



School of Computer Science & IT Devi Ahilya Vishwavidyalaya

SYLLABUS

MCA 2 years

Program Educational Objectives (PEOs)

- PEO 1:** Nurture through in-depth knowledge of fundamental computing concepts and programming along with supplementary skills to bridge industry-academia gap.
- PEO 2:** Develop proficiency in application-based projects through latest technological trends.
- PEO 3:** Inculcate professional values with ethics to pursue career in industry or academia.

Program Specific Outcomes (PSOs)

- PSO 1:** Understand, analyze and develop computer programs in the areas related to algorithm, web design and machine learning for designing efficient software system.
- PSO 2:** Apply standard software engineering practices and strategies in software project development using modern computing tools and techniques with concern for societal, environmental, and cultural aspects.
- PSO 3:** Learn various issues, latest trends in technology development and thereby innovate new ideas and solutions to existing problems.

II - SEMESTER

CS-4305: SOFTWARE ENGINEERING

Course Outcomes (COs):

CO1: Understand the applications of software engineering processes and models.

CO2: Inculcate ability to plan, schedule and estimate software projects.

CO3: Develop skills for analysis and design of software projects using structured and object-oriented approaches.

CO4: Apply testing and quality assurance mechanisms to produce and reliable system.

CO5: Ability to develop software systems using software engineering approaches.

Course Contents

UNIT-I

No. of Hours: 8

Introduction to Software Engineering and Software Processes: Software, Software Classifications and Characteristics, Software Crisis. What is Software Engineering? System Engineering Vs. Software Engineering, Software Engineering Challenges. Software Processes: Process model, Elements and Characteristics of Process model, Process Classification, Software Development Processes: SDLC, Waterfall, Iterative Waterfall, Prototyping, Incremental, Spiral, RAD, Agile Software Development: Principles, Practices & Methods; RUP process, Component-Based Development model etc.

UNIT-II

No. of Hours: 8

Project Management and Planning: Project management essentials, Project success and failures, Project Life Cycle, Project team structure and organization, Software Configuration Management, Risk Management. Project planning activities: Metrics and Measurements, Project Size Estimation, Effort Estimation Techniques, Staffing and Personnel Planning, Project Scheduling and Miscellaneous Plans.

UNIT-III

No. of Hours: 8

Requirements Engineering: Software Requirements, Requirements Engineering Process, Requirements Elicitation. Requirements Analysis: Structured Analysis, Object-oriented Analysis. Requirements Specification, Requirements Validation, and Requirements Management.

UNIT-IV

No. of Hours: 8

Software Design and Coding: Software Design Process, Characteristics of a Good Design, Design Principles, Modular Design (Coupling and Cohesion). Software Architecture. Design Methodologies: Function-oriented Design (Structured Design Methodology in brief). Object-oriented Design using UML, Logical Design.

UNIT-V

No. of Hours: 8

Software Testing, Quality and Maintenance: Testing Fundamentals, Test Planning, Black-Box and White-Box Testing strategy, Levels of Testing, Debugging Approaches. Quality Concept, Quality

Factors, Verification and Validation, Quality Assurance Activities, Quality Standards: Capability Maturity Model (CMM). Software Reliability, Software Maintenance and Reengineering.

Text Books:

1. Software Engineering: Concepts & Practices- Ugrasen Suman, Cengage Learning, 2nd Edition.
2. Fundamentals of Software Engineering- Rajib Mall, PHI, New Delhi.
3. Object Oriented Analysis and Design Using UML, Ugrasen Suman et al, Cengage Learning.

Reference Books:

1. An Integrated Approach to Software Engineering - Pankaj Jalote, Narosa Publishing House.
2. Software Engineering - A practitioner's approach - R. S. Pressman, Tata McGraw-Hill International Editions, New York.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc21_cs65/preview
2. <http://www.rspa.com/spi/>
3. <https://sei.cmu.edu/>
4. <https://dl.acm.org/journal/tosem>

CS-4405: DATABASE MANAGEMENT SYSTEM

Course Outcomes (COs):

CO1: Conceptual Clarity on database systems and their evolution.

CO2: Theoretical foundation of query languages through relational algebra.

CO3: Understand the database design issues from E-R model to Normalization.

CO4: Proficiency in SQL, PLSQL and NoSQL through Case Studies/Real Life Problems.

CO5: Exposure to advance topics like transaction management, concurrency control and physical data storage.

Course Contents

UNIT-I

No. of Hours: 8

Introduction and Relational Model: Advantages of DBMS approach, various views of data, data independence, schema & sub-schema, primary concept of data models, database languages, transaction management, database administrator & user, data dictionary, database structure & architectures. Relational Model: Domains, relation, kind of relation, Relational databases, Various types of keys: candidate, primary, alternate & foreign keys, relational algebra with fundamental and extended operations, modification of database.

