strategies. • To understand how testing methods can be used as an effective tool in quality assurance of software. • To provide skills to design test case plans for testing software. • To provide knowledge of latest testing tools 			
UNIT	Contents	No of Lectures	CO targeted
1	Software testing fundamentals <ul style="list-style-type: none"> • Definition and Objective of Testing • Role of testing and its effects on quality Introduction to Test case design <ul style="list-style-type: none"> • How to identify errors, bugs in the given application. • Design entry and exit criteria for test case, design test cases in excel. • Describe features of a testing method used. 	6	
2	Software Testing Life Cycle <ul style="list-style-type: none"> • Overview of the stages of STLC: • Test planning • Test design • Test execution • Test Plan walkthrough - Live Project document • Practical tips on how to identify the following items for the Test plan:- <ul style="list-style-type: none"> ○ Test strategy ○ Effort Estimation ○ Entry criteria o Exit Criteria ○ Defect Management process 	6	

3	Test Templates creation and use <ul style="list-style-type: none"> • Test scenario template • Test case template • Test plan Defect report Status report...etc • Test scenario creation – what are they, how to write them, why, when etc. • Test documentation review- How to perform Peer reviews • Test cases creation • Test case design techniques <ul style="list-style-type: none"> ○ Boundary Value analysis ○ Equivalence partitioning ○ Error guessing o Types of parameters 	8	
4	Test cases and Test plan <ul style="list-style-type: none"> • Write Test Plan for given application with resources required. Write Test case for a given application. • Prepare Test report for test cases executed. • Write simple programs that make use of loops and control structures. Write test Cases for above programs. 	6	
5	Defect Report <ul style="list-style-type: none"> • Defect Life Cycle • Classification of Defect • Write Defect Report 	4	

Reference Books

1. Software Testing by Ron Patton
2. Effective Methods of Software Testing by William E. Perry
3. Software testing and Quality Assurance by Dr. Satish Ambike

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)

Online Courses:

- https://onlinecourses.nptel.ac.in/noc19_cs71/preview
- <https://www.mygreatlearning.com/academy/learn-for-free/courses/software-testing-fundamentals1>
- <https://www.geeksforgeeks.org/software-testing-basics/>

<p align="center">Course Title: - Compiler Construction Course Type: Major Mandatory Paper(Theory) Course Code: COM36102 Semester- VI</p>			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: Student should have basic knowledge of: <ul style="list-style-type: none"> Knowledge of Automata Theory and Languages. 			
The main objectives of this course are to: <ul style="list-style-type: none"> 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 			
UNIT	Contents	No of Lectures	
1	Introduction <ul style="list-style-type: none"> Definition of Compiler, Aspects of compilation. The structure of Compiler. Phases of Compiler – Lexical Analysis, Syntax Analysis, Semantic Analysis, Intermediate Code generation, code optimization, code generation. Error Handling. Introduction to one pass & Multipass compilers, cross compiler, Bootstrapping. 	8	
2	Lexical Analysis (Scanner) <ul style="list-style-type: none"> Review of Finite automata as a lexical analyzer, Applications of Regular Expressions and Finite Automata (lexical analyzer, searching using RE), Input buffering, Recognition of tokens. LEX: A Lexical analyzer generator (Simple Lex Program) 	4	

3	Syntax Analysis (Parser) <ul style="list-style-type: none"> • Definition, Types of Parsers • Top-Down Parser – • Top-Down Parsing with Backtracking: Method & Problems • Drawbacks of Top-Down parsing with backtracking, • Elimination of Left Recursion (direct & indirect) • Need for Left Factoring & examples • Recursive Descent Parsing: Definition • Implementation of Recursive Descent Parser Using Recursive Procedures • Predictive [LL (1)] Parser (Definition, Model) • Implementation of Predictive Parser [LL (1)] • FIRST & FOLLOW Construction of LL (1) Parsing Table • Parsing of a String using LL (1) Table. • Bottom-Up Parsers • Operator Precedence Parser -Basic Concepts • Operator Precedence Relations form Associativity & Precedence • Operator Precedence Grammar • Algorithm for LEADING & TRAILING (with ex.) • Algorithm for Operator Precedence Parsing (with ex.) • Precedence Functions • Shift Reduce Parser • Reduction, Handle, Handle Pruning • Stack Implementation of Shift Reduce Parser (with examples) • LR Parser: Model, Types [SLR (1), Canonical LR, LALR]-Method & examples. • YACC program sections, simple YACC program for expression evaluation 	8	
---	--	---	--

