UNIVERSITY OF NORTH BENGAL

SYLLABUS

For

M.Sc. in Computer Science

(To be implemented from Session 2015-16)
### Proposed Structure for Syllabus of Four Semester (Full Time) M.Sc. Programme in Computer Science

<table>
<thead>
<tr>
<th>Year</th>
<th>Semester</th>
<th>Paper</th>
<th>Paper Type</th>
<th>Periods / Week</th>
<th>Exam. Marks</th>
<th>Continuing Evaluation</th>
<th>Sessional</th>
<th>Nature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st Year</td>
<td>I</td>
<td>CS 11</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 12</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 13</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 11L</td>
<td>Lab-I</td>
<td>6</td>
<td>50</td>
<td>25</td>
<td>Assignment</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 12L</td>
<td>Lab-II</td>
<td>6</td>
<td>50</td>
<td>25</td>
<td>Assignment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>II</td>
<td>CS 21</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 22</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 23</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 21L</td>
<td>Lab-III</td>
<td>6</td>
<td>50</td>
<td>25</td>
<td>Assignment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 22L</td>
<td>Lab-IV</td>
<td>6</td>
<td>50</td>
<td>25</td>
<td>Assignment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>100</td>
</tr>
<tr>
<td>2nd Year</td>
<td>III</td>
<td>CS 31</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 32E</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 33E</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 31L</td>
<td>Lab-V</td>
<td>6</td>
<td>50</td>
<td>25</td>
<td>Assignments</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS34S</td>
<td>Seminar</td>
<td>6</td>
<td>25</td>
<td>25</td>
<td>Seminar</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>IV</td>
<td>CS 41</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 42E</td>
<td>Theory</td>
<td>3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 41L</td>
<td>Lab-VI</td>
<td>3</td>
<td>50</td>
<td>25</td>
<td>Assignments</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS43P</td>
<td>Project</td>
<td>6</td>
<td>75</td>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS44V</td>
<td>Grand Viva</td>
<td>6</td>
<td>25</td>
<td>25</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>100</td>
</tr>
</tbody>
</table>

| Total Exam Marks | 300x4 | = 1200 |
| Total Sessional Marks | 100x4 | = 400 |
| Grand Total | = 1600 |

Note: One ‘Period’ is of 1 Hr. duration.
<table>
<thead>
<tr>
<th>Year</th>
<th>Semester</th>
<th>Paper</th>
<th>Paper Type</th>
<th>Exam. Marks</th>
<th>Continuing Evaluation (CE)</th>
<th>Sessional</th>
<th>Nature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1st</td>
<td>I</td>
<td>CS 11</td>
<td>Data Structures and Algorithms</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 12</td>
<td>OOPs using JAVA</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 13</td>
<td>Mathematical Foundation in Computer Science</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 11L</td>
<td>Data Structure using Java Lab</td>
<td>50</td>
<td>25</td>
<td>Assignment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 12L</td>
<td>Digital Design Lab</td>
<td>25</td>
<td></td>
<td>Design</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>100</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>II</td>
<td>CS 21</td>
<td>Microprocessors and Advance Computer Architecture</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 22</td>
<td>Software Engineering</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 23E</td>
<td>Elective 1</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 21L</td>
<td>Microprocessor and Interfacing Lab</td>
<td>50</td>
<td>25</td>
<td>Assignment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 22L</td>
<td>Visual Programming Lab</td>
<td>25</td>
<td></td>
<td>Assignment</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>100</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>III</td>
<td>CS 31</td>
<td>Internet and Web Technology</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 32</td>
<td>Data Communication and Computer Networks</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 33E</td>
<td>Elective 2</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 31L</td>
<td>Web Technology Lab</td>
<td>50</td>
<td>25</td>
<td>Assignments</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS34S</td>
<td>Seminar</td>
<td>25</td>
<td></td>
<td>Seminar</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>100</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IV</td>
<td>CS 41</td>
<td>Parallel Computing</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 42E</td>
<td>Elective 3</td>
<td>75</td>
<td>25</td>
<td>Class Test</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS 41L</td>
<td>Parallel Programming Lab</td>
<td>50</td>
<td>25</td>
<td>Assignments</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS43P</td>
<td>Project</td>
<td>75</td>
<td>25</td>
<td>Assignments</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CS44V</td>
<td>Grand Viva</td>
<td>25</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>300</td>
<td>100</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total Exam Marks = 1200
Total Sessional Marks = 400
**Grand Total** = 1600
List of Electives

CS 23E Electives
   E21: Automata Theory & Formal Languages
   E22: Principles of Programming Languages
   E23: Design and Analysis of Algorithms
   E24: Real Time and Embedded Systems

CS 33E Electives
   E31: System Software and Compiler Constructions
   E32: Computer Graphics
   E33: Digital Image Processing and Steganography
   E34: Cloud and Grid Computing
   E35: Mobile and Pervasive Computing

CS 42E Electives
   E41: AI and Expert System
   E42: Data Warehousing & Data Mining
   E43: Soft Computing
   E44: Information Security and Cyber Forensics
   E45: Cryptography and Network Security
   E46: Software Project Management and SQA

** The allotment of electives depends on the availability of Teachers
Detailed Syllabus of Compulsory Papers

CS 11: Data Structures and Algorithms

Introduction to Data Structures: Introduction to basic and user defined data types and their significance, Data types as data structures, user defined data structures, various well known data structures, Abstract Data Types (ADTs).

Introduction to Algorithm: Flowchart to Algorithm, Definition and properties of Algorithm, Top-down and bottom-up approach to algorithm design, Classification of algorithms, Recursive Algorithms, Algorithm Comparison, Analysis of Algorithm, Frequency count, Time and Space Complexity analysis, Big-OH notation, Estimation of Time complexity in Best, Worst and average cases.

Array: Introduction, Need of Array, Single and Multi dimensional Arrays, Memory Representation, Address calculation using Row and Column major ordering, various operations on arrays, Vectors, Applications of Arrays, Matrix operations, sparse polynomial representation and operations, Advantages and Limitations of Arrays.

Linked Lists: Defining, Singly, Doubly and Circular Linked Lists, Implementing linked lists, Operations on Linked lists with creation, insertion and deletion, searching etc., Polynomial representation and manipulation using Linked Lists

Stacks: Introduction to Stack, Stack as ADT, representation and implementation of stack, Stack Operations, Applications of Stack, Conversion and Evaluation of expressions from one notation to another, Generalizing a Stack and its implementation using Templates

Queues: Introduction to Queue, Queue as ADT, Circular, Priority and D-Queues, representation and implementation of queues, Queue Operations, Applications of Queue, Generalizing a queue and its implementation using Templates

Trees: Introduction to Tree and its properties, Tree as ADTs, Different types of Trees, Tree Representation and implementation, AVL Trees, Threaded Trees, Heaps, Spanning trees, Minimum cost Spanning trees, different tree traversal algorithms i.e. in-order, pre-order, post order, Application of Trees.

Graphs: Introduction to Graph, properties and types of graphs, graph representation and implementations, different graph traversal methods, Kruskals, Prims Algorithms, Finding shortest paths in a digraph

Searching Algorithms: Introduction, importance of searching, sequential & binary search algorithms and their implementation, index search, hashing schemes, Complexity Analysis of different searching algorithms

Sorting Algorithms: Introduction and Importance of sorting, Implementation of selection sort, insertion sort, merge sort, quick sort, radix sort, heap sort, topological sorting, Complexity Analysis of different sorting algorithms

References:
2. Debasish Samanta, Classic Data Structures, PHI, 2nd Edition
CS 12: OOPs using JAVA

**Basics of Java:** Introduction, History and Features of Java, Internals of Java Program, Difference between JDK, JRE and JVM, Internal Details of JVM, Variable and Data Type, Unicode System, Naming Convention.

