

V SEM B.Tech

S.No.	Subject Code	SUBJECT	SC	L	T	P	IM	EM	CR
1.	1505501	Web Technologies	PJ	3	1	0	30	70	3
2.	1505502	Computer Networks	PJ	3	1	0	30	70	3
3.	1505503	Software Engineering	PJ	4	0	0	30	70	3
4.	1505504	Compiler Design	PJ	3	1	0	30	70	3
5.		CBCC-I	PJ	3	1	0	30	70	3
	1505505	Advanced Computer Architecture							
	1505506	Multimedia Systems							
	1505507	Principles of Programming Languages							
6.	1504508	Microprocessors	PN	4	0	0	30	70	3
7.	1505509	Web Technologies Lab	PN	0	0	3	50	50	2
8.	1524510	Advanced English & Communication Skills Lab	PN	0	0	3	50	50	2
		Total:		20	4	6	280	520	22

VI SEM B.Tech.

S.No.	Subject Code	SUBJECT	SC	L	T	P	IM	EM	CR
1.	1505601	Object Oriented Analysis & Design	PJ	4	0	0	30	70	3
2.	1505602	Design & Analysis of Algorithms	PJ	4	0	0	30	70	3
3.	1505603	Cryptography & Network Security	PJ	4	0	0	30	70	3
4.	1505604	Data Mining	PJ	3	1	0	30	70	3
5.	1505605	Mobile Application Development	PJ	3	1	0	30	70	3
6.		CBCC-II	PJ	4	0	0	30	70	3
	1505606	Machine Learning							
	1505607	Distributed Systems							
	1505608	Unix & Shell Programming							
7.	1505609	UML & Data Mining Lab	PN	0	0	3	50	50	2
8.	1505610	Mobile Application Development Lab	PN	0	0	3	50	50	2
		Total:		22	2	6	280	520	22

(1505501) WEB TECHNOLOGIES

Course Objectives:

- To introduce client side scripting with Javascript and HTML
- To introduce server side programming with PHP.
- To learn the basic web concepts and Internet protocols

UNIT I

Introduction to Web Technologies: Introduction to Web servers like Apache1.1,IIS XAMPP(Bundle Server), WAMP(Bundle Server), Handling HTTP Request and Response, installations of above servers.

UNIT II

HTML Common tags: List, Tables, images, forms, Frames; Cascading Style sheets;

Introduction to Java Script: Introduction to Java Scripts, Objects in Java Script, Dynamic HTML with Java Script.

UNIT III

Overview of PHP Data types and Concepts: Variables and data types, Operators, Expressions and Statements, Strings, Arrays and Functions.

Overview of Classes, Objects, and Interfaces: Creating instances using Constructors, Controlling access to class members, Extending classes, Abstract classes and methods, using interfaces, Using class destructors, File Handling and Using Exceptions.

UNIT IV

PHP Advanced Concepts: Using Cookies, Using HTTP Headers, Using Sessions, Authenticating users, Using Environment and Configuration variables, Working with Date and Time.

UNIT V

Using Creating and Forms: Understanding Common Form Issues, GET vs. POST, Validating form input, Working with multiple forms, and Preventing Multiple Submissions of a form.

PHP and Database Access: Basic Database Concepts, Connecting to a MYSQL database, Retrieving and Displaying results, Modifying, Updating and Deleting data. MVC architecture.

TEXT BOOKS:

1. Beginning PHP and MySQL, 5th Edition, Jason Gilmore, Apress Publications (Dream tech.).
2. PHP 5 Recipes A problem Solution Approach Lee Babin, Nathan A Good, Frank M.Kromann and Jon Stephens.
3. Web Programming, building internet applications, Chris Bates 3rd edition, WILEY Dreamtech.

REFERENCES:

1. Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP, J.Lee and B.Ware(Addison Wesley) Pearson Education.
2. PHP 6 Fast and Easy Web Development, Julie Meloni and Matt Telles, Cengage Learning Publications.
3. PHP 5.1, I. Bayross and S.Shah, The X Team, SPD.
4. PHP and MySQL by Example, E.Quigley, Prentice Hall (Pearson).
5. PHP Programming solutions, V.Vaswani, TMH.

(1505502) COMPUTER NETWORKS

Course Objectives:

- Study the evolution of computer networks and future direction
- Study the concepts of computer networks from layered perspective
- Study the issues open for research in computer networks

UNIT I

Uses of Computer Networks, Network Hardware, Reference Models: OSI, TCP/IP, Comparison of OSI & TCP/IP reference models.

Example Networks: The Internet, Wireless LANs: 802.11

UNIT II

The Data Link Layer: Data Link Layer design issues, Error Detection and Correction, Elementary Data Link Protocols, Sliding Window Protocols.

The Medium Access Control sub layer : Multiple Access protocols, Ethernet, Data Link Layer Switching.

UNIT III

The Network Layer: Network layer design issues, Routing algorithms : The Optimality Principle, Shortest Path Algorithm, Flooding, Distance Vector Routing, Link State Routing, Hierarchical Routing, Broadcast Routing, Multicast Routing, Anycast Routing,

Congestion control algorithms, Quality of service, IP Addresses, IPv4,IPv6,Tunneling, Fragmentation.

UNIT IV

The Transport Layer: The Transport Service, Elements of Transport Protocols, Congestion Control, The internet transport protocols: UDP, TCP: Introduction to TCP, Service Model, Protocol, Segment Header, Connection Establishment, Connection Release.

UNIT V

Domain Name System (DNS), World Wide Web (WWW), E-mail

TEXT BOOKS:

1. "Computer Networks", 5th edition, 2010, Andrew S. Tanenbaum, David J. Wetherall, Pearson.
2. "Data communications and networking" 5th edition, 2012, Behrouz A. Forouzan, TMH.

REFERENCE BOOKS:

1. "Internetworking with TCP/IP – Principles, protocols, and architecture- Volume 1, Douglas E. Comer, 5th edition, PHI
2. "Computer Networks", 5E, Peterson, Davie, Elsevier.
3. "Introduction to Computer Networks and Cyber Security", Chawan- Hwa Wu, Irwin, CRC Publications.
4. "Computer Networks and Internets with Internet Applications", Comer.

(1505503) SOFTWARE ENGINEERING

Course Objectives:

- To understand the software life cycle models.
- To understand the software requirements and SRS document.
- To understand the importance of modeling and modeling languages.
- To design and develop correct and robust software products.
- To understand the quality control and how to ensure good quality software.
- To understand the planning and estimation of software projects.
- To understand the implementation issues, validation and verification procedures.
- To understand the maintenance of software

UNIT I

Software and Software Engineering: The Nature of Software, The Unique Nature of WebApps, Software Engineering, Software Process, Software Engineering Practice, Software Myths.

Process Models: A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Terminology, Product and Process.

UNIT II

Understanding Requirements: Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.

Requirements Modeling: Requirements Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling.

UNIT III

Design Concepts: Design within the Context of Software Engineering, Design Process, Design Concepts, The Design Model.

Architectural Design: Software Architecture, Architectural Genres, Architectural Styles, Architectural Design.

