### SCHEME OF STUDIES

BCA, MCA Integrated Degree Programme

#### 4th Year

**Semester-VII**

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Course No</th>
<th>Course Name</th>
<th>Periods</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L-T-P</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>CA-1401</td>
<td>Introduction to XML</td>
<td>3-0-0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Elective-3</td>
<td></td>
<td>3-0-0</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>CA1402</td>
<td>Software Testing</td>
<td>3-0-0</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>CA-1403</td>
<td>System and Network administration</td>
<td>3-1-0</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>CA-1404</td>
<td>Advanced Java</td>
<td>3-1-0</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>CA-1405</td>
<td>Information Retrieval Systems</td>
<td>3-1-0</td>
<td>4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Course No</th>
<th>Course Name</th>
<th>Periods</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>L-T-P</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>CA-1451</td>
<td>XML Lab</td>
<td>0-0-2</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>CA-1453</td>
<td>System and Network administration lab</td>
<td>0-0-2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>CA-1454</td>
<td>Advanced Java Lab</td>
<td>0-0-2</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>PD-393</td>
<td>Advanced professional Development</td>
<td>0-0-2</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>PD-491</td>
<td>Co-Curricular Activities</td>
<td>0-0-2</td>
<td>1*</td>
</tr>
</tbody>
</table>

**TOTAL CONTACT HOURS**  
18-3-6(29)

**TOTAL CREDITS**  
25

**FINAL EVALUATION IN GRADES**

(L-T-P-Cr) – Lectures-Tutorials-Practicals-Credits, CW – Class Work  
MSE – Mid-Semester Exam, ESE – End-Semester Exam  
*One credit to be earned in Semester VIII through Co-curricular Activities outside contact hours. However, a student is to register for this course in both the semesters of forth year.*
## SCHEME OF STUDIES
### BCA, MCA Integrated Degree Programme

#### 4th Year
##### Semester-VIII

### THEORY

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Course No.</th>
<th>Course Name</th>
<th>Periods (L-T-P)</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA-1406</td>
<td>Embedded system design</td>
<td>3-1-0</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>CA-1407</td>
<td>Mobile Computing</td>
<td>3-0-0</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>Elective -4</td>
<td>3-0-0</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>CA-1408</td>
<td>Object oriented Software Engineering &amp; UML</td>
<td>3-1-0</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>CA-1409</td>
<td>Introduction to ERP</td>
<td>3-1-0</td>
<td>4</td>
</tr>
</tbody>
</table>

### PRACTICAL / DRAWING / DESIGN

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Course No.</th>
<th>Course Name</th>
<th>Periods (L-T-P)</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA-1456</td>
<td>Embedded system Lab</td>
<td>0-0-2</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>CA-1458</td>
<td>Object Oriented Software Engineering &amp; UML Lab</td>
<td>0-0-2</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>CA-1481</td>
<td>Minor Project -2</td>
<td>0-0-10</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>Co-Curricular Activities</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>PD-491</td>
<td></td>
<td></td>
<td>1*</td>
</tr>
</tbody>
</table>

**TOTAL CONTACT HOURS / TOTAL CREDITS**

15-3-14(32)   25 +1*

**FINAL EVALUATION IN GRADES**

(L-T-P-Cr) – Lectures-Tutorials-Practicals-Credits, CW – Class Work

MSE – Mid-Semester Exam, ESE – End-Semester Exam

*One credit to be earned in Semester-X through Co-curricular Activities outside contact hours.

### LIST OF DEPT. ELECTIVES

#### BCA-MCA INTEGRATED

#### Dept. Electives List 1

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Course No.</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA-1323</td>
<td>Advanced Computer Architecture</td>
</tr>
<tr>
<td>2</td>
<td>CA-1324</td>
<td>Advanced Database Management System</td>
</tr>
<tr>
<td>3</td>
<td>CA-1325</td>
<td>Cryptography &amp; Data Compression</td>
</tr>
</tbody>
</table>

#### Dept. Electives List 2

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Course No.</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA-1326</td>
<td>Expert System</td>
</tr>
<tr>
<td>2</td>
<td>CA-1327</td>
<td>Natural language processing</td>
</tr>
<tr>
<td>3</td>
<td>CA-1328</td>
<td>Digital image processing</td>
</tr>
</tbody>
</table>

#### Dept. Electives List 3

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Course No.</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA-1421</td>
<td>Compiler Design</td>
</tr>
<tr>
<td>2</td>
<td>CA-1422</td>
<td>Soft Computing</td>
</tr>
<tr>
<td>3</td>
<td>CA-1423</td>
<td>Bluetooth Technology</td>
</tr>
</tbody>
</table>

#### Dept. Electives List 4

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Course No.</th>
<th>Course Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CA-1424</td>
<td>Distributed computing</td>
</tr>
<tr>
<td>2</td>
<td>CA-1425</td>
<td>Information Storage &amp; Management</td>
</tr>
<tr>
<td>3</td>
<td>CA-1426</td>
<td>Human computer Interaction</td>
</tr>
<tr>
<td></td>
<td>CA-1427</td>
<td>Android Application development</td>
</tr>
</tbody>
</table>
INTRODUCTION TO XML

L T P Cr
3 0 0 3

PRE-REQUISITES: Knowledge of internet and web development, data mining, computer networks, DBMS.

