CS010 701: Web Technologies

Teaching scheme **Credits:** 4
2 hours lecture and 2 hours tutorial per week

Objectives

- *To impart the new concepts in Web Technologies*
- *To develop understanding about the different technologies used in the World Wide Web including XML, Perl,*

Module I (15hours)

XHTML

Evolution of HTML and XHTML- Standard XHTML Document Structure- Basic Text Markup- Images-Hypertext Links-Lists- Tables- Forms- Frames.

Cascading Style Sheets

Introduction to CSS – Levels of Style Sheets- Style Specification Formats- Selector Forms- Property Value Forms – Font Properties- List Properties – Color- Alignment of Text – Background Images- Span and Div Tags.

Module II (12 hours)

XML

Introduction to SGML – features of XML - XML as a subset of SGML – XML Vs HTML – Views of an XML document - Syntax of XML- XML Document Structure – Namespaces- XML Schemas- simple XML documents – Different forms of markup that can occur in XML documents - Document Type declarations – Creating XML DTDs – Displaying XML Data in HTML browser – Converting XML to HTML with XSL minimalist XSL style sheets – XML applications

Module III (12hours)

Perl

Origin and Use of Perl- Scalars and their Operations – Assignment Statements and Simple Input and Output – Control Statements- Fundamentals of Arrays – Hashes- References- Functions- Pattern Matching – File Input and Output – Simple programs in Perl -Using Perl for CGI Programming.

Module IV (12 hours)

PHP

Origin and Use of PHP- Overview of PHP- General Syntactic Characteristics- Operations and Expressions- Control Statements- Arrays- Functions-Pattern Matching- Form Handling- Files-Cookies-Session Tracking - Simple programs in PHP.

Module V (9 hours)

Rails

Overview of Rails- Document Requests- Processing Forms- Rails Application with Databases – Layouts.

Ajax

Overview of Ajax – Basics of Ajax – Rails with Ajax.

Reference Books

1) Robert W Sebesta, Programming with World Wide Web , 4th ed., Pearson Education ,New Delhi, 2009
2) Deitel & Deitel Internet & World Wide Web *How To Program* 4th ed., Pearson International Edition Education ,New Delhi, 2009
3) Deitel & Deitel, Nieto, Lin, Sadhu, XML How to Program, Pearson Education ,New Delhi, 2011
4) Kogent Learning Solutions Inc, Web Technologies Black Book, Dreamtech Press, New Delhi, 2009
5) Chris Bates, Web Programming Building Internet Applications 3rd ed., Wiley India Edition, New Delhi, 2009
6) Phil Ballard, Michael Moncur, Sams Teach Yourself Ajax, JavaScript and PHP, Pearson Education ,New Delhi, 2009.
7) Achyut S Godbole , Atul Kahate, Web Technologies TCP/IP Architecture and Java Programming, 2nd ed., Tata McGraw Hill Education Private Limited, New Delhi, 2010
8) Pankaj Sharma, Introduction to Web Technology, Katson Books, New Delhi, 2008
9) Bankim Patel, Lal Bihari Barik, Introduction to Web Technology & Internet, Acme Learning Private Limited, New Delhi, 2009

CS010 702: COMPILER CONSTRUCTION

Teaching scheme **Credits:** 4
2 hours lecture and 2 hours tutorial per week

1.) • *To introduce the various techniques involved in the translation of source programs into object programs by a compiler.*
2.) · *To understand the inner working of a compiler using the various data structures used in the translation process.*

Module 1 (12Hrs)

Introduction to compilers:-Phases of a compiler-Analysis and synthesis phases-Lexical analysis and its role-Review of finite automation and Regular Expressions-Specification of tokens using regular expressions-Implementing lexical analyzer using finite automation-Design of lexical analyzer using LEX

Module 2 (12 Hrs)

Syntax analyzer-Role of syntax analyzer-Review of context free grammar-derivation and parse trees-Basic parsing approaches-Top down parsing-Recursive Descent parsing –LL(1) parsing-Bottom up parsing-Shift reduce parsing-Operator precedence parsing-LR parsing-Simple LR, Canonical LR and LALR parsers- Design of syntax analyzer using YACC