UNIT-II

No. of Hours: 12

ER Model and SQL: Basic concept, design issues, mapping constraint, keys, ER diagram, weak & strong entity-sets, specialization & generalization, aggregation, inheritance, design of ER schema, Reduction of ER Schema to tables. SQL: Basic structure of SQL, Set operation, Aggregate functions, Null values, Nested Sub queries, derived relations, views, Modification of database, join relation, Domain, relation & keys, DDL in SQL, integrity rules, Domain rules, Attributes rules, assertion, integrity & SQL. Query Optimization. PL/SQL: Programming concepts, stored procedure, function, packages and triggers. Database connectivity with ODBC/JDBC 9.

UNIT-III

No. of Hours: 6

Functional Dependencies: Basic definitions, Trivial & non trivial dependencies, closure set of dependencies & of attributes, Irreducible set of dependencies, FD diagram. Normalization: Introduction to normalization, non loss decomposition, First, second and third normal forms, dependency preservation, BCNF, Multivalued dependencies and fourth normal form, join dependencies and fifth normal form.

UNIT-IV

No. of Hours: 7

Storage Structure & File Organization: Basic concept, RAID. File Organization. Basic concept of Indexing. Transaction Management: Basic concept, ACID properties, transaction state, Implementation of atomicity & durability, Concurrent execution, Basic idea of serializability. Concurrency & Recovery: Basic idea of concurrency control, Failure Classification, storage structure-

types, stable storage implementation, data access, recovery & Atomicity: log based recovery, deferred database modification, immediate database modification, checkpoints.

UNIT-V

No. of Hours: 7

The concept of NoSQL, Brief history of NoSQL, SQL verses NoSQL, CAP Theorem (Brewer's Theorem), NoSQL pros/cons, Categories of NoSQL database, Production deployment, MongoDB, Key Features, practical with MongoDB.

Text Books:

1. Database System concepts –Henry F. Korth , Tata McGraw Hill 6th Edition.
2. NoSQL Distilled- Pramod J. Sadalage, Martin Fowler, Addison-Wesley, 2013

Reference Books:

1. Fundamentals of Database Systems, Elmasri R, Navathe S, Addison Wesley 6th Ed.
2. An introduction to database system-Bipin C. Desai
3. An introduction to Database System -C.J Date
4. SQL, PL/SQL The programming language of Oracle-Ivan Bayross

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs91
2. https://onlinecourses.swayam2.ac.in/cec19_cs05

CS-4405: DATABASE MANAGEMENT SYSTEM - PRACTICAL

Course Outcomes (COs):

CO1: Knowledge of SQL queries and relational algebra.

CO2: Apply normalization techniques for refining of databases.

CO3: Construct triggers, procedures, function, packages and cursors using PL/SQL.

CO4: Construct database for Case Study/ Real Life Problem.

CO5: Knowledge of MongoDB queries.

Course Contents

UNIT-I

No. of Hours: 8

Oracle Installation.

Create tables according to the following definition

1. Deposit (actno, cname, bname, amount, adate)
2. Branch (bname, city)
3. Customer(cname , city)
4. Borrow(loanno, cname, bname, amount)

actno	cname	bname	amount	adate
100	Anil	VRCE	1000	1-mar-2005
101	Sunil	Ajni	5000	4-jan-2006
102	Mehul	Karolbagh	3500	17-nov-2005
104	Madhuri	Chandani	1200	17-dec-2006
105	Pramod	M.G road	3000	27-mar-2006
106	Sandip	Andheri	2000	31-mar-2006
107	Shivani	Virar	1000	5-sep-2005
108	Kranti	Nehru place	5000	2-jul-2005
109	Naren	Powai	7000	10-aug-2005

Table: Deposit

bname	City
VRCE	Nagpur
Ajni	Nagpur
Karolbagh	Delhi
Chandani	Delhi
Dharmpheth	Nagpur
M.G road	Banglore
Andheri	Bombay
Virar	Bombay
Nehru place	Delhi
Powai	Bombay

Table: Branch

cname	city
Anil	Calcutta
Sunil	Delhi
Mehul	Baroda
Mandar	Patna
Mahuri	Nagpur
Pramod	Nagpur
Sandip	Surat
Shivani	Bombay
Kranti	Bombay
Naren	Bombay

Table: Customer

loanno	cname	bname	Amount
201	Anil	VRCE	1000
206	Mehul	Ajni	5000
311	Sunil	Dharampeth	3000
321	Madhuri	Andheri	2000
375	Pramod	Virar	8000
481	Kranti	Nehru place	3000

Table: Borrow

Selecting Data from Single Table

1. List all the data from table deposit.
2. List all the data from table borrow.
3. List all the data from table customer.
4. List all the data from table branch.
5. Give account no. and amount of depositors.
6. Give cname and account no. of depositors.
7. Give names of customers.
8. Give names of branches.
9. Give names of borrowers
10. Give names of customers living in city Nagpur.
11. Give names of depositors having amount greater than 4000.
12. Give account date of customer Anil.
13. Give names of all branches located in city Bombay.
14. Give names of borrower having loan no. 206.
15. Give names of depositor having account at VCRE.
16. Give names of all branches located in Delhi;
17. Give account number and deposit amount of customer having account opened between 1-12-2005 and 1-6-2005.
18. Give details of customer Anil.
19. Give name of the city where branch Karolbagh is located.

