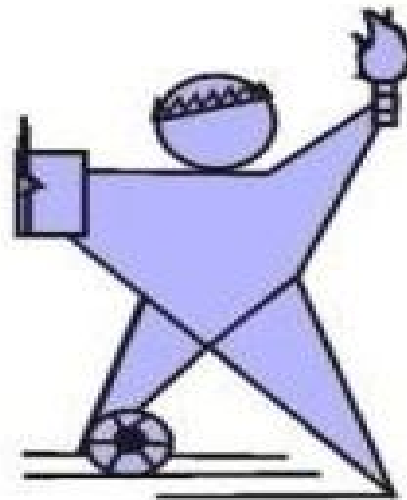


Proposed Scheme & Syllabus of UG Engineering Program

Bachelor of Technology (B.Tech.)

Computer Science & Engineering

2020-21



IPS ACADEMY
INSTITUTE OF ENGINEERING & SCIENCE, INDORE
(A UGC Autonomous Institute affiliated to RGPV)

Name of the Department: **Computer Science & Engineering**

V Semester

**Bachelor of Technology (B.Tech) Computer Science and Engineering/
Computer Engineering/Computer Science & Technology]**

(w.e.f. July, 2019)

S.No.	Subject Code	Category	Subject Name	Maximum Marks Allotted					Total Marks	Contact Hours per week			Total Credits
				Theory			Practical			L	T	P	
				End Sem.	Mid Sem. Exam.	Quiz/ Assignment	End Sem	Term work Lab Work & Sessional					
1.	CS-501	DC	Theory of Computation	70	20	10	30	20	150	3	-	2	4
2.	CS-502	DC	Database Management Systems	70	20	10	30	20	150	3	-	2	4
3.	CS -503	DE	Departmental Elective	70	20	10	-	-	100	3	1	-	4
4.	CS -504	OE	Open Elective	70	20	10	-	-	100	3	-	-	3
5.	CS -505	D Lab	Lab (Linux)	-	-	-	30	20	50	-	-	4	2
6.	CS -506	O/E Lab	Lab (Python)	-	-	-	30	20	50	-	-	4	2
7.	CS -507	IN	Evaluation of Internship-II	-	-	-	-	100	100	-	-	6	3
8.		IN	Internship - III	To be completed anytime during Fifth/Sixth semester. Its evaluation/credit to be added in Seventh Semester.									
9.	CS -508	P	Minor Project- I	-	-	-	-	50	50			4	2
10.	Additional Credits [#]	<i>[#]Additional credits can be earned through successful completion of credit based MOOC's Courses available on SWAYAM platform (MHRD) at respective UG level.</i>											
			Total	280	80	40	120	230	750	12	1	22	24

Departmental Electives	Open Electives
CS 503 (A) Data Analytics	CS 504 (A) Internet and Web Technology *
CS 503 (B) Pattern Recognition	CS 504 (B) Object Oriented Programming **
CS 503 (C) Cyber Security	CS 504 (C) Introduction to Database Management Systems**

1 Hr Lecture	1 Hr Tutorial	2 Hr Practical
1 Credit	1 Credit	1 Credit

Computer Science and Engineering, V-Semester
CS501 Theory of Computation

Course Objective

1. To understand computability, decidability, and complexity through problem solving.
2. To analyze and design abstract model of computation & formal languages
3. To understand and conduct mathematical proofs for computation and algorithms.

Course Outcomes

1. Explain the basic concepts of switching and finite automata theory & languages.
2. Relate practical problems to languages, automata, computability and complexity.
3. Construct abstract models of computing, check their power to recognize the languages and analyze the grammar, its types, simplification and normal form.
4. Interpret rigorously formal mathematical methods to prove properties of languages, grammars and automata.
5. Develop an overview of how automata theory, languages and computation are applicable in engineering application.

Course Contents:

UNIT-I

Introduction of Automata Theory: Examples of automata machines, Finite Automata as a language acceptor and translator, Moore machines and mealy machines, composite machine, Conversion from Mealy to Moore and vice versa.

UNIT-II

Types of Finite Automata: Non-Deterministic Finite Automata (NFA), Deterministic finite automata machines, conversion of NFA to DFA, minimization of automata machines, regular expression, Arden's theorem. Meaning of union, intersection, concatenation and closure, 2-way DFA.

