

DEPARTMENT OF COMPUTER SCIENCE
RATHINAM COLLEGE OF ARTS AND SCIENCE
(AUTONOMOUS)

RathinamTechzone, Pollachi Road, Eachanari, Coimbatore – 641021



Curriculum for M.Sc. Computer Science

I & II Semester

2019 – 2020 Regulation

Vision and Mission of the Institution:

VISION

To emerge as a world renowned Institution that is integrated with Industry to impart Knowledge, Skills, Research Culture and Values in youth who can accelerate the overall development of India.

MISSION

To impart superior quality education at affordable cost, nurture academic and research excellence, maintain eco-friendly and future-ready infrastructure, and create a team of well qualified teaching professionals who can build global competency and employability.

MOTTO

Transform the youth into National Asset.

Vision and Mission of the Department:

VISION

Build a strong research and teaching environment aimed towards betterment of society and industrial needs

MISSION

To provide quality undergraduate and post graduate education in both the theoretical and applied foundations of computer science and prepare the students for a globalised technological trends, knowledge in research towards serving the society.

Program Educational Objectives (PEO)

PEO1	:	To apply hardware and software technologies that provides computing solutions for successful careers in industry/higher education/research.
PEO2	:	To set foundation of mathematics, computer science and problem solving methodology for effective implementation in the area of software services and developments.
PEO3	:	To become entrepreneurs / innovators to apply the principles of system analysis, design, development and project management to address social, technical and business challenges.
PEO4	:	To promote awareness and to acquire leadership qualities with strong communication skills along with professional and ethical values.
PEO5	:	To adopt lifelong learning, act with Integrity and have inter-personal skills needed to engage with commitment towards social responsibilities and maintain high ethical standards.

Mapping of Institute Mission to PEO

Institute Mission	PEO's
To impart superior quality education at affordable cost, nurture academic and research excellence, maintain eco-friendly and future-ready infrastructure, and create a team of well qualified teaching professionals who can build global competency and employability.	PE02, PE04, PE05, PE03

Mapping of Department Mission to PEO

Department Mission	PEO's
To provide quality undergraduate and post graduate education in both the theoretical and applied foundations of computer science and prepare the students for a globalised technological trends, knowledge in research towards serving the society.	PE02, PE04, PE05, PE03

Program Outcomes (PO):

PO1	:	To apply fundamental knowledge of computing, mathematics and science relevance to the discipline.
PO2	:	To design, implement, and evaluate a computer - based system, process, component, or program for various applications.
PO3	:	To use current techniques, skills, and modern tools necessary for research-based knowledge and research methods for the cultural, societal, environmental considerations and demonstrate the knowledge of and need of sustainable development.
PO4	:	To formulate models, design and conduct experiments for interpreting data and critical thinking
PO5	:	To apply ethical principles, commit to professional ethics and responsibilities of the computing practice and its solutions.
PO6	:	Development of emphatic written and verbal communication skills.
PO7	:	To function individually and on teams, including diverse and multidisciplinary, to accomplish a common goal.
PO8	:	Continuous professional development through long term learning.
PO9	:	To solidify the computing principles to apply one's own work, as a member and leader in a team, to manage projects.

Correlation between the POs and the PEOs

Program Outcomes		PEO1	PEO2	PEO3	PEO4	PEO5
PO1	:		√			√
PO2	:	√		√		
PO3	:	√	√		√	√
PO4	:		√	√		√
PO5	:			√	√	√
PO6	:		√		√	√
PO7	:			√	√	
PO8	:	√		√	√	
PO9	:	√		√	√	

Components considered for Course Delivery is listed below:

1. Class room Lecture
2. Laboratory class and demo
3. Assignments
4. Mini Project
5. Project
6. Online Course
7. External Participation
8. Seminar
9. Internship

Mapping of POs with Course Delivery:

Program Outcomes	Course Delivery								
	1	2	3	4	5	6	7	8	9
PO1	√	√	√	√	√				
PO2	√					√	√	√	√
PO3	√	√		√	√				
PO4	√	√		√	√				
PO5		√		√	√				
PO6	√	√					√		
PO7			√				√	√	√
PO8		√					√	√	√
PO9	√			√	√		√	√	√

RATHINAM COLLEGE OF ARTS AND SCIENCE (AUTONOMOUS)

Scheme of curriculum for M.Sc. Computer Science
for the Batch admitted during 2019 - 2020
Board of Studies – Computer Science (PG)