4	Syntax Directed Definition <ul style="list-style-type: none"> ● Syntax Directed Definitions (SDD) ● Inherited & Synthesized Attributes ● Evaluating an SDD at the nodes of a Parse Tree, Example ● Evaluation Orders for SDD's ● Dependency Graph ● Ordering the Evaluation of Attributes ● S-Attributed Definition ● L-Attributed Definition ● Application of SDT ● Construction of syntax trees, ● The Structure of a Type ● Translation Schemes ● Definition, Postfix Translation Scheme 	5	
5	Code Generation and Optimization <ul style="list-style-type: none"> ● Compilation of expression – ● Concepts of operand descriptors and register descriptors with example. ● Intermediate code for expressions – postfix notations, ● Triples, Quadruples and Expression trees. ● Code Optimization – Optimizing transformations – compile time evaluation, elimination of ● common sub expressions, dead code elimination, frequency reduction, strength reduction. ● Three address code ● DAG for Three address code ● The Value-number method for constructing DAG's. ● Definition of basic block, Basic blocks, and flow graphs ● Directed acyclic graph (DAG) representation of basic block. ● Issues in design of code generator. 	5	

Reference Books	
4.	Compilers: Principles, Techniques, and Tools, Alfred V. Aho, Ravi Sethi, Jeffrey D.Ullman, 2004
5.	Principles of Compiler Design By: Alfred V. Aho, Jeffrey D. Ullman, Narosa Publication House, 2002
6.	LEX & YACC, 2nd edition, O'reilly Publication, 2012

Course Title: - Information and Computer Networks Security Course Type: Major Mandatory Paper(Theory) Course Code: COM36103 Semester- VI			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: Student should have basic knowledge of: <ul style="list-style-type: none"> • Basic knowledge of Networking and ISO/OSI model 			
The main objectives of this course are to: <ul style="list-style-type: none"> • To understand some protocols of the application layer used for information transfer. • 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: <ul style="list-style-type: none"> • Student will understand the different protocols of the Application layer. • Develop understanding of technical aspect of Multimedia Systems • Develop various Multimedia Systems applicable in real time. • Identify information security goals. • Understand, compare and apply cryptographic techniques for data security. 			
UNIT	Contents	No of Lectures	
1	Introduction to Computer Networks <ul style="list-style-type: none"> • Computer Networks-Definition, goals and applications, Network Hardware: broadcast and point-to-point, topologies – star, bus, mesh, ring etc. • Network Types LAN, MAN, WAN, Wireless Networks, • Protocols and Standards – Definition of Protocol,standard, • Peer –to-peer and Server – based LAN • Network Software Protocol Hierarchies - layers, protocols, peers, interfaces,network architecture, protocol stack design issues of the layers –addressing, error control, flow control, multiplexing and demultiplexing, routing, Connection-oriented and connectionless service • OSI Reference Model-Functionality of each layer • TCP/IP Reference Model- Introduction to IP, TCP, and UDP, TCP/IP Protocol Suite,Addressing Physical, Logical and Port addresses 	8	

	<ul style="list-style-type: none"> The Data Link Layer-Error Control , CRC (Enough problems should be covered) 	
2	Application Layer <ul style="list-style-type: none"> Domain Name System 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 Electronic Mail- Architecture-First scenario, second scenario, Third scenario, Fourth scenario · User agent-services of user agent, types of UA Format of email , MIME Message transfer agent-SMTP Message Access Agent: POP and IMAP File Transfer FTP-Communication over data control connection,File type,data structure,Transmission mode 	7
3	Cryptography and Network Security <ul style="list-style-type: none"> Terminology: Cryptography, plain text and cipher text, cipher key, categories of cryptography-Symmetric key, asymmetric key Encryption model Symmetric key cryptography Traditional ciphers – substitution cipher, shift cipher, Transposition cipher · Simple Modern ciphers-XOR, Rotation cipher, s-box,p-box Asymmetric key cryptography-RSA 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 Entity Authentication-Passwords, Fixed passwords challenge-response 	9
4	Security in the Internet <ul style="list-style-type: none"> IPSecurity(IPSec) 	6

	<ul style="list-style-type: none"> Two modes · Two security protocols · Services provided by IPSec · Security association · Internet key exchange · Virtual private network ● SSL/TLS <ul style="list-style-type: none"> · SSL services · Four protocols · Transport layer security ● Firewalls <ul style="list-style-type: none"> · Packet filter firewall · Proxy firewall 	
--	---	--