UNIT-III

Grammars: Types of grammar, context sensitive grammar, and context free grammar, regular grammar. Derivation trees, ambiguity in grammar, simplification of context free grammar, conversion of grammar to automata machine and vice versa, Chomsky hierarchy of grammar, killing null and unit productions. Chomsky normal form and Greibach normal form.

UNIT-IV

Push down Automata: example of PDA, deterministic and non-deterministic PDA, conversion of PDA into context free grammar and vice versa, CFG equivalent to PDA, Petrinet model.

UNIT-V

Turing Machine: Techniques for construction. Universal Turing machine Multitap, multihead and multidimensional Turing machine, N-P complete problems. Decidability and Recursively Enumerable Languages, decidability, decidable languages, undecidable languages, Halting problem of Turing machine & the post correspondence problem.

Recommended Books

1. Introduction to Automata Theory Language & Computation, Hopcroft & Ullman, Narosa Publication.
2. Element of the Theory Computation, Lewis & Christors, Pearson.
3. Theory of Computation, Chandrasekhar & Mishra, PHI.
4. Theory of Computation, Wood, Harper & Row.
5. Introduction to Computing Theory, Daniel I-A Cohen, Wiley.

List of Experiments:

1. Design a Program for creating machine that accepts three consecutive one.
2. Design a Program for creating machine that accepts the string always ending with 101.
3. Design a Program for Mode 3 Machine
4. Design a program for accepting decimal number divisible by 2.
5. Design a program for creating a machine which accepts string having equal no. of 1's and 0's.
6. Design a program for creating a machine which count number of 1's and 0's in a given string.
7. Design a Program to find 2's complement of a given binary number.
8. Design a Program which will increment the given binary number by 1.
9. Design a Program to convert NDFA to DFA.
10. Design a Program to create PDA machine that accept the well-formed parenthesis.
11. Design a PDA to accept WCWR where w is any string and WR is reverse of that string and C is a Special symbol.
12. Design a Turing machine that's accepts the following language $anbnc^n$ where $n > 0$.

Computer Science and Engineering, V-Semester

CS -502 Database Management System

Course Objectives:

To understand the fundamental concepts of database management. These concepts include aspects of database design, database languages, and database-system implementation. To give systematic database design approaches covering conceptual design, logical design and an overview of physical design are familiar with the basic issues of transaction processing and concurrency control.

Course Outcomes:

1. Describe basic concepts of DBMS and Explain ER model.
2. Solve queries using Relational Algebra, Relational Calculus and SQL.
3. Explain database schema and discuss the Query optimization methods.
4. Describe transaction processing, concurrency control and recovery technique.
5. Analyze the Various DBMS software like Oracle, SQL/PL SQL etc.

Course Contents:

UNIT I

DBMS Concepts and architecture Introduction, Database approach v/s Traditional file accessing approach, Advantages, of database systems, Data models, Schemas and instances, Data independence, Data Base Language and interfaces, Overall Database Structure, Functions of DBA and designer, ER data model: Entities and attributes, Entity types, Defining the E-R diagram, Concept of Generalization, Aggregation and Specialization. transforming ER diagram into the tables. Various other data models object-oriented data Model, Network data model, and Relational data model, Comparison between the three types of models.

UNIT II

Relational Data models: Domains, Tuples, Attributes, Relations, Characteristics of relations, Keys, Key attributes of relation, Relational database, Schemas, Integrity constraints. Referential integrity, Relational Query languages: SQL-DDL, DML, integrity constraints, Complex queries, various joins, Relational algebra and relational calculus, Relational algebra operations like select, Project, Join, Division, outer union. Types of relational calculus i.e. Tuple oriented and domain oriented relational calculus and its

operations.

UNIT III

Data Base Design: Introduction to normalization, Normal forms, Functional dependency, Decomposition, Dependency preservation and lossless join, problems with null valued and dangling tuples, multivalued dependencies. Query Optimization: Introduction, steps of optimization, various algorithms to implement select, project and join operations of relational algebra, optimization methods: heuristic based, cost estimation based.

UNIT IV

Transaction Processing Concepts: -Transaction system, Testing of Serializability, Serializability of schedules, conflict & view serializable schedule, recoverability, Recovery from transaction failures. Log based recovery. Checkpoints deadlock handling. Concurrency Control Techniques: Concurrency Control, locking Techniques for concurrency control, time stamping protocols for concurrency control, validation based protocol, multiple granularity. Multi version schemes, Recovery with concurrent transaction.