**OOPS Concepts:** Advantage of OOPs, Object and Class, Method Overloading, Constructor, static variable, method and block, this keyword, Inheritance (IS-A), Aggregation and Composition (HAS-A), Method Overriding, Covariant Return Type, super keyword, Instance Initialize block, final keyword, Runtime Polymorphism, static and Dynamic binding, Abstract class and Interface, Downcasting with instance of operator, Package and Access Modifiers, Encapsulation, Object class, Object Cloning, Java Array, Call By Value and Call By Reference, strictfp keyword, Creating API Document.

**String Handling:** Introduction, Immutable String, String Comparison, String Concatenation, Substring, Methods of String class, StringBuffer class, StringBuilder class, Creating Immutable class, toString method, StringTokenizer class.

**Exception Handling:** Introduction, try and catch block, Multiple catch block, Nested try, finally block, throw keyword, Exception Propagation, throws keyword, Exception Handling with Method Overriding, Custom Exception.

**Nested Classes:** Introduction, Member Inner class, Anonymous Inner class, Local Inner class, static nested class, Nested Interface.

**Multithreading:** Introduction, Life Cycle of a Thread, Creating Thread, Thread Scheduler, Sleeping a thread, Joining a thread, Thread Priority, Daemon Thread, Thread Pooling, Thread Group, ShutdownHook, Performing multiple task by multiple thread, Garbage Collection, Runnable class.

**Synchronization:** Synchronization : What and Why?, synchronized method, synchronized block, static synchronization, Deadlock, Inter-thread Communication, Interrupting Thread.

**Input and output:** FileOutputStream & FileInputStream, ByteArrayInputStream & ByteArrayOutputStream, SequenceInputStream, BufferedOutputStream & BufferedInputStream, FileWriter & FileReader, CharArrayWriter, Input from keyboard by InputStreamReader, Input from keyboard by Console, Input from keyboard by Scanner, PrintStream class, PrintWriter class, Compressing and Uncompressing File, Reading and Writing data simultaneously, DataInputStream and DataOutputStream, StreamTokenizer class.

**Serialization:** Serialization & Deserialization, Serialization with IS-A and Has-A, transient keyword.

**Networking:** Socket Programming, URL class, Displaying data of a web page, InetAddress class, DatagramSocket and DatagramPacket, Two way communication.
**AWT and EventHandling:** AWT Controls, Event Handling by 3 ways, Event classes and Listener Interfaces, Adapter classes, Creating Games and Applications

**Swing:** Basics of Swing, JButton class, JRadioButton class, JTextArea class, JComboBox class, JTable class, JColorChooser class, JProgressBar class, JSlider class, Digital Watch, Graphics in swing, Displaying Image, Edit Menu for Notepad, Open Dialog Box, Creating Notepad, Creating Games and applications

**LayoutManagers:** BorderLayout, GridLayout, FlowLayout, BoxLayout, CardLayout

**Applet:** Life Cycle of Applet, Graphics in Applet, Displaying image in Applet, Animation in Applet, EventHandling in Applet, JApplet class, Painting in Applet, Digital Clock in Applet, Analog Clock in Applet, Parameter in Applet, Applet Communication, Creating Games

**Reflection API:** Reflection API, newInstance() & Determining the class object, javap tool, creating javap tool, creating appletviewer, Accessing private method from outside the class

**Collection:** Collection Framework, ArrayList class, LinkedList class, ListIterator interface, HashSet class, LinkedHashSet class, TreeSet class, PriorityQueue class, ArrayDeque class, Map interface, HashMap class, LinkedHashMap class, TreeMap class, Hashtable class, Comparable and Comparator, Properties class

**JDBC:** JDBC Drivers, Steps to connect to the database, Connectivity with Oracle, Connectivity with MySQL, Connectivity with Access without DSN, DriverManager, Connection interface, Statement interface, ResultSet interface, PreparedStatement, ResultSetMetaData, DatabaseMetaData, Storing image, Retrieving image, Storing file, Retrieving file, Stored procedures and functions, Transaction Management, Batch Processing, JDBC New Features, Mini Project

**Java New Features:** Assertion, For-each loop, Varargs, Static Import, Autoboxing and Unboxing, Enum Type, Annotation etc.

**Internationalization:** Internationalization, ResourceBundle class, I18N with Date, I18N with Time, I18N with Number, I18N with Currency

**References:**

5. Dietel & Dietel, JAVA-How to program, Prentice Hall Publication

**CS 13: Mathematical Foundation in Computer Science**

**Computability:** Russell’s paradox, Non-computability and examples of non-computable problems. Permutations, combinations, generation of permutation and combinations, mathematical induction.

**Discrete Probability:** Sample Space, events, axioms, conditional probability, Bayes’ theorem.
Mathematical logic: Formal language, connectives, truth assignments and truth tables, well-formed-formulae, Tautology, contradiction and satisfiability, introduction to predicate calculus.

Combinatorics: Elementary counting principles, the rules of sum and product, Pigeonhole principle, inclusion-exclusion principle, measures of information and mutual information, Combinatorial problems.

Discrete numeric function and generating functions, linear recurrence relations and solutions, Matrix manipulation algorithm.

References:

CS 11L: Data Structure using Java Lab

Students should practice design and implementation of the following using Java:
1. Classes and Objects and Interfaces.
2. Exception handling with user defined Exceptions.
3. String Handling (String Class objects - String Manipulation functions).
4. Streaming
5. Multiple Threads Creation, Thread Synchronization using any application.
6. Reading and Writing Objects using Serialization.
7. Creation of User Interfaces using SWING.
8. Creation and Manipulation of generic objects.
9. Implementation of any Information System using JDBC.
10. Database Connectivity using Java Bean.
11. Abstract Data type Implementation of List - Stack and Queues.
12. Array and Linked List implementation of Stack, Queue, Circular Queue.
13. Set ADT- Bit Vector Implementation
14. Tree Representation and Traversals (preorder, inorder, postorder)
15. Graph Representations and Traversals
17. Spanning Tree Implementation.

CS 12L: Digital Design Lab

Design of simple digital circuits with at least one digital IC.

<table>
<thead>
<tr>
<th>Sessional</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Circuit design</td>
<td>15 marks</td>
</tr>
<tr>
<td>Documentation</td>
<td>5 marks</td>
</tr>
<tr>
<td>Viva-voce</td>
<td>5 marks</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>25 marks</strong></td>
</tr>
</tbody>
</table>
CS 21: Microprocessors and Advance Computer Architecture

Microprocessors: Microcomputer, microprocessor and evolution, Assembly language, microprocessor architecture and its operations, memory input/output, interfacing devices, 8085 based microcomputer system, addressing modes, instruction classification, format, timings and operation status; Instruction set, Data transfer instructions: Arithmetic operations, logic and branch operation, Looping, counting and indexing, 16 bit arithmetic instructions, arithmetic operation related to memory, logic operations: rotate, compare, counters and time delays; Stack, subroutine call and Return instruction, parallel input/output, 8255 programmable peripheral interface, 8253 Programmable timer, The 8085 Interrupts: 8259 programmable interrupt controller, Direct Memory Access, 8257 DMA controller, Restart as software instruction

Advance Computer Architecture: Introduction – Instruction set architecture (ISA) and Hardware system architecture (HSA), family of computer architecture, Different computer architectures, Measuring the quality of computer architecture – generality, applicability, efficiency, ease of use, expandability.

Register transfer language (RTL): differences with programming language, notations, control function, micro-operations, inter-register transfer – serial transfer, parallel transfer, bus transfer: need for having a bus system, design of bus using multiplexers/decoders and tri-state devices.