Component-Level Design: What is a Component, Designing Class-Based Components, Conducting Component-Level Design, Component-Level Design for WebApps.

UNIT IV

User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, Design Evaluation.

Coding and Testing: Coding, Code Review, Software Documentation, Testing, Testing in the Large versus Testing in the Small, Unit Testing, Black-Box Testing, White-Box Testing, Debugging, Program Analysis Tools, Integration Testing, Testing Object-Oriented Programs, System Testing, Some General Issues Associated with Testing.

UNIT V

Software Project Management: Responsibilities of a Software Project Manager, Project Planning, Metrics for Project Size Estimation, Project Estimation Techniques, Empirical Estimation Techniques, COCOMO-A Heuristic Estimation Technique, Halstead's Software Science-An Analytical Technique, Staffing Level Estimation, Scheduling, Organization and Team Structures, Staffing, Risk Management, Software Configuration Management. .

Software Maintenance: Characteristics of Software Maintenance, Software Reverse Engineering, Software Maintenance Process Models, Estimation of Maintenance cost.

TEXT BOOKS :

1. Software Engineering: A practitioner's Approach, Roger S. Pressman, Seventh Edition, 2010, McGrawHill International Edition.
2. Fundamentals of Software Engineering, Rajib Mall, 4th Edition, 2014, PHI.

REFERENCE BOOKS:

1. Software Engineering, Ian Sommerville, Ninth edition, Pearson education.
2. Software Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
3. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India,2010.
4. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
5. Software Engineering1: Abstraction and modeling, Diner Bjorner, Springer International edition, 2006.
6. Software Engineering2: Specification of systems and languages, Diner Bjorner, Springer International edition , 2006.

7. Software Engineering Foundations, Yingxu Wang, Auerbach Publications,2008.
8. Software Engineering Principles and Practice, Hans Van Vliet,3rd edition, John Wiley & Sons Ltd.
9. Software Engineering 3: Domains, Requirements, and Software Design, D. Björner, Springer International Edition.
10. Introduction to Software Engineering, R.J. Leach, CRC Press.

(1505504) COMPILER DESIGN

Course Objectives:

- To understand the basic principles of the compiler, Compiler construction tools and lexical analysis.
- To understand the Concept of Context Free Grammars, Parsing and various Parsing Techniques.
- To understand the process of intermediate code generation.
- To understand the process of Code Generation.
- To understand various Code optimization techniques and runtime environment.

UNIT I

Introduction to Compiling: Compilers, Analysis of the Source program, the phases of a compiler, the cousins of the Compiler, grouping of phases, Compiler construction tools.

Lexical Analysis: The role of the analyzer. Input buffering, Specification of tokens, Recognition of tokens, A language for Specifying Lexical analyzer.

UNIT II

Syntax Analysis: The role of the parser, Context-free grammars, writing a grammar, Top-down parsing, Bottom-up parsing, Operator-precedence parsing, LR parsers, Parser generators.

UNIT III

Syntax Directed Translation: Syntax-directed definitions, Construction of syntax trees, S-attributed definitions, L-attributed definitions.

Type Checking: Type systems, Specification of simple type checker, Equivalence of type expressions, type conversions.

UNIT IV

Run-Time Environments: Source Language issues, storage organization, Storage-allocation strategies, Access to non local names, Symbol tables.

Intermediate Code generation: Intermediate languages, three address code, quadruple, triple and indirect triple.

UNIT V

Code Generation: Issues in the Design of a code generator, The target machine, Basic blocks and flow graphs, Next-use information, A simple code generator, Register allocation and assignment, DAG representation of basic blocks, peephole optimization.

Code Optimization: Introduction, the principle source of optimization.

TEXT BOOK:

1. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman. Compilers-Principles, Techniques and Tools, Pearson Education

REFERENCE BOOKS:

1. Alfred V. Aho, Jeffrey D. Ullman, Principles of Compiler Design. Narosa Publications.
2. K. V. N. Sunitha, Compiler Construction, Pearson Education.
3. K. Muneeswaran, Compiler Design, Oxford university press

B.Tech. V Sem

L T C
3 1 3

**(1505505) ADVANCED COMPUTER ARCHITECTURE
(CBCC-I)**

Course Objective:

- Discuss the concept of parallel processing and the relationship between parallelism and performance
- Understand the organization of computer structures that can be electronically configured and Reconfigured
- Discuss the performance advantages that multithreading can offer along with the factors that make it difficult to derive maximum benefits from this approach

Learning Outcome:

- Realize Parallelism and Parallel architectures
- Ability to use Instruction Level Parallelism
- Ability to use Thread level parallelism

UNIT I

Evolution of Computer Architecture, System Attributes to performance, Shared-Memory Multiprocessors, Distributed- Memory Multiprocessors, A Taxonomy of MIMD Computers, Vector Super computers, SIMD Super computers, PRAM models and PRAM variants.

Conditions of Parallelism- data and resource dependencies, hardware and software parallelism, Program partitioning and Scheduling- Grain sizes and Latency, Grain packing and Scheduling, Static Multi processor scheduling, Program Flow Mechanisms- Control flow vs Data flow, Demand-Driven Mechanisms, comparison of Flow Mechanisms, System interconnect architectures- network properties and routing, static and dynamic connection networks.

UNIT II

Principles of scalable performance:- Performance metrics and measures- Parallelism profile in programs, Mean performance, Efficiency, Utilization and Quality, Benchmarks and Performance Measures, Parallel Processing Applications, Speedup Performance Laws- Amdahl's law, Gustafson's law, Memory bounded speed up model, Scalability metrics and goals,

UNIT III

Bus systems- Back plane bus specification, Addressing and Timing protocols, Arbitration, transaction and interrupt, IEEE future bus and other standards, Shared memory organizations- Interleaved memory organization, band width and fault tolerance, memory allocation schemes, Sequential and Weak Consistency Models.

Linear Pipeline Processors- asynchronous and synchronous models, clocking and timing control, speedup, efficiency, and throughput, Non linear pipeline processors- reservation and latency analysis, collision free scheduling, pipeline schedule optimization, Instruction pipe line design- instruction execution phases, mechanisms for instruction pipelining, dynamic instruction scheduling, branch handling techniques, Static Arithmetic Pipelines.

UNIT IV

Hierarchical bus system, cross bar switch and multiport memory, multistage and combining networks, multistage and combining networks, The cache coherence problem, message passing mechanism- message routing schemes, deadlock virtual channels, flow control strategies, multicast routing algorithms

Vector processing principles- vector instruction types, vector access memory schemes, early super computers, Multi vector multiprocessors- performance directed design rules, architecture of Cray Y-MP,C-90 and MPP, Compound vector operations, vector loops and chaining, SIMD computer organizations

UNIT V

Latency-hiding techniques- shared virtual memory, Prefetching techniques, distributed coherent caches, scalable coherence interface, relaxed memory consistency, Principles of multithreading, The MIT J-Machine, The Tera multiprocessor system, The Evolution of Data flow computers.