UNIT V

Study of Relational Database Management Systems through Oracle/PL SQL

QL/MySQL: Architecture, physical files, memory structures, background process. Concept of table spaces, segments, extents and block. Dedicated server, multi threaded server. SQL queries, Data extraction from single, multiple tables equi- join, non equi-join, self - join, outer join. Usage of like, any, all, exists, in Special operators. Cursor management: nested and parameterized cursors, Oracle exception handling mechanism. Stored procedures, in, out, in out type parameters, usage of parameters in procedures. User defined functions their limitations. Triggers, mutating errors, instead of triggers.

Recommended Books

1. Date C J, "An Introduction To Database System", Pearson Educations
2. Korth, Silbertz, Sudarshan, "Fundamental of Database System", McGraw Hill
3. Rob, "DataBaseSystem:DesignImplementation&Management", Cengage Learning
4. Elmasri, Navathe, "FundamentalsOfDatabaseSystems", Pearson Educations
5. AtulKahate, "IntroductiontoDatabaseManagementSystem", Pearson Educations
6. Oracle9i Database Administration Fundamental-I, Volume-I, Oracle Press, TMH.
7. Paneerselvam, "Data Base Management System", PHI Learning

List of Experiments

1. Introduction to Oracle and SQL
2. Write the queries for Data Definition language(DDL)
3. Write the queries for Data manipulation language(DML)
4. Use of various types of Integrity Constraints
5. Write the queries for Data Control language(DCL)
6. Use of SELECT command with different clauses.
7. Write SQL queries using logical operation (AND, OR, NOT)
8. Write SQL queries for aggregate functions (Max, Min, Sum, Avg, Count)
9. Write SQL queries for group by and Having
10. Write SQL queries for sub queries and nested queries
11. Write an SQL query to implement JOINS
12. Write SQL queries to create views
13. Write program by the use of PL/SQL
14. Design and implementation of any Data base system (like Banking, University etc).

Computer Science and Engineering, V-Semester
Departmental Elective CS- 503 (A) Data Analytics

Course Objective:

To provide an overview of an exciting growing field of big data analytics. To introduce the tools required to manage and analyze big data like Hadoop, MapReduce. To teach the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability. To enable students to have skills that will help them to solve complex real-world problems in for decision support.

Course Outcomes:

1. Understand the basic of data analytics using concepts of statistics and probability, regression.
2. Acquire fundamental enabling techniques and scalable algorithms like Hadoop, Map Reduce in big data analytics.
3. Understand the needs of data processing techniques.
4. Implement the data analytics techniques using Hadoop Map Reduce.
5. Apply the data analytics techniques and tools in real life applications like database, Hive.

Course Contents:

UNIT I

Descriptive Statistics: Probability Distributions, Inferential Statistics, Inferential Statistics through hypothesis tests Regression & ANOVA, Regression ANOVA(Analysis of Variance)

UNIT II

Introduction To Big Data: Big Data and its Importance, Four V's of Big Data, Drivers for Big Data, Introduction to Big Data Analytics, Big Data Analytics applications. Big Data Technologies: Hadoop's Parallel World, Data discovery, Open source technology for Big Data Analytics, cloud and Big Data, Predictive Analytics, Mobile Business Intelligence and Big Data, Crowd Sourcing Analytics, Inter- and Trans-Firewall Analytics, Information Management.

UNIT III

Processing Big Data: Integrating disparate data stores, Mapping data to the programming framework, Connecting and extracting data from storage, Transforming data for processing, subdividing data in preparation for Hadoop Map Reduce.

UNIT IV

Hadoop Map reduce: Employing Hadoop Map Reduce, Creating the components of Hadoop Map Reduce jobs, Distributing data processing across server farms, Executing Hadoop Map Reduce jobs, monitoring the progress of job flows, The Building Blocks of Hadoop Map Reduce Distinguishing Hadoop daemons, Investigating the Hadoop Distributed File System Selecting appropriate execution modes: local, pseudo-distributed, fully distributed.

UNIT V

Big Data Tools And Techniques: Installing and Running Pig, Comparison with Databases, Pig Latin, User- Define Functions, Data Processing Operators, Installing and Running Hive, Hive QL, Querying Data, User-Defined Functions, Oracle Big Data.