Basic computer organization and design – instruction codes, opcode, register configuration, instruction code formats – memory reference, register reference and I/O instructions, timing and control – design of hardware control unit, computer cycle control – fetch, indirect and execute cycles, cases of some instructions including branch and subroutine call, I/O configuration, interrupt cycle, design of accumulator register.

Microprogrammed control, control word, microprogram, control memory, advantages over hardware control, outline of microprogrammed control organization: next address generator (sequencer), control address register, control memory and control data register, address sequencing: associated hardware.

Instruction pipelining, advantages, breaking instruction cycle to achieve pipelining.

Storage technologies, memory array organization, memory hierarchy, cache and virtual memories, associative memory; Preliminary ideas of parallel, superscalar and vector processors.

References:

2. Stallings, William, Computer Organisation an Architecture, PHI.

CS 22: Software Engineering

Introduction to Software Engineering: Program Vs. Software, definition, origin, importance, evolution, paradigm, principles, characteristics of software engineering, software crisis, product and process
Software Processes: SW process and phases, different SDLC models, risk-driven, evolutionary and prototyping approaches, Fourth Generation Techniques, Agile methods, Software Components and CBSD.

Requirement Engineering: Role and skills of system analyst, requirement gathering techniques, problem analysis and tools, feasibility study, software requirements and types, requirement engineering process, elicitation, requirements definition, requirement review and verification, static and dynamic requirement specifications, characteristics of a good SRS, prototype outline for SRS-IEEE


Coding: Programming Languages and types, selection of programming languages, coding standards, guidelines, practices, programming styles, structured and object oriented programming, Information Hiding, Selection of suitable database system, reusability, extensibility, robustness, code documentation, static analysis, symbolic execution, code quality and efficiency

Software Testing: Introduction, software bugs, error, fault, failure, cost of bugs, objectives and purpose of testing, taxonomy of software testing, verification and validation, test case, test data suit preparation, test coverage; testing methodology- functional and structural testing, static and dynamic testing, data testing, state testing, formal reviews, code review checklist, data coverage, code coverage; testing approaches- black box testing techniques, equivalence class partition and boundary value analysis, white box testing techniques, domain and path testing, component testing; level of testing- unit testing, component testing, integration testing, system testing, alpha testing, beta testing, acceptance testing; testing types- configuration testing, compatibility testing, foreign language testing, usability testing, security testing, website testing; automated testing and test tools-benefits of automation and tools, viewers and monitors, drivers, stubs, stress and load tools, analysis tools; test documentation- goal of test planning, test phases, test strategy, resource requirements.

Software Maintenance: Software maintenance and its need, Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering, Maintainability, Documentation to facilitate maintenance


Software Configuration Management- Base Line, SCM process, Version Control, Change Management, Software Configuration Items

Computer Aided Software Engineering- CASE, Tools for Project management Support, Analysis & design, Programming, Prototyping, Maintenance, advantages, limitations, Future of CASE

References:

CS 21L: Microprocessor and Interfacing Lab

Using 8085/8086 microprocessor kits, interfacing with PC, writing and executing assembly language programs using PC, conversion to hex code, downloading to kit, executing programs in the kit from the PC, uploading programs from kit to PC, disassembling.

CS 22L: Visual Programming Lab

Visual Basic, the Language: Familiarizing with Integrated Development Environment, Designing the Loan Calculator, Designing the Math Calculator, Implementing the Loan and Math calculator using multiple forms, Designing a price calculator using check boxes, Creating games using random numbers, Designing Mark sheet, Creating Notepad, Creating a Speed measuring program using scroll bar, Creating an application that can select any shape and also change the border, Creating stop watch using timer control, Using file controls, Creating calendar, Creating thermometer, Simulate dice rolling, Creating tic-tac-toe game.

Database Applications: Connecting visual basic with oracle, Designing the Theatre Booking System, Designing Bus Reservation System.

CS 31: Internet and Web Technology

Internet and Web: Introduction and Evolution of Internet, WWW, Understanding Browsers, Internet protocols and applications i.e. FTP, Telnet, Email, Chat etc.; Semantic Web Information System, Quality Evaluation and Web Engineering and Application Development, Web Design and Development issues, challenges, Web Design Methods; Web Protocols: TCP, IP and HTTP, SMTP, POP3, FTP; Measuring and Evaluating Web Application

Static Web Design with HTML: Introduction, Evolution, Features of HTML, Filenames in HTML, Tools required, Tags and their Types, Attributes, Comments, Structures of HTML tag, Rules for writing a HTML program, starting a HTML document i.e. How to open Notepad, How to open HTML page, Editing the HTML program, Building web pages with different HTML tags, Frames, forms etc, HTML Editors and Tools- Use of different HTML editors and tools like Microsoft Front Page, Dreamweaver etc., Designing Web Application with Web ML and Web
Ratio; **Graphical and Animation Tools**- Use of Different graphical and animation tools like Abode Photoshop, Gif Animator, Macromedia flash etc.


**Client side scripting:** JavaScript, JavaScript Objects, DOM, Java Script, ASP.NET, VB Script

**Server Side Scripting:** Overview of servlets, Servlet API, Servlet life cycle, Servlet based Web Application, Servlet configuration, Running Servlet with database connectivity, Servlet support for cookies, Session tracking; Basics of ASP, JSP, PHP, ASP.NET, Comparison of ASP, PHP and JSP technologies

**JAVA GUI and Database Connectivity:** Generic classes, Generic methods, Applets, Applet life cycle methods, Applets based GUI, GUI components, Basic of Swings, Accessing database with JDBC, basics.

**Enterprise Application Development:** Three Tier Architecture, Working With MVC, JCP, J2EE, Overview of Java beans, XML Based APIs, Application Servers, Presentation Tier and EIS Tier, Java Mail, JMS, Java Transactions, JNDI, Java Authentication and Authorization Services, Java Cryptography

**Hosting Website & Security:** Hosting a Website, Web Security and issues, Firewalls, cyber laws

**Database Integration:** Designing the Databases and linking the web pages with the database using PHP,

**Advanced Topics:** SOA-SOA Basics, Principles, Evolution and implementation; Components and Frameworks- Service and Data Tier, EJB Architecture, Session Beans, Entity Beans, Message Driven Beans, J2EE Connector Architecture, Web Services, J2EE Web Services, Patterns, Presentation, Service Tier and Data Tier Patterns, J2ME, Struts, Hibernate, Spring; Web Services and Service Composition- Web Clients- Browsers, cookies, spiders, search engines and agents, Web Proxies; Web Services- Definition, Design and modeling of web services, Web Services and EAI, Web Services Technologies, web services Architecture, WS-Addressing, Routing, Security, Policy, Web Service invocation framework, Service Coordination and Composition protocols

**References:**
1. A Navarro, Mastering XML, BPB
2. Achyut S Godbole and Atul Kahate, Web Technologies, TMH
4. C. Xavier, Web Technology and Design, TMH
5. David A Chappell, Tyler Jewell, Java Web Services
6. David Busch, Cascating Style Sheets complete, McGrawHill.
8. Ivan Bayross, Sharanam Shah, PHP 5.1 for Beginners (Book/CD-Rom) Paperback – February 15, 2006
9. Ivan Bayross, Web Enabled Commercial Application Development using HTML, DHTML, JavaScript, Perl, CGI, BPB.
11. Patrick Naughton and Herbertz Schildt, Java-2 The complete Reference, TMH.
12. Raj Kamal, Internet and Web Technologies, TMH

CS 32: Data Communication and Computer Networks

Introduction to Communication Systems: Fundamentals of Digital Communication. Communication channel, Data communications and Networking, Data transmission concepts and terminology, Theoretical basis of data communication; analog and digital signals, Modulation-Digital to Analog and Analog to Analog conversion techniques, Interfaces and Modems- DTE-DCE Interface, Modems, Cable modems, bandwidth, channel, baud rate of transmission, asynchronous and synchronous transmission, Transmission modes and medium, broadband and baseband transmission, Problems with digital transmission, Transmission Impairments, Performance criteria of a communication system;