S.No	Sem	Part	Type	Subject	Credit	Hour	Int	Ext	Total
1	1	III	Theory	Core– I Advanced Java Programming	4	7	40	60	100
2	1	III	Theory	Core– II Analysis and Design Algorithm	4	7	40	60	100
3	1	III	Theory	Core– III Shell Programming	4	6	40	60	100
4	1	III	Practical	Core Practical – I Java Programming Lab	4	5	40	60	100
5	1	III	Practical	Core Practical – II Shell Programming Lab	4	5	40	60	100
1	2	III	Theory	Core– IV Web Technology	4	5	40	60	100
2	2	III	Theory	Core– V Advanced Database	4	5	40	60	100
3	2	III	Theory	Core– VI Data mining and Data warehousing	4	5	40	60	100
4	2	III	Theory	Core– VII Advanced Operating System	4	5	40	60	100
5	2	III	Practical	Core Practical – III – Web Technology Lab	4	5	40	60	100
6	2	III	Practical	Core Practical – IV–Advanced Database Lab	4	5	40	60	100
1	3	III	Theory	Core– VIII –Python Programming	4	6	40	60	100
2	3	III	Theory	Elective - I	4	6	40	60	100
3	3	III	Theory	Elective - II	4	6	40	60	100
4	3	III	Practical	Core Practical – IV Python Lab	4	6	40	60	100
5	3	III	Practical	Core Practical V Elective-II Lab	4	6	40	60	100
6	3	III	Practical	Core Practical – VI Industrial Training Report	2	-	50	-	50
1	4	III	Theory	Core - IX Data Analytics	4	5	40	60	100
2	4	III	Theory	Elective- III	4	5	40	60	100
3	4	III	Theory	Elective- IV	4	5	40	60	100
4	4	III	Practical	Core Practical – VII Elective III Lab	4	5	40	60	100
5	4	III	Project	Core Project	8	10	80	120	200
					90	120			2250

List of Electives:

Elective	Subject Name
Elective - I	Advanced Computer Network
	Cryptography and Network Security
	Wireless Sensor Network
Elective - II	Animation Techniques
	Principles of Multimedia
	Computer Graphics
Elective - III	Android Development
	Ajax Programming
	Big Data
Elective - IV	Software Engineering
	Software Testing
	Software Project Management

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Core - Advanced Java Programming	4	7	0	0	Theory

Introduction: Learning the java Programming with oops concepts and other methodology for the purpose of practicing.

Course Outcome:

CO1	:	To understand and explore the basics of JDK Environment & tools also with OOPs
CO2	:	Understand the basics of java, control structure and Array concepts in java.
CO3	:	To learn the functions of Classes and Objects.
CO4	:	To familiarize the Packages also collections
CO5	:	Understanding the File and Exception Handling methods in java environment.

Unit I: [12 Periods]

Introduction to Java -Features of java -JDK Environment & tools like(java, javac, appletviewer, javadoc, jdb) - OOPs Concepts Class, Abstraction , Encapsulation, Inheritance, Polymorphism - Memory allocation for objects -Constructor -Implementation of Inheritance Simple, Multilevel-Interfaces -Abstract classes and methods -Implementation of Polymorphism -Method Overloading, Method Overriding -Nested and Inner classes-Modifiers and Access Control

Unit II: [12 Periods]

Packages -Packages Concept -Creating user defined packages -Java Built in packages -java.lang->math - java.util->Random, Date, Hashtable -Wrapper classes .Collection: Interfaces -Collection -List -Set-SortedSet-Enumeration -Iterator-ListIterator. Working with maps -Map interface-Map classes -HashMap -TreeMap.

Unit III: [12 Periods]

File and Exception Handling :Exception -Exception types -Using try catch and multiple catch Nested try, throw , throws and finally -Creating user defined Exceptions .File Handling :Stream ByteStream Classes - CharacterStream Classes -File IO basics -File operations -Creating file -Reading file(character, byte) - Writing file (character, byte) .

Unit IV: [12 Periods]

JDBC Object: The Concept of JDBC- JDBC Driver Types- JDBC Packages- Database Connection. JDBC and Embedded SQL: Model Programs- Tables- Indexing- Inserting Data into Tables- Inserting Data into Tables.

Unit V:

[12 Periods]

Introducing Swing- The Origins and Design Philosophy of Swing- Components and Containers- Layout Managers- A First Simple Swing Program- Use JButton- Create a JCheckBox- Create a Swing Applet. JQuery:

Introduction to JQuery - Validation using JQuery - JQuery Forms - JQuery Examples.

AJAX: Introduction to AJAX - Servlet and JSP with AJAX - Interacting with database .

Text Book :

1. Herbert Schildt ,”The Complete Reference – JAVA”, Seventh Edition,MGH.
2. E Balgurusamy , “Programming with JAVA-3e” third Edition TMH, 2014

Reference :

1. James Keogh ,“J2ME: The Complete Reference”, McGraw-Hill/Osborne,2008.
2. James Gosling, Bill Joy, Guy Steele, Gilad Bracha, and Alex Buckle,”The Java Tutorials“, Sixth Edition ,TMH ,2004
3. Bruce Eckel,”Thinking in Java ',3rd edition,TMH,2002.