Recommended Books

1. Michael Minelli, Michehe Chambers, “Big Data, Big Analytics: Emerging Business IntelligenceandAnalyticTrendsforToday’sBusiness”,1stEdition,AmbigaDhiraj,WielyCIO Series,2013.
2. ArvindSathi,“BigDataAnalytics:DisruptiveTechnologiesforChangingtheGame”,1stEdition , IBM Corporation, 2012.1. Rajaraman, A., Ullman, J. D., Mining of Massive Datasets, Cambridge University Press, United Kingdom,2012
3. Berman, J.J., Principles of Big Data: Preparing, Sharing and Analyzing Complex
4. Information,MorganKaufmann,2014
5. Barlow,M.,Real-TimeBigDataAnalytics:EmergingArchitecture,OReilly,2013
6. Schonberger,V.M.,KennethCukier,K.,BigData,JohnMurrayPublishers,2013
7. BillFranks,“TamingtheBigDataTidalWave:FindingOpportunitiesinHugeData
8. StreamswithAdvancedAnalytics”,1stEdition,WileyandSASBusinessSeries,2012.

Computer Science and Engineering, V-Semester
Departmental Elective CS- 503 (B) Pattern Recognition

Course Objective:

To introduce the fundamental algorithms for pattern recognition and instigate the various classification and clustering techniques. Students will learn the fundamentals of pattern recognition and its relevance to classical and modern problems. To be able to identify where, when and how pattern recognition can be applied.

Course Outcomes:

1. Understand the concept of a pattern and the basic approach to the development of pattern recognition
2. Acquire the knowledge of classification, its types and normalization.
3. Understand unsupervised learning & clustering in pattern recognition
4. Understand the basic methods of feature extraction, feature evaluation, and data mining
5. Understand soft computing and fuzzy classification for recent advancements in pattern recognition.

Course Contents:

UNIT-I

Introduction – Definitions, data sets for Pattern, Application Areas and Examples of pattern recognition, Design principles of pattern recognition system, Classification and clustering, supervised Learning, unsupervised learning and adaptation, Pattern recognition approaches, Decision Boundaries, Decision region , Metric spaces, distances.

UNIT -II

Classification: introduction, application of classification, types of classification, decision tree, naïve Bayes, logistic regression , support vector machine, random forest, K Nearest Neighbor Classifier and variants, Efficient algorithms for nearest neighbor classification, Different Approaches to Prototype Selection, Combination of Classifiers, Training set, test set, standardization and normalization.

UNIT – III

Different Paradigms of Pattern Recognition, Representations of Patterns and Classes, Unsupervised Learning & Clustering: Criterion functions for clustering, Clustering Techniques: Iterative square -error partitional clustering – K means, hierarchical clustering,

Cluster validation.

UNIT -IV

introduction of feature extraction and feature selection, types of feature extraction , Problem statement and Uses, Algorithms - Branch and bound algorithm, sequential forward / backward selection algorithms, (l, r) algorithm.

UNIT -V

Recent advances in Pattern Recognition, Structural PR, SVMs, FCM, Soft computing and Neuro-fuzzy techniques, and real-life examples, Histograms rules, Density Estimation, Nearest Neighbor Rule, Fuzzy classification.

Recommended Books:

1. Richard O. Duda, Peter E. Hart and David G. Stork, "Pattern Classification", 2nd Edition, John Wiley,2006.
2. C.M. Bishop, "PatternRecognitionandMachineLearning",Springer,2009.
3. S. Theodoridis and. Koutroumbas, "Pattern Recognition", 4th Edition, academic Press,2009.
4. Robert Schalk off, "pattern Recognition: statistical, structural and neural approaches", JohnWiley&sons,Inc,2007.

Computer Science and Engineering, V-Semester
Departmental Elective CS- 503 (C) Cyber Security

Course Objective:

Analyze and resolve security issues in networks and computer systems to secure an IT infrastructure. Design, develop, test and evaluate secure software. Develop policies and procedures to manage enterprise security risks.

Course Outcomes:

1. Define and explain the concepts of cyber crime and its classification.
2. Delineate the components online frauds, intrusions, virtual crimes and hacking.
3. Knowledge of different act's in cyber security
4. List the various parts of IT act related to electronic records.
5. Knowledge of different Cyber Security tools.

Course Contents:

UNIT-I

Introduction of Cyber Crime, Challenges of cyber crime, Classifications of Cybercrimes: E-Mail Spoofing, Spamming, Internet Time Theft, Salami attack/Salami Technique,

UNIT-II

Web jacking, Online Frauds, Software Piracy, Computer Network Intrusions, Password Sniffing, Identity Theft, cyber terrorism, Virtual Crime, Perception of cyber criminals: hackers, insurgents and extremist group etc. Web servers were hacking, session hijacking.