Module 3 (12 Hrs)

Semantic analysis-Need for semantic analysis-Syntax directed definitions-S attributed definitions- L- attributed definitions-Translation schemes-Type system and Type checking-Design of a simple type checker

Storage Management:-Memory allocation strategies (static, stack and heap allocations)-Memory allocation in block structured languages-Accessing local and non local data-Array allocation and access-Procedure calls-Parameter passing methods-Runtime stack and storage management

Module 4(12 Hrs)

Synthesis phase:-Intermediate Code Generation (ICG)-Need for ICG-IC Formats-3 Address code-Triples and quadruples

Code optimization:-Need for code optimizer-Basic blocks and program flow graph-Machne dependent and machine independent optimizations-Optimization transformations-Local and global optimizations

Module 5(12 Hrs)

Code Generation-Basic issues in code generation-Data descriptors-Expression trees-Generating target code from expression trees-Symbol table handling-Symbol table requirements and organization. Error handling-Types of errors-Compile time errors and recovery-Runtime errors-Runtime Error Handling ,Cross Compilers and Incremental Compilers(Brief idea only)

Reference Books

1.) .Aho A Ravi Sethi and J D Ullman, Compilers Principles Techniques and Tools,Addison Wesley
2.) Kenneth C Louden, “Compiler Construction Principles and Practice”, Cenage Learning Indian Edition
3.) D M Dhamdhare, System programming and operating system, Tata McGraw Hill & Company
4.) Tremblay and Sorenson, The Theory and Practice of Compiler Writing - Tata McGraw Hill & Company

Mahatma Gandhi University

CS010 703: COMPUTER GRAPHICS

Teaching scheme **Credits:** 3

2 hours lecture and 1 hour tutorial per week

Objectives:-

To understand the basic concepts of Computer Graphics & display techniques.

Module I (3 Hrs)

Introduction: Applications of Computer Graphics, Raster scan and Random scan displays **[1]**– Video Display Devices, Display files – graphical input & output devices-Flat panel displays, Hardcopy Output Devices, Physical Interactive Devices , Data generation devices.**[2]**

Module II (10 Hrs)

2D Graphics: Output primitives-Line drawing algorithms – DDA, Bresenham's – Bresenham's Circle drawing algorithm – Other curves,polynomials and spline curves-2D viewing transformation-clipping-Cohen-Sutherland line clipping –polygon clipping-2D Transformations**[1]**

Module III (12 Hrs)

3D Graphics: 3D Transformations, 3D display methods, 3D Object Representation – Polygon Surfaces – Curved lines and surfaces-Quadric surfaces – Spline Representations – Cubic Spline Interpolation Methods-Bezier Curves and Surfaces – B-Spline Curves and Surfaces, Sweep representation,Octrees.**[1]**

Module IV (10 Hrs)

3D Rendering: Three-Dimensional Viewing – Projections **[3]**, Visible Surface Detection – Classification of Visible surface detection algorithms – Back-face Detection, Depth- Buffer Method, Scan-line Method. **[1,3]**

Module V (10 Hrs)

Rendering: Surface Rendering Methods- Basic illumination Models – Polygon–rendering Methods,Interpolative shading methods-Constant shading, Gouraud shading,Phong shading, Texture Mapping.**[3]**

Fractal Geometry Methods – Classification of Fractals – Self-Squaring Fractals, Ray Tracing and Ray Casting.**[1]**

REFERENCES:

1. Computer Graphics (C version) - Donald Hearn & Pauline Baker (Pearson Education Asia)
2. Procedural Elements for Computer Graphics –David F. Rogers, TATA McGraw Hill edition-second edition.
3. Computer Graphics - Zhigang Xiang & Roy A Plastack, Schaum's Series McGraw Hill edition.

CS010 704 : Object Oriented Modeling and Design

Teaching scheme **Credits:** 3
2 hours lecture and 1 hour tutorial per week

Objective

- *To impart ideas on building systems through the object oriented modelling approach using the Unified Modelling Language.*

Module 1 (10 hours)

Introduction: object oriented development-modeling concepts – object oriented methodology – models – object oriented themes-Object Modeling– links and associations – advanced links and association concepts – generalization and inheritance - grouping constructs – a sample object model

Advanced Object Modeling: aggregation – abstract classes – generalization as extension and restriction – multiple inheritance – metadata – candidate keys – constraints.