Unit 1 - XML Fundamentals
Introducing the eXtensible Markup Language (XML); Extending and Adopting Markup Languages From SGML to XML and XHTML; Benefits and Drawbacks of XML; Representing Mixed Data and Context with XML; Creating an XML Document; Defining Structure; Rules for Well-Formed and Valid XML; Changing XML Documents

Unit 2 - XML Syntax & Namespaces
Tag Attributes and Naming Rules; Empty and Non-Empty Elements; Processing Instructions for XML; Accessing Data from XML Elements; XML Namespaces; Prefixes and Declarations; Default and Multiple Namespaces

Unit 3 - XML Document Type Definition (DTD)
XML DTD as an XML Schema; Creating a DTD; Element Conditions and Quantifiers; Referencing DTD Declarations; Validating DTD Compliance

Unit 4 - XML Schema Definition (XSD)
Element and Attribute Declarations; Simple, Complex, and Built-in Types; Named and Anonymous Types; Associating XML with a Schema; Validating XSD Compliance

Unit 5 - XQuery and XPath
Why XQuery and XPath; XPath Nodes and Syntax; Seven Node Types; Node Paths and Predicates; Node Axes and Functions; XQuery Structure and Usage; XPath and XSD in XQuery; Terms and Syntax; Selecting and Filtering Elements

Unit 6 - Publishing XML
Stylesheet Languages; Using Style Sheets with XML; Page Layout with Cascading Style Sheets (CSS); CSS Syntax and Classes; Introduction to XSL

Unit 7 - Applying XML
XML and Web Services; HTML with XML; XML and e-Commerce; XML Databases; Storing Binary Data in XML; XML parser.

TEXT BOOK

REFERENCE Books
**OBJECTIVE** To develop deep understanding about computer software testing methodologies and tools

**PRE-REQUISITES** Knowledge of programming, software engineering,

1. **FUNDAMENTALS AND TESTING TYPES:** First, second and later cycles of testing, Objectives and limits of testing, Overview of software development stages, Planning and Design stages and testing during these stages. Glass box code, Regression and Black box testing, Software errors, Categories of software error

2. **REPORTING AND ANALYZING BUGS:** Problem reports, Content and Characteristics of Problem Report, analysis and Tactics for analyzing a reproducible bug, Making a bug reproducible

3. **PROBLEM TRACKING SYSTEM:** Objective of Problem Tracking System, tasks of the system, Problem tracking overview, users of the tracking system, mechanics of the database

4. **TEST CASE DESIGN:** Characteristics of a good test, equivalence classes and boundary values, visible state transitions, Race conditions and other time dependencies, load testing. Error guessing, Function equivalence testing, Regression Testing, General issues in configuration testing, printer testing

5. **LOCALIZATION AND USER MANUALS TESTING:** Translated text expands, Character sets, Keyboards, Text filters, Loading, saving, importing, and exporting high and low ASCII, Operating system Language, Hot keys, Error message identifiers, Hyphenation rules, Spelling rules, Sorting Rules, Uppercase and Lowercase conversion, Printers, Sizes of paper, CPU’s and video, Rodents, Data formats and setup options, Rulers and measurements, Culture-bound Graphics and output, European product compatibility, Memory availability, automated testing, Testing User Manuals, Effective documentation, documentation tester’s objective, How testing documentation contributes to software reliability

6. **TESTING TOOLS AND TEST PLANNING:** Fundamental tools, Automated acceptance and regression tests, standards, Translucent box testing Overall objective of the test plan: product or tool? Detailed objective, type of test, strategy for developing components of test planning documents, components of test planning documents, documenting test materials

7. **MANAGEMENT ISSUES OF TESTING:** Software Development tradeoffs and models, Quality-related costs, The development time line, Product design, alpha, Pre-beta, Beta, User Interface freeze, Pre-final, Final integrity testing, Project post-mortems, Legal consequences of defective software, Managing and role of a testing group, independent test agencies

**TEXT BOOK**

**REFERENCE BOOKS**
OBJECTIVE
To lay a strong foundation for overall system and network management

PRE-REQUISITES
Knowledge of computer organization and architecture, operating system, computer networks

1. INTRODUCTION TO SYSTEMS AND NETWORK ADMINISTRATION: Scope of systems and network administration; goals of systems and network administration; system components and their management

2. OPERATING SYSTEMS UTILITIES: Windows and Unix variants; file systems and standards (UFS, NFS, NTFS); processes and job control; privileged, user and group accounts; logs and audits; advanced scanning concepts and tools; advanced sniffer.

3. HOST MANAGEMENT: Booting and shutting down of an operating system; formatting, partitioning and building a file system; file system layout; concept of swap space; OS installation; installation and configuration of devices and drivers.

4. SERVER CONFIGURATION & TROUBLESHOOTING: Linux/Windows server configuration; superuser/ administrator privileges; user management, controlling user resources; disk space allocation and quotas; process management (monitoring, killing/stopping, monitoring activity); file system repair, backup and restoration; integrating multiple operating systems; system sharing; authentication process.