UNIT-III

Cyber Crime and Criminal justice: Concept of Cyber Crime and the IT Act, 2000, Hacking, Teenage Web Vandals, Cyber Fraud and Cheating, Defamation, Harassment and E-mail Abuse, Other IT Act Offences, Monetary Penalties, jurisdiction and Cyber Crimes, Nature of Criminality, Strategies to tackle Cyber Crime and Trends.

UNIT-IV

The Indian Evidence Act of 1872 v. Information Technology Act, 2000: Status of Electronic Records as Evidence, Proof and Management of Electronic Records; Relevancy, Admissibility and Probative Value of E-Evidence, Proving Digital Signatures, Proof of Electronic Agreements, Proving Electronic Messages.

UNIT-V

Tools and Methods in Cybercrime: Proxy Servers and Anonymizers, Password Cracking, Key loggers and Spyware, virus and worms, Trojan Horses, Backdoors, DoS and DDoS Attacks , Buffer and Overflow, Attack on Wireless Networks, Phishing : Method of Phishing, Phishing Techniques. Introduction to KALI Linux

Recommended Books:

1. Principles of Cyber crime, Jonathan Clough Cambridge University Press
2. John R. Vacca, Computer Forensics: Computer Crime Scene Investigation, 2ndEdition, CharlesRiverMedia,2005
3. Cyber Law Simplified, Vivek Sood, Pub: TMH.
4. Cyber Security by Nina Godbole, Sunit Belapure Pub: Wiley-India
5. Information Warfare: Corporate attack and defense in digital world, William Hutchinson, Mathew Warren, Elsevier.
6. Cyber Laws and IT Protection, Harish Chander, Pub: PHI.

Computer Science and Engineering, V-Semester
Open Elective CS- 504 (A) Internet and Web Technology

Course Objective:

The students will be able to: • Analyze a web page and identify its elements and attributes. Create web pages using XHTML and Cascading Style Sheets. Build dynamic web pages using JavaScript, PHP and MySQL. Create XML documents and Schemas.

Course Outcomes:

1. Describe the concepts of WWW including browser and HTTP protocol.
2. List the various HTML tags and use them to develop the user-friendly web pages.
3. Define the CSS with its types and use them to provide the styles to the web pages at various levels.
4. Develop the modern web pages using the XML, HTML, CSS along with php as Per need of applications.
5. Use server side scripting with PHP to generate the web pages dynamically using the database connectivity.

Course Contents:

UNIT-I

Introduction: Concept of WWW, Internet and WWW, HTTP Protocol : Request and Response, Web browser and Web servers, Features of Web 2.0 Web Design: Concepts of effective web design, Web design issues including Browser, Bandwidth and Cache, Display resolution, Look and Feel of the Web site, Page Layout and linking, User centric design, Sitemap, Planning and publishing website, Designing effective navigation.

UNIT-II

HTML :Basics of HTML, formatting and fonts, commenting code, color, hyperlink, lists, tables, images, forms, XHTML, Meta tags, Character entities, frames and frame sets, Browser architecture and Web site structure. Overview and features of HTML5

UNIT-III

Style sheets : Need for CSS, introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, manipulating texts, using fonts, borders and boxes, margins, padding lists, positioning using CSS, CSS2, Overview and features of CSS3

JavaScript : Client side scripting with JavaScript, variables, functions, conditions, loops

and

repetition, Pop up boxes, Advance JavaScript: Javascript and objects, JavaScript own objects, the DOM and web browser environments, Manipulation using DOM, forms and validations, HTML : Combining HTML, CSS and JavaScript, Events and buttons

UNIT-IV

XML : Introduction to XML, uses of XML, simple XML, XML key components, DTD and Schemas, Using XML with application. Transforming XML using XSL and XSLT PHP: Introduction and basic syntax of PHP, decision and looping with examples, PHP and HTML, Arrays, Functions, Browser control and detection, string, Form processing, Files, Advance Features: Cookies and Sessions, Object Oriented Programming with PHP

UNIT-V

PHP and MySQL: Basic commands with PHP examples, Connection to server, creating database, selecting a database, listing database, listing table names, creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHP my admin and database bugs, Introduction of Bootstrap and jQuery

Recommended Books:

1. Developing Web Applications, Ralph Moseley and M. T. Savaliya, Wiley-India
2. Web Technologies, Black Book, dreamtech Press
3. HTML 5, Black Book, dreamtech Press
4. Web Design, Joel Sklar, Cengage Learning
5. Developing Web Applications in PHP and AJAX, Harwani, McGraw Hill
6. Internet and World Wide Web How to program, P.J. Deitel & H.M. Deitel, Pearson

Computer Science and Engineering, V-Semester
Open Elective CS- 504 (B) Object Oriented Programming

Course Objective:

Students will get the knowledge programming basics with data types, control structure and array. Also understand the concepts of object-oriented programming to students with a background in the procedural paradigm.