Module 2 (10 hours)

Dynamic modeling: Events and states – Operations – Nested state diagrams – Concurrency – Advanced dynamic modeling concepts – A sample dynamic model – Relationship of Object and Dynamic models.

Functional modeling: Functional models – Data Flow Diagrams - Specifying operations – Constraints – A sample functional model – Relation of functional to Object and Dynamic models.

Module 3 (10 hours)

Analysis: Analysis in object modeling, dynamic modeling and functional modeling, Adding operations- Iterating the analysis

System Design: Breaking system into subsystems - Identifying concurrency-allocating subsystems to processors and tasks, managing of data stores. Handling of global resources- handling boundary conditions-Common Architectural Frameworks

Module 4 (8 hours)

Object Design: Overview of Object design – Combining the three models – Designing algorithms – Design optimization – Implementation of control – Adjustment of inheritance - Design of association – Object representation – Physical packaging – Documenting design decisions-Comparison of methodologies

Module 5 (7 hours)

Unified Modeling language: Introduction, UML Diagrams – Class diagrams, Sequence diagrams, Object diagrams, Deployment diagrams, Use case diagrams, State diagrams, Activity diagram, Component diagrams – Case Study.

Reference Book

1. Object Oriented Modeling and Design -James Rumbaugh, Prentice Hall India
2. UML Distilled – Martin Fowler, Addison Wesley
3. Object- oriented Systems analysis and design using UML- 4th ed., Simon Bennet,Stephen McRobb, Ray Farmer. TMH.
4. Object Oriented Analysis and Design with Applications - Grady Booch, Pearson Education Asia

CS010 705: PRINCIPLES OF PROGRAMMING LANGUAGES

Teaching scheme **Credits:** 3

2 hours lecture and 1 hour tutorial per week.

Objectives

- *To provide an overview of the key paradigms used in developing modern programming languages.*
- *To explore the implementation details of languages to provide an understanding of the source program and its execution behavior.*

Module I (9 Hours)

Introduction – Role of programming languages - Programming domains - Language evaluation criteria - Influence on language design - Implementation methods - Virtual computers - Bindings - Concept of binding.

Module II (9 Hours)

Data types - Implementation of data types - Primitive, User defined – Names –Variables - Type checking - Strong Typing - Type compatibility -Scope – Lifetime - Referencing environments - Named constants – Virtualization - Heap management.

Module III (8 Hours)

Expressions , Assignments and Control Structures – Arithmetic expressions – Assignment statements-Compound statements - Selection statements - Iterative statements – Unconditional branching – Guarded commands.

Module IV (10 Hours)

Subprograms-Fundamentals-Design issues-Local Referencing Environment-Parameter passing methods –Subprogram names as parameters – Overloaded Subprograms – Generic Subprograms – Separate & independent compilation – Design issues for functions – Accessing non-local environments – User defined overloaded operators – Co-routines.

Module V (9 Hours)

Implementation of Subprograms – General semantics of calls & returns- Activation Records – Blocks – Recursion

Exceptions and Programming Paradigms - Exception handling in C++, Java, PL/I, Ada , Fundamentals of Functional programming language – Examples – LISP Interpreter -Overview of Logic programming - Basic elements of Prolog.

References

1. Robert W. Sebesta , “Concepts of Programming Languages” 4th Ed,2001.
2. Ravi Sethi ”Programming Languages-concepts and constructs”, Addison Wesley, 2nd Ed,1996.
3. Terrence W. Pratt , “Programming Languages” , Prentice Hall, 9th Ed,1996.
4. Michael L. Scott, “Programming Language Pragmatics” ,Elsevier, New Delhi,2009.
5. Thomson Learning, Kenneth .C. Louden, “Programming Languages: Principles And Practices” , 2nd Ed,2011.
6. Bjarne StroutStrup ,”Design and Evolution of C++”, Addison Wesley,1991.
7. James Gosling, “Java Programming Language “, Addison Wesley,2000.