5. NETWORK ADMINISTRATION: Introduction to network administration approaches; addressing and subnetting: fixed vs. variable masks, VLAN principles and configuration, routing concepts, static and dynamic routing; routing protocols (RIP, OSPF, BGP).

6. ADVANCED NETWORK MANAGEMENT SERVICES: Configuring a Linux/Windows box as a router; dial-up configuration & authentication: PPP, RAS; configuring a DNS server; configuring Sendmail service; configuring a web server; configuring a proxy server; TCP/IP troubleshooting (ping, traceroute, ifconfig, netstat, ipconfig, network management).

7. NETWORK SECURITY: Security planning; categories of security; access control and monitoring; wrappers; firewalls: filtering rules, detection and prevention of Denial of Service (DOS) attacks; automatic identification of configuration loop holes; security information resources: cert, installing and upgrading system software, use of scripting tools.

REFERENCE BOOKS
8. Subramaniam S., "Network Management: Principles & Practice", Addison Wesley,
OBJECTIVE
To relay the theoretical and practical knowledge of Advanced Java programming language

PRE-REQUISITES
Basic Knowledge of programming language and object oriented programming

1. CORE JAVA: Introduction to Java; Data types; variables; operators; Arrays; Control Statements; Classes & Methods; Inheritance; Exception Handling; Multithreading; Collections; I/O streams; AWT & Applet Programming; Swings.
2. NETWORK PROGRAMMING: Networking basics; Socket; port; Proxy servers; Internet addressing and URL; java.net – networking classes and interfaces; Implementing TCP/IP based Server and Client. Classes to be covered Socket; ServerSocket; IP Address; URL connections
3. JAVA DATABASE CONNECTIVITY: Types of JDBC Drivers; Writing JDBC applications using select; insert; delete; update; Types of Statement objects (Statement; PreparedStatement and CallableStatement); ResultSet; ResultsetMetaData; Inserting and updating records.
4. REMOTE METHOD INVOCATION AND JAVA BEANS: Introduction of RMI & Architecture; Implementing RMI Methods; Introduction to Java Bean; Rules for writing a Simple Bean; Using Beans to Build an Application.
5. SERVLETS: Configuring directory structure for a web application; Servlet API Overview; Writing and running Simple Servlet. Servlet Life; Cycle; GenericServlet and HttpServlet; ServletConfig & ServletContext; Writing servlet to Handle Get and Post Methods; Reading user request data; Concept of cookie; Reading and writing cookies
6. JAVA SERVER PAGES: Why JSP? JSP Directives; writing simple JSP page; Scripting Elements; JSP Actions: JSP & Java Beans; JSP Actions: include; forward and plugin; Managing sessions using JSP; JSP & Databases; Error Handling in JSP; Writing custom tags; Different scopes in a JSP page; Using JDBC in JSP; Study and Development of a Web Application and an Assignment.
7. INTRODUCTION TO STRUCTS: A Web Application Framework – struts-config.xml; Understanding MVC architecture; ActionServlet; ActionForm; ActionMapping; Action classes.

REFERENCE BOOKS
7. O’Reilly, “Servlet and JSP”.
OBJECTIVE
The main objective behind this course is to lay a strong foundation for overall knowledge of information retrieval systems.

1. INTRODUCTION: Definition; Objectives; Functional Overview; Relationship to DBMS; Digital libraries and Data Warehouses.
2. INFORMATION RETRIEVAL SYSTEM CAPABILITIES: Search; Browse; Miscellaneous.
3. CATALOGING AND INDEXING: Objectives; Indexing Process; Automatic Indexing; Information Extraction.
4. DATA STRUCTURES: Introduction; Stemming Algorithms; Inverted file structures; N-gram data structure; PAT data structure; Signature file structure; Hypertext data structure.
5. AUTOMATIC INDEXING: Classes of automatic indexing; Statistical indexing; Natural language; Concept indexing; Hypertext linkages.
6. DOCUMENT AND TERM CLUSTERING: Introduction; Thesaurus generation; Item clustering; Hierarchy of clusters.
7. USER SEARCH TECHNIQUES: Search statements and binding; Similarity measures and ranking; Relevance feedback; Selective dissemination of information search; Weighted searches of Boolean systems; Searching the Internet and hypertext. Information Visualization: Introduction; Cognition and perception; Information visualization technologies.

TEXT BOOK

REFERENCE BOOKS

OBJECTIVE
Embedded computers are found everywhere from home appliances to automobiles to medical devices. Designing an embedded computing system is a challenging task because the requirements include manufacturing cost, performance, power consumption, user interface, hard deadlines and rich functionality. The objective is to teach embedded system design process which includes requirements, specification, architecture, components and system integration phases.

The course will have real-life design examples to illustrate the design process and the students are encouraged to design embedded systems to gain experience. PRE-REQUISITES
Knowledge of logic design, assembly language programming, computer organization and architecture, microprocessors and interfacing, operating systems

1. INTRODUCTION: Introduction, overview, design process; instruction set architecture; CISC and RISC instruction set; architecture; basic embedded processor/ microcontroller architecture; memory system architecture; I/O subsystem; co-processors and hardware accelerators; processor performance enhancement; 16 & 32 bit microprocessor and micro-controller and DSP hardware with reference to embedded system.