Course Outcomes:

1. State the basic concept of programming
2. Identify and demonstrate concept of decision control structure and operators.
3. Demonstrate the working of different types of arrays.
4. Analyze structural programming and its use.
5. the fundamental concepts in object-oriented approach.

Course Contents:

UNIT-I

Basics of programming: Character set, Constants, Variables, keywords, identifiers literals. Instructions: Type Declaration Instruction, arithmetic Integer Long Short, Signed unsigned, storage classes, Integer and Float Conversions, type conversion in assignment, hierarchy of operations.

UNIT –II

Decision control structure: control instructions, if, if-else, use of logical operator, hierarchy of logical operators, arithmetic operators, relational operators, assignment operators, increment and decrement operators, conditional operators, bit wise operators, special operators, "&,*,,>," "sizeof" Loops control structure: while loop, for loop, do – while loop, odd loop, nested loop, break, continue, case control structure, go to, exit statement.

UNIT-III

Array: what are arrays , array initialization, bound checking 1D array, 2D array initialization of 1D and 2D array, memory map of 1D and 2D array, Multidimensional array. Strings: what are strings, standard library string function strlen(), strcpy(), strcat(), strcmp(), 2D array of characters.

UNIT-IV

Structure: Why use structure, declaration of structure, accessing structure elements, how structure elements are stored, array of structure, uses of structure. Preprocessor: features of Preprocessor, macro expansion, macro with arguments, file inclusion, conditional, #if, #elif, miscellaneous directives, #include, #define, directives, #undef, #pragma directives. Union: Union definition & declaration, accessing a union member, union of structures, initialization of union member, uses of union, use of user defined data types.

UNIT-V

Introduction: Basic concepts of OOP: object, class, data abstraction, data encapsulation, inheritance, polymorphism, Static and dynamic binding, message passing, benefits of OOP's, disadvantage of OOP's, application of OOP's, a simple program, anatomy of program, creating a source file, compiling and Linking.

Recommended Books:

1. David Parsons; Object oriented programming with C++; BPB publication
2. Object oriented programming in C++ by Robert Lafore: Galgotia
3. Balagurusamy; Object oriented programming with C++; TMH
4. Java Complete Reference: Herbert Schildt, Mc Graw Hill
5. Hubbard; Programming in C++ (Schaum); TMH
6. Mastering C++ by Venugopal, TMH

Computer Science and Engineering, V-Semester
Open Elective CS- 504 (C) Introduction to Database Management Systems

Course Objective:

To understand the fundamental concepts of database management. These concepts include aspects of database design, database languages, and database-system implementation. To give systematic database design approaches covering conceptual design, logical design and an overview of physical design. Includes the concepts of SQL queries to manage database efficiently

Course Outcomes:

1. Understand the different concept of database system.
2. Study the database designs, database modeling, relational, hierarchical, and network models
3. Understand and use data definition language to query, update, and manage a database
4. Demonstrate the use of data manipulation language
5. Demonstrate the use different SQL queries with the concepts of joins and also understand the ANSI SQL

Course Contents:

UNIT-I

Database Management System Concepts: Introduction, Significance of Database, Database System Applications; Data Independence; Data Modeling for a Database; Entities and their Attributes, Entities, Attributes, Relationships and Relationships Types, Advantages and Disadvantages of Database Management System, DBMS Vs RDBMS.

UNIT-II

Database Models and Implementation: Data Model and Types of Data Model, Relational Data Model, Hierarchical Model, Network Data Model, Object/Relational Model, Object-Oriented Model; Entity-Relationship Model, Modeling using E-R Diagrams, Notation used in E-R Model, Relationships and Relationship Types; Associative Database Model

UNIT-III

SQL : Data Definition Language : Categories of SQL Commands; Data Definition Language ; Create table , Drop table and Alter Table . Primary Key , Foreign Key, Truncate Table,

Index, Cursor.