Text Books :

1. KAI Hwang & Naresh Jotwani, “Advanced Computer Architecture- Parallelism, Scalability, Programmability” Second Edition, Mc Graw Hill Publishing

Reference Books :

1. Hennessy Patterson, "Computer Architecture- A Quantitative Approach" Fifth Edition, Elsevier
2. Kai Hwang, "Advanced Computer Architecture- Parallelism, Scalability, Programmability", TMH.
3. Computer Architecture, Concepts and Evolutions, Garrit A Blaauw, PEA.

B.Tech. V Sem

L T C
3 1 3

(1505506) MULTIMEDIA SYSTEMS
(CBCC-I)

Course Objectives:

- To adapt the architecture for design of multimedia system.
- To solve issues related to multimedia file handling.
- To adopt hypermedia standards in developing multimedia applications.
- Know the basics of creating multimedia applications.

UNIT I

Multimedia: Definition, Where to use multimedia, Medium, Main properties of multimedia system, Traditional data stream characteristics, Data stream characteristics for continuous media, Information units, **Sound/Audio:** Basic sound concepts, Music, Speech, **Images/Graphics:** Basic concepts, Computer image processing.

UNIT II

Video and Animation: Basic concepts, Television, Computer based animation, **Data Compression:** Storage space, Coding requirements, Source, Entropy and Hybrid coding, Some basic compression techniques, JPEG, H.261,(Px64), MPEG, DVI.

UNIT III

Computer Technology: Communication Architecture, Multimedia Workstation, **Multimedia Operating Systems:** Introduction, Real time, Resource management, Process management, **Multimedia Communication systems:** Application Subsystem, Transport subsystem.

UNIT IV

Database Systems: Multimedia Database Management System, Characteristics of an MDBMS, Data Analysis, Data Structure, Operations on data, Integration in a Database model, **Documents, Hypertext and MHEG:** Documents, Hypertext and Hypermedia, Document Architecture ODA, MHEG.

UNIT V

Synchronization: Introduction, Notion of Synchronization, Presentation requirements, Reference model for multimedia synchronization, Synchronization specifications, **Multimedia Applications:** Introduction, Media Presentation, Media Composition, Media Integration, Media Communication, Media Consumption, and Media Entertainment.

TEXT BOOKS :

1. “ Multimedia: Computing, Communications and Applications”, Ralf Steinmetz and Klara Nahrstedt, Pearson Education.
2. “Multimedia: Making It work:, Tay Vaughan, Pearson Education.

REFERENCE BOOKS:

1. “ Multimedia System design “, Prabhat K. Andheigh, Kiran Thakrar, THM
2. “ Multimedia Systems”, Koegel Buford, Pearson Education
3. “ Fundamentals of Multimedia , Ze-Nian Li, Mark.S.Drew
4. “ Multimedia Communication Systems: Techniques, standards and networks, K.R.Rao,D.Milovanovic

B. Tech V Sem.

L T C

3 1 3

(1505507) PRINCIPLES OF PROGRAMMING LANGUAGES

(CBCC-I)

Course Objectives:

- To study various programming paradigms.
- To provide conceptual understanding of High level language design and implementation.
- To introduce the power of scripting languages

Course Outcomes:

- Ability to select appropriate programming language for problem solving
- Ability to design new programming language.

UNIT I

Introduction: Software Development Process, Language and Software Development Environments, Language and Software Design Models, Language and Computer Architecture, Programming Language Qualities, A brief Historical Perspective.

Syntax and Semantics: Language Definition, Language Processing, Variables, Routines, Aliasing and Overloading, Run-time Structure.

Unit II

Structuring the data: Built-in types and primitive types, Data aggregates and type constructors, User-defined types and abstract data types, Type Systems, The type Structure of representative languages, Implementation Models

Unit III

Structuring the Computation: Expressions and Statements, Conditional Execution and Iteration, Routines, Exceptions, Pattern Matching, Nondeterminism and Backtracking, Event-driven computations, Concurrent Computations.

Structuring the Program: Software Design Methods, Concepts in Support of Modularity, Language Features for Programming in the Large, Generic Units.

Unit IV

Object-Oriented Languages: Concepts of Object-oriented Programming, Inheritances and the type system, Object-oriented features in programming languages

Unit V :

Functional Programming Languages: Characteristics of imperative languages, Mathematical and programming functions, Principles of Functional Programming, Representative Functional Languages, Functional Programming in C++

Logic and Rule-based Languages: —What|| versus —how||: Specification versus implementation, Principles of Logic Programming, PROLOG, Functional Programming versus Logic Programming, Rule-based Languages

TEXT BOOK:

1. Programming Language Concepts||, Carlo Ghezzi, Mehdi Jazayeri, WILEY Publications.
Third Edition, 2014

REFERENCE BOOKS:

1. Concepts of Programming Languages, Tenth Edition, Robert W. Sebesta, Pearson Education.
2. Programming Languages Principles and Paradigms, Second Edition, Allen B. Tucker, Robert E. Noonan, McGraw Hill Education.
3. Introduction to Programming Languages, Aravind Kumar Bansal, CRC Press.

(1504508) MICROPROCESSORS

Course Objectives:

- To learn the architecture and instruction set of 16 bit Microprocessors
- To learn the instruction set of 16 bit microprocessor and solve problems using the same

UNIT-I

Development of microprocessors 8086 microprocessors – Architecture, Pin configuration, Instruction set, Addressing modes, Interrupt system. System timing of 8086 – clock cycle, machine cycle and instruction cycle, timing diagram for simple instructions, generation of delays.

UNIT-II

Assembler, Assembler directives, Assembly language programs (8086) with Assembler directives for addition, subtraction, multiplication, division etc., sorting and searching, bit manipulation, programs using look-up tables, stages of software development, modular programming, debugging and documentation.

UNIT-III

Data transfer schemes – Synchronous, Asynchronous, Interrupt driven and DMA type schemes, USART (8251) and its interfacing, Programmable interrupt controller (8259) and its interfacing, Programmable DMA controller and its interfacing, Data communication standards, RS – 232 Serial interface standard, IEEE – 488 GPIB standard.

UNIT-IV

Memory interfacing to 8086 – Interfacing various types of RAM and ROM chips, Address decoding techniques. Interfacing ADC and DAC to 8086 systems, Data acquisition, Waveform generation, Traffic light controller, Stepper motor control, temperature measurement and control.

UNIT-V

Introduction to 80386 and 80486 microprocessors, different modes of operation – protected mode, virtual mode. Introduction to Pentium processor – special Pentium registers, Pentium memory management, introduction to the Pentium pro– microprocessor.

TEXT BOOKS:

1. Barry B Brey: The INTEL Microprocessors 8086/8088, 80186/80188/80286, 80386, 80486, Pentium and Pentium processor, Architecture, Programming and Interfacing 4th edition (PHI).
2. Hall Douglas V. Microprocessor and Interfacing TMH.
3. Ram B : Fundamentals of Microprocessors and Micro Computers, Dhanpat Rai & sons.
4. Mukhopadhyay, Microprocessor, Microcomputer and their Applications, Narosa Publishing House. 5. J.Uffembeqe, the 8086/8088 family – Design, Programming and Interfacing, PH

B.Tech. V Sem

L T P

0 0 3

(1505509) WEBTECHNOLOGIES LAB

Course Objectives:

- To introduce client side scripting with Java script and HTML
- To introduce server side programming with PHP.
- To learn the basic web concepts and Internet protocols

Course Outcomes:

- Ability to create dynamic and interactive web sites.
- Gain knowledge of client side scripting using java script and HTML.
- Able to do server side programming with PHP.