2. REAL TIME OPERATING SYSTEMS: Real time operating system overview; basic features of an operating system, kernel features; processes and threads, context switching; scheduling, inter-process communication; real-time memory management; I/O processes; exposure to Windows CE, QNX, micro kernels and \O/S of introduction to process models; interrupt routines in an RTOS environment; encapsulating semaphores and queues; hard real-time scheduling considerations; saving memory space.

3. DESIGNING EMBEDDED COMPUTING PLATFORM: Using CPU bus, memory devices and their characteristics, I/O devices, component interfacing, memory interfacing; I/O device interfacing, interfacing protocols, designing with processors: system architecture, hardware design, FPGA based design; implementation: development environment, debugging techniques, design examples: data compressor, alarm clock.

4. PROGRAMMING EMBEDDED SYSTEMS: Program design, programming languages, use of high level languages, programming and run-time environment, basic compilation techniques, analysis and optimization of execution time, analysis and optimization of energy and power, analysis and optimization of program size, program validation and testing

5. NETWORK BASED EMBEDDED APPLICATIONS: Network fundamentals, layers and protocols, network architectures, distributed embedded architectures, elements of protocol design, high level protocol design languages, network based design, internet-enabled systems: protocols for industrial and control applications; internetworking protocols; wireless applications

6. EMBEDDED CONTROL APPLICATIONS: Introduction, open-loop and closed loop control systems; PID controllers, fuzzy logic controller; application examples: washing machine, automotive systems, auto-focusing digital camera, air-conditioner

7. EMBEDDED SYSTEM DEVELOPMENT: Design methodologies; architectural design; design examples: telephone PBX, PDA, set-top box, elevator control system, ATM system, fault-tolerance techniques, reliability evaluation techniques

TEXT BOOK
Simon David E., —An Embedded System Primerl, Addison-Wesley, 1999
REFERENCE BOOKS
OBJECTIVE
Recent developments in portable devices and high-bandwidth, ubiquitous wireless networks has made mobile computing a reality. Indeed, it is widely predicted that within the next few years' access to Internet services will be primarily from wireless devices, with desktop browsing the exception. Such predictions are based on the huge growth in the wireless phone market and the success of wireless data services. This course will help in understanding fundamental concepts, current developments in mobile communication systems and wireless computer networks.

PRE-REQUISITES
Computer Networks and wireless communication

1. **INTRODUCTION TO WIRELESS TRANSMISSION**: Applications, A short history of wireless communication, Frequency for radio transmission, Signals, Antennas, Signal propagation, Multiplexing, Modulation, Spread spectrum, Cellular systems.
2. **MEDIUM ACCESS CONTROL**: Motivation for a specialized MAC: Hidden and Exposed terminals. Near and Far terminals; SDMA, FDMA, TDMA: Fixed TDM, Classical Aloha, Slotted Aloha, Carrier sense multiple access, Demand assigned multiple access, PRMA packet reservation multiple access, Reservation TDMA, Multiple access with collision avoidance, Polling, Inhibit sense multiple access; CDMA: Spread Aloha multiple access
4. **SATELLITE & BROADCAST SYSTEMS**: History, Applications, Basics: GEO, LEO, MEO; Routing, Localization, Handover, Examples, Cyclic repetition of data, Digital audio, broadcasting: Multimedia object transfer protocol; Digital video broadcasting
7. **MOBILE TRANSPORT LAYER**: Traditional TCP: Congestion control, Slow start, Fast retransmit/fast recovery, Implications on mobility; Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit/fast recovery, Transmission/time-out freezing, Selective retransmission, Transaction oriented TCP

REFERENCE BOOKS
1. Jochen Schiller, “Mobile Communications”, Addison Wesley/Pearson Education, 2005
OBJECTIVE
To provide adequate knowledge about the different types of system software available & to introduce the object oriented concepts to the programming skills.

PRE-REQUISITES
Understanding of object orientation and knowledge of software engineering

1. REVIEW OF OBJECT ORIENTED SYSTEMS: Design objects, class hierarchy, inheritance, polymorphism, object relationships and associations, aggregations and object containment, object persistence, meta classes, object oriented systems development life cycle, Software development process, object oriented systems development: a use case driven approach.
2. OBJECT ORIENTED ANALYSIS: Analysis process, use case driven object oriented analysis, use-case model, object classification, theory, different approaches for identifying classes, classes, responsibilities and collaborators, identifying object relationships, attributes and methods, super sub class relationships, A- part of relationships aggregation, class responsibilities, object responsibilities.
3. OBJECT ORIENTED DESIGN: Object oriented design process, corollaries, design axioms, design patterns, object oriented design philosophy
4. METHODOLOGY FOR OBJECT ORIENTED DESIGN: Object modeling technique as software engineering methodology, Rumbaugh methodology, Jacobson Methodology, Booch Methodology
5. UNIFIED APPROACH FOR OBJECT ORIENTED DESIGN: Patterns, Frameworks, the unified approach, unified modeling language (UML).
6. UML: Why we model, types of models, principles of modelling, object oriented modelling, object oriented concepts, UML notation, object oriented analysis: use case diagrams, interaction diagrams, activity diagrams, object oriented design: class diagrams, object diagrams, state diagrams, collaboration diagrams, post-testing: deployment diagrams, patterns, frameworks
7. USING UML FOR OOD: UML object constraint language, designing classes: the process, class visibility, refining attributes, designing methods ad protocols, packages and managing classes, designing interface objects, view layer interface design, macro and micro level interface design process