UNIT-II

No. of Hours: 8

Join and Cartesian product

1. Give name of customer having living city Bombay and branch city Delhi.
2. Give name of customers having same living city as their branch city.
3. Give names of customers who are borrowers as well as depositors and having living city Nagpur.
4. Give names of customers who are depositors and having the same branch city as that of Sunil.
5. Give the names of depositors having the same city as that of Shivani and having deposit amount greater than 2000.
6. Give the names of borrowers having deposit amount greater than 1000 and the loan amount greater than 2000.
7. Give names of depositors having the same branch as the branch of sunil.
8. Give names of borrowers having loan amount greater than the loan amount of Anil.
9. Give the names of customers living city where branch of depositor sunil is located.
10. Give loan no and loan amount of borrower having the same branch as that of depositor sunil.
11. Give loan no., loan amount, account no. and deposit amount of customers living in city Nagpur.
12. Give loan no., loan amount , account no. and deposit amount of customers having deposit branch located in Delhi
13. Give loan no., loan amount, account no., deposit amount, branch name, branch city and living city of Pramod.
14. Give deposit details and loan details of customer in the city where Pramod is living.
15. Give name of depositors having the same branch city as that of sunil and having the same living city as that of Anil.
16. Give names of depositors having amount greater than 1000 and having the same living city as Pramod.
17. Give city of customer having the same branch city as that of Pramod.
18. Give branch city and living city of Pramod.
19. Give branch city of Sunil or branch city of Anil.
20. Give the living city of Anil and living city of Sunil.

Set operations

1. List all the customers who are depositors but not borrowers.
2. List all the customers who are both depositors and borrowers.
3. List all the customers, along with their amount, who are either borrowers or depositors and living in city Nagpur.
4. List all the depositors having in all the branches where Sunil is having account.
5. List all the customers living in city Nagpur and having branch city Bombay or Delhi.
6. List all the depositors living in city Nagpur.
7. List all the depositors living in city Nagpur and having branch in city Delhi.
8. List the branch cities of Anil and Sunil.
9. List the borrowers having branch city same as that of Sunil.
10. List the customer having deposit greater than 1000 and loan less than 10000.

11. List the borrowers having branch city same as that of Sunil.
12. List the cities of depositors having branch VRCE.
13. List the depositors having the same living city as that of Sunil and the same branch city as that of Anil.
14. List the depositors having amount less than 8000 and living in the same city as Ms. Shivani.
15. List all the customers who are both depositors and borrowers and living in the same city as Anil.
16. List all the cities where branches of Anil and Sunil are located.
17. List all the customer names and the amount for depositors living in the city where either Anil or Sunil is living.
18. List the amount for the depositors living in the city where Anil is living.
19. List the cities which are either branch city of Anil or living city of Sunil.
20. List the customers who are borrowers and depositors and having living city Bombay and branch city same as that of Sandip.
21. List the customers who are both borrowers and depositors and having the same branch city as that of Anil.

UNIT-III

No. of Hours: 8

1. List total loan.
 2. List total deposit
 3. List total loan taken from Andheri branch.
 4. List total deposit of customer having account date later than 1-Jan-2006.
 5. List total deposit of customers living in city Nagpur.
 6. List maximum deposit of customer living in Bombay.
 7. List total deposit of customers having branch city Delhi.
 8. List total deposit of customers living in the city where Sunil is living.
 9. Count total number of branch cities.
 10. Count total number of customer cities.
 11. Give branch name and branch-wise deposit.
 12. Give city name and city wise deposit.
 13. Give city wise name and branch wise deposit.
 14. Give the branch wise deposit of customer after account date 1-jan -2006.
 15. Give branch wise loan of customer living in Nagpur.
 16. Count total no. of customers.
 17. Count total no. of depositor branch wise.
 18. Give maximum loan from branch VRCE.
 19. Give living city wise loan of borrowers.
 20. Give the number of customers who are depositor as well as borrowers.
- Group by and having Clause
1. List the branches having sum of deposit more than 4000.
 2. List the branches having a sum deposit more than 1000 and location in city Bombay.

3. List the names of customers having deposit in the branches where the average deposit is more than 1000.
4. List the name of customers having maximum deposit.
5. List the name of customers having maximum deposit in the customers living in Nagpur.
6. List the name of branch having highest number of depositors.
7. Count the number of depositors living in Nagpur.
8. Give the name of customers in Powai branch having more deposit than all customer VRCE branch.
9. Give names of customers in Karolbagh branch having more deposit than any other in Virar branch.
10. Give names of customers having highest deposit in the branch where Sunil is having deposit.
11. Give the highest deposit of the city where branch of Sunil is located.
12. Give names of customers having more deposit than the average deopait in their respective branches.
13. Give names of customers having maximum deposit among deposits of Nagpur for branch VRCE.
14. Give the name of branch where name of depositors is less than 2.
15. Give name and city having more customers living in than Nagpur.
16. Give names of branches having the number of depositors more than the number of borrowers.
17. Give the names of customers living in the city where the maximum numbers of depositors are located.
18. Give the name of cities in which the maximum numbers of branches are located.
19. Give the names of borrowers having the same branch city and highest borrower.
20. Count the number of customers living in the city where branch is located.