Hardware and Software required:

1. A working computer system with either Windows or Linux
2. A web browser either IE or Firefox
3. Apache web server or IIS Web server
4. A database either Mysql or Oracle

LIST OF EXPERIMENTS

1. Write a program in HTML to display different styles of heading text.
2. Write a HTML program to design their class time table using table tags.
3. Write a program to display the processes to be followed for a patient when he enters for a complete checkup. Use ordered lists and unordered lists.
4. To create a webpage with the following using HTML
 - To embed an image in web page
 - To fix the hot spots.
 - Show all the related information when a hot spot is clicked in the map.

5. Use “Anchor” tag to write a program for displaying various Menus.
6. Design the following static web pages required for an online book store web site.

a) HOME PAGE:

The static home page must contain three **frames**.

Top frame: Logo and the college name and links to Home page, Login page, Registration page, Catalogue page and Cart page (the description of these pages will be given below).

Left frame: At least four links for navigation, which will display the catalogue of respective links. For e.g.: When you click the link —CSE|| the catalogue for CSE Books should be displayed in the Right frame.

Right frame: The pages to the links in the left frame must be loaded. Validate the registration, user login pages using JavaScript.

b) LOGIN PAGE

c) CATOLOGUE PAGE

The catalogue page should contain the details of all the books available in the web site in a table.

The details should contain the following:

1. Snap shot of Cover Page.
2. Author Name.
3. Publisher.
4. Price.
5. Add to cart button.

d) CART PAGE

The cart page contains the details about the books which are added to the cart.

e) REGISTRATION PAGE:

Create a —registration form —with the following fields

- 1) Name (Text field)
- 2) Password (password field)

- 3) E-mail id (text field)
- 4) Phone number (text field)
- 5) Sex (radio button)
- 6) Date of birth (3 select boxes)
- 7) Languages known (check boxes – English, Telugu, Hindi, Tamil)
- 8) Address (text area)

7. Write *JavaScript* to validate the following fields of the above registration page.

1. Name (Name should contains alphabets and the length should not be less than 6 characters).
2. Password (Password should not be less than 6 characters length).
3. E-mail id (should not contain any invalid and must follow the standard pattern name@domain.com)
4. Phone number (Phone number should contain 10 digits only).

8. Design the webpage by applying the different styles using inline, external & internal style sheets.

9. Design a web page using **CSS (Cascading Style Sheets)** which includes the following:

1) Use different font, styles:

In the style definition you define how each selector should work (font, color etc.). Then, in the body of your pages, you refer to these selectors to activate the styles.

2) Set a background image for both the page and single elements on the page.

3) Control the repetition of the image with the background-repeat property. As background-repeat: repeat Tiles the image until the entire page is filled, just like an ordinary background image in plain HTML.

4) Define styles for links as

A:link

A:visited

A:active

A: hover

5) Work with layers

10. To create a php program to demonstrate the different predefined functions in Array and Math.

11. Install IIS web server and APACHE.

While installation assign port number 4040 to IIS and 8080 to APACHE. Make sure that these ports are available i.e., no other process is using this port.

Access the above developed static web pages for books web site, using these servers by putting the web pages developed in program-6.

12. User Authentication :

Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Write a PHP for doing the following.

1. Create a Cookie and add these four user id's and passwords to this Cookie.

2. Read the user id and passwords entered in the Login form (program 6) and authenticate with the values (user id and passwords) available in the cookies.

If he is a valid user (i.e., user-name and password match) you should welcome him by name (user-name) else you should display — you are not an authenticated user “. Use init-parameters to do this.

13. Install a database (Mysql or Oracle).

Create a table which should contain at least the following fields: name, password, email-id, phone number (these should hold the data from the registration form).

Write a PHP program to connect to that database and extract data from the tables and display them. Experiment with various SQL queries.

Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page

14. Write a PHP which does the following job:

Insert the details of the 3 or 4 users who register with the web site by using registration. Authenticate the user when he submits the login form using the user name and password from the database.

(1524510) ADVANCED ENGLISH COMMUNICATION SKILLS LAB

1. Introduction

The Advanced English Language Skills Lab introduced at the 3rd year B.Tech level is considered essential for the student for focusing on his/her career. At this stage it is imperative for the student to start preparing for the ever growing competition in the job market. In this scenario, in order to be on par with the best, he/she needs to improve his/her Communication and soft skills

This course focuses on the practical aspects of English incorporating all the four (LRSW) skills relevant to the requirements of the prospective employers in view of globalization. The proposed course will enable the students to perform the following:

- Intensive reading to improve comprehension and communication
- Attentive listening for better understanding
- Write project/research/technical reports
- Write Resume' to attract attention
- Discuss ideas / opinions for better solutions
- Face interviews confidently
- Gather information, organize ideas, and present them effectively before an audience
- To help the students cultivate the habit of reading passages from the computer monitor, thus providing them with the required ability to face computer-based competitive exams such GRE, TOEFL,CAT, GMAT etc.

2. Objectives:

Keeping in mind the previous exposure of the student to English, this lab focuses on improving the student's proficiency in English at all levels. The lab intends to train students to use language effectively, to participate in group discussions, to help them face interviews, and sharpen public speaking skills and enhance the confidence of the student by exposing him/her to various situations and contexts which he/she would face in his/her career

3 Syllabus

The following course content is prescribed for the Advanced Communication Skills Lab:

Reading Comprehension -- Reading for facts, guessing meanings from context, speed reading, scanning, skimming for building vocabulary(synonyms and antonyms, one word substitutes, prefixes and suffixes, idioms and phrases.)

Listening Comprehension-- Listening for understanding, so as to respond relevantly and appropriately to people of different backgrounds and dialects in various personal and professional situations.

Technical Report Writing—Types of formats and styles, subject matter, organization, clarity, coherence and style, data-collection, tools, analysis

Resume' Writing—Structure, format and style, planning, defining the career objective, projecting one's strengths, and skills, creative self marketing, cover letter

Group Discussion-- Communicating views and opinions, discussing, intervening. providing solutions on any given topic across a cross-section of individuals,(keeping an eye on modulation of voice, clarity, body language, relevance, fluency and coherence) in personal and professional lives.

Interview Skills—Concept and process, pre-interview planning, mannerisms, body language, organizing, answering strategies, interview through tele and video-conferencing

Technical Presentations (Oral)— Collection of data, planning, preparation, type, style and format ,use of props, attracting audience, voice modulation, clarity, body language, asking queries.

4. Minimum Requirements

The English Language Lab shall have two parts:

The Computer aided Language Lab for 60 students with 60 systems, one master console, LAN facility and English language software for self-study by learners.