CS010 706L01 : Real Time Systems
(Common to IT010 706L04 Real Time Systems)

Teaching scheme **Credits:** 4
2 hours lecture and 2 hour tutorial per week

Objectives

- *to learn , real-time operating systems, task scheduling, communication, fault tolerant techniques and , programming languages*

Module 1 (12 hours)

Introduction to Real Time Systems: Structure of real time systems, real time computer, task classes – Periodic, Aperiodic, critical, Non-critical, definition of real time systems – real time systems, embedded systems - Hard real time systems, soft real time systems, real time design issues.

Module 2 (12 hours)

Task Assignment and Scheduling: Uniprocessor scheduling algorithms –Rate monotonic Scheduling, Preemptive Earliest Deadline First (EDF), IRIS Tasks. Scheduling Aperiodic and Sporadic jobs in Priority Driven Sytems, Task Assignment-Utilization Balancing algorithm, Next Fit Algorithm for RM scheduling, Bin Packing for EDF, Myopic Offline Scheduling(MOS), Focused Addressing and Bidding, Buddy strategy. Fault Tolerant scheduling.

Module 3 (12 hours)

Communication – Communication Media and message sending topologies, network architecture issues, protocols – contention – based, token - based, stop and go multi loop, polled bus, hierarchal round robin, fault tolerant routing – clocks and synchronization– fault tolerant synchronization in hardware, synchronization in software.

Module 4 (12 hours)

Fault tolerance – definition, cause of failure, fault types, fault detection and containment, redundancy – hardware, software, time, information, integrated failure handling. Reliability Evaluation techniques- Obtaining parameter values, Reliability models for Hardware redundancy, software error models.

Module 5 (12 hours)

Programming Languages and Real Time databases – Desired language characteristics, Data Typing, Control Structures. Real time databases, characteristics, main memory databases, Transaction, Disk schedule algorithms, Databases for hard real time systems, maintaining serialization constituency.

References

1. Real Time Systems - C.M Krishna, Kang G. Shini (Tata McGraw Hill)
2. Real Time Systems- Jane W.S. Liu(Pearson)

CS010 706L02: DATA MINING AND DATA WAREHOUSING

Teaching scheme **Credits:** 4
2 hours lecture and 2 hours tutorial per week

Objectives

- *To impart an introduction to Data Mining.*
- *To develop basic knowledge of how data is transformed to Data Warehouses .*

Module I (12 hours)

Data Mining- Data Mining Funtionalities-Classification of Data Mining Systems-Data Mining Task Primitives- Major Issues in Data Mining
Data Preprocessing- Descriptive Data Summarization- Data Cleaning- Data Integration and Transformation- Data Reduction- Data Discretization and Concept Hierarchy Generation

Module II (14 hours)

Data Warehouse- A Multidimensional Data Model- Data Warehouse Architecture- Data Warehouse Implementation
Data Cube Computation and Data Generalization- Efficient methods for Data Cube Computation- Data Cube and OLAP Technology- Attribute Oriented Induction

Module III (10 hours)

Mining Frequent Patterns-Associations- Correlations-Basic Concepts-Efficient and Scalable Frequent Itemset Mining methods- Mining various kinds of Association Rules- From Association Mining to Correlation Analysis- Constraint Based Association Mining.

Module IV (12 hours)

Classification and Prediction- Issues regarding Classification and Prediction- Classification by Decision Tree Induction- Bayesian Classification – Rule Based Classification- Classification by Backpropagation- Support Vector Machines- Classification by Association Rule Analysis- Learning from Neighbors- Prediction- Accuracy and Error measures- Evaluating the accuracy of a Predictor- Ensemble methods- Model Selection.