UNIT-IV

No. of Hours: 8

1. Give 10% interest to all depositors.
2. Give 10% interest to all depositors having branch vrce.
3. Give 10% interest to all depositors living in Nagpur.
4. Give 10% interest to all depositors living in Nagpur and having branch in city Bombay.
5. Add hundred rupees to the deposit of Anil and assign it to Sunil.
6. Change the deposit of VCRE branch to 1000 and change the branch as VCRE_Ambhazari.
7. Assign to all deposit of Anil the maximum deposit from VRCE branch.
8. Change the living city of VRCE branch borrowers to Nagpur.
9. Update deposit of Anil give him maximum deposit from depositors in living city Nagpur.
10. Deposit the sum of the deposits of Sunil and Vijay in an account of Anil.
11. Transfer Rs 10 from the account of Anil to the account of Sunil.
12. Transfer Rs 10 from the account of Anil to the account of Sunil if both are having the same branch.
13. Transfer Rs 10 from the account of Madhuri to the account of Parmod if both are lining in Nagpur.
14. Delete from customer.
15. Delete depositors of branches having number of customer between 1 and 3.
16. Delete branches having average deposit less than 5000.
17. Delete branches having maximum loan more than 5000.

18. Delete branches having deposit from Nagpur.
19. Delete deposit of Anil and Sunil if both are having branch Virar.
20. Delete deposit of Anil and Sunil if both are having living city Nagpur.
21. Delete deposit of Anil and Sunil if both are having same living city.
22. Delete deposit of Anil and Sunil if they are having less deposit than Vijay.
23. Delete deposit of Vijay.
24. Delete deposit of Ajay if Vijay is not a depositor.
25. Delete customer from Bombay city.
26. Delete depositors if the branch is Virar and depositor name is Ajay.
27. Delete depositors having deposit less than 500.
28. Delete borrower having loan more than 10000.
29. Delete borrower having loan more than 10000 and branch Karolbagh.
30. Delete the names of those depositors of VRCE branch who live in the city Bombay.
31. Delete borrower having branch name Chandani.
32. Delete borrower of branches having average loan less than 1000.
33. Delete borrower of branches having the minimum number of customers.

UNIT-V

No. of Hours: 8

1. PLSQL code to find sum of two numbers
2. PLSQL code to print Fibonacci series
3. PLSQL code to find whether given number is even or odd
4. Create function to display square of given number.
5. Create procedure to display division of student.
6. Create package having one procedure to add two numbers and one function to subtract two numbers.
7. Demonstrate cursor.
8. Demonstrate trigger.
9. CASE STUDY: STUDENT PROGRESS MONITORING SYSTEM
A database is to be designed for a college to monitor students' progress throughout their course of study. The students are reading for a degree (such as BA, BCA(Hons) MSc, etc) within the framework of the modular system. The college provides a number of modules, each being characterised by its code, title, credit value, module leader, teaching staff and the department they come from. A module is co-ordinated by a module leader who shares teaching duties with one or more lecturers. A lecturer may teach (and be a module leader for) more than one module. Students are free to choose any module they wish but the following rules must be observed: some modules require pre-requisites modules and some degree programmes have compulsory modules. The database is also to contain some information about students including their numbers, names, addresses, degrees they read for, and their past performance (i.e. modules taken and examination results).
10. Illustration of Where Clause, AND, OR operations in MongoDB.
11. Execute the Commands of MongoDB and operations in MongoDB : Insert, Query, Update, Delete and Projection. (Note: use any collection)

Text Books:

1. NoSQL Distilled- Pramod J. Sadalage, Martin Fowler, Addison-Wesley, 2013

2. SQL, PL/SQL The programming language of Oracle-Ivan Bayross

Reference Books:

1. Oracle Database 10g: The Complete reference, Kevin Loney.

Online Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs91

2. https://onlinecourses.swayam2.ac.in/cec19_cs05

CS-5216: DESIGN AND ANALYSIS OF ALGORITHMS

Course Outcomes (COs):

CO1: Understand complexity representation in terms of asymptotic notations.

CO2: Derive and solve recurrences describing the performance of divide-and-conquer algorithms.

CO3: Compare and analyse different data structures and sorting algorithms.

CO4: Understand and analyse major graph algorithms

CO5: Understand and apply different algorithm design paradigms such as greedy, dynamic programming, backtracking, branch and bound for effective problem-solving.

Course Contents

UNIT-I

No. of Hours: 8

Introduction: Algorithms, analysing algorithms, Complexity of algorithms, Growth of functions, Asymptotic Notations, Recurrence Relations and their Solution Methods.

UNIT-II

No. of Hours: 8

Sorting and Order Statistics - Divide-Conquer approach with Quick sort, Merge sort, Comparison of sorting algorithms, sorting in linear time (Counting, Radix, Bucket sort), Heap Sort.