Mapping of Courses and POs:

S- Strong Coorelation M – Medium Coorelation B – Blank

Mapping of Course Outcomes with Program Outcomes:

Course Outcomes	Program Outcomes								
	P01	PO2	P03	P04	P05	P06	P07	P08	P09
CO1	H	-	L	L	-	M	-	H	H
CO2	H	-	M	L	-	-	L	H	H
CO3	M	-	L	L	-	-	M	L	M
CO4	H	-	L	L	-	-	-	L	L
CO5	-	-	L	L	-	L	-	M	M

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Analysis and Design Algorithm	4	7	0	0	Theory

Introduction :

To introduce the classic algorithms in various domains and study the different techniques for designing efficient algorithms.

Course Outcome:

CO1	:	Ability to analyze the performance of algorithms.
CO2	:	Ability to choose appropriate algorithm design techniques for solving problems.
CO3	:	Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs.
CO4	:	To understand the variations among tractable and intractable problems.
CO5	:	To understand NP-Hard and NP-Complete problems

Unit I:

[12 Periods]

Introduction - The role of Algorithms in Computing – Algorithms, Algorithm Specification- Performance Analysis- Randomized Algorithm – Elementary Data Structures.

Unit II:

[12 Periods]

Divide and Conquer - General Method, Binary Search, Finding the Maximum and Minimum, Merge Sort, Quick Sort, Selection sort - Strassen's Matrix Multiplication. Greedy Method - Knapsack Problem, Minimum Spanning Trees, Prim's algorithm and Kruskal's algorithm.

Unit III:

[12 Periods]

Dynamic programming Method - Optimal Binary Search Trees, Traveling Salesman Problem, Longest Common Subsequence Back Tracking - Introduction - The 8-queens problem, Sum of Subsets. Branch and Bound General Method - Traveling Salesman Problem

Unit IV:

[12 Periods]

Graph Algorithms- Representation of Graph, Depth First Search, Breadth first search. Single Source shortest path – Dijkstra's Algorithm and Bellman Ford Algorithm. All Pair Shortest Path – Floyd-Warshall Algorithm. Lower Bound Theory - Comparison trees for sorting and searching.

Unit V:

[12 Periods]

NP-Hard and NP-Complete problems - Basic Concepts, NP-Hard graph problems, NP-Hard Scheduling problems - NP-Hard code generation problems - some simplified NP-Hard problems.

Reference :

1. Cormen T H, Leiserson C E, Rivest R L and Stein, Clifford, Introduction to Algorithms, PHI, 2nd Edition, 2009.
2. Horowitz E and Sahni S. Fundamentals of Computer Algorithms, Computer Science Press, 2008.
3. Essential Reading / Recommended Reading
4. Gelder Van Allen and Baase Sara, Computer Algorithms – Introduction to Design and Analysis, Addison Wesley, 3rd Edition, 2002.
5. Aho A V, Hopcroft J E and Ullman J D., The Design and Analysis of Computer Algorithms, Addison Wesley Publishing House, 1983.
6. Dromey, R.G., How to solve it by Computer, Prentice-Hall International, 2006.

Mapping of Course Outcomes with Program Outcomes:

Course Outcomes	Program Outcomes							
	P01	PO2	P03	P04	P05	P06	P07	P08
CO1	H	H		L	L	H	L	H
CO2	H	L	H	H	L	L	L	L
CO3	H	H	L	L	H	H	L	H
CO4	L	H		H	H	L		H
CO5	H	H	L	H	L	L	L	H

H - High ; M- Medium ; L- Low

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Shell Programming	4	6	-	-	Theory

Introduction: To give students knowledge about File systems and use of basic Commands and Shell programming.

Course Outcome:

CO1	: Understanding the basic set of commands and utilities in Linux/UNIX systems.
CO2	: To learn to develop software for Linux/UNIX systems.
CO3	: To learn the important Linux/UNIX library functions and system calls
CO4	: To obtain a foundation for an advanced course in operating systems
CO5	: To understand the concept of web processing and file systems in linux.

Unit-I : [12 periods]

Introduction - Printing in the terminal - Playing with variables and environment variables -F unction to prepend to environment variables-Math with the shell-Playing with file descriptors and redirection - Arrays and associative arrays-Visiting aliases-Grabbing information about the terminal-Getting and setting dates and delays-Grabbing information about the terminal-Getting and setting dates and delays- Debugging the Script- Functions and arguments-reading the output of a sequence of commands in a variable- Reading n character without pressing the return key- Field separator and iterators.