Text Books

- 1.Computer Networks-II ,Vision Publication
- 2.Computer Networks-II, Nirali Publication

Reference Books

1. Data communications and networking by Behrouz Forouzan 4th/5th edition, McGraw Hill Pvt Ltd.
2. Computer Networks by Andrew S Tanenbaum, 4th/5th edition, Pearson Education
3. Cryptography and Network Security: Principles and Practice, William Stallings, 7th edition, Pearson Education
4. Network Security Essentials: Applications and Standards (For VTU), William Stallings, 3rd edition, Pearson Education

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)

Online Courses:Computer Networks, Network Security,Cryptography

<p align="center">Course Title: - Practical on Operating System Course Type: Major Mandatory Paper(Practical) Course Code: COM36104 Course Title: Practical Course Based on Operating System Semester- VI</p>			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Practicals 10	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
<p>Prerequisites: Student should have basic knowledge of:</p> <ul style="list-style-type: none"> • Understand concepts like process scheduling, memory allocation, and file systems • Understanding the importance of algorithms in efficient resource management. 			
<p>The main objectives of this course are to:</p> <ul style="list-style-type: none"> • To understand the concept of process scheduling with the help of simulation. • To study the concept demand paging concepts in the operating system. • To understand the working of the operating system shell. 			
<p>Expected Course Outcome:</p> <ul style="list-style-type: none"> • To learn and understand the concept of processes and threads. • Analyze and implement process scheduling algorithms (e.g., FCFS, SJF, Priority, Round Robin). • To learn disk scheduling algorithms (e.g., FCFS, SSTF, SCAN, C-SCAN) 			
<p>List of Assignments:</p> <ul style="list-style-type: none"> • Operations on processes : (2 slot) • (Create a child process using fork() and commands like exec(),execv() and execvp()). • Simulation of Operating System Shell and its working (commands)(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, MFU(4 slots) 			
UNIT	Contents		Sessions
1	<p>Process Management</p> <ul style="list-style-type: none"> • Implementation of CPU scheduling algorithms: • First-Come, First-Served (FCFS) • Shortest Job Next (SJN) • Round Robin (RR) • Priority Scheduling • Implementing process creation using system calls (e.g., fork() in 		3

	Unix/Linux).	
2	Deadlocks Simulation of the Banker's Algorithm for deadlock avoidance. <ul style="list-style-type: none"> Simulation of Deadlock Detection and Recovery algorithms. Implementing Deadlock Prevention mechanisms. 	2
3	Memory Management <ul style="list-style-type: none"> Simulation of Page Replacement Algorithms: FIFO (First-In-First-Out) LRU (Least Recently Used) Optimal Page Replacement. Demonstrating Segmentation using program simulation. 	2
4	Disk Scheduling <ul style="list-style-type: none"> Simulation of Disk Scheduling Algorithms FCFS (First-Come-First-Served) SSTF (Shortest Seek Time First) SCAN and C-SCAN. Visualization of seek time differences for each algorithm. 	3

Text Books
1. Operating System-I, Parijat publication ISBN-978-93-90769-28-5 2. Operating System-II, Nirali Prakshan ISBN-978-93-5451-253-7

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. Advanced Concepts in Operating Systems, M Singhal and NG Shivaratri, Tata McGraw Hill Inc, 2001 4. The 'C' Odyssey, UNIX-the open boundless C, Meeta Gandhi, Tilak Shetty, Rajiv Shah, BPB publication

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)
1. https://www.javatpoint.com/operating-system 2. https://www.w3schools.com/#gsc.tab=0&gsc.q=operating%20system 3. https://onlinecourses.nptel.ac.in/noc24_cs108/preview