Advanced Data Structures: Red-Black Tree-Properties, Insertion, B-Trees-Creation, Insertion, and Deletion. Introduction to Binomial Heaps - Merge, Union Operation, Fibonacci Heaps - Insertion, Finding Minimum Key, Union.

UNIT-III

No. of Hours: 8

Graph: Graph Traversal-Breadth First Search, Depth First Search

Greedy Methods: Fractional Knapsack, Activity Selection Problem. Minimum Spanning trees – Prim’s and Kruskal’s algorithms. Single source shortest paths - Dijkstra’s algorithms.

UNIT-IV

No. of Hours: 8

Dynamic Programming: 0/1 Knapsack, Longest Common Subsequence (LCS), Matrix Chain Multiplication. Edit distance, All pair shortest paths – Warshal’s and Floyd’s algorithms, Optimal binary search tree, Bellman-Ford algorithms.

UNIT-V

No. of Hours: 8

Backtracking: Graph Coloring, n-Queen Problem, Sum of Subset, Branch and Bound: Travelling Salesman Problem, Introduction to P, NP, NP-complete, NP-Hard.

Text Books:

1. Thomas H. Cormen, Charles E. Leiserson and Ronald L. Rivest (2022), Introduction to Algorithms, Fourth edition, Prentice Hall of India.

2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran (2008), “Fundamentals of Computer Algorithms”, Orient Longman Pvt. Ltd.

Reference Books:

1. Gilles Brassard Paul Bratley (1996), "Fundamentals of Algorithms", Prentice Hall.
2. Anany Levitin (2009), "An Introduction to Design and Analysis of Algorithms", Pearson.

Online Resources:

1. NPTEL Course Video Lectures on "Design and Analysis of Algorithms" - By Prof. Abhiram G Ranade, Prof. Ajit A Diwan, Prof. Sundar Viswanathan IIT Bombay
2. NPTEL Course Video Lectures on "Design and Analysis of Algorithms" - Prof. Madhavan Mukund, Chennai Mathematical Institute.

CS-5216: DESIGN AND ANALYSIS OF ALGORITHMS - PRACTICAL

Course Outcomes (COs):

CO1: Understand complexity representation in terms of asymptotic notations.

CO2: Derive and solve recurrences describing the performance of divide-and-conquer algorithms.

CO3: Compare and analyse different data structures and sorting algorithms.

CO4: Understand and analyse major graph algorithms.

CO5: Understand and apply different algorithm design paradigms such as greedy, dynamic programming, backtracking, branch and bound for effective problem-solving.

Course Contents

UNIT-I

No. of Hours: 8

Implement the following sorting algorithm for a given set of elements and determine the time required to sort the elements. The elements can be read from a file or can be generated using the random number generator.

- Quick sort
- Merge sort
- Counting, Radix, Bucket sort
- Heap Sort

UNIT-II

No. of Hours: 8

Write programs to implement the following data structures:

- Red-Black Tree
- B-Trees

UNIT-III

No. of Hours: 8

Write programs to print all the nodes reachable from a given starting node in a graph using the following traversal methods:

- Breadth First Search (BFS)
- Depth First Search (DFS)

Write programs to find the optimal solution for the following problems using the Greedy Method:

- Fractional Knapsack
- Activity Selection Problem.
- Minimum Spanning trees – Prim's and Kruskal's algorithms.
- Single source shortest paths - Dijkstra's algorithms

UNIT-IV

No. of Hours: 8

Write programs to find the optimal solution for the following problems using the Dynamic Programming approach:

- 0/1 Knapsack
- Longest Common Subsequence (LCS)
- Matrix Chain Multiplication
- Edit distance
- All pair shortest paths – Warshal’s and Floyd’s algorithms
- Optimal binary search tree
- Bellman-Ford algorithms

UNIT-V

No. of Hours: 8

Write programs to find the optimal solution for the following problems using the Backtracking approach:

- Graph Coloring
- n-Queen Problem
- Sum of Subset

Write a program to find the optimal solution for the following problems using the Branch and Bound method:

- Travelling Salesman Problem

Text Books:

1. Thomas H. Cormen, Charles E. Leiserson and Ronald L. Rivest (2022), Introduction to Algorithms, Fourth edition, Prentice Hall of India
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran (2008), “Fundamentals of Computer Algorithms”, Orient Longman Pvt. Ltd.

Reference Books:

1. Gilles Brassard Paul Bratley (1996),” Fundamentals of Algorithms”, Prentice Hall.
2. Anany Levitin (2009), “An Introduction to Design and Analysis of Algorithms”, Pearson.

Online Resources:

1. NPTEL Course Video Lectures on "Design and Analysis of Algorithms" - By Prof. Abhiram G Ranade, Prof. Ajit A Diwan, Prof. Sundar Viswanathan IIT Bombay
2. NPTEL Course Video Lectures on "Design and Analysis of Algorithms" - Prof. Madhavan Mukund, Chennai Mathematical Institute.