Unit-II : [12 periods]

Concatenating with cat-Recording and playing back of terminal sessions-Finding files and file listing-Playing with xargs-Translating with tr-Checksum and verification-Cryptographic tools and hashes-Sorting unique and duplicates-Temporary file naming and random numbers-Splitting files and data-Slicing filenames based on extension-Renaming and moving files in bulk-Spell checking and dictionary manipulation-Automating interactive input-Making commands quicker by running parallel processes.

Unit-III : [12 periods]

Generating files of any size-The intersection and set difference (A-B) on text files-Finding and deleting duplicate files-Working with file permissions, ownership, and the sticky bit-Making files immutable-Generating blank files in bulk-Finding symbolic links and their targets-Enumerating file type statistics-Using loopback files-Creating ISO files and hybrid ISO-Finding the difference between files, patching-Using head and tail for printing the last or first 10 lines-Listing only directories – alternative methods-Fast command-line navigation using pushd and popd-Counting the number of lines, words, and characters in a file-Printing the directory tree.

Unit-IV : [12 periods]

Using regular expressions - Searching and mining a text inside a file with grep-Cutting a file column-wise with cut-Using sed to perform text replacement-Using awk for advanced text processing-Finding the frequency of words used in a given file-Compressing or decompressing JavaScript-Merging multiple files as columns-Printing the nth word or column in a file or line-Printing text between line numbers or patterns-Printing lines in the reverse order-Parsing e-mail addresses and URLs from a text-Removing a sentence in a file containing a word-Replacing a pattern with text in all the files in a directory-Text slicing and parameter operations.

Unit-V : [12 periods]

Downloading from a web page- Downloading a web page as plain text- A primer on cURL- Accessing unread Gmail e-mails from the command line- Parsing data from a website - Image crawler and downloader - Posting to a web page and reading response- Archiving with tar- Archiving with cpio- Compressing data with gzip- Archiving and compressing with zip- Faster archiving with pbzip2- Creating filesystems with compression- Backup snapshots with rsync.

Text Book :

1. “Linux Shell Scripting Cookbook”, by Shantanu Tushar & Sarath Lakshman, Published by Packt Publishing Ltd., Second Edition.

Reference Book:

1. Eric Foster Johnson, John C. Welch and Micah Anderson, “ Beginning Shell Scripting”, Wiley, 2005.
2. Carl Albing, JP Vossen and Cameron Newham, “Bash Cookbook”, O’Reilly Media, 2007.
3. Richard Blum, “Linux Command Line and Shell Scripting Bible”, Wiley, 2008.

Mapping of Course Outcomes with Program Outcomes:

Course Outcomes	Program Outcomes							
	P01	P02	P03	P04	P05	P06	P07	P08
CO1	H	L	-	L	-	L	-	L
CO2	H	L	-	L	-	-	-	L
CO3	H	L	-	L	-	-	-	L
CO4	H	M	L	L	-	-	-	L
CO5	H	L	-	L	-	-	-	L

H - High ; M- Medium ; L- Low

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Practical – I Java Programming Lab	4	0	0	5	Practical

Course Outcome

CO1	: Implement and know the concept of Java Data Base Connectivity.
CO2	: Development of web based components using Servlets.
CO3	: Java application development using Java Beans.
CO4	: Using JSP to dynamically generate HTML, XML or other types of documents in response to a Web client request.
CO5	: The server and the client communicate and pass information back and forth by using RMI.

Course Outcome:

1. Write a java Program that displays two text boxes for entering a students' Roll-no and Name with appropriate labels and buttons.
2. Write a Java program that makes a connection with database using JDBC and prints metadata of this connection.
3. Write a Program to insert, update and delete Student Information using JDBC.
4. Write a java program for two way TCP communication for server and client. It should look like a simple chat application.
5. Write a java program for UDP Communication where client will send name of country and server will return the capital of that country.
6. Write a simple servlet program that just generates plain text.
7. Write a simple servlet program which displays cookie id.
8. Write a JSP program for basic arithmetic functions.
9. Write a JSP program to display a String.
10. Write a JSP program to create check boxes.
11. Write a program to generate plain text using java beans?
12. Create a simple calculator application that demonstrates the use of RMI. You are not required to create GUI.