UNIT-IV

SQL DML :Data Manipulation Language, Insert Statement, Multiple Inserts, Delete Statement, Delete with conditions , Update statement, Update with Conditions , Merge Statement,

UNIT-V

SELECT . SQL queries, Data extraction from single, multiple tables equi-join, non equi-join, self-join, outer join. Usage of like, any, all, exists, in Special operators. Hierarchical queries, inline queries, flashback queries. Introduction of ANSI SQL, anonymous block, nested anonymous block, branching and looping constructs in ANSI SQL.

Recommended Books:

1. Date C J, "An Introduction To Database System", Pearson Educations
2. Korth, Silbertz, Sudarshan, "Fundamental of Database System", McGraw Hill
3. Rob, " Data Base System: Design Implementation & Management", Cengage Learning
4. Elmasri, Navathe, "Fundamentals Of Database Systems", Pearson Educations
5. Atul Kahate , " Introduction to Database Management System", Pearson Educations
6. Oracle 9i Database Administration Fundamental-I, Volume I, Oracle Press, TMH.
7. Paneerselvam, "Data Base Management System", PHI Learning

Computer Science and Engineering, V-Semester
CS505 Linux (LAB)

Course Objective:

The student should be able to: Identify and use UNIX/**Linux** utilities to create and manage simple file processing operations, organize directory structures with appropriate security, and develop shell scripts to perform more complex tasks.

Course Outcomes:

1. Understand the system calls
2. Compare between ANSI C AND C++ AND POSIX standards
3. Mapping the relationship between UNIX Kernel support for files
4. Understand Kernel support for process creation and termination and memory allocation

Course Contents:

UNIT-I

Overview of Unix/Linux: Concepts, Unix/Linux Installation Process, Hardware Requirements for Unix/Linux, Advantages of Unix/Linux, Reasons for Popularity and Success of Linux/Unix Operating System, Features of Linux/Unix Operating System, Kernel, Kernel Functions, The Shell Basic Commands, Shell Programming: Shell Variables, Branching Control Structures, Loop-Control Structure, Continue and break Statements, Sleep Command, Debugging Script. Use of Linux as web- server, file server, directory server, application server, DNS server, SMTP server, Firewall, Proxy server.

UNIT-II

File System: Definition of File System, Defining Geometry, Disk Controller, Solaris File System, Disk Based File Systems, Network-Based File Systems, Virtual File systems, UFS File System, The Boot Block, The Super Block, The Inode, Tuning File System, Repairing File System.

UNIT-III

Process Control: Viewing a Process, Command to display Process, Process Attributes, Process States, Process Fields, PS Commands options, PGREP, PRSTAT, CDE Process Manager, Scheduling Process, Scheduling Priorities, Changing the Priority of a time-sharing process, Killing Process.

UNIT-IV

System Security: Physical Security, Controlling System Access, Restricted Shells Controlling File Access, File Access Commands, Access Control List(ACLs), Setting ACL Entries, Modifying ACL entries on a file, Deleting ACL entries on a file, Restricting FTP, Securing Super User Access, Restricting Root Access, Monitoring super user Access, TCP Wrappers.

UNIT-V

Dynamic Host Configuration Protocol: Introduction, DHCP Leased Time, DHCP Scopes, DHCP IP Address, Allocation Types, Planning DHCP Deployment, DHCP Configuration files, Automatic Startup of DHCP Server, Configuration of DHCP Clients, Manually Configuring the DHCP.

Case Study: -

Installation of Linux, Customization of Linux, Installation of SAMBA, APACHE, TOMCAT, Send MAIL, Postfix, Implementation of DNS, LDAP services, Firewall, Proxy server

List of Experiments: -

Experiment 1 (Basic commands)

- a) Installation of Unix/Linux operating system.
- b) Study of logging/logout details.
- c) Study of Unix/Linux general purpose utility command list obtained from (man, who, cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown) commands.
- d) Study of vi editor.
- e) Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.
- f) Study of Unix/Linux file system (tree structure).
- g) Study of .bashrc, /etc/bashrc and Environment variables.

Experiment 2 (C programs)

Write a C program to check whether the given string is palindrome or not using Command line substitution.

II. Write a C program to emulate the UNIX ls-l command.

III. Write a C program to check the given integer is prime or not.

IV. Write a C program to display Largest of three numbers.

V. Write a C program to check whether the given number is Avogadro number or not.

VI. Write a C program to find the Factorial of a given number.

VII. Write a C program that accept two integers as its arguments and computes the value of first number raised to the power of second number.