The Communication Skills Lab with movable chairs and audio-visual aids with a P.A System, a TV, A digital stereo-audio and video system, Camcorder etc

System Requirement (Hardware Component):

Computer network with LAN with a minimum of 60 multimedia systems with the following specifications:

P-IV Processor, Speed-2.8 GHz, RAM_512 MB minimum, Hard Disk-80 GB, Headphones

Prescribed Software: Walden and K-van solutions.

Books Suggested for English Language Lab Library (to be located within the lab in addition to the CDs of the text book which are loaded on the systems):

1. **Technical writing and professional communication, Huckin and Olsen** Tata Mc Graw-Hil 2009.

2. **Speaking about Science**, A Manual for Creating Clear Presentations by **Scott Morgan and Barrett Whitener**, Cambridge University press, 2006
3. **Books on TOEFL/GRE/GMAT/CAT/ IELTS** by **Barron's/DELTA/Cambridge University Press**.
4. **Handbook for Technical Writing** by David A McMurrey & Joanne Buckely CENGAGE Learning 2008
5. **Technical Communication** by Meenakshi Raman & Sangeeta Sharma, Oxford University Press 2009.
6. **The ACE of Soft Skills** by Gopal Ramesh and Mahadevan Ramesh, Pearson Education, 2010
7. **Cambridge English for Job-Hunting** by Colm Downes, Cambridge University Press, 2008
8. **Resume's and Interviews** by M.Ashraf Rizvi, Tata Mc Graw-Hill, 2008
9. **From Campus To Corporate** by KK Ramachandran and KK Karthick, Macmillan Publishers India Ltd, 2010
10. **English Language Communication : A Reader cum Lab Manual** Dr A Ramakrishna Rao, Dr G Natanam & Prof SA Sankaranarayanan, Anuradha Publications, Chennai 2008.
11. **Managing Soft Skills** by K R Lakshminarayan and T.Muruguvel, Sci-Tech Publications, 2010
12. **Business Communication** by John X Wang, CRC Press, Special Indian Edition,2008

B.Tech. VI Sem

L T C
4 0 3

(1505601) OBJECT ORIENTED ANALYSIS & DESIGN

Course Objectives:

- To understand the Object oriented life cycle.
- To know how to identify objects, relationships, Services and attributes through UML.
- To understand different UML diagrams.
- To know object oriented design process, software quality and usability.

UNIT I

Introduction to UML: Importance of Modeling, Principles of Modeling, Object Oriented Modeling, Conceptual Model of the UML, Architecture.

UNIT II

Basic Structural Modeling : Classes, Relationships, Common Mechanisms, and Diagrams
Interfaces, Types and Roles, Packages.

Class and Object Diagrams : Terms, Concepts, Modeling Techniques for Class and Object
Diagrams.

UNIT III

Basic Behavioral Modeling-I : Interactions, Interaction Diagrams.

Basic Behavioral Modeling-II : Use Cases, Use Case Diagrams, Activity Diagrams.

UNIT IV

Advanced Behavioral Modeling : Events and Signals, State Machines, Processes and Threads,
Time and Space, State Chart Diagrams.

UNIT V

Architectural Modeling : Component, Deployment, Component Diagrams and Deployment
Diagrams, Systems and Models.

Case Study : The Unified Library Application.

TEXT BOOKS:

1. The Unified Modeling Language User Guide, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.
2. UML 2 Toolkit, Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado,

WILEY-Dreamtech India Pvt. Ltd.

REFERENCES:

1. Fundamentals of Object Oriented Design in UML, Meilir Page- Jones, Pearson Education.
2. Modeling Software Systems Using UML2, Pascal Roques, Wiley- Dreamtech India Pvt. Ltd.
3. Object Oriented Analysis and Design, Atul Kahate, The McGraw- Hill Companies.
4. Object-Oriented Analysis and Design with the Unified Process, John W. Satzinger, Robert B Jackson and Stephen D Burd, Cengage Learning.
5. Learning UML 2.0, Russ Miles and Kim Hamilton, O'Reilly, SPD.
6. Applying UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.
7. UML and C++, R.C.Lee and W.M.Tepfenhart, PHI.
8. Object Oriented Analysis, Design and Implementation, B.Dathan and S.Ramnath, Universities Press.
9. OO Design with UML and Java, K.Barclay, J.Savage, Elsevier.
10. Mark Priestley: Practical Object-Oriented Design with UML, TMH.

B.Tech. VI Sem

L T C
4 0 3

(1505602) DESIGN AND ANALYSIS OF ALGORITHMS

Course Objectives:

- To introduce the basic concepts of algorithms.
- To introduce the mathematical aspects and analysis of algorithms.
- To introduce sorting and searching algorithms.
- To introduce various algorithmic techniques.
- To introduce algorithm design methods.

UNIT-I

Introduction: Algorithm, Performance analysis.

Asymptotic Notations: Big Oh Notation, Omega Notation, Theta Notation, Little Oh Notation, Basic efficiency classes, Performance Measurement.

Brute Force Method: Selection sort, Bubble sort, Sequential search, Brute force string matching.

UNIT-II

Divide-and Conquer: The general method, Binary search, Finding the maximum and minimum, Merge sort, quick sort, Strassen's matrix multiplication.

Greedy Method: The general method, Knapsack problem, Job sequencing with deadlines, Minimum Spanning trees, Huffman codes.

UNIT-III

Dynamic Programming: Multistage graph, Single source shortest path, All Pair Shortest Path, Optimal Binary Search trees, 0/1 Knapsack, Reliability Design, Traveling Salesman Problem.

UNIT-IV

Search and Traversal techniques: Techniques for binary tree, Techniques for Graphs, Connected components and spanning tree, Biconnected components and DFS.

Backtracking: The general method, 8-Queens Problem, sum of subsets, Graph Coloring, Hamilton cycles.

UNIT-V

Branch and Bounds: General method, 0/1 Knapsack, LC Branch and Bound solution FIFO Branch and Bound solution, Traveling salesperson problem.

Limitations of Algorithm power: Lower bound arguments, decision trees, P, NP.

NP-Hard and NP-Completeness: Basic concepts, Nondeterministic algorithms, The Classes –NP – hard and NP complete, Cook's theorem.

TEXT BOOKS:

1. Horowitz E. Sahani S and Rajasekaran S: "Fundamentals of Computer Algorithm", Galgotia Publications.
2. Anany Levitin, "Introduction to the Design & Analysis, of Algorithms", Pearson Education, 2nd Edition, 2008.

REFERENCE BOOKS:

1. Aho, Hopcroft, Ulman, "The Design and Analysis of Computer Algorithm", Pearson Education, 2000.
2. Parag H. Dave, Himanshu B. Dave "Design and Analysis of Algorithms" Pearson Education, 2008.
3. Cormen T H, Leiserson C.E, Rivest R.L, and Stein C, Introduction to Algorithms, 2nd edition , Prentice –Hall of India, 2001.
4. R.c.T. Lee , S.S.Tseng, R.C.Chang and T.Tsai , Introduction to Design and analysis of Algorithms A strategic approach , MC Graw Hill.

(1505603) CRYPTOGRAPHY & NETWORK SECURITY

Course Objectives:

- Extensive, thorough and significant understanding of the concepts, issues, principles and theories of computer network security
- Identifying the suitable points for applying security features for network traffic
- Understanding the various cryptographic algorithms and implementation of the same.
- Understanding the various attacks, security mechanisms and services.