Mapping of Course Outcomes with Program Outcomes:

H - High ; M- Medium ; L- Low

Course Outcomes	Program Outcomes							
	P01	PO2	P03	P04	P05	P06	P07	P08
CO1	M	-	H	H	M	-	M	H
CO2	L	-	M	L	M	-	M	H
CO3	M	-	L	H	H	-	M	H
CO4	M	-	H	M	L	-	-	H
CO5	H	-	L	L	M	-	M	H

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Practical – II– Shell Programming Lab	4	-	-	5	Practical Lab

Course Outcome:

CO1	: Comfortably use basic UNIX/Linux commands from the command line
CO2	: Organize and manage their files within the UNIX/Linux file system. And organize and manage their processes within UNIX/Linux
CO3	: Usefully combine UNIX/Linux tools using features such as filters, pipes, redirection, and regular expressions.
CO4	: Customize their UNIX/Linux working environment
CO5	: Know how to use UNIX/Linux resources to find additional information about UNIX/Linux commands

1. Write a Shell program to identify the Current Shell and length of the String.
2. Write a Shell program to Count backwards for 100 to 0 Using Loops.
3. Write a Shell program to Search File name using regular Expression.
4. Write a Shell program for sorting unique and duplicate Text Files.
5. Write a Shell program to perform operation using intersection, difference and set difference.
6. Write a Shell program to find and duplicate File in a directory of files.
7. Write a Shell program to perform Silent output for grep.
8. Write a Shell program to Printing lines before and after text matches.
9. Write a Shell program for printing text between line numbers or patterns.
10. Write a Shell program for Parsing e-mail addresses and URLs from text.
11. write a shell script that will take an input file and remove identical lines (or duplicate lines from the file)
12. Write a shell script that takes a name of a folder, and delete all sub folders of size 0

Mapping of Course Outcomes with Program Outcomes:

Course Outcomes	Program Outcomes								
	P01	PO2	P03	P04	P05	P06	P07	P08	P09
CO1	H	L	-	L	-	L	-	L	-
CO2	H	L	-	L	-	-	-	L	-
CO3	H	L	-	L	-	-	-	L	-
CO4	H	M	L	L	-	-	-	L	-
CO5	H	L	-	L	-	-	-	L	-

H - High ; M- Medium ; L- Low

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Web Technology	4	5	1	0	Theory

Introduction :

This Subject is useful for Making own Web page and how to host own web site on internet. Along with that Students will also learn about the protocols involve in internet technology.

Course Outcome:

CO1	:	Students are able to develop a dynamic webpage by the use of java script and DHTML.
CO2	:	Students will be able to write a well formed / valid XML document
CO3	:	Students will be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.
CO4	:	Students will be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database.
CO5	:	Students will be able to write a server side java application called JSP to catch form data sent from client and store it on database.

Unit - I : [12 periods]

HTML Common tags- Block Level and Inline Elements - Lists - Tables - Images - Forms – Frames - Cascading Style sheets - CSS Properties - Java Script: Introduction to Java Script - Objects in Java Script - Dynamic HTML with Java Script.

Unit – II : [12 periods]

Data Base - Database Schema - A Brief Overview Of The JDBC Process, JDBC Driver Types, JDBC Packages, Database Connection, Associating The JDBC-ODBC Bridge With Database, Creating, Inserting, Updating And Deleting Data In Database Tables, Result Set, Metadata.

Unit – III: [12 periods]

Tomcat web server, Introduction to Servlets: Servlets, the Advantage of Servlets over “Traditional” CGI, Basic Servlet Structure, Simple Servlet Generating Plain Text, Compiling and Installing the Servlet, Invoking the Servlet, Lifecycle of a Servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Context Parameters, Handling Http Request & Responses, Using Cookies-Session Tracking, Servlet with JDBC.

Unit – IV: [12 periods]

The Problem with Servlet. The Anatomy of a JSP Page, JSP Processing, JSP Application Development: Generating Dynamic Content, Using Scripting Elements, Implicit JSP Objects, Declaring Variables and Methods , Sharing Data Between JSP pages, Users Passing Control and Data between Pages, JSP application design with JDBC, JSP Application Design with MVC.

Unit – V:

[12 periods]

Basics of PHP, Functions, Error Handling, Interaction between PHP and MySQL, Database using Forms, Using PHP to manipulate and Retrieve Data in MySQL.

Textbook:

1. Jon Duckett “Beginning Web Programming” WROX.
2. Marty Hall and Larry Brown “Core Servlets and Java Server pages Vol. 1: Core Technologies”, Pearson.

Reference :

1. Dan Woods and Gautam Guliani, “Open Source for the Enterprise: Managing Risks, Reaping Rewards”, O’Reilly, Shroff Publishers and Distributors, 2005.
2. Sebesta, “Programming world wide web” Pearson.
3. Dietel and Nieto, “Internet and World Wide Web – How to program”, PHI/Pearson Education Asia.