CS-5613: COMPUTER NETWORKS

Course Outcomes (COs):

- CO1:** Learn the basic concepts of data communication, computer networks and Internet.
- CO2:** Develop the problem-solving skills required for communication between devices by understanding the design issues and services of various layers in a network architecture.
- CO3:** Apply the knowledge in developing robust and secure IT applications.
- CO4:** Understanding of specific implemented protocols covering the application layer, transport layer, network layer, and link layer of the Internet (TCP/IP) stack.
- CO5:** Understanding security issues.
-

Course Contents

UNIT-I

No. of Hours: 8

Introduction: Overview, Applications of Computer Networks; Network Hardware – LAN, MAN, WAN and topologies; LAN components – File server, Workstations, Network Adapter Cards; Network Software - Protocol hierarchies, Design issues for the layers, Connection Oriented and Connection less services, Service primitives, Relationship between Services and Protocols; Switching Techniques – Circuit Switching and Packet Switching; Reference models – OSI and TCP/IP, comparison and critique of OSI and TCP/IP reference models.

Physical Layer: data communication fundamentals – Signal, bandwidth, data rate, modulation; Guided Transmission media – Twisted pair, Coaxial Cable and Fibre Optics; Wireless transmission – EM spectrum, radio transmission, Microwave transmission, Infrared transmission.

UNIT-II

No. of Hours: 8

Data Link Layer: Design issues – Services, Framing, Error Control and Flow Control; Error Detection Techniques – Parity Check and Cyclic Redundancy Check (CRC); Error Correction Technique – Hamming code; Elementary Data Link Protocols – Unrestricted Simplex Protocol, Simplex Stop-and-Wait Protocol, Sliding Window protocols:- One-Bit Sliding Window Protocol, protocol using Go Back N and Selective Repeat; HDLC protocol; Data link layer in the Internet – PPP.

UNIT-III

No. of Hours: 8

Medium Access Sublayer: Channel Allocation problem; Multiple access protocols: Pure Aloha, Slotted Aloha, CSMA Protocols, CSMA/CD, Collision-Free Protocols, CSMA/CA; IEEE MAC Sublayer protocols - 802.2, 802.3; High speed LANs – Fast Ethernet, FDDI; Wireless LANs: IEEE 802.11, IEEE 802.16; Data Link Layer Switching – Bridges and Switches, their difference with Repeaters, Hubs, Routers and Gateways.

UNIT-IV

No. of Hours: 8

Network Layer: Design issues; Routing algorithms - Optimality Principle, Shortest Path Routing, Flooding, Distance Vector Routing, Link State Routing, Hierarchical Routing, Broadcasting Routing, Multicast Routing; Internetworking; The Network Layer in the Internet - Internet Protocol, Internet addressing and Internet Control protocols.

UNIT-V

No. of Hours: 8

Transport Layer: Transport Service; Elements of transport protocols - Addressing, Connection establishment, Connection release, Flow control and Buffering, Multiplexing; The Internet Transport Protocols - UDP and TCP. Application layer: Client Server Architecture, DNS, WWW and HTTP, E-mail Protocols (SMTP, POP3, IMAP, MIME), FTP, TELNET.

Network Security: Cryptography, Symmetric Key Algorithms, Public key Algorithms and Digital Signatures.

Text Books:

1. Andrew S. Tanenbaum, "Computer Networks", Fourth Edition, Pearson Education.

Reference Books:

1. Behrouz A. Forouzan , "Data Communications and Networking", Fourth Edition, McGraw-Hill Publication.

2. William Stallings, "Data and Computer Communications", Eighth Edition, Pearson Education.

3. James F. Kurose & Keith W. Rose "Computer Networking: A Top Down Approach", Fourth Edition, Pearson Education.

4. Douglas E. Comer, "Computer Networks and Internets", Fourth Edition, Pearson Education.

Online Resources:

CS-5617: INTERNET AND WEB TECHNOLOGY

Course Outcomes (COs):

- CO1:** Use syntax and semantics of java programming language.
- CO2:** Develop reusable programs using the concepts of inheritance, polymorphism, interfaces and packages.
- CO3:** Apply the concepts of Multithreading and Exception handling to develop efficient and error free codes.
- CO4:** Developing application using advance java concepts.
- CO5:** Understanding the concepts of MVC architecture.
-

Course Contents

UNIT-I

No. of Hours: 8

Features of Java, Object-oriented programming overview, Introduction of Java Technologies, How to write simple Java programs, Data Types, Variables, Memory concepts, control statements, looping, Method CallStack and Activation Record, Argument Promotion and Casting, Scope of declaration and Method Overloading, StringHandling: The String constructors, String operators, Character Extraction, String comparison, String Buffer. Arrays: Declaring and Creating Arrays, Enhanced for Statement, Passing Arrays to Method, Multidimensional Arrays, Variable-Length Argument lists, Using Command-line Arguments

UNIT-II

No. of Hours: 8

Inheritance: Extending classes & related things, Packages and Interfaces: Defining a Package, Understanding CLASSPATH, Access Protection, Importing packages, creating own packages Exception Handling: Introduction, overview of doing it and keywords used, when to use it, Multithreading: What are threads, The java Thread model, Thread priorities, Thread life cycle, Thread Synchronization, Applets: Applet basics, Applet Architecture, Applet life cycle methods, Database connectivity: JDBC, The design of JDBC, Typical uses of JDBC

UNIT-III

No. of Hours: 8

Introduction to HTTP, webServer and application Servers, Installation of Application servers, Config files, Web.xml. JavaServlet, Servlet Development Process, Deployment Descriptors, Generic Servlet, Lifecycle of Servlet, Servlet Packages, Classes, Interfaces, and Methods, Handling Forms with Servlet, Various methods of Session Handling, various elements of deployment descriptors.