Experiment 3 (Shell scripts)

- I. Write a shell script program to display list of user currently logged in.
- II. Write a shell script program to display "HELLO WORLD".
- III. Write a shell script program to develop a scientific calculator.
- IV. Write a shell Script program to check whether the given number is even or odd.
- V. Shell script Program to search whether element is present is in the list or not.

Experiment 4 (Shell scripts and sed)

- I. Shell script program to check whether given file is a directory or not.
- II. Shell script program to count number of files in a Directory.
- III. Shell script program to copy contents of one file to another.
- IV. Create directory, write contents on that and Copy to a suitable location in your home directory.
- V. Use a pipeline and command substitution to set the length of a line in file to a variable.
- VI. Write a program using sed command to print duplicated lines of Input.

Experiment 5 (grep, awk, perl scripts)

- I. (a) Write a grep/egrep script to find the number of words character, words and lines in a file.
(b) Write an awk script to develop a Fibonacci series.
- II. (a) Write a perl script to compute the power of a given number.
(b) Write an awk script to display the pattern of given string or number.
- III. (a) Write a perl script to check a number is prime or not.
(b) Write an egrep script to display list of files in the directory.

Experiment 6 (programming)

- I. Write a shell script program to display the process attributes.
- II. Write a shell script to change the priority of processes.
- III. Write a shell script to change the ownership of processes. IV. Write a program to send back a process from foreground.
- V. Write a program to retrieve a process from background.
- VI. Write a program to create a Zombie process.
- VII. Write a program to create a child process and allow the parent to display "parent" and the child to display "child" on the screen.

Experiment 7 (Shell script programming)

- I. Write a shell script program to check variable attributes of file and processes.
- II. Write a shell script program to check and list attributes of processes.
- III. Shell Script program to implement read, write, and execute permissions.
- IV. Shell Script program for changing process priority.

Experiment 8 (Installations)

- I. Installation of VirtualBox (VMWare) on a PC having other operating system.
- II. Installation of Cygwin on a PC having other operating system.
- III. Installation of NS2 on a PC having Unix/Linux operating system.
- IV. Installation of Unix/Linux packages.
- V. Installation of SAMBA, APACHE, TOMCAT.
- VI. Implementation of DNS, LDAP services,
- VII. Study & installation of Firewall & Proxy server

Recommended Books:

1. Venkatesh Murthy, "Introduction to Unix & Shell", Pearson Edu
2. Forouzan, "Unix & Shell Programming", Cengage Learning
3. Sumitab Das, "Unix Concept & Application", TMH
4. Gopalan, Shivaselvan, "Beginners Guide to Unix " PHI Learning
5. Venkatachalam, "Linux Programming Tools Unveil`ed", BS Publication.
6. Richard Peterson, "Linux Complete Reference", TMH
7. Richard Peterson, "Unix Complete Reference", TMH

Computer Science and Engineering, V-Semester

CS506- Python

Course Objective:

1. To acquire programming skills in core Python.
2. To acquire Object Oriented Skills in Python
3. To develop the skill of designing Graphical user Interfaces in Python
4. To develop the ability to write database applications in Python

Course Outcomes:

1. Describing about simple computational problems using Python programs.
2. Explain lists, tuples and dictionaries in Python for representing compound data
3. Apply sorting concepts to solve real world problem
4. Make use of engineering mathematics and command line arguments
5. Solve real world business applications using advance concepts of python library and files handling

List Of Experiments:

1. To write a Python program to find GCD of two numbers.
2. To write a Python program to implement a stack and queue using a list data-structure.
3. To write a Python program to find the exponentiation of a number.
4. To write a Python Program to find the maximum from a list of numbers.
5. To write a Python Program to perform Linear Search
6. To write a Python Program to perform binary search.
7. To write a Python Program to perform selection sort.
8. To write a Python Program to perform insertion sort.
9. To write a Python Program to perform Merge sort.
10. To write a Python program to find first n prime numbers.
11. To write a Python program to multiply matrices.
12. To write a Python program for command line arguments.
13. To write a Python program to find the most frequent words in a text read from a file.
14. Write a random number generator that generates random numbers between 1 and 6 (simulates a dice).
15. Remove all the lines that contain the character `a` in a file and write it to another file.

Recommended Books:

1. Timothy A. Budd: Exploring python, McGraw-Hill Education.
2. R. Nageshwar Rao ,”Python Programming” ,Wiley India
3. Allen B. Downey; Think Python, O'Reilly Media, Inc.