Mapping of Course Outcomes with Program Outcomes:

Course Outcomes	Program Outcomes											
	P01	PO2	P03	P04	P05	P06	P07	P08	P09	P010	P011	P012
CO1	H	-	L	L	-	M	-	H	H	-	M	H
CO2	H	-	M	L	-	-	L	H	H	-	M	H
CO3	M	-	L	L	-	-	M	L	M	-	M	H
CO4	H	-	L	L	-	-	-	L	L	-	-	H
CO5	-	-	L	L	-	L	-	M	M	-	M	H

H - High ; M- Medium ; L- Low

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Advanced Database	4	5			Theory

Introduction: To study the concept of advanced database techniques can be able to critically assess new developments in database technology, Interpret and explain the impact of emerging database standards.

Course Outcome

CO1	An ability to create a SQL database
CO2	Ability to analyse the transaction in database
CO3	Ability to create table in Sql
CO4	Gain Knowledge of data warehousing
CO5	Gain Knowledge of data Mining

Unit I

[12 periods]

Overview : PL/SQL – Introduction to PL/SQL – Declare, begin statements, Variables, Control Structure, PL/SQL Transactions – Savepoint, Cursor, PL/SQL Database Objects – Procedures, Functions, Packages, Triggers. Programmatic SQL – Embedded SQL, Dynamic SQL, and ODBC Standard.

Unit II

[12 periods]

Transaction processing and concurrency control: Definition of Transaction and ACID properties. Transaction Processing - Transaction-processing monitors, transactional workflows, main- memory databases, real-time transaction systems, long-duration transactions, transaction management in multi-databases. Concurrency Control – Locks, Optimistic Concurrency Control (Backward and Forward validations), Timestamping Concurrency Control.

Unit III

[12 periods]

Object-based databases and xml : Object-based databases – Complex data types, structured types and inheritance in SQL, table inheritance, array and multiset types in SQL, object identity and reference types in SQL, implementing O-R features, Persistent programming languages, OO vs OR. XML – Structure of XML, Document Schema, Querying and Transformation, API in XML, XML applications.

Unit IV

[12 periods]

DBMS : Concepts, Benefits and Problems, DW Architecture – Operational Data, load manager, meta data, DW Data flows – inflow, upflow, meta flow, DW tools and technologies – Extraction, cleansing and transformation tools, DW DBMS, admin and management tools, data marts – reasons and issues.

Unit V

[12 periods]

Olap and NoSQL: On-line Analytical Processing – OLAP BenchMarks, applications, benefits, tools, categories, extensions to SQL- The Nosql Movements – Basic Concepts and techniques – key/value stores.

Text Book

1. Dr.Radványi Tibor “Advanced DBMS” Kiadó © Dr. Radványi Tibor, 2011

Reference Book

1. Korth, Sudarshan, Silberschatz “Database System Concepts”
2. Ramamohanarao Kotagiri and P Radha Krishna , “Advances in Databases: Concepts, Systems and Applications: 12th International Conference on Database Systems for Advanced Applications”
3. Vijay Atluri and Sushil Jajodia , “Multilevel Secure Transaction Processing (Advances in Database Systems)”
4. Charu C Aggarwal and, “Managing and Mining Graph Data (Advances in Database Systems)”

Mapping of Course Outcome and Program Outcome

Course Outcome	Program Outcome							
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	H	--	--	-	L	-	M	--
CO2	-	M	M	-	M	--	H	M
CO3	H	-	H	H	-	H	-	L
CO4	-	H	-	H	H	-	H	-
CO5	M	-	M	-	L	M	-	H

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Core- VI Data mining and Data warehousing	4	5	0	0	Theory

Introduction: This subject gives the Knowledge of Fundamentals of Data warehousing, Data mining and Data Mining Techniques.

Course Outcome

CO1	: To explain the core concepts of the Data Warehousing. This Explain about the Concept of Different Types of Data warehouse and its features.
CO2	: To discuss Data Mining Techniques and issues.
CO3	: To analyze various Association Rules in Data Warehousing.
CO4	: To understand various Clustering techniques.
CO5	: To deploy applications of Web Mining.

UNIT I

[12 periods]

Data Warehousing: Introduction – Definition – Multidimensional Data Model - OLAP Operations – Warehouse Schema – Data warehousing Architecture – Metadata – OLAP Engine - Data Warehouse Backend Process.

UNIT II

[12 periods]

Data Mining: Definition – Comparison with other fields – DM Techniques – Issues - Application Areas.

UNIT III

[12 periods]

Association Rules: Methods – A Priori algorithm – Partition Algorithm – Pincer-Search Algorithm – Border Algorithm – Generalized Association Rules with Item constraints.

UNIT IV

[12 periods]

Clustering Techniques : Clustering Paradigms – Partitioning Algorithms – CLARA – CLARANS- Hierarchical Clustering – DBSCAN – Categorical Clustering Algorithms – STIRR. Decision Trees: Tree Construction Principle – Best Split – Splitting Indices – Splitting Criteria CART – ID3.

UNIT V

[12 periods]

Web Mining: Introduction – Web Content Mining – Web Structure Mining – Web Usage Mining – Text Mining – Hierarchy of Categories – Text Clustering.

Text Book:

1. Arun K Pujari, Data Mining Techniques, Universities Press, 2001.

Reference book:

1. "Data Mining: Concepts and Techniques" by Jewie Han, Michelins Kamber, Jianpei Morgan Kaufmann Publishers, Third Edition, 2012.
2. "Introduction to Data Mining" by Pang-Ning Tan, Michael Steinbach, Vipin Kumar, Pearson India Education 2016.

Mapping of Course Outcomes with Program Outcomes:

Course Outcomes	Program Outcomes								
	P01	PO2	P03	P04	P05	P06	P07	P08	P09
CO1	M	-	L	L	-	M	-	H	H
CO2	L	-	M	M	-	-	L	H	H
CO3	M	-	M	M	H	M	M	M	M
CO4	H	-	L	L	H	-	-	H	L
CO5	-	-	H	H	-	L	-	M	M

H - High ; M- Medium ; L- Low

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Advanced Operating System	4	5	0	0	Theory

Introduction :

Advanced Operating Systems is a post graduate-level course that addresses a broad range of topics in operating system design and implementation.

Course Outcome:

CO1	:	Knowledge about advanced concepts in OS
CO2	:	Ability to develop OS for distributed systems
CO3	:	Extrapolate the interactions among the various components of computing systems
CO4	:	Master system resources sharing among the users
CO5	:	Ability to develop modules for mobile devices

UNIT I

[12 periods]

Multiprocessor Operating Systems: System Architectures- Structures of OS – OS design issues – Process synchronization – Process Scheduling and Allocation- memory management.

UNIT II

[12 periods]

Distributed Operating Systems: System Architectures- Design issues – Communication models – clock synchronization – mutual exclusion – election algorithms- Distributed Deadlock detection

UNIT III

[12 periods]

Distributed scheduling - Distributed shared memory - Distributed File system – Multimedia file systems - File placement - Caching

UNIT IV

[12 periods]

Database Operating Systems: Requirements of Database OS – Transaction process model – Synchronization primitives - Concurrency control algorithms

UNIT V

[12 periods]

Mobile Operating Systems: ARM and Intel architectures - Power Management - Mobile OS Architectures - Underlying OS - Kernel structure and native level programming - Runtime issues- Approaches to power management

TEXT BOOK:

1. M Singhal and NG Shivaratri , Advanced Concepts in Operating Systems, Tata McGraw Hill Inc, 2001.

REFERENCE BOOK

1. A S Tanenbaum, Distributed Operating Systems, Pearson Education Asia, 2001
2. Source Wikipedia, Mobile Operating Systems, General Books LLC, 2010
3. P. K. Sinha, "Distributed Operating Systems" Pearson Education, 1998.
4. Distributed Operating Systems – The Logical Design by A. Goscinski, AW
5. Modern Operating Systems by A. S. Tanenbaum, Pearson Education

Mapping of Course Outcomes with Program Outcomes:

Course Outcomes	Program Outcomes							
	P01	P02	P03	P04	P05	P06	P07	P08
CO1	H	L	-	M	-	-	-	L
CO2	M	L	-	L	-	-	-	L
CO3	H	-	-	L	-	-	-	L
CO4	H	L	-	H	H	H	H	L
CO5	H	L	-	L	-	-	-	L

H - High ; M- Medium ; L- Low

Semester II:

Subject Code	Subject Title	Credit	Lecture	Tutorial	Practical	Type
	Core Practical – III - Web Technology Lab	4	-	-	4	Lab

Course Outcome:

CO1	: Conceptualize and plan an internet-based business that applies appropriate business models and web technologies.
CO2	: To Select and apply markup languages for processing, identifying, and presenting of information in web pages.
CO3	: Design websites using appropriate security principles, focusing specifically on the vulnerabilities inherent in common web implementations.
CO4	: Incorporate best practices in navigation, usability and written content to design websites that give users easy access to the information.
CO5	: To Create a static website using HTML and add dynamic functionality to it by using java Script.