Encoding and Decoding: data encoding and modulation techniques, Line and Block coding, Scrambling techniques, pulse code modulation, Variable length codes, transmission errors - error handling mechanisms, Error detection codes, Information Theory -Measure of Information, Entropy, Discrete and Continuous channel, Shannon's encoding algorithms, Shannon-Hantlly Theorem, Data compression;

Bandwidth utilization techniques: Multiplexing, Frequency division, Time division and Wave division multiplexing, Spread spectrum concepts Baseband data transmission, Baseband pulseshaping, Inter Symbol Interface (ISI), Dubinary Baseband PAM, System Many signaling schemes, Equalisation, Synchronisation Scrambler and Unscrambler; Band-pass data transmission system ASK, PSK, FAK, DPSK &PSK, MSK, Modulation schemes coherent and Non Coherent detector. Probability of Error (PE), Performance Analysis and Comparison; Synchronous and Asynchronous transmission, Modem, Serial interface, Circuit Switching, Packet Switching, Hybrid switching, Architecture of computer network, OSI model, Data communication protocols.

Introduction to Computer Network: Uses of Computer Networks, Types of Computer Networks, OSI Reference Model, Example Networks;

Physical Layer: Data and signal fundamentals, Transmission impairments, Attenuation, Distortion, Noise, Data rate limits for noisy and noiseless channels, Performance; Digital Transmission – Problems with digital transmission, Different line coding schemes, Block coding schemes, Scrambling techniques, Analog to digital encoding. Analog Transmission; Transmission Media - Guided (wired) media – Twisted pair cable, Coaxial cable and Fibre optic cable, Unguided (wireless) media – Different propagation modes, Radio waves, Terrestrial microwaves, Satellite communication; Concept of multiplexing, Frequency division multiplexing, Time division multiplexing – Synchronous and Statistical time division multiplexing, Handling variable length data, Pulse stuffing. Concept of spreading spectrum, Frequency hopping spread spectrum and Direct sequence spread spectrum;

Data Link Layer: Link Layer Services, Error detection and Correction Techniques, Multi Access Protocols, Link Layer Addressing, Ethernet, Hubs, Switches and Switches, Point to Point Protocol, Asynchronous Transfer Mode, Multiprotocol Label Switching;
Network Layer: Introduction, Virtual Circuit and Datagram Networks, IP Addressing, Subnetting, Routing Algorithms (Link State, Distance Vector, Hierarchical), Routing in the Internet (RIP, OSPF, BGP), Broadcast and Multicast Routing Algorithms, Routers, ICMP, IPv6;
Transport Layer: Introduction to Transport Layer Services, Connectionless Transport: UDP, Principles of Reliable Data Transfer, Connection Oriented Transport: TCP, Principles of Congestion Control, TCP, Congestion Control, Sockets, Quality of services (QOS);
Application Layer: Web and HTTP, Domain Name Space (DNS), Electronic Mail (SMTP, MIME, IMAP, POP3), File Transfer Protocol, Cryptography

References:

CS 31L: Web Technology Lab

Students are to practically implement the following types of applications:
1. Web programming with HTML tags, CSS for styling, Page layout
2. Develop webpage using JavaScript for client side programming and HTML forms
3. Using The DOM and the JavaScript object models
4. Website optimization: crunching HTML, using CSS to replace HTML and light-weight graphics to speed up websites
5. Creating XML file with XML DTD and XML schema, SAX, XSL
6. Web site creation with PHP for server side programming for storing current date-time using cookies and for storing page views using sessions
7. Web application development using Servlet/ PHP/ JSP/ ASP.NET
8. Working with PHP and MySQL.
9. Constructing dynamic server-side web pages using JSF and integrate the Web application with any of the other Java2 Enterprise Edition application server methodologies such as Enterprise Java Beans, JavaMail, and SOAP.
10. Developing Java Enterprise Applications Using EJB3 Session beans, entity beans and message-driven beans.
11. Working with JNDI, JDBC and JMS.
12. Application development using J2ME.
15. Simple xml based applications using DOM, SAX and XSL.
16. Basic Java programming covering objects, inheritance, polymorphism, interfaces, packages and exception handling.
17. String handling programs and regular expression programs.
18. Creation of applet based GUIs.
19. Application involving applet based GUI, JDBC, Servlet, JSP/PHP, cookies and session tracking.
20. Designing typical website for different types of organizations

CS34S: Seminar

Seminar topics will be allotted to the students of each group by the teacher concerned.

CS 41: Parallel Computing


Introduction to Parallel Computing: Motivation, What is Parallel Computing and Why to Use? Concurrent, Parallel, Distributed computing, interacting with hardware- Composite Capabilities, How Do Languages and Environments Assist with These Tasks? Applications of Parallel Computing, RAM and PRAM model, PRAM pseudocode, Data vs. Task parallelism,

Parallel Computers Architectures: Modifications to the Von-Neumann Model, Memory Barriers, Memory Hierarchy and organization, Different types of memory access-UMA and NUMA, Shared memory, distributed memory and distributed shared memory architectures, Cache Coherence and Memory Consistency, classification of parallel computers, Flynn's Classical Taxonomy, ILP, Multi-threaded architectures and TLP, Pipeline Parallelism, I/O Operations; Overheads- Hardware System Architecture, Costs of Operations; Parallel Architecture Design Tradeoffs and Future Directions, SIMD Processors, Systolic Processors, Cluster Computing, Grid and Cloud Computing, Multicore Systems, GPU computing, Synchronization and Mutual Exclusion; Scalability and Load Balance,

Interconnection Networks: Introduction, Communication Between Parallel Processors, Classification of Interconnection Networks by Logical Topologies, Interconnection Network Switch Architecture, Routing Mechanisms for Interconnection Networks,


References:
5. Introduction to Parallel Processing by M. Sasi Kumar, Dinesh Shikhare P. Raviprakash - PHI Publication.
7. An Introduction to Parallel Programming by Peter S. Pancheo, 2011.

CS 41L: Parallel Programming Lab

Students are to learn at least one parallel programming language/extensions suitable to different parallel programming models and should practice the implementation of programs like the followings:

**Basic Applications**: sending and receiving data to/from multiple processing nodes, Calculation the value of PI, Finding Partial Sum, Average, mean squared deviation, curve fitting, numerical integration, traveling salesman problem, Gaussian elimination, Discrete event time simulation

**Search Algorithms for Discrete Optimization Problems**: Sequential Search Algorithms, Search Overhead Factor, Parallel Depth-First Search, Parallel Best-First Search, Speedup Anomalies in Parallel Search Algorithms

**Sorting**: Issues in Sorting on Parallel Computers, Sorting Networks, Bubble Sort and its Variants, Quick sort

**Dense Matrix Algorithms**: Matrix-Vector Multiplication, Matrix-Matrix Multiplication, dense matrix algorithms, sparse matrix algorithms, Solving a System of Linear Equations

**Graph Algorithms**: Definitions and Representation, Minimum Spanning Tree: Prim's Algorithm, Single-Source Shortest Paths: Dijkstra's Algorithm, All-Pairs Shortest Paths

**Fast Fourier Transform**: The Serial Algorithm, The Binary-Exchange Algorithm, The Transpose Algorithm, Cost-Effectiveness of Parallel FFT Algorithms

CS43P: Project

Students have to undergo a project on real problems at the department.