UNIT I

Computer Security concepts, The OSI Security Architecture, Security attacks, Security services and Security mechanisms, A model for Network Security, Classical encryption techniques-symmetric cipher model, substitution ciphers, transposition ciphers, Steganography, Modern Stream ciphers.

UNIT II

Modern Block Ciphers: Block ciphers principles, Data encryption standard (DES), Strength of DES, Block cipher modes of operations, AES, RC4

Introduction to Number theory : Integer Arithmetic, Modular Arithmetic, Linear Congruence, Algebraic Structures, $GF(2^n)$ Fields, Primes, Factorization, Chinese remainder Theorem, Quadratic Congruence.

UNIT III

Public-key cryptography : Principles of public-key cryptography, RSA Algorithm, Diffie-Hellman Key Exchange, ElGamal cryptographic system.

Cryptographic Hash functions: Applications of Cryptographic Hash functions, Requirements and security, Hash functions based on Cipher Block Chaining, Secure Hash Algorithm (SHA).

UNIT IV

Message Authentication Codes: Message authentication Requirements, Message authentication functions, Message authentication codes, security of MACs, HMAC.

Digital Signatures: Digital Signatures, Schnorr Digital Signature Scheme, Digital Signature Standard

UNIT V

User Authentication: Remote user Authentication Principles, Kerberos

Electronic mail security: Pretty Good Privacy (PGP), S/MIME

Worms, Viruses, Firewalls

TEXT BOOKS:

1. Cryptography and network Security by Fourth edition, Stallings, PHI/Pearson
2. Cryptography & Network Security by Behrouz A. Forouzan, TMH.

REFERENCE BOOKS:

1. Network Security: The complete reference by Robert Bragg, Mark Rhodes, TMH
2. Computer Security Basics by Rick Lehtinen, Deborah Russell & G.T.Gangemi Sr., SPD O'REILLY.

(1505604) DATA MINING

Course Objectives:

- To learn the concepts of database technology evolutionary path which has led to the need for data mining and its applications
- To learn Data mining algorithms to build analytical applications

UNIT I

Introduction: What is Data Mining, Motivating Challenges, The Origins of Data Mining, Data Mining Tasks.

Data: Types of Data, Data Quality, Data Preprocessing, Measures of Similarity and Dissimilarity.

Exploring Data: Summary Statistics, OLAP and Multidimensional Data Analysis

UNIT II

Basic Concepts, Decision Trees, and Model Evaluation: Preliminaries, General Approach to Solving a Classification Problem, Decision Tree Induction, Model Overfitting, Evaluating the Performance of a Classifier, Methods for Comparing Classifiers.

UNIT III

Classification-Alternative techniques: Rule-Based Classifier, Nearest-Neighbor Classifiers, Bayesian Classifiers, Artificial Neural Networks, Support Vector Machines, Ensemble Methods, Class Imbalance Problem, Multiclass Problem

UNIT IV

Association Analysis- Basic Concepts and Algorithms: Problem Definition, Frequent Item set Generation, Rule Generation, Compact Representation of Frequent Item sets, Alternative Methods for Generating Frequent Item sets, FP-Growth Algorithm, Evaluation of Association Patterns, Effect of Skewed Support Distribution.

UNIT V

Cluster Analysis- Basic Concepts and Algorithms: Overview, k-means, Agglomerative Hierarchical Clustering, DBSCAN, Cluster Evaluation.

Cluster Analysis-Additional Issues and Algorithms: Characteristics of Data, Clusters, and Clustering Algorithms, Prototype-Based Clustering, Density-Based Clustering, Graph-Based Clustering-Minimum Spanning Tree (MST) Clustering, Chameleon, Scalable Clustering Algorithms-Scalability-General Issues and Approaches, BIRCH, CURE

TEXT BOOKS :

1. Introduction to Data Mining, Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Pearson
2. Data Mining: Concepts and Techniques, Jiawei Han and Micheline Kamber, Morgan Kaufmann Publishers, Elsevier, Second Edition, 2006

REFERENCE BOOKS :

1. Data Mining Principles & Applications, T.V. Suresh Kumar, B. Eswara Reddy, Jagadish S Kallimani, Elsevier.
2. Data Mining Techniques and Applications an Introduction, Hongbo Du, Cengage Learning
3. Data Mining Techniques, Arun K Pujari, Second Edition, Universities Press
4. Data Mining, Pudi, Oxford University Press

B.Tech. VI Sem

L T C
3 1 3

(1505605) MOBILE APPLICATION DEVELOPMENT

Course Objectives:

- To Understand fundamentals of android operating systems.
- To learn the internals of the Android OS
- To learn the Mobile application development using the Android SDK.

UNIT 1

Android Introduction and Basics: The Android 4.1 jelly Bean SDK, Understanding the Android Software Stack, installing the Android SDK, Creating Android Virtual Devices, Creating the First Android Project, Using the Text view Control, Using the Android Emulator, The Android Debug Bridge(ADB), Launching Android Applications on a Handset.

UNIT 2

Basic Widgets: Understanding the Role of Android Application Components, Understanding the Utility of Android API, Overview of the Android Project Files, Understanding Activities, Role of the Android Manifest File, Creating the User Interface, Commonly Used Layouts and Controls, Event Handling, Displaying Messages Through Toast, Creating and Starting an Activity, Using the Edit Text Control, Choosing Options with Checkbox, Choosing Mutually Exclusive Items Using Radio Buttons

UNIT 3

Building Blocks for Android Application Design: Introduction to Layouts, Linear Layout, Relative Layout, Absolute Layout, Using Image View, Frame Layout, Table Layout, Grid Layout, Adapting to Screen orientation.

Utilizing Resources and Media: Resources, Creating Values Resources, Using Drawable Resources, Switching States with Toggle Buttons, Creating an Images Switcher Application, Scrolling Through Scroll View, playing Audio, Playing Video, Displaying Progress with Progress Bar, Using Assets.

UNIT 4

Using Selection widgets and Debugging: Using List View, Using the Spinner control, Using the GridView Control, Creating an Image Gallery Using the ViewPager Control, Using the Debugging Tool: Dalvik Debug Monitor Service(DDMS), Debugging Application, Using the Debug Perspective.

Displaying And Fetching Information Using Dialogs and Fragments: What Are Dialogs?, Selecting the Date and Time in One Application, Fragments, Creating Fragments with java Code, Creating Special Fragments

UNIT 5

Building Menus and Storing Data: Creating Interface Menus and Action Bars, Menus and Their Types, Creating Menus Through XML, Creating Menus Through Coding, Applying a Context Menu to a List View, Using the Action Bar, Replacing a Menu with the Action Bar, Creating a Tabbed Action Bar, Creating a Drop-Down List Action Bar.