1. Write an HTML code to display your education details in a tabular format.
2. Write an HTML code to display your CV on a web page.
3. Write an HTML code to create a Home page having three links: About Us, Our Services and Contact Us. Create separate web pages for the three links.
4. Write an HTML code to create a login form. On submitting the form, the user should get navigated to a profile page.
5. Write an HTML code to create a Registration Form. On submitting the form, the user should be asked to login with this new credentials.
6. Write an HTML code to create your Institute website, Department Website and Tutorial website for specific subject.
7. Write an HTML code to illustrate the usage of the following:
 - Ordered List
 - Unordered List
 - Definition List
8. Write an HTML code to create a frameset having header, navigation and content sections.
9. Write an HTML code to demonstrate the usage of inline CSS.

10. Write an HTML code to demonstrate the usage of internal CSS.
11. Write an HTML code to demonstrate the usage of external CSS.
12. Write a Java script to prompt for users name and display it on the screen.
13. Design HTML form for keeping student record and validate it using Java script.
14. Write programs using Java script for Web Page to display browsers information.

Mapping of Course Outcomes with Program Outcomes:

Course Outcomes	Program Outcomes								
	P01	P02	P03	P04	P05	P06	P07	P08	P09
CO1	L	-	M	L	L	M	L	M	H
CO2	H	-	L	L	L	-	L	L	L
CO3	H	-	M	M	L	L	M	L	M
CO4	L	-	M	M	L	-	L	L	M
CO5	H	-	H	L	L	L	-	H	H

H - High ; M- Medium ; L- Low

Semester – II

Subject Code	Subject Title	Lecture	Tutorial	Practical	Credit	Type
	Core Practical – Advanced Database Lab	0	0	4	4	Practical

1. Database Schema for a customer-sale scenario

Customer(**Cust id : integer**, cust_name: string)
Item(**item id: integer**, item_name: string, price: integer)
Sale(**bill no: integer**, bill_data: date, **cust_id: integer**, **item_id: integer**, qty_sold: integer)

For the above schema, perform the following—

- Create the tables with the appropriate integrity constraints
- Insert around 10 records in each of the tables
- List all the bills for the current date with the customer names and item numbers
- List the total Bill details with the quantity sold, price of the item and the final amount
- List the details of the customer who have bought a product which has a price>200
- Give a count of how many products have been bought by each customer
- Give a list of products bought by a customer having cust_id as 5
- List the item details which are sold as of today
- Create a view which lists out the bill_no, bill_date, cust_id, item_id, price, qty_sold, amount
- Create a view which lists the daily sales date wise for the last one week

2. Database Schema for a Student Library scenario

Student(**Stud no : integer**, Stud_name: string)
Membership(**Mem no: integer**, **Stud no: integer**)
Book(**book no: integer**, book_name:string, author: string)
Iss_rec(**iss no:integer**, iss_date: date, **Mem no: integer**, **book no: integer**)

For the above schema, perform the following—

- Create the tables with the appropriate integrity constraints
- Insert around 10 records in each of the tables
- List all the student names with their membership numbers
- List all the issues for the current date with student and Book names
- List the details of students who borrowed book whose author is CJDATE
- Give a count of how many books have been bought by each student
- Give a list of books taken by student with stud_no as 5
- List the book details which are issued as of today
- Create a view which lists out the iss_no, iss_date, stud_name, book name
- Create a view which lists the daily issues-date wise for the last one week

3. Database Schema for a Employee-pay scenario

employee(**emp id : integer**, emp_name: string)
department(**dept id: integer**, dept_name:string)
paydetails(**emp_id : integer**, **dept_id: integer**, basic: integer, deductions: integer, additions: integer, DOJ: date)
payroll(**emp_id : integer**, pay_date: date)

For the above schema, perform the following—

- a. Create the tables with the appropriate integrity constraints
 - b. Insert around 10 records in each of the tables
 - c. List the employee details department wise
 - d. List all the employee names who joined after particular date
 - e. List the details of employees whose basic salary is between 10,000 and 20,000
 - f. Give a count of how many employees are working in each department
 - g. Give a names of the employees whose netsalary>10,000
 - h. List the details for an employee_id=5
 - i. Create a view which lists out the emp_name, department, basic, dedeuctions, netsalary
 - j. Create a view which lists the emp_name and his netsalary
4. Write a program to find largest number from the given three numbers.
 5. Write a program to check whether the given number is Armstrong or not
 6. Write a program to generate all prime numbers below 100.
 7. Write a program to demonstrate predefined exceptions
 8. Create a cursor, which displays all employee numbers and names from the EMP table.
 9. Create a cursor, which update the salaries of all employees as per the given data.
 10. Create a procedure to find reverse of a given number
 11. Create a procedure to demonstrate IN, OUT and INOUT parameters
 12. Create a function to check whether given string is palindrome or not.
 13. Create a trigger before/after update on employee table for each row/statement.
 14. Create a trigger before/after delete on employee table for each row/statement.
 15. Create a trigger before/after insert on employee table for each row/statement.