CS44V: Grand Viva

Questions will be asked from the subjects taught in the entire course.
Detailed Syllabus of Elective Papers

CS 23E Electives

E21: Automata Theory & Formal Languages

**Introduction to the theory of computation:** Symbol, alphabet, sets, relations and functions, strings and languages. *Finite state machines:* Finite automata definition & description, transition system, DFA, NFA, equivalence of DFA and NFA, Conversion of NFA to DFA, finite automata with outputs, Moore machine, Mealy machine, equivalence between Moore and Mealy machines, Chomsky Hierarchy of languages.

**Regular expressions and regular grammars:** Regular expressions, equivalence of regular expressions and FA, Regular sets and properties: Regular set, Pumping lemma for regular sets, closure properties of regular sets, Regular grammars, Right linear and Left linear grammar, equivalence between Regular linear grammar and FA inter conversion between RE and RG.

**Context free languages:** Introduction, context free grammars, derivation trees, single factum of context free grammars, leftmost and rightmost derivations, ambiguity in CFG, simplification of CFG, normal forms-Chomsky normal form CNF, Greibach normal form GNF, Enumeration of properties of CFL.

**Pushdown automata:** Definition, model, acceptance of CFL, deterministic PDA, nondeterministic PDA, the pumping lemma for CFL’s, closure properties of CFL’s, A context Free Grammar corresponding to a given context free grammar, equivalence of CFL and PDA

**Turing machines:** Definition, model, representation of TM, design of TM, Computable Languages and Functions of Turing Machines, Techniques of turing machine construction, types of TM, Universal Turing machine, computable languages and function, Halting Problem, Modifications of Turing machine, Church’s Hypothesis, Linear bounded automata and context sensitive languages, Introduction of DCFL and DPDA, Decidability of problems.

**Computability & Recursion:** Basic definition of computable and non-computable functions, primitive Recursive, Recursive and partial Recursive functions, RICE theorem and Greibach theorem, PCP and un decidability, Properties of recursive & non recursive enumerable languages, post correspondence problem

References:

3. Introduction to Formal Languages, Automata theory and Computation Kamala Krithivasan, Rama R. Pearson Education
4. An Introduction to Formal Languages and automata by Peter Liz.
5. Introductory theory of Computer Science”- V.Krishnamurthy (EWP)
7. Mathematical theory of computation By Mannaz
8. Theory of Computer Science by KLP Mishra & N.Chandra Sekaran, PHI

E22: Principles of Programming Languages

Compilation and Virtual Machines, programming environments, The halting problem and computability, Turing completeness;

**Syntax and Semantics**: formal specifications, general Problem of describing Syntax and Semantics, formal methods of describing syntax - BNF, EBNF for common programming languages features, tokenizing versus parsing, recursive descent parsers; parse trees, one-token lookahead parsing, abstract syntax trees; ambiguous grammars, attribute grammars, denotational semantics and axiomatic semantics for common programming language features, attributes and binding; scope; symbol tables; allocation and storage classes; variables; pointers

**Data types**: Introduction, primitive, character, user defined, array, associative, record, union, pointer and reference types, design and implementation uses related to these types. Names, Variable, concept of binding, type checking, strong typing, type equivalence and compatibility, type systems; type inference; type coercion, named constants, variable initialization,

**Expressions and Statements**: Arithmetic relational and Boolean expressions, Short circuit evaluation mixed mode assignment, Assignment Statements, Control Structures – Statement Level, Compound Statements, Selection, Iteration, Unconditional Statements, guarded commands.

**Control**: expressions, selection, loops, go-to, parameters, activation records for function calls

**Subprograms and Blocks**: Fundamentals of sub-programs, Scope and lifetime of variable, static and dynamic scope, Design issues of subprograms and operations, local referencing environments, parameter passing methods, overloaded sub-programs, generic sub-programs, parameters that are sub-program names, design issues for functions user defined overloaded operators, co routines.

**Abstract Data types**: Abstractions and encapsulation, introductions to data abstraction, design issues, language examples, C++ parameterized ADT, object oriented programming in small talk, C++, Java, C#, Ada 95 Concurrency: Subprogram level concurrency, semaphores, monitors, massage passing, Java threads, C# threads.

**Exception handling**: Exceptions, exception Propagation, Exception handler in Ada, C++ and Java.

**Functional Programming Languages**: Introduction, functional algorithms; tail-recursion; fundamentals of FPL, LISP, ML, Haskell, application of Functional Programming Languages and comparison of functional and imperative Languages. Scripting Language: Pragmatics, Key Concepts, Case Study : Python – Values and Types, Variables , Storage and Control, Bindings and Scope, Procedural Abstraction, Data Abstraction, Separate Compilation, Module Library, lambda calculus -conversions, Church-Rosser theorem, fixed-points,

**Object-oriented Programming**: Polymorphism, Exceptions, Lazy evaluation, Reflection, Inheritance and subtyping, Concurrency and synchronization (“threads”)”

**Logic Programming Language**: Introduction and overview of logic programming, basic elements of prolog, application of logic programming, Horn clause logic, resolution and unification

**References**:
6. Krishnamurthi, Programming Languages: Application and Interpretation, ver.

**E23: Design and Analysis of Algorithms**

**Introduction**: Framework for Algorithms Analysis, Asymptotic Notations, Basics, Euclid's algorithm, Problem, Instance, RAM model, RAM model

**Asymptotic complexity**: Some stylistic issues Analysis of Algorithms, O(log n) bit model , Principles of Algorithm Design, Finding Maximum and Minimum
Algorithm Design Techniques: Introduction to Iterative techniques, Divide and conquer, dynamic programming, greedy algorithms.

Divide and Conquer Technique: Introduction, Binary Search, Merge Sort, Quick Sort, Multiplication of Large Integers, Sorting, Median Finding, Surfing Lower Bounds, Closest Pair, Strassen's Matrix Multiplication algorithm.

Dynamic Programming: Combinatorial Search, Longest common subsequence, 0-1 Knapsack Problem, Matrix chain multiplication or Optimal search trees, A machine scheduling problem, shortest path, Travelling salesman problem.

Greedy Algorithms: Introduction, Set of Intervals, Minimum spanning tree, Union find, Set cover, Knapsack problem, Fractional Knapsack, Huffman Coding, Pattern Matching

Searching and Sorting Techniques: Review of elementary sorting techniques – selection sort, bubble sort, insertion sort; more sorting techniques – quick sort, heap sort, shell sort; external sorting; Comparison Tree, Lower bound on comparison-based sorting, Sorting in Linear Time, Counting Sort, Radix Sort.

Lower bounding techniques: Decision Trees, Adversaries.


Graphs: Analysis of Graph algorithms Depth-First Search and its applications, minimum Spanning Trees and Shortest Paths.

String Processing: KMP, Boyre-Moore, Robin Karp algorithms.

Introduction to randomized algorithms: Random numbers, randomized Qsort, randomly Built BST, Advanced Techniques to analyze algorithms: Use and study advanced data structures unionfind (Disjoint Set Structure), Fibonacci heaps.

Complexity Analysis: Complexity measures, Worst, Best and Average Case, Upper and Lower bounds, Order Notations, Introduction to Branch and Bound and backtracking techniques.


References:
E24: Real Time and Embedded Systems

**Introduction:** Embedded Systems, Challenges of Embedded Systems, fundamental components, examples of embedded systems, hardware fundamentals, gates, timing diagrams, memory, DMA, busses, interrupts, schematics, build process of embedded systems, examples.

**Embedded System Design And Implementation:** Requirements of an embedded system, Meeting real time constraints, Multi-state systems and function sequences, architecture styles and patterns, design methodologies and practices, implementation aspects and choices, 8051/89c51 and Advanced Processor Architectures, Memory Organization and Real world Interfacing, Memory access procedure, types of memory, memory management methods, Pointer related issues, polling versus interrupts, types of interrupts, interrupt latency, reentrancy, interrupt priority, programmable interrupt controllers, interrupt service routines.