Course Title: - Practical on Blockchain Technology Course Type: Major Mandatory(Practical) Course Code: COM36105 Semester- VI			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Practical s: 30 hrs	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: Student should have basic knowledge of: <ul style="list-style-type: none"> • Programming like Python, Solidity, Postman, Flask • Data Structures, Networking and foundation of cryptography 			
The main objectives of this course are to: <ul style="list-style-type: none"> • Understand Blockchain fundamentals • Perform hands-on implementation. • Explore cryptocurrency techniques 			
Course Outcomes On completion of the course, student will be able to: <ul style="list-style-type: none"> • Implement Blockchain Networks • Implement Blockchain Solutions for Specific Use Cases • Design smart contracts 			
UNIT	Contents		No. of Sessions
1	<ul style="list-style-type: none"> • Create a simple ledger in Excel/Google Sheets to simulate how a blockchain works. • Understand the working of blockchain using Ganache, Truffle 		1
2	<ul style="list-style-type: none"> • Demonstration of Blockchain: https://andersbrownworth.com/blockchain • Write a Simple Python program to create a Block class that contains index, timestamp, and previous hash. Connect the blocks to create a Blockchain. 		2
3	<ul style="list-style-type: none"> • Use Python to create blockchain and use postman and flask for live working 		2
4	<ul style="list-style-type: none"> • Write and deploy a basic Ethereum smart contracts using Remix IDE for the following <ol style="list-style-type: none"> 1. Basic Bank operations 2. Maintaining Certificate 3. Voting System 4. Lottery System 		5

Text Books
1. Blockchain Technology : Nirali Publication 2. Blockchain Technology : Vision Publication

Reference Books
1. Mastering Blockchain by Imran Bashir, Third Edition, Packt Publication 2. Waterhole, The Science of the Blockchain 3. Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System 4. Mastering Ethereum: Building Smart Contracts and DAPPS, by Andreas Antonopoulos, Dr. Gavid Wood, O'reilly Publication

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)
1. https://www.investopedia.com/terms/b/blockchain.asp 2. Udemy courses: https://www.udemy.com/topic/blockchain/?srsltid=AfmBOorGdTlcUT1gJLf8nJvZyStI-P hPvoUX5WLPSzweLIces-d6pJbb 3. NPTEL Course: Blockchain and its Applications

Course Title: - BlockChain Technology Course Type: Major Elective(Theory) Course Code: COM36106 Semester- VI			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: Student should have basic knowledge of: <ul style="list-style-type: none"> • Programming and Data Structures • Networking and foundation of cryptography 			
The main objectives of this course are to: <ul style="list-style-type: none"> • Understand the fundamentals of Blockchain • Explore the applications and emerging trends in Blockchain • Develop and deploy smart contracts • Address the challenges and security 			
Course Outcomes On completion of the course, student will be able to: <ul style="list-style-type: none"> • Develop and deploy Blockchain solutions • Apply Blockchain Technology to real-world use cases • Analyze Blockchain Architectures and Algorithms 			
UNIT	Contents		No of Lectures
1	Introduction to Blockchain Technology <ul style="list-style-type: none"> • Definition and History of Blockchain • Key Characteristics: Decentralization, immutability, transparency, and security • Blockchain Vs Database • Layered Blockchain Architecture • Blockchain Structure: Blocks, Chains, and Nodes • Types of Blockchains: Public, Private, and Consortium • Applications of Blockchain in industries • Overview of Distributed Ledger Technology (DLT) 		5
2	Blockchain Platforms <ul style="list-style-type: none"> • Overview of Popular Platforms: Bitcoin, Ethereum, Hyperledger, and Corda • Comparing Features and Use Cases of Different Platforms • Tokenization and Cryptocurrency Basics • Cryptography (private and public keys, Hashing & Digital Signature) 		6

	<ul style="list-style-type: none"> Consensus Mechanisms: Proof of Work (PoW), Proof of Stake (PoS), and others 	
3	Working of Blockchain <ul style="list-style-type: none"> Understanding SHA256 Hash Immutable Ledger Distributed P2P Network How Mining Works? (The NONCE and Cryptographic Puzzle) Byzantine Fault Tolerance Consensus Protocols: Proof of Work, Proof of State, Défense Against Attackers, Competing Chains Blockchain Demo 	9
4	Smart Contracts <ul style="list-style-type: none"> Ethereum Definition Ethereum Virtual Machine, Ether, Gas Smart contracts and their role in blockchain DApps Decentralized Autonomous Organizations (DAO) Hard and Soft Forks Structure of a Solidity Smart Contract: Contract declaration, State variables, functions, modifiers, and events. Data Types and Control Structures: Basic types: uint, int, string, bool, address. Arrays, mappings, and structs. If/else, loops, and error handling. 	10
Text Books		
1. Blockchain Technology : Nirali Publication 2. Blockchain Technology : Vision Publication		