UNIT-IV

No. of Hours: 8

JSP Basics: JSP lifecycle, Directives, scripting elements, standard actions, implicit objects. Connection of JSP and Servlet with different database viz. Oracle, MS-SQL Server, MySQL, java.sql Package, Querying database, adding records, deleting records, modifying records, types of Statement,

Separating Business Logic and Presentation Logic, Building and using JavaBean, Session handling in JSP, Types of errors and exceptions handling.

UNIT-V

No. of Hours: 8

MVC Architecture Introduction to Remote Method Invocation, Introduction to Enterprise Java Bean, Types of EJB, Creating and working with Session Bean

Text Books:

1. Java2:The Complete Reference by Herbert Schildt, Tata McGraw-Hill, 8th Edition, 2011.
2. K. Mukhar, "Beginning Java EE 5: From Novice to Professional", Wrox Press.

Reference Books:

1. The Java Programming Language, Ken Arnold, James Gosling, David Holmes, 3rd Edition, Person Education, 2000
2. Head First Java, Kathy Sierra, Bert Bates, O'Reilly Publication, 2nd Edition, 2005

Online Resources:

1. NPTEL Course Video Lectures on "Programming in java", By Prof. Debasis Samant, IIT Kharagpur
2. Swayam Course Video Lectures on "Web Technology", By Prof. Dr. Ashutosh Kumar Bhatt

CS-5617: INTERNET AND WEB TECHNOLOGY-PRACTICAL

Course Outcomes (COs):

CO1: Able to write real world problem using core java concepts

CO2: Apply java concepts like inheritance, polymorphism, interfaces and packages in programming.

CO3: Apply the concepts of Multithreading and Exception handling to develop efficient and error free codes

CO4: Developing web application using advance java concepts like jdbc, servlet and jsp.

CO5: Understanding the concepts of MVC architecture.

Course Contents

UNIT-I

No. of Hours: 8

1. Write a Java program to print 'Hello' on screen and your name on a separate line.
2. Write a Java program to print the sum of two numbers.
3. Write a Java program to divide two numbers and print them on the screen.
4. Write a Java program that takes two numbers as input and displays the product of two numbers.
5. Write a Java program to print the sum (addition), multiply, subtract, divide and remainder of two numbers.
6. Write a Java program to print the area and perimeter of a circle.
7. Write a Java program that takes three numbers as input to calculate and print the average of the numbers.
8. Write a Java program to swap two variables values.
9. Write a Java program to swap two variables values without using third variable.
10. Write a Java program to check whether Java is installed on your computer.
11. Write a Java program to get a number from the user and print whether it is positive or negative.
12. Write a Java program that takes three numbers from the user and prints the greatest number.
13. Write a Java program to find the number of days in a month.
14. Write a Java program that requires the user to enter a single character from the alphabet. Print Vowel or Consonant, depending on user input. If the user input is not a letter (between a and z or A and Z), or is a string of length > 1, print an error message.
15. Write a Java program that takes a year from the user and prints whether it is a leap year or not.
16. Write a program to check whether a number is divisible by 5 and 11 or not.
17. Write a program to check whether a number is even or odd.

18. Write a program to input marks of five subjects Physics, Chemistry, Biology, Mathematics and Computer. Calculate percentage and grade.
19. Write a program to input basic salary of an employee and calculate its Gross salary according to following:
- Basic Salary \leq 10000 : HRA = 20%, DA = 80%
Basic Salary \leq 20000 : HRA = 25%, DA = 90%
Basic Salary $>$ 20000 : HRA = 30%, DA = 95%
20. Write a program to input electricity unit charges and calculate total electricity bill according to the given condition:
- For first 50 units Rs. 0.50/unit
For next 150 units Rs. 0.75/unit
For next 250 units Rs. 1.20/unit
For unit above 250 Rs. 1.50/unit
An additional surcharge of 20% is added to the bill
21. If his basic salary is less than Rs. 1500, then HRA = 10% of basic salary and DA = 90% of basic salary. If his salary is either equal to or above Rs. 1500, then HRA = Rs. 500 and DA = 98% of basic salary. If the employee's salary is input through the keyboard write a program to find his gross salary. Write a program to calculate the salary as per the following table
Taken From Original file