REFERENCE BOOKS
OBJECTIVE
To provide knowledge about the enterprise resource planning tools, models and techniques

PRE-REQUISITES
Knowledge of internet and web development, data mining, computer networks, software engineering

Unit 1. ENTERPRISE RESOURCE PLANNING – ERP overview; need of ERP; growth of ERP; benefit; Proper and improper ERP implementation; data ware housing; data mining; OLAP.

Unit 2. ERP AND RELATED TECHNOLOGIES: Business process reengineering (BPR); management information system (MIS); decision support systems (DSS); executive support systems (ESS); data warehousing, data mining; online analytical processing (OLTP); supply chain management (SCM); customer relationship management (CRM).

Unit 3. ERP MODULES AND VENDORS: Finance; production planning, control & maintenance ; sales & distribution; human resource management (HRM); inventory control system; quality management; ERP market

Unit 4. ERP IMPLEMENTATION LIFE CYCLE: evaluation and selection of ERP package ;project planning; implementation team training & testing ; end user training & going live; post evaluation & maintenance; introduction to hidden costs, vendors, consultant employees.

Unit 5. ERP & E-COMM, FUTURE DIRECTIVES: in ERP, ERP and internet, critical factors guiding selection and evaluation, strategies for successful implementation, impediments and initiatives to achieve success, critical success and failure factors, integrating ERP into organizational culture.

Unit 6. ERP CASE STUDIES: Post implementation review of ERP packages in manufacturing, services, and other organizations; using ERP tool: either sap or oracle format to case study.


TEXT BOOK

REFERENCE BOOKS
LIST OF EXPERIMENTS

1. Creating XML documents
2. Creating DTD to validate XML documents
3. Creating Schemas to validate XML documents
4. Creating Simple XML documents
5. Working with Elements
6. Working with Attributes
7. Processing XML documents with DOM
8. Creating Valid Documents
9. Declaring Document Elements
10. Working with Attributes Types
11. Creating Schemas documents
12. Working with simple types
13. Working with complex types
14. Displaying data with XSLT
15. Using XPath functions

REFERENCE BOOKS

2. XML 1.1 Bible, 3rd Ed (Reference), Harold, Elliotte Rusty; Wiley.

LIST OF EXPERIMENTS

1. Management of the users and the domain.
2. Configuring DHCP.
3. Setting up the local security policy.
4. Start and stop services from user window and command prompt.
5. Use of event viewer.
6. Use of the performance monitor.
7. Management of the IIS and FJP server.
8. Setting up of local area network.
10. Use of utilities: Ping, Tracert, Netstat, Net, IP configuration, Path ping
11. Use of network monitor.
12. Setting up of a DNS.
13. Setting up and use “Terminal Clinet Services”.

REFERENCE BOOKS

LIST OF EXPERIMENTS
1. Create an User-defined exception class using the extends keyword. Write a constructor for this class that takes a string argument and stores it inside the object with a string handle. Write a method that prints out the stored string. Create a try-catch clause to exercise the created exception.
2. Create a Java program using thread
3. Write a program to accept two names as command line parameters. Check whether each of them exist in c:\java directory. If it exists, display its name and size, else, display the message that it does not exist. Further, if the extension of the file is “html” then it has to be deleted.
4. Write a java program to accept text from the user until he types “end” in a newline. The name of the file to store the contents should be accepted from the user at the command line. If no command line parameters are provided, the program should print error message and exit. Ensure if the same file has been created by listing the contents of the same.
5. Create an applet to obtain the list of fonts available with the current Java working environment.
6. Write an animation applet that makes an image appear from left to right, in stages.
7. Create a simple, non-editable combo box with a list of items, when selected one of the items, will display the string to the console and also printing the string which is being deselected, i.e., the string which already been selected.
8. Write a program that takes as input the telephone bill amount for 12 months and calculate the average telephone bill for the year. The program should also display the least and the maximum bill amounts.
9. Write a program through which the insert statement can be given at runtime. Use the program to insert the following test data in the master and details tables.
10. Write a program to delete a particular row according to required information.
11. Write a Echoserver and echo-client program that displays whatever is typed in the server on to the client using sockets.
12. Use socket programming to design a client/server application that takes the password as input and checks whether it is correct. The program should print the appropriate message.
13. Using beans to build an application
14. Using servlet develop a java program (Database connectivity)
15. Using RMI develop a client/server frame

OBJECTIVE
To equip the students with the basics of law, accounting, corporate policies, and ethics; the general awareness useful in leading a well informed life.