**Programming Concepts and Embedded Programming in C, C++ and Java:** Software Programming In Assembly Language (ALP) and in High-level Language C, Object-Oriented Programming, Embedded Programming in C++, Embedded Programming in Java.

**Embedded Software Development Tools:** Host and target machines, cross compilers, linker and locators for embedded software, Emulators and debuggers, address resolution, locating program components, initialized data and constant strings, PROM programmers, ROM emulators, Flash memory.

**References:**
2. Raj Kamal, Microcontroller, 2nd Indian Print
7. Frank Vahid & Tony Givargus, Embedded System Design, Willey Publication

**CS 33E Electives**

E31: System Software and Compiler Constructions

**System Software:** Introduction, Definition, Role and Functions, characteristics, types

**Assembler:** Introduction, functions, features, design of one pass and two pass assemblers;
**Macroprocessors:** Introduction, functions, features and design;

**Loader and Linkers:** Basic Concepts of Linkers and Loader Functions, Boot Loaders, Linking Loaders, Linkage Editors, Dynamic Linking

**Compiler:** Introduction to Compiler. Different phases and passes of compiler
Compiler Structure, Analysis-synthesis model of compilation, various phases of a compiler, tool based approach to compiler construction.

**Lexical Analysis:** Role of Lexical Analyzer, Interface with input, parser and symbol table, Input Buffering, Specification of Tokens, lexeme and patterns; difficulties in lexical analysis; error reporting; Finite state machines and regular expressions and their applications to lexical analysis, regular definition, transition diagrams, Lex., Review of regular languages, design and implementation of a lexical analyzer,

**Syntax Analysis:** Role of the parser, Formal and context free grammars(CFGs) and their application to syntax analysis, ambiguity, associatively, precedence, Derivation and parse trees, Top Down parsing, LL(1) grammars, recursive descent parsing, transformation on the grammars, predictive parsing, bottom up parsing, Shift Reduce Parsing, LR(0) grammars, operator precedence grammars, LR parsing algorithms and LR parsers, Yacc.

**Syntax directed translation and Definitions:** Syntax directed definitions, Construction of syntax trees, Top down and bottom up approaches, dependency graph, data types, mixed mode expression; subscripted variables, evaluation order and sequencing statement, Inherited and synthesized attributes, bottom up and top down evaluation of attributes, L- and S-attributed definitions.

**Type Checking:** Type system, type expressions, structural and name equivalence of types, type conversion.

**Run Time System Environments:** Source Language issues, Storage organization, Storage Allocation strategies, activation tree, activation record, parameter passing, symbol table, dynamic storage allocation, Access to non-local names, Parameter passing mechanism

**Intermediate Code Generation:** Intermediate languages, Intermediate Graphical representations, Three address code, Implementation of three address statements (Quadruples, Triples, Indirect triples), translation of declarations, assignments, control flow, Boolean expressions and procedure calls, implementational issues.

**Code Optimization and generation:** Introduction and Issues, Basic blocks and flow graphs, Transformation of basic blocks, DAG representation of basic blocks, code generation from dags, Loops in flow graph, Principle sources of optimization, Peephole optimization, machine dependent and machine independent optimization techniques, Issues in the design of code generator, Register allocation and assignment, code generation, specifications of machine.

**Subroutines and functions:** parameters called by address, by name and by value, subroutines with side effects.

**References:**
9. C. N. Fischer and R. J. LeBlanc, Crafting a compiler with C, Pearson Education.
13. Systems Programming and Operating Systems D. M. Dhamdhere, TMH

**E32: Computer Graphics**


**2D Transformations:** Transformations and Matrices, Transformation Conventions, 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translations and Homogeneous Coordinates, Rotation, Reflection, Scaling, Combined Transformation, Transformation of Points, Transformation of The Unit Square, Solid Body Transformations, Rotation About an Arbitrary Point, Reflection through an Arbitrary Line, A Geometric Interpretation of Homogeneous Coordinates, The Window-to-Viewport Transformations


**Viewing in 3D:** Stages in 3D viewing, Canonical View Volume (CVV), specifying an Arbitrary 3D View, Examples of 3D Viewing, The Mathematics of Planar Geometric Projections, Combined transformation matrices for projections and viewing, Coordinate Systems and matrices, camera model and viewing pyramid, Scan conversion-Lines, circles and Ellipses; Filling polygons and clipping algorithms, Scan Converting Lines, Mid-point criteria, Problems of Aliasing, end-point ordering and clipping lines, Scan Converting Circles, Scan Converting Ellipses, Filling Polygons, edge data structure, Clipping Lines algorithms Cyrus-Beck, Cohen Sutherland and Liang-Barsky, Clipping Polygons, problem with multiple components.

**Solid Modeling:** Representing Solids, Regularized Boolean Set Operations, Primitive Instancing, Sweep Representations, Spatial-Partitioning Representations, Octree representation, B-Reps, Constructive Solid Geometry, Comparison of Representations

**Visible-Surface Determination:** Techniques for efficient Visible-Surface Algorithms, Categories of algorithms, Back face removal, The z-Buffer Algorithm, Scan-line method, Painters algorithms (depth sorting), Area sub-division method, BSP trees, Visible-Surface Ray Tracing, comparison of the methods

**Illumination and Shading:** Illumination and Shading Models for Polygons, Reflectance properties of surfaces, Ambient, Specular and Diffuse reflections, Atmospheric attenuation, Phongs model, Gouraud shading, some examples.

**Plane Curves and Surfaces:** Curve Representation, Nonparametric Curves, Parametric Curves, Parametric Representation of a Circle, Parametric Representation of an Ellipse, Parametric Representation of a Parabola, Parametric Representation of a Hyperbola, A Procedure for using Conic Sections, The General Conic Equation; Representation of Space Curves, Cubic Splines,
Bezier Curves, B-spline Curves, B-spline Curve Fit, B-spline Curve Subdivision, Parametric Cubic Curves, Quadric Surfaces. Bezier Surfaces

**Graphics Programming using OPENGL:** Why OpenGL, Features in OpenGL, OpenGL operations, Abstractions in OpenGL GL, GLU & GLUT, 3D viewing pipeline, viewing matrix specifications, a few examples and demos of OpenGL programs.

**Miscellaneous topics:** Why Realism? Aliasing and Anti-aliasing, texture bump mapping, Animation methods, methods of controlling animation, soft modeling of objects, image based rendering, Fundamental Difficulties.

**Image Manipulation and Storage:** What is an Image? Digital image file formats, Image compression standard JPEG, Image Processing - Digital image enhancement, contrast stretching, Histogram Equalization, smoothing and median Filtering.

**References:**

**E33: Digital Image Processing and Steganography**

**Fundamentals of Image Processing:** Image Acquisition, Image Model, Sampling, Quantization, Relationship between pixels and distance measurement, connectivity, Image Geometry, Photographic film, Light, Brightness adaption and discrimination, Perspective Projection, Spatial Domain Filtering. Color fundamentals, color models (RGB, CMY, HIS), formulation, color complements, color slicing, tone and color corrections, image file formats

**Image Filtering:** Spatial Domain Filtering- Intensity transformations, contrast stretching, histogram equalization, Correlation and convolution, Smoothing filters, sharpening filters, gradient and Laplacian; Frequency domain Filtering- Hotelling Transform, Fourier Transforms and properties, FFT, Convolution, Correlation, 2-D sampling, Discrete Cosine Transform, Frequency domain filtering, Inverse filtering, Least squares filtering. Recursive filtering

**Image Compression:** Encoder-Decoder model, Types of redundancies, Lossy and Lossless compression, Entropy of an information source, Shannon's 1st Theorem, Huffman Coding, Arithmetic Coding, Golomb Coding, LZW coding, Transform Coding, Sub-image size selection, blocking artifacts, DCT implementation using FFT, Run length coding, FAX compression (CCITT Group-3 and Group-4), Symbol-based coding, JBIG-2, Bit-plane encoding, Bit-allocation, Zonal Coding, Threshold Coding, JPEG, Lossless predictive coding, Lossy predictive coding, Motion Compensation; Wavelet based Image Compression- Expansion of functions, Multi-resolution analysis, Scaling functions, MRA refinement equation, Wavelet series expansion, Discrete Wavelet Transform (DWT), Continuous Wavelet Transform, Fast Wavelet Transform, 2-D wavelet Transform, JPEG-2000 encoding, Digital Image Watermarking; Fidelity criterion- MSE, PSNR, Compression ratio,

Morphological Image Processing: Basics, SE, Erosion, Dilation, Opening, Closing, Hit-or-Miss Transform, Boundary Detection, Hole filling, Connected components, convex hull, thinning, thickening, skeletons, pruning, Geodesic Dilation, Erosion, Reconstruction by dilation and erosion.