Module V (12 hours)

Cluster Analysis- Types of Data in Cluster Analysis- Catagorization of Major Clustering methods- Partitioning methods- Hierarchical methods- Density based methods- Grid based methods- Model based Clustering methods- Clustering High Dimensional Data- Constraint based Cluster Analysis- Outlier analysis

Reference Books

1) Jiawei Han, Micheline Kamber, Data Mining Concepts and Techniques, 2nd edtn. , Elsevier New Delhi 2010
2) Alex Berson, Stephen J. Smith, Data Warehousing, Data Mining & OLAP Tata McGraw-Hill Publishing Company Limited, New Delhi, 2008
3) Pieter Adriaans, Dolf Zantinge, Data Mining, Pearson Education Ltd., New Delhi, 2008
4) Thomas W Miller, Data and Text Mining, A Business Applications Approach, Pearson Education Ltd., New Delhi, 2008
5) Galit Shmueli, Nitin R. Patel, Peter C. Bruce, Data Mining for Business Intelligence, Wiley India Pvt. Ltd.,New Delhi 2009.

CS010 706L03: Operating System Kernel Design
(common to IT010 706L05 Operating System Kernel Design)

Teaching scheme **Credits:** 4
2 hours lecture and 2 hours tutorial per week

Objectives

- *To provide knowledge about the operating system working principles.*
- *To discuss most of the significant data structures and algorithms used in the kernel.*

Module I (13 hours)

Basic Operating System Concepts – Kernel – Types: monolithic, microkernel – An Overview of Unix Kernels-The Process/Kernel Model, Reentrant Kernels – Signals sending and receiving – System calls – System Call Handler and Service Routines - Interrupts and Exceptions - Interrupt Handling - The Timer Interrupt Handler.

Module II (13 hours)

Processes - Process Descriptor - Process State, Process relationship – Creating Processes - Process Termination - Process Scheduling – Scheduling algorithm – SMP Scheduler.
Kernel Synchronization - Synchronization Techniques - Process Communication - System V IPC.

Module III (10 hours)

Paging in Linux - Memory Management - Page Frame Management - The Buddy System Algorithm - The Process's Address Space - The Memory Descriptor - Memory Regions - Page Fault Exception Handler.

Module IV (14 hours)

Overview of the Unix File System - The Virtual File System - role of the VFS - VFS Data Structures – File system Mounting.
The Ext2 File system - Disk Data Structures - Creating the File system - Data Blocks Addressing - Allocating a Data Block.

Module V (10 hours)

Managing I/O Devices - Associating Files with I/O Devices - Device Drivers - Character Device - Block Device.
Disk Caches - Buffer Cache - Writing Dirty Buffers to Disk - Page Cache.

Reference Books

1) Daniel P. Bovet, Marco Cesati, *Understanding the Linux Kernel*, First ed., O'Reilly, 2000
2) M Bech et al., *Linux Kernel Internals*, 2nd ed., Addison-Wesley, 1998
3) Maurice J. Bach, *The Design of the Unix Operating System*, First Edition, Pearson Education, 1999.
4) Abraham Silberschatz, Peter B.Galvin and Greg Gagne, "*Operating System Concepts*", John Wiley & Sons Inc, 8th Edition 2010.

CS010 706L04 : Digital image processing

Teaching scheme **Credits:** 4
2 hours lecture and 2 hour tutorial per week

Objectives

- *To learn the image fundamentals and mathematical transforms necessary for image processing.*
- *To learn the image enhancement techniques and image restoration procedures.*
- *To learn the image segmentation and representation techniques.*

Module I (14 hours)

Digital image representation : Elements of digital image processing systems - Image digitizers & scanners - Elements of visual perception - Brightness & contrast - colour perception & processing - pixel based transformation – geometric transformation – image file formats
Image sampling & Quantization - Two dimensional Sampling theorem - Reconstruction of image from its samples – Aliasing

Module II (14 hours)

Image Transforms : Two dimensional DFT & its properties - Walsh Transform, Hadamard Transform, Discrete Cosine Transform, Haar, Slant, and Karhunen – Loeve transforms

Module III (10 hours)

Image Enhancement : Point processing - Histogram processing - Spatial Filtering – image subtraction - image averaging - Enhancement in the frequency domain - colour Image processing.

Module IV (12 hours)

Image Restoration : Degradation model – Diagonalization of circulant matrices - Inverse filtering - Wiener filter methods – Constrained least mean square filtering
Image Coding & Compression- basic principles Image compression: Run length coding , predictive coding ,Basics of Image compression standards:

Module V (10 hours)

Image analysis : Segmentation – Thresholding – point, line and edge detection – Boundary detection - Region Based segmentation - image reconstruction – radon transform – projection theorem – convolution filter back projection - Fourier reconstruction method – applications of image processing.

References

1. Rafael C. Gonzalez - Richard E. Woods, *Digital Image Processing*, Pearson Education
2. Dutta Majumdar - *Digital Image Processing and Applications,* PHI
3. Madhuri A. Joshi – Digital Image Processing,PHI, New Delhi,2010
4. Anil K. Jain - Fundamentals of Digital Image processing," Prentice Hall India, 1989.
5. William K. Pratt - Digital Image Processing, John Wiley and sons, New delhi, 2010.
6. S.Jayaraman, S. Esakkiarajan. T. Veerakumar- Digital Image Processing,TMH,New Delhi, 2010.
7. Rosenfield and A. C. Kak - Digital Picture Processing, 2nd edition, Vols. 1 & 2,
    a. Academic Press, New York, 1982.

CS010 706L05: DATA PROCESSING AND FILE STRUCTURES

Teaching scheme

Credits: 4

2 hours lecture and 2 hour tutorial per week

Objectives

- *To develop an understanding about basic concepts of data processing in mainframe system.*
- *To enable the students to learn the detailed features of COBOL, database concepts.*

Module I (10 hours)

Introduction to mainframe system

Introduction—Evolution of Mainframe Systems, Introduction to COBOL & JCL, COBOL/JCL Relation ,Compiling and Linking Programs in Mainframes, VSAM—VSAM Data Sets—Mainframes Operating Systems(over view), z/OS , OS/2 , MVS --Features

Module II (14 hours)

Programming Concept

Mainframe P**rogramming**—Introduction to COBOL, Structure of COBOL Programs, COBOL words, Identification and Environment Division, Configuration Section, Input-output Section, Data Division, Level Structure— File section, Assign to clause, Working Storage section-Editing, Special-names paragraph, Usage clause—Synchronized, Justified, Redefines, Renames clauses

Module III (11hours)

Data Processing Concept

Procedure division—Data movement, Arithmetic, Sequence control , Input/Output Conditional verbs, Group moves, Compute verb, Conditions, Table handling, Occur clause—Perform verb, Set verb, Writing simple COBOL programs

Module IV (14 hours)

File Handling in Mainframes

File types — Sequential, Direct, Indexed files, Using Files in COBOL Programs, File Manipulation Verbs, **JCL Basics**—Writing to disk, DSN, DISP, Unit, Space, DCB Parameters, Job statement and Parameters –Positional and keyword Parameters, EXEC statement, EXEC Parameters, Concept of Compile and Run JCL s.

Module V (11 hours)

DataBase Concepts

Introduction to DB2—Relational DBMS Concept, Writing DB2/COBOL programs, Compilation and Binding of DB2 Programs , Concepts of DBRM, Bind JCL, Introduction

to CICS – Case study (library information system in COBOL/JCL/DB2—to be taken along with all modules as example)

Reference Books

1. M K Roy, D Ghosh Dastidar ,*Cobol Programming* ,Tata McGraw Hill,New Delhi,1999,Second Edition
2. M K Roy, D Ghosh Dastidar ,*Cobol Programming : problems & Solutions*, Tata McGraw Hill, New Delhi
3. Saba Zamir**,** Chander Ranade ,*The MVS JCL Primer (J Ranade IBM Series)*, McGraw-Hill
4. C.J. date, Colin J White, *A Guide to DB2*, Pearson Education , New Delhi,4th Edition, 2006.
5. Craig S. Mullins, *DB2 Developers Guide*, Pearson education , New Delhi, 5th Edition,2008
6. Andreas S Philippakis, Leonard J Kazmier ,*Information System through COBOL*, McGraw-Hill

CS010 706L06 CLIENT SERVER ARCHITECTURE AND APPLICATIONS

Teaching scheme **Credits:** 4
2 hours lecture and 2 hour tutorial per week

Objectives

- *To impart an introduction Client-Server system.*
- *To develop basic knowledge on securing Client-Server system.*
- *To have exposure to applications of Client-Server system.*

Pre-requisites: *Computer Networks and Operating Systems*

Module I (10 hours)

Introduction: History-uses-Client Server Computing& Heterogeneous Computing Cross Platform Computing Distributed Computing - The costs of Client Server Computing - Advantages and Disadvantages - Client Server Databases.

Module II (12 hours)

Design: Fundamentals of client server design - Managing the interaction of client and server - Communications Techniques protocols & Client server interaction protocols - Preparing applications for client server - Optimizing applications for client server - Example client server implementations - Request acceptance dispatching - Execution of requests - Client server interaction using message.

Module III (14 hours)

Multitasking: Multi programming vs multitasking - Processor - Advantages and draw backs of multiple processor - Child and parent processor - Case study Novell Netware and Windows NT - Developing server applications - Threads - Server communication model.

Module IV (12 hours)

Synchronization: Scheduling implementations - processing queues - context switching pre-emptive systems - critical sections - mutual exclusion - semaphores – semaphore implementations in NT & Netware

Module V (12 hours)

Communications: Network communication - Inter process communication - Building portable client server applications - Introduction to Client/server security concepts- Secure client/server communications – password security at system level and application level

Reference Books

1. Jeffrey D.Schank, *" Novell's Guide to Client-Server Application & Architecture"* Novell Press.
2. Robert Orfali,Dan Harkey, Jeri Edwards,"Clien/Server Survival Guide",Wiley-India Edition,Third Edition,2007
3. Dawna Travis Dewire, "*Client Server Computing*", McGraw Hill
4. W.H.Inman,*"Developing Client Server Applications"* , BPB
5. Joe Salemi, "*Guide to Client Server Databases*", BPB.
6. David Vaskevitch, "*Client Server Strategies*",Galgotia.
7. Peter T.Davis, "Securing Client/Server Computer Networks", McGraw Hill
8. Subhash Chandra Yadav, Sanjay Kumar Singh,"An Introduction to Client/Server Computing", New Age International Publishers,2009

CS010 707: Systems Programming Lab

Teaching scheme **Credits:** 2
3 hours practical per week

Objectives

- *To familiarize the design of all phases of compilers up to a stage of intermediate code generation.*
- *To enable the students to design and implement modern compilers for any environment.*

Section 1 (Compiler Design)

1. Design of a Lexical Analyzer using Finite Automation (including Symbol table)
(The program should be designed for a specific number of keywords, identifiers, numbers,
operators, punctuators etc. Finite automata should be designed for each type of token)
2. Design of lexical analyzer using LEX
3. Design of recursive descent and LL (1) parsers (including syntax tree)
(The programme should be designed for a subset of PL features (For example Arithmetic
expressions with operators +, -, *, /, ↑ etc)
4. Implementation of Operator precedence Parsing (including syntax tree)
5. Design of parser for arithmetic expressions using YACC
6. Design of a simple type checker (For eg for the primitive types of C)
7. Generation of IC for arithmetic expressions
8. Simple code optimization strategies (For example Constant folding, Loop invariant elimination, common sub expression elimination etc)
9. Design of a code generator for arithmetic expressions using Expression tree
(The program should take a set of IC as the input and produce the target code for some
machine such as Intel 8086 Microprocessor)
10. Writing a simple Compiler for a subset of Language features

Section 2:-

1.Design of 2-Pass Assembler(The Program should be designed for the generation for machine code of any simple processor such as Intel 8005)
2.Design of Absolute Loader
3.Design of Macro Pre-processor(The program should be designed for a simple preprocessor such as the # define in C)

4 Design of Device Drivers (Implementation of Simple Device Drivers such as one for the PC Speaker.)

Remark:

At Least 8 experiments from Section 1 and 2 experiments from section

CS010 708: Networking Lab

Teaching scheme **Credits:** 2
3 hours practical per week

Objectives

- *To provide experience on design, testing, and analysis of Java Programs.*
- *To acquaint the students with the Networking Protocols and Communication using ports and sockets.*

1) Basic Java Programming
2) Programs to create Applets
3) Programs to create Graphic User Interfaces
4) Programs to implement Client and Server Sockets
5) Programs for Chatting using TCP and UDP
6) Programs for Remote Procedure Call
7) Programs for Remote Method Invocation
8) Programs to interface with XML
9) Programs to implement Sliding Window Protocols
10) Programs for Multicasting
11) Programs to interface with Databases
12) Programs for Image Processing
13) Programs in Perl and PHP
14) Programs to create Dynamic Web Pages

Any experiment according to the syllabus of CS010 602 Internet Computing, CS010604 Computer Networks, CS010701 Web Technologies may be substituted subjected to permission from competent authority.

CS 010 709 Seminar

Teaching scheme **credits: 2**

2 hours practical per week

The seminar power point presentation shall be fundamentals oriented and advanced topics in the appropriate branch of engineering with references of minimum seven latest international journal papers having high impact factor.

Each presentation is to be planned for duration of 25 minutes including a question answer session of five to ten minutes.

The student's internal marks for seminar will be out of 50. The marks will be awarded based on the presentation of the seminar by the students before an evaluation committee consists of a minimum of 4 faculty members. Apportioning of the marks towards various aspects of seminar (extent of literature survey, presentation skill, communication skill, etc.) may be decided by the seminar evaluation committee.

A bona fide report on seminar shall be submitted at the end of the semester. This report shall include, in addition to the presentation materials, all relevant supplementary materials along with detailed answers to all the questions asked/clarifications sought during presentation. All references must be given toward the end of the report. The seminar report should also be submitted for the viva-voce examination at the end of eighth semester.

For Seminar, the minimum for a pass shall be 50% of the total marks assigned to the seminar.

CS 010 710 Project Work

Teaching scheme **credits: 1**

1 hour practical per week

Project work, in general, means design and development of a system with clearly specified objectives. The project is intended to be a challenge to intellectual and innovative abilities and to give students the opportunity to synthesize and apply the knowledge and analytical skills learned in the different disciplines.

The project shall be a prototype; backed by analysis and simulation etc. No project can be deemed to be complete without having an assessment of the extent to which the objectives are met. This is to be done through proper test and evaluation, in the case of developmental work, or through proper reviews in the case of experimental investigations.

- The project work has to be started in the seventh semester and to be continued on to eighth semester.
- Project work is to be done by student groups. Maximum of four students only are permitted in any one group.
- Projects are expected to be proposed by the students. They may also be proposed by faculty member (Guide) or jointly by student and faculty member.
- Students are expected to finalise project themes/titles with the assistance of an identified faculty member as project guide during the first week of the seventh semester.

The progress from concept to final implementation and testing, through problem definition and the selection of alternative solutions is monitored. Students build self confidence, demonstrate independence, and develop professionalism by successfully completing the project.

Each student shall maintain a project work book. At the beginning of the project, students are required to submit a project plan in the project book. The plan should not exceed 600 words but should cover the following matters.

- ❖ Relevance of the project proposed
- ❖ Literature survey
- ❖ Objectives
- ❖ Statement of how the objectives are to be tackled

- Time schedule
- Cost estimate

These proposals are to be screened by the evaluation committee (EC- minimum of 3 faculty members including the guide) constituted by the head of department, which will include a Chairman and the EC will evaluates the suitability and feasibility of the project proposal. The EC can accept, accept with modification, request a resubmission, or reject a project proposal.

Every activity done as part of project work is to be recorded in the project book, as and when it is done. Project guide shall go through these records periodically, and give suggestions/comments in writing in the same book.

The students have to submit an interim report, along with project work book showing details of the work carried out by him/her and a power point presentation at the end of the 7^{th} semester to EC. The EC can accept, accept with modification, request a resubmission, or extension of the project.

The student's internal marks for project will be out of 50, in which 30 marks will be based on day to day performance assessed by the guide. Balance 20 marks will be awarded based on the presentation of the project by the students before an evaluation committee consists of a minimum of 3 faculty members including the guide.

For Project, the minimum for a pass shall be 50% of the total marks assigned to the Project work.