1. LAW FOR THE LAYMAN: Indian Judiciary System; Intellectual Property Rights (IPR); labour laws; employee rights; human rights; criminal laws, civil rights.
2. BASICS OF ACCOUNTING: Credit-Debit transactions; balance sheet; ledgers; receipts & vouchers; P & L statement; exercises.
3. MONEY MANAGEMENT: Types of taxes; how to manage taxes; investment options; an overview of stocks & shares; savings options; understanding important terms (depreciation, VAT, education cess).
4. CORPORATE RULES & POLICIES: The need; advantages; illustrations of certain rules & policies followed by selected corporate; code of conduct.
5. RIGHTS & DUTIES: An overview of the Indian constitution; fundamental rights & duties; directive principles of state policy; societal values; ideologies of some famous personalities.
6. TECHNOLOGY, POLITICS & RELIGIONS IN INDIA: various religions and their teachings; political developments in India; history of science & technology.
7. **HUMAN VALUES**: Ethics at work place; human values; morals & ethics; professional ethics; case studies.

**REFERENCE BOOKS**


<table>
<thead>
<tr>
<th>PD-491</th>
<th>CO-CURRICULAR ACTIVITIES</th>
<th>L</th>
<th>T</th>
<th>P</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

**OBJECTIVE**

To help the students in their all round growth and acquire attributes like team spirit, organizational ability, leadership qualities, etc.

**OPERATION**

The students are to take part in Co-curricular activities outside contact hours through clubs/ societies spread over all the three terms of the year. They are required to register for this course in each term and their performance will be evaluated in last term of the year.

<table>
<thead>
<tr>
<th>CA-11481</th>
<th>MINOR PROJECT -2</th>
<th>L-T-P</th>
<th>Cr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0-0-10</td>
<td>5</td>
</tr>
</tbody>
</table>

The project involves in-depth study on the topic, design, development, analysis fabrication and/or experimental work – Hardware and/or Software. It is intended to give an opportunity to a student to apply his knowledge to solve real-life problem. The student has to select a project work based on a topic of interest.
OBJECTIVE
To lay adequate foundation for design and development of compiler and other system software tools such as linkers, debuggers, assemblers, etc.

PRE-REQUISITES
Knowledge of data structures, basic programming concepts, theory of computations and operating systems

1. **INTRODUCTION:** Evolution of Components Systems Programming, Assemblers, Loaders, Linkers, Macros, Compilers, Software tools: Text editors, Interpreters, program generators, Testing software, Programming environment (such as Integrated Development Editors)

2. **SYSTEM SOFTWARE SPECIFICS:** Compiler: Brief overview of compilation process, Incremental compiler, structure of compiler: its different phases, Compiler construction tools. Assembler: Problem statement, single phase and two phase assembler, symbol table; Loader schemes, compile and go Loader, general loader schemes, absolute loader, Subroutine linkage, Reallocating loader, Direct linkage Loader, Binders, Linking loader, overlays

3. **LEXICAL AND SYNTAX ANALYSIS:** Role of lexical analyzer, design of lexical analyzer, regular expressions, Specification and recognition of tokens, input buffering, a language specifying lexical analyzer. Finite automata, conversion from regular expression to finite automata, and vice versa, minimizing number of states of DFA, Implementation of lexical analyzer. Syntax Analysis: Role of parsers, context free grammars, definition of parsing

4. **PARSING TECHNIQUE:** Shift- reduce parsing, operator precedence parsing, top down parsing, predictive parsing. LR parsers, SLR, LALR and Canonical LR parser

5. **SYNTAX DIRECTED TRANSLATIONS:** Syntax directed definition, construction of syntax trees, syntax directed translation scheme, implementation of syntax directed translation, three address code, quadruples and triples

6. **SYMBOL TABLE & ERROR DETECTION AND RECOVERY:** Symbol tables, its contents and data structure for symbol tables; trees, arrays, linked lists, hash tables. Errors, lexical phase error, syntactic phase error, semantic error

7. **CODE OPTIMIZATION AND CODE GENERATION:** Code generation, forms of objects code, machine dependent code, optimization, register allocation for temporary and user defined variables

**REFERENCE BOOKS**
OBJECTIVE
To introduce about incorporating more mathematical approach (beyond conventional logic system) into the artificial intelligence approaches for problem solving such as fuzzy logic, genetic algorithms, etc.

2. OPTIMIZATION: Least-square methods for system identification, recursive least square estimator; LSE for nonlinear models; derivative based optimization: descent methods, Newton’s method, conjugate gradient methods; nonlinear least-squares problems: Gauss Newton method, Levenberg-Markwardt method.
3. NEURAL NETWORKS: Different architectures; back-propagation algorithm; hybrid learning rule; supervised learning- perceptrons, back-propagation multilayer perceptrons, radial basis function networks; unsupervised learning – competitive learning network, Kohonen self-organizing networks, the Hopfield network.
4. FUZZY SET THEORY: Basic definition and terminology; basic concepts of fuzzy logic; set theoretic operators; membership functions: formulation and parameterization; fuzzy union, intersection and complement; fuzzy rules and fuzzy reasoning; fuzzy inference systems: Mamdani and Sugeno fuzzy models.
5. NEURO-FUZZY MODELLING: Adaptive neuro-fuzzy inference systems; neuro-fuzzy controller-feedback control; back-propagation through time and real-time recurrent learning; gradient-free optimization.
6. NEURO-FUZZY CONTROLLER IN ENGINEERING APPLICATIONS: Analytical issues in fuzzy logic control; fuzzy logic in intelligent agents.
7. GENETIC ALGORITHMS: Basics of genetic algorithms; design issues in genetic algorithm; genetic modeling; sequential and parallel genetic algorithms.