Image Segmentation: Definition, Detection of Discontinuities, Point, line detection, Edge detection, Edge linking, local processing, regional processing, Hough transform, Iterative and Multivariable thresholding, Otsu's method, Moving averages, Boundary detection based techniques; Characteristics of segmentation, Pixel based, Region based and histogram based segmentation methods, segmentation by sub region aggregation, split and merge technique, Watershed segmentation. Use of motion in segmentation (spatial domain technique only).

Image Enhancement: Spatial Domain Methods- Arithmetic and Analytical operations, pixel or point operations, size operations) Smoothing filters Mean, Median, Mode filters. Low pass filters, high pass filters, sharpening filters; Frequency Domain Method- Design of Low Pass, High Pass, Edge enhancement, Sharpening filters in frequency domain, Buffer Worth Filter, Homomorphic filters in frequency domain and spatial domain.

Steganography: Introduction, importance, steganography related issues and techniques, Application

References:
3. Rafael C Gonzalez, Richard E Woods; Digital Image Processing, Pearson Education
5. B Chanda & D Dutta Majumder; Digital Image Processing and Analysis, PHI.

E34: Cloud and Grid Computing
Introduction to Grid Computing: What is a grid? Infrastructure of hardware and software, Main Projects and Applications, The Open Grid Forum, International Grid Trust Federation; Grid Architecture, Overview of Resource Managers, Overview of Grid Systems; Application Management: Grid Application Description Languages, Application Partitioning, Meta-scheduling, Mapping, Monitoring; Web Services, Grid Portals,

Cloud Computing Overview: What is a cloud, Definition of cloud, Characteristics of cloud, Why use clouds, How clouds are changing, Driving factors towards cloud, Comparing grid with cloud, Public clouds (commercial), Cloud Computing and SOA, Enterprise Cloud drivers and adoption trends, Typical Cloud Enterprise workloads, Cloud service models/types, Cloud deployment models, Cloud ROI models, Cloud reference architectures, Cloud standards, Technology providers vs, Cloud providers vs. Cloud vendors, Planning Cloud transformations

Cloud service delivery: Cloud service, Cloud service model architectures, Infrastructure as a service (IaaS) architecture, Platform as a service (PaaS) architecture, Platform as a service (PaaS), Software as a service (SaaS) architecture, Examples of SaaS applications, Business Process as a Service (BPaaS) Architecture, Trade-off in cost to install versus, Common cloud management platform reference architecture: Architecture overview diagram, Common cloud management platform.

Cloud deployment scenarios: Cloud deployment models, Public clouds, Hybrid clouds, Community, Virtual private clouds, Vertical and special purpose, Migration paths for cloud, Selection criteria for cloud deployment, Case study example: IBM Smart Cloud

security models and related patterns; Virtualization and multitenancy, Internal security breaches, Data corruption or loss, User account and service hijacking, Steps to reduce cloud security breaches, Steps to reduce cloud security breaches; Identity detection, forensics and management, What is SSL? Cloud security in mainstream vendor solutions; Mainstream Cloud security offerings: security assessment, secure Cloud architecture design; Design a secure Cloud architecture to support the deployment of a secure version of the course project application.

References:

E35: Mobile and Pervasive Computing
Mobile Computing: Introduction, Differences between Mobile Communication and Mobile Computing, Contexts and Names; Functions, Applications and Services, Design Considerations, Integration of Wireless and Wired Networks Standards Bodies;


3g and 4g Cellular Networks: Migration to 3G Networks, IMT 2000 and UMTS, UMTS Architecture, User Equipment Radio Network Subsystem, UTRAN Node, B RNC functions, USIM Protocol Stack, CS and PS Domains, IMS Architecture, Handover 3.5G and 3.9G, a brief discussion 4G LAN and Cellular Networks, LTE Control Plane, NAS and RRC User Plane, PDCP, RLC and MAC WiMax IEEE 802.16d/e WiMax Internetworking with 3GPP

Context Aware Computing: Adaptability Mechanisms for Adaptation, Functionality and Data Transcoding, Location Aware Computing, Location Representation, Localization Techniques, Triangulation and Scene Analysis, De-launay Triangulation and Voronoi graphs, Types of Context, Role of Mobile Middleware, Adaptation and Agents, Service Discovery Middleware;

Mobile computing environment: Functions-architecture-design considerations, content architecture, CC/PP exchange protocol, context manager; Data management in WAECoda file system, caching schemes, Mobility QOS, Security in mobile computing.
Handoff in wireless mobile networks: reference model-handoff schemes, Location management in cellular networks, Mobility models, location and tracking management schemes, time, movement, profile and distance based update strategies, ALI technologies

Open protocols: Service discovery technologies- SDP, Jini, SLP, UdnP protocols, data synchronization, SyncML framework, Context aware mobile services, Context aware sensor networks, addressing and communications, Context aware security

Pervasive Computing: Basics, Vision and Principles; Characteristics- interaction transparency, context aware, automated experience capture; Architecture for pervasive computing, Pervasive devices, Categories of Pervasive Devices, embedded controls, smart sensors and actuators, Context communication and access services

Application Development: Three tier architecture, MVC Architecture, Memory Management, Information Access Devices, PDAs and Smart Phones, Smart Cards and Embedded Controls, J2ME Programming for CLDC, GUI in MIDP Application Development ON Android and iPhone.

References:

CS 42E Electives
E41: AI and Expert System

Scope of AI: Games, theorem proving, natural language processing, vision and speech processing, robotics, expert systems, AI techniques-search knowledge, abstraction.

Problem solving: State space search- Production systems; Search space control-Depth first search, breadth first search, heuristic search – Hill climbing, best first search, branch and bound; Minimax search- Alpha-Beta cutoffs.


Rule Based Systems- Forward reasoning, Conflict resolution, Backward reasoning- Use of no backtrack.


Handling uncertainty: Probabilistic reasoning, Use of certainty factors, Fuzzy logic.


Expert Systems: Need and justification for expert systems, Knowledge acquisition

Case studies: MYCIN, RI.

References:

E42: Data Warehousing & Data Mining

Introduction to Data Mining: Definition of data mining ,Data Mining functionalities, Classification of data mining systems , Data Mining Applications, Architectures of data mining systems, Data mining class comparison.

Data Mining Algorithms: Concept Description: Definition, Data Generalization and Summarization –Based Characterization, Mining Descriptive Statistical Measures in Large Databases; Mining Association Rules: Association Rule Mining, Market Basket Analysis, Association Rule Classification, The Apriori Algorithm, Mining Multilevel Association Rules, Constraint-Based Association Mining, Sequential mining

Classification and Prediction: What is Classification and Prediction? Data Classification Process, Issues Regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Classification Based on Association Rule Mining, Other Classification Methods Cluster Analysis: What is Cluster Analysis? Types of Data in Cluster Analysis, Categorization of Clustering Methods, Partitioning methods

Introduction to Data Warehousing: Introduction to Decision Support System: DSS Definition, History of DSS, Ingredients of DSS, Data and Model Management, DSS Knowledge base, User Interfaces, The DSS Users, Categories and Classes of DSSs Need for data warehousing, Operational & informational data, Data Warehouse Definition and characteristics, Operational Data Stores

Data warehouse Components: Architectural components, Data Preprocessing: Why Preprocess Data? Data Cleaning Techniques, Data Integration and Transformation, Data Reduction Techniques, Discretization and Concept Hierarchy, Generation for numeric and categorical data, Significant role of metadata, Building a Data warehouse, Benefits of Data Warehousing.

References:
1. Jiawei Han, Micheline Kamber; Data Mining: Concepts and Techniques, Morgan Kaufmann, ISBN 1558609016, 2006.
2. Paul Punnian; Data Warehousing Fundamentals, John Wiley Pub
3. Alex Berson, S.J. Smith; Data Warehousing, Data Mining and OLAP, TMH
5. Ralph Kimball; The Data Warehouse Lifecycle toolkit, John Wiley

E43: Soft Computing
Soft Computing: Introduction of soft computing, soft computing vs. hard computing, various types of soft computing techniques, applications of soft computing; Artificial Intelligence : Introduction, Various types of production systems, characteristics of production systems, breadth first search, depth first search techniques, other Search Techniques like hill Climbing, Best first Search, A* algorithm, AO* Algorithms and various types of control strategies; Knowledge representation issues, Prepositional and predicate logic, monotonic and non monotonic reasoning, forward Reasoning, backward reasoning, Weak & Strong Slot & filler structures, NLP.
Artificial Neural Networks: Neuron, Nerve structure and synapse, Artificial Neuron and its model, activation functions, Neural network architecture: single layer and multilayer feed forward networks, recurrent networks.Various learning techniques; perception and convergence rule, Auto-associative and hetero-associative memory; Propagation Networks- introduction, Counter propagation network, architecture, functioning & its characteristics, Back Propagation Networks - Architecture: perceptron model, solution, single layer artificial neural network, multilayer perception model; back propagation learning methods, effect of learning rule co-efficient ;back propagation algorithm, factors affecting backpropagation training, applications; Hopfield/Recurrent network, configuration, stability constraints, associative memory, and characteristics, limitations and applications; Hopfield v/s Boltzman machine; Adaptive Resonance Theory: Architecture, classifications, Implementation and training; Associative Memory.
Genetic Algorithm(GA): Basic concepts, working principle, procedures of GA, flow chart of GA, Genetic representations, (encoding) Initialization and selection, fitness function, reproduction, Genetic modeling: Genetic operators, Inheritance operator, cross over, inversion & deletion, mutation operator, Bitwise operator; Generational Cycle, Convergence of GA, Applications & advances in GA, Differences & similarities between GA & other traditional method
References:
2. Siman Haykin, "Neural Networks” Prentice Hall of India.
5. J. Yen and R. Langari., Fuzzy Logic, Intelligence, Control and Information, Pearson Education.

E44: Information Security and Cyber Forensics


Network Defense tools: Firewalls and Packet Filters: Firewall Basics, Packet Filter Vs Firewall, How a Firewall Protects a Network, Packet Characteristic to Filter, Stateless Vs Stateful Firewalls, Network Address Translation (NAT) and Port Forwarding, the basic of Virtual Private Networks, Linux Firewall, Windows Firewall, Snort: Introduction Detection System


Cyber Forensics: Firewalls and Packet Filters, password Cracking, Keyloggers and Spyware, Virus and Warms, Trojan and backdoors, Steganography, DOS and DDOS attack, SQL injection, Buffer Overflow, Attack on wireless Networks

27001, Basics of Indian Evidence ACT IPC and CrPC, Electronic Communication Privacy ACT, Legal Policies.

References:

E45: Cryptography & Network Security
Introduction to Classical Cryptosystems: Introduction, Need and importance of Cryptography, Classical Cryptosystems, Introduction to symmetric and asymmetric cryptography, Cryptanalysis of Classical Cryptosystems, Shannons Theory
Symmetric Key Ciphers and Cryptanalysis: Introduction, Symmetric Key Ciphers, Modern Block Ciphers- DES, AES; Linear Cryptanalysis, Differential Cryptanalysis, Other Cryptanalytic Techniques, Overview on S-Box Design Principles, Modes of operation of Block Ciphers
Hash Functions and MACs: Hash functions, The Merkle Damgard Construction, Message Authentication Codes (MACs)
Asymmetric Key Ciphers and Cryptanalysis: Construction and Cryptanalysis, RSA Cryptosystem, Different Attacks & Remedies on RSA, Semantic Security of RSA, The Discrete Logarithm Problem (DLP), Diffie Hellman Key Exchange algorithm, The ElGamal Encryption Algorithm, Massey-Omura; Construction and Cryptanalysis, Cryptanalysis of DLP
Digital Signatures: Introduction, Signature schemes, Authentication Protocols, Digital Signature Standards (DSS), Proxy Signatures
Network Security: Secret Sharing Schemes, Network Protocols, Kerberos, Pretty Good Privacy (PGP), Secure Socket Layer (SSL), Intruders and Viruses, Firewalls
Primality Testing: Primality Testing, Quadratic Residues, Randomized Primality Test & Deterministic Polynomial Time Algorithm

References:
10. Wenbo Mao, "Modern Cryptography, Theory & Practice", Pearson Education.

E46: Software Project Management and SQA


SW Project Planning: Defining the problems, developing a solutions strategy, planning the development process, activity involved in SW project planning, Steps in SW project planning, planning an organizational structures.

SPM Activities and Activity Planning: Objectives, Project Schedule, Sequencing and Scheduling Activities, Umbrella Activities- Metrics, Configuration Management, Software Quality Assurance; In Stream Activities- Project Initiation, Project Planning, Execution and Tracking, Project Wind up, Concept of Process/Project Database, Network Planning Models i.e PERT and CPM, Shortening Project Duration


Metrics: Need for Software Metrics, Classification of Software Metrics: Product Metrics (Size
Metrics, Complexity Metrics, Halstead’s Product Metrics, Quality Metrics), and Process metrics (Empirical Models, Statistical Models, Theory-based Models, Composite Models, and Reliability Models)

**Managing People and Organizing Teams:** Introduction, Becoming a Team, Organizational and team Structures, Team Management, Client Relationship Management, Case Studies.

**SW Quality Fundamentals:** SW quality concept- what and why? Benefits and importance, SW Quality models i.e. McCall, Boehm, FURPS, Dromey, ISO 9001, 9126 etc., Cost of Poor quality, SQA: Introduction, roles and benefits, SQA and quality control, SQA planning and activities, SQA process framework i.e. ISO, CMM, Six-Sigma, TMMi, People CMM etc. and their relevance to Project Management

**Fundamentals of Software Quality Assurance:** Ethical Basis for Software Quality, Total Quality Management Principles, Software Processes and Methodologies.


**Quality Metric System:** Concepts, Measurement Theory, Software Quality Metrics, importance and categories of metrics, Metrics Program (GQM), Designing Software Measurement Programs, Complexity Metrics and Models, Organizational Learning, Improving Quality with Methodologies, Structured/Information Engineering, commonly used metrics i.e. Process, Product and Resource metrics.

**Test Management:** Recap of SW Testing fundamentals, Test Management and activities involved, Evaluation of Test Effectiveness, release management, Test management tools

**Tools for Quality Improvement:** Basic quality control tools, check sheet, C&E diagram, Pareto diagram, histogram, Scatter Plot, Run Chart, Control Chart, orthogonal defect classification,

**References:**