Reference Books
1. Mastering Blockchain by Imran Bashir, Third Edition, Packt Publication 2. Waterhole, The Science of the Blockchain 3. Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System 4. Mastering Ethereum: Building Smart Contracts and DAPPS, by Andreas Antonopoulos, Dr. Gavid Wood, O'reilly Publication

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)

1. <https://www.investopedia.com/terms/b/blockchain.asp>

2. Udemy courses:

Delving into Blockchain : A Comprehensive

Blockchain A-Z: Build a Blockchain

Blockchain Programming

3. NPTEL Course: Blockchain and its Applications

<p align="center"> Course Title: - Operating System Course Type: Major Elective(Theory) Course Code: COM36107 Semester- VI </p>			
Teaching Scheme: 2 Hours / Week	No. of Credits: 2	No. of Lectures: 30	Examination Scheme: CIE: 20 Marks ESE: 30 Marks
Prerequisites: Student should have basic knowledge of: <ul style="list-style-type: none"> Understand concepts like process scheduling, memory allocation, and file systems Understanding the importance of algorithms in efficient resource management. 			
The main objectives of this course are to: <ul style="list-style-type: none"> Explore the purpose of operating systems in managing hardware and software resources. Learn how the OS serves as an intermediary between users and computer hardware. 			
Expected Course Outcomes <ul style="list-style-type: none"> To understand the core concepts, design principles, and functionalities of operating systems (OS). To Understand the architecture and key components of modern operating systems To understand and learn how to evaluate algorithms for memory allocation and page replacement 			
UNIT	Contents		No of Lectures
1	Introduction to System Programming & Operating System <ul style="list-style-type: none"> Introduction To System Programming Types of program – System program and Application program. Difference between system programming and application programming. Elements of Programming environment - Editor, Preprocessor, Assembler, Compiler, Interpreter, Linker and Loader, Debugger, Device drivers Editors, Assembler Introduction to Operating Systems <ul style="list-style-type: none"> Definition and Purpose Types of Operating Systems <ul style="list-style-type: none"> Batch OS Time-sharing OS Distributed OS Real-time OS 		6

	<ul style="list-style-type: none"> • System Components and Structure • System Calls and Operating System Services • 	
2	Process Management <ul style="list-style-type: none"> • Process Concept <ul style="list-style-type: none"> Process States Process Control Block (PCB) • Threads and Multithreading <ul style="list-style-type: none"> Benefits of Multithreading User vs. Kernel Threads • Process Scheduling <ul style="list-style-type: none"> Types of Schedulers Scheduling Criteria and Algorithms FCFS, SJF, Round-Robin, Priority Scheduling • Context Switching 	5
3	Synchronization and Deadlocks <ul style="list-style-type: none"> • Interprocess Communication (IPC) <ul style="list-style-type: none"> Shared Memory vs Message Passing • Synchronization Mechanisms <ul style="list-style-type: none"> Critical Section Problem Semaphores, Mutex, Monitors Classic Problems of Synchronization – The bounded buffer problem, The reader writer problem, The dining philosopher problem • Deadlocks <ul style="list-style-type: none"> Deadlock Prevention, Avoidance, Detection, and Recovery Banker's Algorithm 	7
4	Memory Management <ul style="list-style-type: none"> • 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, MFU 	6

5	I/O Systems <ul style="list-style-type: none"> ● I/O Hardware and Software Interrupt Handling Device Drivers ● Disk Scheduling: FCFS (First-Come, First-Served) SSTF (Shortest Seek Time First) SCAN, C-SCAN Algorithms DMA (Direct Memory Access) ● Security Concepts: Threats and Attacks User Authentication ● Protection: Goals and Principles of Protection Access Control Mechanisms (ACLs and Capabilities) Security Policies ● Basics of Cryptography in Operating System 	6
---	--	---

Text Books

1. Operating System-I, Parijat publication ISBN-978-93-90769-28-5
2. Operating System-II, Nirali Prakshan ISBN-978-93-5451-253-7

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. Advanced Concepts in Operating Systems, M Singhal and NG Shivaratri, Tata McGraw Hill Inc, 2001
4. The 'C' Odyssey, UNIX-the open boundless C, Meeta Gandhi, Tilak Shetty, Rajiv Shah, BPB publication

Related Online Content (MOOCS, SWAYAM, NPTEL, WEBSITES etc.)

1. <https://www.javatpoint.com/operating-system>
2. <https://www.w3schools.com/#gsc.tab=0&gsc.q=operating%20system>
3. https://onlinecourses.nptel.ac.in/noc24_cs108/preview