REFERENCE BOOKS
5. Sivanandam, “Introduction to Neural Networks with MATLAB 6.0”, Tata McGraw-Hill
OBJECTIVE
There is a long-standing interest from an increasing number of disciplines in migrating interactive computer graphics away from traditional keyboard/monitor/mouse interaction and out into the broader environment. Applications commonly obtain data from disparate media sources (e.g., sound, video, network data feeds, sensors, etc.); process this data in real-time, and ultimately (re)present information in different forms. Movement becomes color, video controls sound, and light drives motion.

1 INTRODUCTION TO WIRELESS TECHNOLOGIES: WAP services; serial and parallel communication; asynchronous and synchronous communication; FDM; TDM; TFM; spread spectrum technology.
2 INTRODUCTION TO BLUETOOTH: Specification; core protocols; cable replacement protocol Bluetooth radio: type of antenna; antenna parameters; frequency hoping.
3 BLUETOOTH NETWORKING: Wireless networking; wireless network types; devices roles and states; Ad-hoc Network; scatternet
4 CONNECTION ESTABLISHMENT PROCEDURE: Notable aspects of connection establishment; mode of connection; Bluetooth security; security architecture; security level of services; profile and usage model; generic access profile (gap), SDA, serial port profile; secondary Bluetooth profile.
5 HARDWARE: Bluetooth implementation; baseband overview; packet format; transmission buffers; protocol implementation: link manager protocol; logical link control adaptation protocol; host control interface; protocol interaction with layers
6 PROGRAMMING WITH JAVA: Java programming; J2ME architecture; Java; Bluetooth package interface; classes; exceptions, Java. Obex package: interfaces, classes
7 BLUETOOTH SERVICES REGISTRATION AND SEARCH APPLICATION: Bluetooth Client and Server Application. Overview of IRDA, homeRF, Wireless LANS, JINI

REFERENCE BOOKS

OBJECTIVE
This course will introduce the algorithms and technologies of distributed systems. It will teach both fundamentals as well as systems where these fundamentals are applied in practice.

PREREQUISITES
Knowledge of databases, networking, operating system and web technologies

1. DISTRIBUTED COMPUTING: History, forms of computing; strengths and weaknesses of distributed computing; OS basics; network basics; software engineering basics; CLIENT SERVER PARADIGM: issues, software engineering for a network service, connection oriented and connectionless servers, iterative server and concurrent server, stateful servers.
2. INTERPROCESS COMMUNICATION: Archetypal IPC program interface; event synchronization; timeouts and threading; deadlock and timeouts; data representation, data encoding; text based protocols, request response protocols; event and sequence diagram; connection vs. connectionless IPC.
3. DISTRIBUTED COMPUTING PARADIGMS AND SOCKET API: Paradigms; abstraction; socket metaphor; diagram socket API, stream mode socket API; sockets with non-blocking I/O; secure socket API
4. GROUP COMMUNICATION: Unicasting; multicasting, archetypal multicast API; connection oriented and connectionless; reliable, unreliable multicast; Java basic multicast API.
5. DISTRIBUTED OBJECTS: Message passing vs. distributed objects; archetypal distributed object architecture; distributed object systems; remote procedure calls; Java RMI architecture; API for Java RMI; Advanced RMI: Client callback, stub downloading, RMI security manager; allowing for stub downloading
6. SIMPLE OBJECT ACCESS PROTOCOL: SOAP request, SOAP response; Apache SOAP; invoking web service; implementing web service
7. ADVANCED DISTRIBUTED COMPUTING PARADIGMS: Message queue system paradigm; mobile agents; network service; object spaces

REFERENCE BOOKS
OBJECTIVE
Using a “building block” approach, the ISM curriculum provides a core understanding of storage technologies and progresses into system architectures, introduction to networked storage, and introduction to information availability. The course provides a comprehensive introduction to data storage technology fundamentals. Students will gain knowledge of the core logical and physical components that make up a storage systems infrastructure.

PRE-REQUISITES
Knowledge of computer networks

1. **INTRODUCTION**: Meeting today’s data storage needs - data creation; data creation: individuals, business; categories of data: data storage models; common data storage media and solutions - tape storage systems, optical data storage, disk based storage
2. **DATA CENTER INFRASTRUCTURE**: Example; key requirements of storage systems management activities
3. **STORAGE SYSTEMS ARCHITECTURE**: Storage system environment; components of a host; connectivity; physical disks; RAID array; disk storage systems; data flow exercise
4. **NETWORKED STORAGE**: Direct Attached Storage (DAS), Network Attached Storage (NAS), Fiber Channel Storage Area Network (FC SAN), IP Storage Area Network (IP SAN), Content Addressed Storage (CAS)
5. **BUSINESS CONTINUITY**: Introduction, overview, backup and recovery, local replication, remote replication.
6. **MONITORING AND MANAGING THE DATA CENTER**: Areas of the data center to monitor; considerations for monitoring the data center; techniques for managing the data center.
7. **SECURING STORAGE AND STORAGE VIRTUALIZATION**: Securing the storage infrastructure; virtualization technologies.