Using Databases: Using the SQLiteOpenHelperclasss, Accessing Databases with the ADB, Creating a Data Entry Form,

Text Books

1. Android Programming by B.M Harwani, Pearson Education, 2013.

Reference Text Books:

1. Android application Development for Java Programmers, James C Sheusi, Cengage Learning

2. Android In Action by w.Frank Ableson, Robi Sen, Chris King, C. Enrique Ortiz., Dreamtech.

3. Beginning Android 4 Application Development, by Wei-Meng Lee , Wiley India.

(1505506) MACHINE LEARNING
(CBCC-II)

Course Objectives:

- To understand the basic theory underlying machine learning.
- To be able to formulate machine learning problems corresponding to different applications.
- To understand a range of machine learning algorithms along with their strengths and weaknesses.
- To be able to apply machine learning algorithms to solve problems of moderate complexity.

Course Outcomes:

- Ability to understand what is learning and why it is essential to the design of intelligent machines.
- Ability to design and implement various machine learning algorithms in a wide range of real-world applications.
- Acquire knowledge deep learning and be able to implement deep learning models for language, vision, speech, decision making, and more.

UNIT I

What is Machine Learning?, Examples of machine learning applications, supervised Learning: learning a class from examples, Vapnik- Chervonenkis dimension, probably approximately correct learning, noise, learning multiple classes, regression, model selection and generalization, dimensions of a supervised machine learning algorithm. Decision Tree Learning: Introduction, Decisions Tree representation, Appropriate problems for decision tree learning, the basic decision tree learning algorithm, Hypothesis space search in decision tree learning, Inductive bias in decision tree learning, issues in decision tree learning, Artificial Neural Networks: Introduction, Neural Network Representation – Problems – Perceptrons – Multilayer Networks and Back Propagation Algorithm, Remarks on the BACKPROPGRATION Algorithm, An illustrative Example: Face Recognition, Advanced Topics in Artificial Neural Networks.

UNIT II

Evaluating Hypotheses: Motivation, Estimating hypothesis accuracy, basics of sampling theory, a general approach for deriving confidence intervals, differences in error of two hypothesis, comparing learning algorithms, Bayesian Learning: Introduction, Bayes Theorem, Bayes Theorem and Concept Learning, Maximum Likelihood and least squared error hypothesis, Maximum Likelihood hypothesis for predicting probabilities, Minimum Description Length Principle, Bayes Optimal Classifier, Gibbs Algorithm , Naïve Bayes Classifier , Bayesian Belief Network, EM Algorithm

UNIT III

Dimensionality Reduction: Introduction, Subset selection, principle component analysis, feature embedding, factor analysis, singular value decomposition and matrix factorization, multidimensional scaling, linear discriminant analysis, canonical correlation analysis, Isomap, Locally linear embedding, laplacian eigenmaps, Clustering: Introduction, Mixture densities, K-Means clustering, Expectations Maximization algorithm, Mixture of latent variable models, supervised learning after clustering, spectral clustering, Hierarchical clustering, Choosing the number of clusters, Nonparametric Methods: Introduction, Non Parametric density estimation, generalization to multivariate data, nonparametric classification, condensed nearest neighbor, Distance based classification, outlier detection, Nonparametric regression: smoothing models, how to choose the smoothing parameter.

UNIT IV

Linear Discrimination: Introduction, Generalizing the linear model, geometry of the linear discrimination, pair wise separation, parametric discrimination revisited, gradient descent, logistic discrimination, discrimination by regression, learning to rank, Multilayer Perceptrons: Introduction, the perceptron, training a perceptron, learning Boolean functions, multilayer perceptrons, MLP as a universal approximator, Back propagation algorithm, Training procedures, Tuning the network size, Bayesian view of learning, dimensionality reduction, learning time, deep learning.

UNIT V

Kernel Machines: Introduction, Optimal separating hyperplane, the non separable case: Soft Margin Hyperplane, ν -SVM, kernel Trick, Vectorial kernels, defining kernels, multiple kernel learning, multicast kernel machines, kernel machines for regression, kernel machines for ranking, one-class kernel machines, large margin nearest neighbor classifier, kernel dimensionality reduction, Graphical models: Introduction, Canonical cases for conditional independence, generative models, d separation, belief propagation, undirected Graphs: Markov Random files, Learning the structure of a graphical model, influence diagrams.

TEXT BOOKS:

1. Machine Learning by Tom M. Mitchell, Mc Graw Hill Education, Indian Edition, 2016.
2. Introduction to Machine learning, Ethem Alpaydin, PHI, 3rd Edition, 2014

REFERENCE BOOK:

1. Machine Learning: An Algorithmic Perspective, Stephen Marsland, Taylor & Francis, CRC Press Book.

B.Tech. VI Sem

L T C
4 0 3

(1505607) DISTRIBUTED SYSTEMS
(CBCC-II)

Course Objectives:

- To understand the basic concepts of distributed systems, challenges & trends in distributed systems.
- To provide an adequate idea on clock synchronization algorithms and coordination among processes in distributed systems.
- To learn the techniques of remote procedure call, remote method invocation, and group communications.
- To gain the knowledge on distributed file system, naming systems, transactions, distributed transactions and concurrency control techniques.

UNIT-I

Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Trends in distributed systems, Focus on Resource Sharing, Challenges.

System Models: Introduction, Physical models, Architectural Models, Fundamental Models.

UNIT-II

Time and Global States: Introduction, Clocks, Events and Process States, Synchronizing Physical Clocks, Logical Time and Logical Clocks, Global States, Distributed Debugging.

Coordination and Agreement: Introduction, Distributed Mutual Exclusion, Elections, Coordination and agreement in group communication,.

UNIT-III

InterProcess Communication: Introduction, The API for the Internet Protocols, External Data Representation and Marshalling, Multicast communication, Network virtualization: Overlay Networks.

Remote Invocation: Introduction, Request Reply Protocols, Remote Procedure Call, Remote Method Invocation.

UNIT-IV

Distributed File Systems: Introduction, File Service Architecture, Case Study 1: Sun Network File System, Case Study 2: The Andrew File System.

Name Services: Introduction, Name Services and the Domain Name System, Directory Services, Case Study: Global Name Services.

UNIT- V

Transactions and Concurrency Control: Introduction, Transactions, Nested Transactions, Locks, Optimistic Concurrency Control, Timestamp Ordering, Comparison of Methods for Concurrency Control.

Distributed Transactions: Introduction, Flat and Nested Distributed Transactions, Atomic Commit Protocols, Concurrency Control in Distributed Transactions, Distributed Deadlocks, Transaction Recovery.

TEXT BOOK

1. Distributed Systems, Concepts and Design, George Coulouris, J Dollimore and Tim Kindberg, Gordon Blair, Pearson Education, 5th Edition.

REFERENCE BOOKS

2. Distributed Systems, Principles and Paradigms, Andrew S. Tanenbaum, Maarten Van Steen, 2nd Edition, PHI.
3. Distributed Systems, An Algorithm Approach, Sukumar Ghosh, Chapman&Hall/CRC, Taylor & Fransis Group, 2007.
4. Distributed Operating Systems, Andrew S. Tanenbaum, Pearson Education.

B.Tech. VI Sem

L T P
0 0 3

(1505608) UNIX INTERNALS AND SHELL PROGRAMMING
(CBCC-II)

Course Objectives:

- To provide overview of unix operating system and its architecture.
- To provide an adequate idea on internal representation of files and system calls.
- To introduce shell programming

UNIT I

Introduction to the kernel: - Architecture of the UNIX, The buffer cache.

Internal representation of files:- Inodes, accessing blocks, releasing blocks, structure of regular files, conversion of a path name to an Inode, Inode assignment to a new file, allocation of disk-block.

UNIT II

System calls for the file systems: - Open, Read, Write, and Close.

PIPES:-the pipe system call, opening a named pipes, reading and writing pipes, closing pipes, DUP, mounting and amounting file system, LINK, UNLINK.

UNIT III

The structure of processes: - process states and transitions, layout of system memory, the context of a process, saving the context of the process, manipulation of the process address space.

Process Control: - process creation, signals, process termination, awaiting process termination, the user id of a process, changing the size of the process.

UNIT IV

Shell Programming: - Study of different types of Shells like C Shell, Bourne Shell etc. Shell variable, Shell Script, Shell Command.

Looping and Making choices: - For Loop, While and Until, Passing Arguments to Scripts. Programming in different shells.

UNIT V

Inter Process Communication: - Process Tracing, Network Communication, Sockets.

Multiprocessor System: - Problem of multiprocessor systems, Solution with Master and Slave Processor, Solution with Semaphores.

TEXT BOOKS:

1. The Design of the UNIX Operating S Paperback – 2015 by Maurice J Bach, Pearson Education India.
2. Advanced UNIX: A Programmer Guide by Stephen Prata, Howard W. Sams, 1987.

REFERENCE BOOKS:

1. William Stallings, Operating Systems: Internals and Design principles, Fifth Edition, Prentice Hall, 2005.
2. Understanding the LINUX Kernel, Daniel P. Bovet and Marco ceasti, O'RELLY publications, 2005.

B.Tech. VI Semester

L T P
0 0 3

(1505609) DATA MINING AND UML LAB

PART-A: Data Mining

Course Objective:

•□The different **data mining models** and techniques will be discussed in this course. Data mining and data warehousing applications in bioinformatics will also be explored.

Credit Risk Assessment

Description: The business of banks is making loans. Assessing the credit worthiness of an applicant is of crucial importance. You have to develop a system to help a loan officer decide whether the credit of a customer is good, or bad. A bank's business rules regarding loans must consider two opposing factors. On the one hand, a bank wants to make as many loans as possible. Interest on these loans is the banks profit source. On the other hand, a bank cannot afford to make too many bad loans. Too many bad loans could lead to the collapse of the bank. The bank's loan policy must involve a compromise: not too strict, and not too lenient.

To do the assignment, you first and foremost need some knowledge about the world of credit. You can acquire such knowledge in a number of ways.

1. Knowledge Engineering. Find a loan officer who is willing to talk. Interview her and try to represent her knowledge in the form of production rules.
2. Books. Find some training manuals for loan officers or perhaps a suitable text book on finance. Translate this knowledge from text form to production rule form.
3. Common sense. Imagine yourself as a loan officer and make up reasonable rules which can be used to judge the credit worthiness of a loan applicant.
4. Case histories. Find records of actual cases where competent loan officers correctly judged when, and when not to, approve a loan application.

The German Credit Data:

Actual historical credit data is not always easy to come by because of confidentiality rules. Here is one such dataset, consisting of 1000 actual cases collected in Germany. credit dataset (original) Excel spreadsheet version of the German credit data (Down load from web).

In spite of the fact that the data is German, you should probably make use of it for this assignment. (Unless you really can consult a real loan officer !)

A few notes on the German dataset

- DM stands for Deutsche Mark, the cents Canadian (but looks and acts like a quarter).
- Owns telephone. German phone rate so fewer people own telephones.
- foreignhere_ areworkermillionsofthese. In Germany (many from Turkey). It is very hard to get German citizenship if you were not born of German parents.
- There are 20 attributes used to classify the applicant into one of two categories, good or bad.

Subtasks : (Turn in your answers to the following tasks)

1. List all the categorical (or nominal) attributes and the real-valued attributes separately.
2. What attributes do you think might be crucial in making the credit assessment? Come up with some simple rules in plain English using your selected attributes.
3. One type of model that you can create is a Decision Tree - train a Decision Tree using the complete dataset as the training data. Report the model obtained after training.
4. Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly ? (This is also called testing on the training set) Why do you think you cannot get 100 % training accuracy ?
5. Is testing on the training set as you did above a good idea ? Why or Why not ?
6. One approach for solving the problem encountered in the previous question is using cross validation ? Describe what is cross-validation briefly. Train a Decision Tree again using cross-validation and report your results. Does your accuracy increase/decrease ? Why ? (10 marks)

PART-B: UML

Course Objectives:

- Practice the notation for representing various UML diagrams
- Analyze and design the problem by representing using UML diagrams
- Become familiar with all phases of OOAD

Course Outcomes:

- Find Solutions to the problems using object oriented approach
- Represent using UML notation and interact with the customer to refine the UML diagrams

The Lab is not just about using UML notation. The analysis and design part has to be given importance. Students have to analyze, design and model the following using UML notation.

Students have to describe the problem in their own words, identify the classes, and their collaboration in solving the problem. Give paper justification for selection. Student has to draw the following diagrams using UML notation.

1. Use case diagram
2. Class diagram
3. Sequence diagram
4. Collaboration diagram
5. State diagram
6. Activity diagram
7. Component diagram
8. Deployment diagram

Problems that may be considered are

1. Library Management System
2. ATM System

TEXT BOOKS:

1. CCNA study guide by Cannon, Caudle, Chiarella, Cenage Learning, Indian Edition.
2. “Object Oriented Analysis and Design with Application”, Grady BOOCH, Robert A, Maksimchuk, Michael W, ENGLE, Bobbi J.Young, Jim Conallen, Kellia Houston, Pearson 3rd Edition, 2013.
3. “The Unified Modeling Language User Guide”, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson 12th Impression, 2012.

B.Tech. VI Semester

L T P
0 0 3

(1505610) MOBILE APPLICATION DEVELOPMENT LABORATORY

Course Objectives:

- To understand fundamentals of android operating systems.
- Illustrate the various components, layouts and views in creating android applications.
- To understand fundamentals of android programming.

Course Outcomes:

- Create data sharing with different applications.
- Develop applications using services and publishing android applications.
- To demonstrate their skills of using Android software development tools.

1. Setting up the Development Environment

1.1 Installation of JDK and Setting path

1.2 Downloading and Installing Android Studio

2. Creating "Hello World" Application and viewing the output through emulator.

3. Creating the Application by using Activity class

i) onCreate()

ii) onStart()

iii) onResume()

iv) onPause()

v) onStop()

vi) onDestroy()

vii) onRestart()

4. Create the Application using the Edit Text control.

5. Create the Application Choosing options.

i) CheckBox

ii) RadioButton

iii) RadioGroup

iv) Spinner

6. Create the applications using different layouts.

i) Linear Layout

ii) Relative Layout

iii) Absolute Layout

7. Create the application for doing arithmetic operations. (Calculator)

8. Create the application to play the audio and video clips.

9. Create the application by using menus and action bar.