UNIT-II

No. of Hours: 8

1. Write a java program to create class called Employee with methods called work() and getSalary(). Create a subclass called HRManager that overrides the work() method and adds a new method called addEmployee().
2. Write a java program to create a class known as Person with methods called getFirstName() and getLastName(). Create a subclass called Employee that adds a new method named getEmployeeID() and override the getLastName() methods to include the employee's job title.
3. Write a java program to create a class known as "BankAccount" with methods called deposit() and withdraw(). Create a subclass called SavingsAccount that overrides the withdraw() method to prevent withdraws if the account balance falls below one hundred.
4. Write a java program to create interface shape with getArea() method. Create three classes Rectangle, Circle and Triangle that implements the shape interface. Implement the getArea() method for each of the three classes.
5. Write a java program to create interface sortable with a method sort() that sorts an array of integers in ascending order. Create two classes Bubblesort and Selectionsort that implement the sortable interface and provide their own implementations of the sort() method.
6. Write a Java program to create an abstract class Animal with an abstract method called sound(). Create subclasses Lion and Tiger that extend the Animal class and implement the sound() method to make a specific sound for each animal.
7. Write a Java program to create an abstract class Employee with abstract methods calculateSalary() and displayInfo(). Create subclasses Manager and Programmer that extend the

Employee class and implement the respective methods to calculate salary and display information for each role

8. Write a Java program to create an abstract class Employee with abstract methods calculateSalary() and displayInfo(). Create subclasses Manager and Programmer that extend the Employee class and implement the respective methods to calculate salary and display information for each role
9. Write a Java program to create a class called "Cat" with instance variables name and age. Implement a default constructor that initializes the name to "Unknown" and the age to 0. Print the values of the variables.
10. Write a Java program to create a class called Student with instance variables studentId, studentName, and grade. Implement a default constructor and a parameterized constructor that takes all three instance variables. Use constructor chaining to initialize the variables. Print the values of the variables
11. Write a Java program to create a basic Java thread that prints "Hello, World!" when executed
12. Write a Java program that performs matrix multiplication using multiple threads.
13. Write a Java program that creates a bank account with concurrent deposits and withdrawals using threads.
14. Write a Java program to create a producer-consumer scenario using the wait() and notify() methods for thread synchronization.
15. Write a Java program to demonstrate Semaphore usage for thread synchronization.
16. Write a simple applet program to display "Hello Applet" Message.
17. Write a applet program to show its life cycle methods.
18. Write a applet program for revolving banner.
19. Connect database to Java program
20. Program to create database table using Java
21. Java Program to insert, update, delete & select records
22. Program to delete record from database
23. Program to execute batch of SQL statements
24. Program to insert date in oracle database
25. Program to insert & read image
26. Program to commit & rollback
27. Program to execute SQL select query

UNIT-III

No. of Hours: 8

1. Write a program to create a servlet that displays the welcome message.
2. Create a simple interest calculator web page using a servlet.
3. Develop a dynamic web application using servlets to create and display cookies.
4. Create a registration form and display the values entered by the user in another page using servlets.
5. Write a program using servlets to display data from a database table.

6. Display a servlet that fetches init parameters from web.xml displaying user information.
7. Write a program using servlets to create and display context parameters.
8. Develop a dynamic web application using servlets to demonstrate doGet(), and doPost() methods.
9. Write an application using servlets to show how forward() method works.
10. Also, demonstrate the use of include() method in servlets.
11. Furthermore, show how sendRedirect() method works using servlets.
12. In order to demonstrate the working of servlet filters create a dynamic web application.

UNIT-IV

No. of Hours: 8

1. Write a program in JSP to display the given text in a paragraph in different font sizes using a loop.
2. Create a dynamic web application in JSP to find the grade according to the marks entered by the user using the switch statement.
3. Build an application in JSP to calculate factorial of a number using recursion.
4. Write a program to create a JSP method that accepts a number and returns its double.
5. Use Page directive in JSP to display the current date.
6. Create a summation method in JSP that accepts an array and display the sum of its element.
7. Write a program to create a JSP page called alert message JSP that displays alert messages if username & password fields are blank and displays welcome user otherwise.
8. Develop an application to demonstrate the include directive in JSP to display the current date from another JSP page.
9. Write a program in JSP to display the properties of an object using the useBean directive.
10. Build an application in JSP that redirects to another page.

UNIT-V

No. of Hours: 8

1. Write java program to demonstrate the functionality of RMI.
2. Write java program to show the use of EJB.
3. Write java program to demonstrate the functionality of MVC.
4. Write java program to show the use of session.

Text Books:

1. Java2:The Complete Reference by HerbertSchildt,TataMcGraw-Hill,8th Edition, 2011.
2. K.Mukhar,“BeginningJavaEE5:From Novice to Professional”, WroxPress.

Reference Books:

1. The Java Programming Language, Ken Arnold, James Gosling, David Holmes, 3rd Edition, Person Education, 2000
2. Head First Java,Kathy, Sierra, Bert Bates, O’Reilly Publication, 2ndEdition,2005

Online Resources:

1. NPTEL Course Video Lectures on "Programming in java" , By Prof. Debasis Samant, IIT Kharagpur
2. Swayam Course Video Lectures on "Web Technology", By Prof. Dr. Ashutosh Kumar Bhatt