REFERENCE BOOKS
OBJECTIVE  To acquaint with the analysis, design and development aspects of enhancing interactions between human and computer system keeping in view the behavioural and psychological factors of any human

PRE-REQUISITES  Knowledge of computer organization and architecture, software engineering, computer graphics and multimedia technologies

1. INTRODUCTION: Introduction to Human-Computer Interaction (HCI); history; human factors of interactive software – goals of software engineering, goals of user interface design; motivation for human factors in design; accommodation of human diversity.

2. HUMAN INFORMATION PROCESSING: Human memory; thinking – reasoning and problem solving; skill acquisition; mental models; decision making; computer system interfaces: mechanics of input and output devices, review of computer architecture; performance characteristics of humans and systems; review of computer graphics

3. PRINCIPLES BEHIND HUMAN – SYSTEM INTERACTION: Paradigms of interaction; principles to support usability

4. USER CENTERED DESIGN OVERVIEW: Software development life cycle – actual, three pillars of design; usability engineering; iterative design and prototyping; design rationale; usability testing

5. TASK ANALYSIS: Basic concepts, task decomposition; knowledge based analysis; entity-relationship based analysis; sources of information; uses of task analysis

6. SYSTEM DESIGN: Use cases; scenarios; structuring information; information architecture; process flows, wireframes, mock-ups, comps.

7. DESIGN FOR UNIVERSAL ACCESS: Access concepts; accessible software; factors driving software accessibility; universal accessibility principles, guidelines and recommendations; case studies


REFERENCE BOOKS
  3. Fulleton Tracy, Swain Christopher, and Hoffman Steve, —Game Design Workshop: Designing, Prototyping, and Playtesting Games‖, CMP Books, USA, 2004
  4. Garrett Jesse James, —A Visual Vocabulary for Information Architecture‖, JJG.Net, USA, 200
OBJECTIVE:

Upon completion of this course the student will know the basic concepts and technique of developing applications for the Android phone, Be able to use the SDK and other development tools, Know the basic concepts of Android phone features and capabilities.

PRE-REQUISITES: previous programming experience in Java

Unit 1: Getting Started with Android Programming

Unit 2: Activities and Intents
Understanding Activities, Applying Styles and Themes to Activity, Hiding the Activity Title, Displaying a Dialog Window, Displaying a Progress Dialog, Linking Activities Using intents, Resolving Intent Filter Collision, Returning Results from an Intent, Passing Data Using an Intent Object, Calling Built-in Applications Using intents, Understanding the Intent Object, Using Intent Filters, Adding Categories, Displaying notifications

Unit 3: Getting to know the Android user interface
Understanding the Components of a Screen, Views and ViewGroups, Linear Layout, Absolute Layout, Table Layout, RelativeLayout, Frame Layout, ScrollView, Adapting to Display Orientation, Anchoring Views, Resizing and Repositioning, Managing Changes to Screen Orientation, Persisting State Information during Changes in Configuration, Detecting Orientation Changes, Controlling the Orientation of the Activity, Creating the User Interface Programmatically, Listening for UI notifications, Overriding Methods Defined in an Activity Registering Events for Views, Button, ImageButton, EditText, CheckBox, ToggleButton, RadioButton, and RadioGroup Views, Progress Bar View, Auto Complete Text View View

Unit 4: Storing and Retrieving Data
Saving and Loading User Preferences, Persisting Data to Files, using the file system, Saving to Internal Storage, Saving to External Storage (SD Card), Choosing the Best Storage Option Using Static Resource, Creating and Using Databases, Creating the DBAdapter Helper Class, Using the Database Programmatically, Upgrading the Database, Pre-Creating the Database, Bundling the Database with an Application, working with content provider classes

Chapter 5: Messaging and networking
SMS Messaging, Sending SMS Messages Programmatically, Getting Feedback After Sending the Message, Sending SMS Messages Using Intent, Receiving SMS Messages, Updating an Activity from a BroadcastReceiver, Invoking an Activity from a , BroadcastReceiver, Caveats and Warnings, Sending e-Mail, Networking, Downloading Binary Data, Downloading Text Files, Accessing Web Services, Performing Asynchronous Calls

Unit 6: Graphics, Animation and Multimedia

Unit 7: Developing Android Services & Publishing Android Applications
Creating Your Own Services, Performing Long-Running Tasks in a Service, Performing Repeated Tasks in a Service, Executing Asynchronous Tasks on Separate Threads Using Intent Service, Communicating between a Service and an Activity, Binding Activities to Services, Preparing for Publishing, Digitally Signing Your Android application
Applications, Deploying APK Files, Using the adb exe Tool, Using a Web Server, Publishing on the Android Market, Creating a Developer Profile, Submitting Your Apps

Books: