Syllabus
BE (Computer Engineering)
2008 Course
Preamble

Dear Students and Teachers, we, the Board of Studies Computer Engineering, are very happy to present BE Computer Engineering Syllabus (2008 course). We are sure you will find this syllabus interesting and challenging. The objective of the syllabus is to inculcate reading, innovative thinking, technical-writing and presentation habits in the students. Development of skills and imparting knowledge is also important part of syllabus. To achieve these objectives teachers are encouraged to take home tutorials, class tutorials, students presentations and invited talks in addition to the lectures conducted by teachers. Review and interactive discussions on home tutorials, classroom tutorials and students presentation along with review of recent advances in the subject is a must. There is mountainous growth in the areas of computer applications, subjects, technology and algorithms. The computer equipments are becoming very faster, granular with concurrent architectures and operating systems environments. Along with the Industry, the open-source movement has also contributed to the great extent. The education must withstand with this growth and demand of innovation and be the front-runner to lead the path of progress and innovation. We give you best wishes to take this syllabus and travel the path of success.

Board of Studies
Computer Engineering
## BE (COMPUTER ENGINEERING) - 2008 COURSE STRUCTURE

### Term-I

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Teaching Scheme</th>
<th>Examination Scheme</th>
<th>Total Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>410441</td>
<td>Design &amp; Analysis of Algorithms</td>
<td>04</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>410442</td>
<td>Principles of Compiler Design</td>
<td>04</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>410443</td>
<td>Object Oriented Modeling &amp; Design</td>
<td>04, 02</td>
<td>100 25</td>
<td>50 175</td>
</tr>
<tr>
<td>410444</td>
<td>Elective-I</td>
<td>03, 02</td>
<td>100 25</td>
<td>50 175</td>
</tr>
<tr>
<td>410445</td>
<td>Elective-II</td>
<td>03</td>
<td>100</td>
<td>--- 100</td>
</tr>
<tr>
<td>410446</td>
<td>Computer Laboratory-I</td>
<td>--- 04</td>
<td>---</td>
<td>--- 50</td>
</tr>
<tr>
<td>410447</td>
<td>Project Work</td>
<td>--- 02</td>
<td>---</td>
<td>--- 50</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>18, 10</td>
<td>500 100</td>
<td>50 100 750</td>
</tr>
</tbody>
</table>

#### Total of Part I (A) 28 Hrs 750

### Term II

<table>
<thead>
<tr>
<th>Subject Code</th>
<th>Subject</th>
<th>Teaching Scheme</th>
<th>Examination Scheme</th>
<th>Total Marks</th>
</tr>
</thead>
<tbody>
<tr>
<td>410448</td>
<td>Distributed Operating Systems</td>
<td>04</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>410449</td>
<td>Advanced Computer Architecture</td>
<td>04</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>410450</td>
<td>Elective-III</td>
<td>04, 02</td>
<td>100 50</td>
<td>50 200</td>
</tr>
<tr>
<td>410451</td>
<td>Elective-IV</td>
<td>04</td>
<td>100</td>
<td>--- 100</td>
</tr>
<tr>
<td>410452</td>
<td>Computer Laboratory II</td>
<td>--- 04</td>
<td>---</td>
<td>50 100</td>
</tr>
<tr>
<td>410447</td>
<td>Project Work</td>
<td>--- 06</td>
<td>100</td>
<td>50 150</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td></td>
<td>16, 12</td>
<td>400 200</td>
<td>50 100 750</td>
</tr>
</tbody>
</table>

#### Total of Part II (B) 28 Hrs 750

| Grand Total (A+B) | 1500 |

Th: Theory    TW: Term Work    Pr: Practical    Or: Oral
### Elective I
1) Image Processing  
2) Design & Analysis of Computer Networks  
3) Artificial Intelligence  
4) Software Architecture

### Elective II
1) Multimedia Systems  
2) Mobile Computing  
3) Embedded Systems  
4) Software Testing & Quality Assurance

### Elective III
1) Pattern Recognition  
2) High Performance networks  
3) Neural Networks  
4) Advanced Databases

### Elective IV
1) VLSI & Digital System Design  
2) Operations Research  
3) Cloud Computing  
4) Information Security  
-or Open Elective

Open Elective: Any other Electives that are being taught in Term II under the Faculty of Engineering or individual college and Industry, together, can define new elective using framework of Elective IV defined in syllabus structure and GET IT APPROVED FROM BOARD OF STUDIES COMPUTER ENGINEERING AND OTHER NECESSARY STATUTORY SYSTEMS IN THE UNIVERSITY OF PUNE BEFORE 30th DECEMBER.

The BE Project Term work assessment for Term I will be done by selecting panel of examiners amongst senior teachers of Computer Engineering. Existing prevailing practices are followed for all remaining examinations and assessment work.
### 410441: Design and Analysis of Algorithms

**Teaching Scheme:**

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit II</td>
<td>Dynamic Programming, Study of different ways to implement Knapsack Problem. Implementation of OBST, Traveling Salesperson Problem. General Strategy, Multistage graphs, OBST, 0/1 Knapsack, Traveling Salesperson Problem, Flow Shop Scheduling.</td>
<td>6</td>
</tr>
<tr>
<td>Unit III</td>
<td><strong>Backtracking:</strong> General Strategy, 8 Queen's problem, Graph Coloring, Hamiltonian Cycles, 0/1 Knapsack. <strong>Branch and Bound:</strong> General Strategy, 0/1 Knapsack, Traveling Salesperson Problem. Design of N Queen's problem, Hamiltonian Cycles</td>
<td>6</td>
</tr>
<tr>
<td>Unit IV</td>
<td><strong>Basic Concepts:</strong> Non deterministic algorithms, The classes NP Hard and NP Complete, Cook's Theorem, NP Hard graph problems: Clique Decision problem, Node cover Decision problem, Chromatic number decision problem, Directed Hamiltonian Cycle Problem, TSP Decision problem, AND/OR Graph decision problem, <strong>NP-Hard Scheduling problems:</strong> Scheduling Identical processors, Flow shop scheduling, Job shop scheduling. NP-Hard Scheduling. Study of NP-Hard and NP-COMPLETE problems. Solving NP-COMPLETE problem.</td>
<td>6</td>
</tr>
<tr>
<td>Unit V</td>
<td>Parallel Algorithms, Study of different graph problems. Implementation of different sorting problems on multiprocessor system. Computational Model, Basic Techniques and Algorithms, Complete Binary Tree, Pointer Doubling, Prefix Computation, Selection, Merging, Sorting, Graph Problems.</td>
<td>6</td>
</tr>
<tr>
<td>Unit VI</td>
<td>Case Studies of Algorithmic Designs &amp; Applications, Implementation of Huffman Problem. Deadlock detection and avoidance implementation. Image edge detection algorithms, Resource allocation algorithm with deadlock avoidance, Heuristic search algorithm, Coding theory algorithm</td>
<td>6</td>
</tr>
</tbody>
</table>
(Huffman coding), Sorting & Convex hulls algorithm. Review and interactive discussions on home tutorials, classroom tutorials and students presentation. Review of recent advances in the subject.

Text Books:
2. Bressard, "Fundamental of Algorithm.", PHI

Reference Books:
1. Thomas H Cormen and Charles E.L Leiserson, "Introduction to Algorithm" PHI
### 41442 Principles of Compiler Design

**Teaching Scheme**
Lectures: 4 Hrs/Week

**Examination Scheme**
Theory: 100 marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
<th>Teaching Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unit I</strong></td>
<td>Introduction to lexical Analysis and Syntax analysis. Review of lexical analysis: alphabet, token, lexical error, Block schematic of lexical analyser, Introduction to phases of Compiler, Lexical Analysis. Automatic construction of lexical analyser-(LEX), LEX specification and features. Introduction: Role of parsers, Top down-RD parser, Predictive parsers, LL (k) parsers, Bottom up Parsers - Operator precedence parsers, shift-Reduce: SLR, LR (k), LALR etc. using ambiguous grammars. Error detection and recovery, Automatic construction of parsers (YACC), YACC specifications.</td>
<td>13</td>
</tr>
<tr>
<td><strong>Unit II</strong></td>
<td>Introduction to Semantic Analysis and syntax directed translation, Need of semantic analysis, type checking and type conversion Syntax directed definitions, construction of syntax trees, bottom-up evaluation of S-attributed Definitions, L-attributed definitions, Top-down translation,</td>
<td>06</td>
</tr>
<tr>
<td><strong>Unit III</strong></td>
<td>Declarations, assignment statements, iterative statements, case statements, arrays, structures, conditional statements, Boolean expressions, back patching, procedure calls, Intermediate Code Generation, Intermediate languages, Intermediate languages programming structures, Intermediate code generation using YACC. Review and interactive discussions on home tutorials, classroom tutorials and students presentation. Review of recent advances in the subject.</td>
<td>06</td>
</tr>
<tr>
<td><strong>Unit IV</strong></td>
<td>Run-time Storage organization, Storage organization and allocation strategies, Source language issues, Storage organization and allocation strategies for block structured and non block structured languages, Activation record, variable-length data, procedure parameters, nested procedures, access to non-local names, procedure Call and return, static and dynamic scope. Symbol Table organization and management Run-time Storage organization</td>
<td>05</td>
</tr>
<tr>
<td><strong>Unit V</strong></td>
<td>Introduction: Issues in code generation, Basic blocks and flow graphs, next-use information, Target machine description, Register allocation and assignment, Dag representation of basic blocks, Peephole optimization, Generating code from a DAG, Dynamic programming, Code generator concept.</td>
<td>06</td>
</tr>
<tr>
<td><strong>Unit VI</strong></td>
<td>Introduction and Classification of code optimization, Principle sources Of Optimization, optimization of basic blocks, Introduction, Loops in flow graphs, Optimizing transformations:</td>
<td>06</td>
</tr>
</tbody>
</table>

Text books:

References Books:
# 410443 Object Oriented Modeling and Design

## Teaching Scheme
- Lectures: 4 Hours /Week
- Practical: 2 Hours/Week

## Examination Scheme:
- Theory: 100 Marks
- Term Work: 25 Marks
- Oral: 50 Marks

<table>
<thead>
<tr>
<th>Units</th>
<th>Contents</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIT I</td>
<td>What is modeling? Object Oriented Thinking, History of UML</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Building Blocks of UML, OCL: what &amp; why, expression syntax</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Introduction to OMG standards MDA, XMI, UML 2.0. RUP emphasizing Inception,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Elaboration, Construction, Transition Phases. 4+1 Architecture,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>UML Meta model, Extensibility mechanisms like stereotypes, tagged values,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>constraints and profiles. MDA, XMI, UML 2.0</td>
<td></td>
</tr>
<tr>
<td>UNIT II</td>
<td>Requirements modeling: Gathering &amp; classifying requirements,</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Requirements metamodel, Requirements workflow</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Detail Use case specifications, Use cases &amp; RUP, Advanced use-case</td>
<td></td>
</tr>
<tr>
<td></td>
<td>modeling, Activity diagrams. Review and interactive discussions on home</td>
<td></td>
</tr>
<tr>
<td></td>
<td>tutorials, classroom tutorials and students presentation. Review of</td>
<td></td>
</tr>
<tr>
<td></td>
<td>recent advances in the subject. Use-case modeling (actors, use cases,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>relationships),</td>
<td></td>
</tr>
<tr>
<td>UNIT III</td>
<td>Package diagrams, Basic Class diagrams, Need, purpose &amp; application of</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>UML diagrams, CRC method, Advanced Class diagrams, Object diagrams,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Composite structure diagrams, Illustrations of Package diagrams, Basic</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Class diagrams, Object diagrams, Composite structure diagrams</td>
<td></td>
</tr>
<tr>
<td>UNIT IV</td>
<td>Need, purpose &amp; application of sequence diagrams, Communication diagrams,</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Interaction Overview diagrams, Timing Diagram, State Machine diagrams,</td>
<td></td>
</tr>
<tr>
<td>UNIT V</td>
<td>Component Diagram, Interfaces and ports, Deployment diagrams, Need,</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>purpose &amp; application of above diagrams two, three tier architecture,</td>
<td></td>
</tr>
<tr>
<td>UNIT VI</td>
<td>Concept of Forward Engineering and Reverse Engineering of UML Diagrams,</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Design patterns, Forward Engineering and Reverse Engineering of all</td>
<td></td>
</tr>
<tr>
<td></td>
<td>diagrams of UML 2.0. UML diagrams for following design patterns: Singleton</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Abstract factory, Façade, proxy, Iterator, Observer. Design pattern</td>
<td></td>
</tr>
<tr>
<td></td>
<td>examples</td>
<td></td>
</tr>
</tbody>
</table>
Text Books:

3. Mike O’Docherty, “Object Oriented Analysis & Design”, Wieley India

Reference Books:

2. Tom Pender, “UML 2 Bible”, wiely publication,
4. Dan Pilone, Neil Pitman "UML 2.0 in a Nutshell”, (In a Nutshell (O'Reilly)) Paperback)
Object Oriented Modeling and Design Lab

Objectives of the Laboratory:

• To learn how to understand the requirements of a system, its scope.
• To learn good design, good modeling practices, document them and be able to discuss the pros and cons of your designs and models.
• To learn issues in modeling large, complex systems with example hypothetical systems
• To learn concepts, best practices in software, firmware development today and explore UML 2.0 Basic And Advanced concepts and notation for the same:
• To use UML 2.0 diagrams for modeling different aspects of a system throughout the SDLC lifecycle.
• To model an entire system using UML 2.0
• To learn effective use of any CASE TOOL for UML 2.0.

Outcomes:
1) Students will be able to document SRS.
2) Students will be able to develop structural models.
3) Students will be able to develop behavioral models.
4) Students will be able to create Architectural models.
5) Students will be able to select the set of appropriate diagrams to develop a system model.
6) Students will be skilled in a CASE Tool

To meet above objectives teachers will help students choose a hypothetical system preferably either a commercial, web based or embedded system for modeling. The students will try and identify scope of such a system as realistically as possible. Students will learn to draw, discuss different UML 2.0 diagrams, their concepts, notation, advanced notation, forward and reverse engineering aspects. As far as possible draw as many diagrams for one single system, unless they are not applicable for the chosen system in which case other systems may be chosen for specific diagrams. Any 8 diagrams can be drawn using tool, the other diagrams can be drawn on paper. Optionally one may draw Interaction overview diagrams, timing diagrams, and composite structure diagrams, object diagrams for your system as study assignments, paper based assignment or in cases relevant even in CASE TOOL

The faculties are advised to use Project Based Learning [PBL] to conduct this lab.
## 410444: Elective I: Image Processing

### Teaching Scheme:
- **Teaching:** 3 Hrs/Week
- **Practical:** 2 Hrs/week

### Examination Scheme
- **Theory:** 100 Marks
- **Term Work:** 25 Marks
- **Oral:** 50 Marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit I</td>
<td>What is digital image processing? Origin, usage and application of image processing. Fundamental steps and component of image processing system. Introduction to Human Visual System. Digital representation of images (monochrome &amp; color). Elements of matrix theory, Digital Imaging Hardware &amp; Software.</td>
<td>6</td>
</tr>
<tr>
<td>Unit II</td>
<td>Basic image preprocessing (contrast enhancement, simple noise reduction, color balancing), Spatial transformation Gray Level liner and non-linear transformation, Histogram Processing, Hadamard and Walsh transformation. Image enhancement in spatial and frequency domain: basic fundamental, smoothing and sharpening domain filters. Sampling &amp; Quantization.</td>
<td>6</td>
</tr>
<tr>
<td>Unit III</td>
<td>Image Processing filters, Image Segmentation &amp; Analysis, Implementation Feature extraction: Edges, Lines &amp; corners detection, Texture &amp; shape measures. Segmentation &amp; thresholding, region extraction, edge (Canny) &amp; region based approach, use of motion in segmentation. Feature extraction- Edges, Lines &amp; corners detection, Texture &amp; shape measures.</td>
<td>6</td>
</tr>
<tr>
<td>Unit IV</td>
<td>Image Restoration &amp; Reconstruction. Introduction, Model of Image degradation, Noise Models, Classification of image restoration techniques, Blind-deconvolution techniques, Lucy Richardson Filtering, Wiener Filtering.</td>
<td>6</td>
</tr>
<tr>
<td>Unit V</td>
<td>Image Compression &amp; Object Recognition. Introduction to Image Compression and its need, Coding Redundancy, Classification of Compression Techniques (Lossy and Losless - JPEG, RLE, Huffman, Shannon fano), Scalar &amp; Vector Quantization. Introduction to Object Recognition, Object Representation (Signatures, Boundary Skeleton), Simple Boundary Descriptors, Regional descriptors(Texture).</td>
<td>6</td>
</tr>
</tbody>
</table>
Applications-
Medical Image Processing, Face detection, Iris Recognition

Text books:
2. Digital Image Processing, R.C. Gonzalez, R.R. Woods(TMH)

References Books:
4. Introduction to Digital Image Processing with MATLAB, Alasdair McAndrew, Cengage Learning

Assignment for Image Processing (Using Object oriented techniques/OpenCV)

1) Write a program to create a simple image file, save the same in .tiff format, and display it.
2) Write a program for image transformation, (recommended using Hadamard or Walsh transformation method or latest advanced transformation method).
3) Write a program using derivative filtering technique for edge detection and further thinning the edge.
4) Write a program for image enhancement, use suitable algorithm for
   a. Histogram equalization & specification,
   b. Local enhancement,
   c. Smoothing and
   d. Sharpening
5) Write a program to illustrate
   a. Constrained image restoration, and
   b. Unconstrained image restoration
6) Implement any of the following application/ or any Image processing application
   using MATLAB/OpenCV
   a. Medical Image Processing
   b. Face detection
   c. Iris Recognition

(Note: Students will submit the term work in the form of journal. The journal will contain minimum six assignments. Oral examination will be based on term work submitted.)
### Elective I: Design and Analysis of Computer Networks

#### Teaching Scheme:
- **Teaching:** 3 Hrs/Week
- **Practical:** 2 Hrs/week

#### Examination Scheme:
- **Theory:** 100 Marks
  - **Term Work:** 25 Marks
  - **Oral:** 50 Marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
</table>
| **Unit I** | Introduction, Review of background concepts of computer networks and data communication  
Review of discrete random variables, point processes  
Delay Models- Little’s theorem, M/M/1 Queue System, M/G/1 system, generalized state dependent arrival and service system, Markov chains, Network of queues, open queuing system, closed queuing system, Independence assumption. | 6 |
| **Unit II** | Introduction, Resource constraints and their metrics, common design techniques, performance analysis and tuning, summarization of various techniques (e.g. multiplexing, pipelining, virtualization, randomization, soft state etc), Network design, Analysis and design of switching fabrics, Access networks and Backbone design. Modeling and simulation techniques, System Design, Study and analysis of System Design, Implementation of System Design, Simulation techniques of System Design | 6 |
| **Unit III** | Introduction, Requirements, Fundamental choices, scheduling best effort connections and guaranteed service connections, comparison of various approaches, Open loop and closed loop flow control, comparison of closed loop schemes (stop and wait, static window, TCP Tahoe, TCP Vegas, ATM Forum, Credit based etc.), Study of flow control and scheduling, Implementation aspects of open loop and closed loop flow control. | 6 |
| **Unit IV** | Introduction and Framework for Traffic management, Traffic Models, Traffic classes, Renegotiation, Signaling, Admission control, peak load pricing, capacity planning, QOS (ATM Forum and IETF approaches), ATM Forum and IETF approaches | 6 |
| **Unit V** | Router/Switch architectures and algorithms  
Introduction and overview of router architecture and hardware Prefix match IP lookup, IP trace back, Packet scheduling algorithms and RED/AQM, Router algorithms for network measurements, Implementation aspects of Router/Switch architectures and algorithms | 6 |
| **Unit VI** | Functions and responsibilities, Network planning and implementation, Sub-netting, Bandwidth management, security issues, Tools for BW and security management, Security Issues and tools for BW | 6 |
Text Books:
1. An Engineering approach to computer Networking by S.Keshav, Pearson education
2. Network Algorithms by George Varghese, Morgan Kaufmann (For Unit 5)

Reference Books:
1. Data networks by D.Berteskas and R Gallagar, Printice Hall
2. Design and Analysis of Communication Networks By V Ahuja, McGraw Hill

Design 6 suitable assignments based on following topics for the purpose of Design and Analysis of Computer Networks Laboratory
1. Study of Network Simulators, Open Virtual Network Tools
2. Use of Little’s theorem or Marcov Chains for design and analysis of network queues.
3. Using simulation tools and test data perform design and analysis of access network and backbone design
4. Analysis of Flow control and scheduling algorithms using test data
5. Network Analysis based decision system for Admission control, peak load pricing, capacity planning
6. Design and analysis of router Algorithm for network measurements using network simulation tools
7. Network planning and implementation: Bandwidth issue, Security Issues
### 410444: Elective I: Artificial Intelligence

<table>
<thead>
<tr>
<th>Teaching Scheme</th>
<th>Examination Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teaching: 3Hrs/Week</td>
<td>Theory: 100 Marks</td>
</tr>
<tr>
<td>Practical: 2 Hrs/week</td>
<td>Term Work: 25 Marks</td>
</tr>
<tr>
<td></td>
<td>Oral: 50 Marks</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
</table>
| Unit IV | Logical agents and Classical Planning, Study and comparison of knowledge representation structures. Implementation aspects of Backtracking algorithm and forward and backward chaining  
| --- | --- |
| Unit V | Quantifying Uncertainty:  
Acting under Uncertainty, Basic Probability Notation, Inference Using Full Joint Distributions, Bayes' Rule and Its Use, Representing Knowledge in an Uncertain Domain, Other Approaches to Uncertain Reasoning, Rule-based methods for uncertain reasoning, Representing vagueness: Fuzzy sets and fuzzy logic, Study of fuzzy logic and Decision trees, Implementation aspects of Decision trees  
Learning from Examples:  
Forms of Learning, Supervised Learning, Learning Decision Trees, The decision tree representation, Expressiveness of decision trees, Inducing decision trees from examples.  
Unit VI | Natural language processing and Expert systems,  
Natural Language Processing:  
Language Models, Steps in NLP, Syntactic Analysis (Parsing), Semantic interpretation, Discourse and pragmatic Processing, Text Classification. Discourse and pragmatic Processing, Implementation aspects of Syntactic Analysis (Parsing)  
Expert Systems:  
What is Expert system, Components of Expert System, Case studies on Expert System |
| --- | --- |
2. Prolog Programming for A.I. by Bratko, TMH
4. Artificial Intelligence and Intelligent Systems by Padhy, Oxford University Press,

**Artificial Intelligence Laboratory Assignments:**

2. Implementation of MinMax Search Procedure with alpha beta pruning for finding the solutions of games.
5. Implementation of Solving Uncertainty Problems like Weather forecasting
6. Implementation of Parsing methods to categorize the text
   Or
7. Implementation of Mini Expert System
## 420444 Elective I: Software Architecture

### Teaching Scheme
- Lectures: 3 Hrs/Week
- Practical: 2 Hrs/week

### Examination Scheme
- Theory: 100 marks
- Oral: 50 Marks
- Term Work: 25 Marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
<th>Nos. Hrs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit I</td>
<td>Introduction to Software Architecture, Architecture Business Cycle, What is software architecture, software architecture requirements, Architecture structures and views, Documenting software architectures, Opportunities and Advances in software architectures.</td>
<td>6</td>
</tr>
<tr>
<td>Unit II</td>
<td>Introduction to Quality Attributes, Need of quality attributes, Understanding quality attributes, architecture and quality attributes, achieving quality attributes. Case study of quality attributes in software architecture templates. Deriving Quality Attributes for software architectures</td>
<td>6</td>
</tr>
<tr>
<td>Unit III</td>
<td>Design Patterns: history, principles and expectations. Study of a number of representative patterns like Singleton, Factory, Adaptor, Facade, Proxy, Iterator, Observer, Mediator, composite, chain of. Ways of using patterns Case studies of patterns in software architecture</td>
<td>6</td>
</tr>
<tr>
<td>Unit IV</td>
<td>Introduction to Middleware, Types of Middleware, Application servers, Introduction to Java EE, Introduction to Java EE technologies like JMS, JDBC, RPC, RMI, SOCKET. EJB 3.0 Architecture, Entity, Session, Message beans, XML, XSLT. Specifications and characteristics of Middleware technologies. Recent advances in Middleware technologies</td>
<td>6</td>
</tr>
<tr>
<td>Unit V</td>
<td>Introduction to three tier and n-Tier Web Architectures, XML, Client side technologies HTML, DHTML, Java Applets, Active X controls, DOM, AJAX. Need of Client side technology in multi-tier architectures Examples of three tier and n-tier architectures, client side technologies Case study of mobile or portable client side technologies.</td>
<td>6</td>
</tr>
<tr>
<td>Unit VI</td>
<td>Need of server side technology in multi-tier architectures, Java Web Services, Server side technologies: JSP, JSF, SOA,MVC. Java Servlets, struts</td>
<td>6</td>
</tr>
</tbody>
</table>
Text Book:

2. Erich Gamma, "Design Patterns"
4. xyz, PHI Publications

Reference Book:

1. James L. Weaver, Kevin Mukhar, "Beginning J2EE 1.4: From Novice to Professional

Software Architecture Laboratory

Objectives:
- Understand various non-functional requirements that lead to need for good Architecture and good Design.
- Explore various design Patterns and learn to implement them
- Understand how to design and partly implement Client Server systems using components and web services
- Explore implementations of some of the underlying technologies of distributed client server applications

Faculty can set assignments based on Microsoft / Java world / OR Open source based languages, platforms, Middleware, APIs.

Design Patterns

1. Study any two patterns and SUBMIT a design pattern specification these in a standard format along with their UML diagrams. The specification must include Problem/ Issue, Audience/ Context, Forces, Solution, Discussion/ Consequences/ Implementation, Related Patterns, Example Instances, References.

2. Implement an ITERATOR/OBSERVER or any two patterns pattern in language of your choice and submit it along with a write-up with its specification.

3. Study and submit a report for any of the MVC based Frameworks (J2EE/EJB)
Implement a web application for a system covering

4. A representative paper design of the above system using navigation, components, interfaces, its deployment issue with UML

5. Develop dynamic and interactive web client using JSP, Scripting/Applets/ActiveX controls

6. Develop server side programming with database handling/ servlets /EJB/webservice on server side with security aspects covered.
# 410445: Elective II: Multimedia Systems

<table>
<thead>
<tr>
<th>Teaching Scheme</th>
<th>Examination Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teaching: 3Hrs/Week</td>
<td>Theory: 100 Marks</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unit III</strong></td>
<td>Multimedia Audio: Data structures used in audio files, Characteristics of sound waves, psycho, digital audio, MIDI and MIDI File format, CD and DVD formats. Audio file formats: WAV, VOC, AVI, MPEG Audio Audio compression: Compression in audio PCM,DM, DPCM Study of different audio file formats and compression techniques Programming considerations for audio compression.</td>
<td>6</td>
</tr>
<tr>
<td>Analysis of video formats, compression and streaming</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------------------------------------------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Unit V</strong> Animation and Multimedia Languages, Learn to use OpenGL</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Animation:</strong> Basics of animation, types of animation, principles of animation, techniques of animation, Creating animation.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>OpenGL:</strong> Open GL over windows/Linux, Extension, programming languages, SDK, shadowing techniques, rendering, Programming aspects in creating simple animation using OpenGL</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Unit VI</strong> Multimedia Communication and applications, Study of Multimedia networking, Quality of data transmission, Multimedia over IP, Media on Demand</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Applications</strong> Media Entertainment, Media consumption, web-based applications, e-learning and education, Students Presentations: Different Multimedia applications, Analysis and development of Multimedia application</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Text Book**

3. Ranjan Parekh, "Principles of Multimedia", TMH.

**References:**

4. “DeMustified Video”
410445 Elective II: Mobile Computing

Teaching Scheme: 3Hrs/Week

Marking Scheme:
Theory: 100 Marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit I</td>
<td>Applications, history of mobile communication, introduction to GSM system, GSM background, GSM operational and technical requirements. cell layout and frequency planning, mobile station, base station systems, switching sub systems, home locations, register, Visiting Location Register (VLR), equipment identity register, echo canceller. GSM network structure, Recent Advances and application Standards in Mobile OS</td>
<td>6</td>
</tr>
<tr>
<td>Unit II</td>
<td>Time and Frequency Domains representations, structure of TDMA slot with frame; Time organization of signaling channels, frequency hopping. TDMA standards and Applications, Time Organization of signaling Channels</td>
<td>6</td>
</tr>
<tr>
<td>Unit III</td>
<td>Mobility Management, Signaling protocols, steps in formation of a call, location updates, MS-PSTN call, PSTN-MS call, MS-MS call, call handover. Functioning and types of PSTN networks</td>
<td>6</td>
</tr>
<tr>
<td>Unit IV</td>
<td>Security issues in mobile computing, Authentication, encryption, Characteristics of SIM, equipment identification, Review and interactive discussions on home tutorials, classroom tutorials and students presentation. Review of recent advances in the subject. Security Application development for Mobile OS</td>
<td>6</td>
</tr>
<tr>
<td>Unit V</td>
<td>Multiplexing issues in time and frequency domains, FDMA, TDMA, CDMA,</td>
<td>6</td>
</tr>
<tr>
<td>Unit VI</td>
<td>Physical layer, data link layer, MAP Protocols, MTP3, SCCP, TCAP protocol, message formation, MAP protocol-MAP protocol for MM, MAP protocol for basic service support. Application layers RR-layer, MM-layer, CC-Layer, API’s for mobile application development</td>
<td>6</td>
</tr>
</tbody>
</table>

Text Books

1. Asha Mehrotra, GSM System Engg., Artech House

Reference Books

2. Jochen Schiller, Mobile Communication, Pearson Education Asia
### 410445: Elective II: Embedded Systems

**Teaching Scheme:**
Teaching 3Hrs/Week

**Examination Scheme**
Theory: 100 Marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit II</td>
<td>System hardware, Interrupt structure and Applications, ARM7 Processor - Architecture, Programmer’s model, Modes of operation, Interrupt Structure and Applications, Comparison with ARM9, Details of Components of Embedded Systems - Management of Power Supply, Clocking Unit, Real Time Clock and Timers, Reset Circuitry and Watchdog Timer, Structural Units of Processor, Processor and Memory Selection, Memory Map Of Embedded System, Interfacing Processors, Memories and I/O, Processor and Memory Selection, Memory Map Of Embedded System</td>
<td>06</td>
</tr>
<tr>
<td>Unit III</td>
<td>I/O interfacing and Communication Buses, Serial Data Communication using USB/CAN/RS-232C and Comparison, I/O devices, ADC / DAC, Optical Devices such as LED / LCD Display devices, Opto-Isolator, Relay &amp; stepper motor, Timers/Counters. Parallel v/s serial communication, Parallel ports their uses in device interfacing, Different serial communication Protocols - RS232C, RS 485, I2C, CAN, &amp; USB - Protocol Architecture, topology, different Packets, Communication Cycle, Arbitration, Applications, Their comparison and area of application</td>
<td>06</td>
</tr>
<tr>
<td>Unit IV</td>
<td>Programming concepts, Embedded System Programming, C /C++/Java programming concepts, Software development cycle, Assembly language V/s High Level Language and its suitability for applications development, C program elements – Micros and functions, data types, data structures, modifiers, statements, loops and pointers, queues and stacks, List &amp; Order List and their use in the implementation of Embedded System software, Brief overview of Converting Embedded System Programming In C++ &amp; Java, Process of Converting assembly language program and C language program to ROM image</td>
<td>06</td>
</tr>
</tbody>
</table>
Difference between Compliers & Cross Compliers. Embedded System testing, simulation and debugging tools – simulators, laboratory tools-Oscilloscope, Logic Analyzer and In-Circuit Emulator.

|---|---|

<table>
<thead>
<tr>
<th>Unit VI</th>
<th>Exemplary Operating Systems and Representative Embedded Systems, Design of tasks and synchronization and IPCs Examples of Real Time OS, embedded System OS and Handheld OS – Windows XP Embedded, Embedded Linux, VxWorks, MicroC/OS II, Palm OS, Symbian OS - Their features and applications, Representative Embedded Systems – Digital Thermometer, Handheld computer, Navigation system, IP phone, Software Defined Radio, Design Examples and case study of – Mobile Phone, Digital camera, Automatic Cruise control System, their Block diagram, class diagrams, Design of tasks and synchronization and IPCs. (Features, IPC used, Scheduling, Applications), Design of tasks and IPCs.</th>
</tr>
</thead>
</table>

**Text Books**


**Reference Book:**

# 410445 Elective II: Software Testing and Quality Assurance

### Teaching Scheme
- Lectures: 3 Hrs/Week

### Examination Scheme
- Theory: 100 marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unit II</strong></td>
<td>Introduction, Need of black box testing, Black box testing Concept, Requirement Analysis, Test case design criteria, Testing Methods, requirement based testing, Positive &amp; negative testing, Boundary value analysis, Equivalence Partitioning class, state based or graph based, cause effect graph based, error guessing, documentation testing &amp; domain testing, design of test cases. Case studies of Black-Box testing.</td>
<td>06</td>
</tr>
<tr>
<td><strong>Unit III</strong></td>
<td>Introduction, Need of white box testing, Testing types, Test adequacy criteria, static testing by humans, Structure - logic coverage criteria, Basis path testing, Graph metrics, Loop Testing, Data flow testing, Mutation Testing, Design of test cases. Testing of Object oriented systems, Challenges in White box testing, Case-study of White-Box testing</td>
<td>06</td>
</tr>
<tr>
<td><strong>Unit IV</strong></td>
<td>Test organization, Structure of testing, Measurement tools, Testing metrics: Type of metric – Project, Progress, Productivity, Metric plan, Goal Question metric model, Measurement in small &amp; large system. Other Software Testing: GUI testing, Validation testing, Regression testing, Scenario testing, Specification based testing, Adhoc testing, Sanity testing, Smoke testing, Random Testing. Advances in Software Testing Methods</td>
<td>06</td>
</tr>
<tr>
<td><strong>Unit V</strong></td>
<td>Software quality, Quality attribute, Quality Assurance, Quality control &amp; assurance, Methods of quality management, Cost of quality, Quality management, Quality factor, Quality management &amp; project management, Software quality metrics-TQM, Six Sigma, ISO, SQA Model.</td>
<td>04</td>
</tr>
<tr>
<td><strong>Unit VI</strong></td>
<td>Manual testing, Automated Testing Tools &amp; Case studies, Study of Testing tools (QTP, Rational Robot, Winrunner, Loadrunner), Case studies based on Web based, GUI testing, Manual testing Vs Automated testing, Automated Testing Tools Case studies</td>
<td>04</td>
</tr>
</tbody>
</table>
Text books:

References Books:
Principles of Compiler Design Assignments

LEX
1. Assignment to understand basic syntax of LEX specifications, built-in functions and Variables.
2. Implement a lexical analyzer for a subset of C using LEX Implementation should support Error handling,

YACC
1. Assignment to understand basic syntax of YACC specifications, built-in functions and Variables
2. Write an ambiguous CFG to recognize an infix expression and implement a parser that recognizes the infix expression using YACC. Provide the details of all conflicting entries in the parser table generated by LEX and YACC and how they have been resolved

Intermediate Code Generation
3. Write an attributed translation grammar to recognize declarations of simple variables, "for", assignment, if, if-else statements as per syntax of C or Pascal and generate equivalent three address code for the given input made up of constructs mentioned above using LEX and YACC. Write a code to store the identifiers from the input in a symbol table and also to record other relevant information about the identifiers. Display all records stored in the symbol table.

Code Optimization
4. Assignment to optimize the generated equivalent three-address code in 3rd assignment.

Code Generation
5. Generate the target code for the optimized code in 4th assignment.
410447 Project Work

Objectives:
To expose students with project-product development cycle using industrial experience, use of state-of-art technologies. To encourage and expose students for participation in National/International paper presentation activities. Exposure to Learning and knowledge access techniques using Conferences, Journal papers and participation in research activities.

Term I
Teaching Scheme:
Practical: 2 Hours/Week

Examination Scheme:
Term Work: 50 Marks

Term I project term work

1. Internal guide allocation for the BE Project: Assistant Professor in computer engineering can guide the BE Project. In case of non-availability of qualified guide, HOD can allocate the senior teacher as a guide, without compromising the quality of the work. The Project laboratory of 4 project groups (3 to 4 students in one group) constituting one laboratory batch (2 hrs per week), be allocated to the guide.

   The project group will submit the synopsis including title of the project, Technical Key Words (Ref. ACM Keywords) and relevant mathematics associated with the Project, names of at least two conferences, where papers can be published, Review of Conference/Journal papers (at least 10 papers + White papers or web references, (if any)) supporting the project idea, Plan of project execution using planner or alike project management tool. (Recommended dates: 3 weeks after Commencement of the Term). Preferably, the projects are Industry Sponsored or part of high level research/ Sponsored Research Project that are not conducted for any award of the educational degree.

2. The project conduct and procedures are amended as detailed below:-

   Problem statement feasibility assessment using, satisfiability analysis and NP-Hard, NP-Complete or P type using modern
algebra and relevant mathematical models.(recommended date of submission:- 8 weeks before term end)

3. Use of above to identify objects, morphism, overloading, functions and functional relations and any other dependencies. (recommended submission date:- 6 weeks before term end)

4. Functional dependency graphs and relevant UML diagrams or other necessities.(recommended submission date:- 3 weeks before term end)

5. Testing of problem statement using generated test data (using mathematical models, Function testing principals) selection and appropriate use of testing tools, testing of UML diagram's reliability. (recommended submission date:- two weeks before term end)

6. The index of submission must cover above mentioned 5 heads in addition to the instructions by the guide.

7. Students must submit atleast one technical paper on the project design in the conferences/workshops in IITs, Central Universities or UoP Conferences or equivalent International Conferences.

8. A Report consisting of problem definition, literature survey, platform choice, SRS (System Requirement Specification) Document in specific format and high-level design document along with Annex A: Laboratory assignments on Project Analysis of Algorithmic Design, Annex B: Laboratory assignments on Project Quality and Reliability Testing of Project Design at the end of term-I and Annex C: Project Planner and progress report. The term work at the end of Term-I shall be assessed and evaluated for 50 marks by the panel of examiners in the subject (Internal (preferably guide) and external examiner from Computer Department of Engineering Colleges).

**Term-I Project Laboratory Assignments**

1. Project problem statement feasibility assessment using NP-Hard, NP-Complete or satisfiability issues using modern algebra and/or relevant mathematical models.
2. Use of above to identify objects, morphisms, overloading in functions \textit{(if any)}, and functional relations and any other dependencies \textit{(as per requirements)}.

3. Use of above to draw functional dependency graphs and relevant UML diagrams or other necessities using appropriate tools.

4. Testing of project problem statement using generated test data (using mathematical models, GUI, Function testing principals, \textit{if any}) selection and appropriate use of testing tools, testing of UML diagram's reliability.
# 410448: Distributed Operating Systems

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
</table>
**Message Passing** - Introduction, Desirable features of good message passing system, Issues in IPC by Message passing, RPC, RMI Synchronization, Buffering, Multidatagram messages, Encoding and decoding of message data, Process addressing, Failure Handling, Group communication, Case Study: RMI, CORBA. Advances in Distributed Systems | 7               |
| **Unit II**   | **Architecture of Distributed System** – Introduction, Motivations, Concepts of Distributed System, Process Synchronization, System architecture types, Distributed operating system, NOS, Middleware Communication Networks, Communication primitives, Architectural models of Distributed System  
**Synchronization**  
Introduction, Inherent Limitations of a Distributed System, Lamport’s logical clock, Vector clock, Global states, Concept of Process, Process Migration, Threads Clock synchronization, Event ordering, Mutual Exclusion, Deadlock, Election Algorithms  
Issues in Designing Distributed System and role of middleware in Distributed System | 6               |
| **Unit III**  | **Distributed Mutual Exclusion** – Introduction, Classification of mutual exclusion algorithms, Preliminaries, A simple solution to distributed mutual exclusion, non token based algorithms, Ricart Agrawala algorithm, Token based algorithms, Suzuki Kasami’s broadcast algorithms  
**Distributed Deadlock detection** – Introduction, Preliminaries, Deadlock handling strategies, Issues in deadlock detection and resolution, Control organizations for distributed deadlock detection, Centralized deadlock detection algorithms, Distributed deadlock detection algorithms, Avoidance and Prevention algorithms, Hierarchical deadlock detection algorithms  
**Agreement Protocols** – Introduction, System Model, Classification of agreement problems, Solutions to the Byzantine Agreement problem, Applications of Agreement algorithm, Distributed Synchronization and | 8               |
<table>
<thead>
<tr>
<th>Unit</th>
<th>Topic</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>IV</td>
<td>Distributed Resource Management, Concepts of File System, Scheduling Algorithms</td>
<td>8</td>
</tr>
</tbody>
</table>
|      | **Distributed File System**  
Introduction, Architecture, Mechanisms for building distributed file system, Design issues, Case studies, Log structured file systems, Google FS |   |
|      | **Distributed Shared Memory** – Introduction, Architecture and Motivation, algorithms for implementing DSM, Memory Coherence, Coherence protocols, Design issues, Case studies-Linda |   |
|      | **Distributed Scheduling** – Introduction, Motivation, Issues in load distribution, Components of load distributing algorithms, Stability, Load distributing algorithms, Performance Comparison, Selecting a suitable load sharing algorithms, Requirements for load distributing, Load sharing policies, Task migration |   |
|      | Distributed File System: Comparisons |   |
| V | Recovery and Security mechanism, Concepts of Database system Security, Basic concepts of Recovery and Types of Failures | 8 |
|      | **Recovery** – Introduction, Basic concepts, Classification of failures, Backward and forward error recovery, Backward error recovery, Recovery in concurrent systems, Consistent set of checkpoints, Synchronous and Asynchronous check pointing and recovery |   |
|      | **Access and Flow control** – Introduction, Preliminaries, Access matrix Model, Implementations of Access Matrix, Safety in Access matrix model, advanced models of protection, Case studies-Unix OS |   |
|      | Distributed Fault tolerance and Security, Advances in Recovery and security mechanisms |   |
|      | **Grid Computing**  
|      | **SOA**: Basic SOA Definition, Overview of SOA, SOA and Web Services, Service Oriented Grid, SOA Design and Development, Advantages and Future of SOA |   |
|      | Grid computing, Cloud and SOA |   |
Text books:

Reference Books:
1. “Distributed System Principles and Paradigms”, Andrew S. Tanenbaum, 2\textsuperscript{nd} edition, PHI
2. “Distributed Systems”, Colouris, 3\textsuperscript{rd} Edition
# Advanced Computer Architecture

**Teaching Scheme:**
Theory: 4 Hrs/Week

**Examination Scheme**
Theory: 100 Marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
</table>
| Unit I | Overview of Parallel Processing and Pipelining Processing, study and comparison of uni-processors and parallel processors. Conventional and EPIC architecture. Evolution of parallel processors, future trends and there architecture.  
**Overview of Parallel Processing and Pipelining Processing**  
**Principles of scalable performance:** Performance Metrics and Measures, Speedup Performance Laws, Programming aspects for Intel Itanium Processor | 6               |
| Unit II| Principles and implementation of Pipelining, Classification of pipelining processors, Pipeline Architecture, Study and comparison of processors with and without pipelining. General pipelining reservation table, Design aspect of Arithmetic and Instruction pipelining, Pipelining hazards and resolving techniques, Data buffering techniques, Job sequencing and Collision, Advanced pipelining techniques, loop unrolling techniques, out of order execution, software scheduling, trace scheduling, Predicated execution, Speculative loading, Register Stack Engine, Software pipelining, VLIW (Very Long Instruction Word) processor, Case study: Super scalar Architecture- Pentium, Ultra SPARC. Super scalar architecture of Pentium, Ultra SPARC, Advances in pipeline architectures. Implementation issues of a program on any pipelined processor their analysis. | 8               |
| Unit III| Study and comparison of Vector and array processors, Vector and Array Processor, Basic vector architecture, Issues in Vector Processing, Vector performance modeling, vectorizers and optimizers, Case study: Cray Arch.  
**SIMD Computer Organization** Masking and Data network mechanism, Inter PE  
Communication, Interconnection networks of SIMD, Static Vs Dynamic network, cube hyper cube and Mesh Interconnection network.  
**Parallel Algorithms For Array Processors:** Matrix Multiplication, Sorting, SIMD computer organization. Implementation issues of Matrix multiplication and sorting on array processor and their analysis. | 8               |
### Unit IV
Microprocessor Architectures, study and comparison of Loosely and Tightly coupled multiprocessors.

Loosely and Tightly coupled multiprocessors, Processor characteristics of multiprocessors, Inter Processor communication network, Time shared bus, Crossbar switch, Multiport Memory Model, Memory contention and arbitration techniques, Cache coherency and bus snooping, Massively Parallel Processors (MPP), Cow’s and NOW’s Cluster and Network of Work Stations, Chip Multiprocessing (CMP), Case Study of IBM Power4 Processor Inter Processor Communication and Synchronization, Implementation issues of a program on multiprocessor system.

### Unit V

**Parallel Programming Techniques:** Message passing program development, Synchronous and asynchronous message passing, Message passing parallel programming, Shared Memory Programming, Data Parallel Programming. Implementation issues of a multithreaded program.

### Unit VI
Parallel software issues, study of parallel programming concepts.

a) Parallel algorithms for multiprocessors, classification of parallel algorithms, performance of parallel algorithms

b) Operating systems for multiprocessors systems, Message passing libraries for parallel programming interface, PVM (in distributed memory system), Message Passing Interfaces (MPI), Threads (in shared memory system)
c) Parallel Programming Languages : Fortran 90, Occam, C-Linda, CCC etc.

### Text Books

### References:
1. V.Rajaraman, L Sivaram Murthy, "Parallel Computers", PHI.
5. Richard Y. Kain, Advanced Computer Architecture
### 410450: Elective III: Pattern Recognition

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unit I</strong></td>
<td>Basics of pattern recognition Features, Feature Vectors and Classifiers, Supervised versus Unsupervised Pattern Recognition, Comparison of Supervised and unsupervised Pattern Recognition, Feature Vectors and Classifiers</td>
<td>7</td>
</tr>
<tr>
<td><strong>Unit II</strong></td>
<td>Bayesian decision theory: Classifiers, Linear Discriminant Functions and Decision Hyperplanes, Perceptron, Least Squares Methods, Mean Square Estimation Revisited, Review and interactive discussions on home tutorials, classroom tutorials and students presentation. Review of recent advances in the subject. Challenges in Bayesian decision theory</td>
<td>7</td>
</tr>
<tr>
<td><strong>Unit III</strong></td>
<td>Parameter estimation methods, Maximum-Likelihood estimation, Gaussian mixture models, Expectation-maximization method, Bayesian estimation.</td>
<td>7</td>
</tr>
<tr>
<td><strong>Unit IV</strong></td>
<td>Context-dependent classification, Sequential Pattern classification, Context-dependent classification Discrete hidden Markov models, Continuous density hidden Markov models, Dimension reduction methods, Fisher discriminant analysis, Principal component analysis,</td>
<td>7</td>
</tr>
<tr>
<td><strong>Unit V</strong></td>
<td>Non-parametric techniques for density estimation, Parzen-window method, K-Nearest Neighbour method, Linear discriminant function based classifiers, Perceptron, Support vector machines.</td>
<td>7</td>
</tr>
<tr>
<td><strong>Unit VI</strong></td>
<td>Non-metric methods for pattern classification, Non-numeric data or nominal data, Decision trees, Unsupervised learning and clustering, Criterion functions for clustering, Algorithms for clustering: K-means, Hierarchical and other methods, Cluster validation, Review and interactive discussions on home tutorials, classroom tutorials and students presentation. Review of recent advances in the subject. Implementation issues in algorithms for clustering</td>
<td>7</td>
</tr>
</tbody>
</table>

**Text Books:**

**Reference Books:**

**Pattern Recognition Practical Assignment Topic (design six suitable assignments based on the following topic)**

1. Write program to build a Bayesian classifier
2. Write a program to recognize line patterns in a given set of points.
3. Implement Image Block matching by 2D Log Search and Brute-force approach
4. Write a program to implement Edit Distance Problem
5. Exploring difference between color spaces and color-based image segmentation. **Experiments with edge detection:**
   - Exploring different feature spaces - using Fourier shape descriptors, experimenting with wavelet transform, template matching.
   - Using and interpretation of ROC curves, experiments with PCA and ICA.
<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
</table>
| Unit I | Introduction and Applications of Gigabit Ethernet  
Business Drivers and need of it, Architecture and Overview of Gigabit Ethernet,  
Gigabit Ethernet Media Access Control (Half Duplex operation Vs Full Duplex Operation),  
Gigabit Ethernet Physical layer (1000Base X and 1000Base T),  
Applications of Gigabit Ethernet, Ethernet summary Migration from 10Mbps to Gigabit Ethernet, Network Design using Gigabit Ethernet a case study,  
Architecture and Overview of Gigabit Ethernet. Gigabit Ethernet Physical layer (1000Base X and 1000Base T) | 7 |
| Unit II | Integrated Services Digital Network, ISDN Interfaces and functions, Conceptual overview of Signaling System Numbers, Frame relay  
LAPF protocol, Concept of DLCI and its significance, Frame Relay Congestion control, ISDN Interfaces and functions (transmission structure, U-N Configuration, Protocol architecture etc) Congestion Notification (FECN, BECN and CLLM)  
Conceptual view of ISDN and ISDN standards, ISDN Interfaces and functions (transmission structure, U-N Configuration, Protocol architecture etc), ISDN Data Link layer (LAPD protocol, terminal adoption, I.465/v.120 ),ISDN Network layer (Overview, basic call control using Q.931), ISDN services, Conceptual overview of Signaling System Number 7  
**Frame Relay:** Frame relay Vs X.25, Frame relay Protocols and services (protocol architecture, Frame Mode call control), LAPF protocol (Both Core and Control), Concept of DLCI and its significance, Frame Relay Congestion control, Need, Congestion Control frame Work, Network use of CIR, and DE bit, Congestion Notification (FECN,BECN and CLLM). | 6 |
| Unit III | Driving forces and need, B-ISDN standards and services, B-ISDN Functional Architecture, B-ISDN Transmission structure, B-ISDN protocol architecture, SONET/SDH and comparison with other available standards  
**ATM:** Overview, ATM protocol architecture, Virtual Channels and Virtual Path Switching, Detail Functionality of ATM Layer( Cell structure, HEC, Cell Delineation etc),ATM Adoption layer (need, different types and comparison),ATM traffic and Congestion control ,Requirements, ATM service categories ,ATM traffic descriptors, ATM QOS parameters, Classical IP over ATM,ATM in LAN environment (LANE). | 8 |
| Unit IV | VDSL and DSL background and technological capabilities and its standards, Architecture, Conceptual overview of VDSL, Background and technological capabilities, Standards and associations, Architecture, Conceptual overview of VDSL, Deployment Case study, Market status | 8 |
and future.

<table>
<thead>
<tr>
<th>Unit V</th>
<th>Introduction to MPLS and QOS, Network Components of MPLS, working RSVP protocol, MPLS network Components, MPLS basic working, Applications, IETF approach, RSVP protocol, Integrated &amp; differential Services Framework.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Unit VI</td>
<td>WiFi &amp; WiMax Basics, IEEE Standards, Physical layer specifications, Introduction of WiFi and WiMax, WiMax Standards MAC protocol and operation, WiMax physical and MAC layer overviews Introduction to IEEE 802.11 standards, Physical layer specifications, MAC protocol and operation, Comparison of various standards. Implementing WiFi Network (Including configuration), Introduction to various WiMax Standards, WiMax physical and MAC layer overviews, OFDM and OFDMA, QOS and WiMax.</td>
</tr>
</tbody>
</table>

Text Books:

2. Rich Seifert, “Gigabit Ethernet” (Addison Wesley Inc.)

Reference Books:

2. Fundamental of WiMAX by Jeffery Andrews, Arunabha Ghosh and Rias Muhammad Pearson Education (For Unit VI)

Design 6 suitable assignment form the following topics for HPN Laboratory

1. Survey of Open Source Simulation tools for HPN, Virtual Network Laboratory for HPN
2. Network Scalability (Servers) using Gigabit Ethernet
3. Implementation of ISDN (U-N interface, Router configuration, Costing etc.)
4. Frame relay Congestion Control
5. Performance study of RSVP protocol using virtual HPN test-bed.
6. New node detection and authentication implementation in WiFi environment or simulated WiFi environment
7. Latency time measurements in different HPN Server-Client graphs configurations in simulated scalable environment (For example, HPN organization using bipartite graph)
### 410450 Elective III: Neural Network

**Teaching Scheme:** 4 Hrs. / Week  
**Practical:** 2 Hrs/Week  

**Exam Scheme:**  
**Paper:** 100 Marks  
**Term Work:** 50 Marks  
**Oral:** 50 Marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Unit I</strong></td>
<td>Introduction to Neural Networks, Biological Neural Networks, Characteristics of Neural Networks, Models of Neuron, Basic data structures: mapping of vector spaces, clusters, principal components. Basic Learning Rules, Recent advances in Neural Networks</td>
<td>6</td>
</tr>
<tr>
<td><strong>Unit II</strong></td>
<td>The Perceptron and its learning law. Classification of linearly separable patterns. Adaptive networks, Supervised Learning Neural Networks, Single layer and multi layer perceptrons, Radial basis function networks, Modular neural networks, Adaline and madaline</td>
<td>8</td>
</tr>
<tr>
<td><strong>Unit III</strong></td>
<td>Stochastic Processes and Neural Networks and Stimulated Annealing, Analysis of pattern storage Networks, Analysis of linear auto adaptive feed forward networks, Boltzman machine.</td>
<td>8</td>
</tr>
<tr>
<td><strong>Unit IV</strong></td>
<td>Analysis of linear auto adaptive feed forward networks, Feedback Neural Networks, Analysis of linear auto adaptive feed forward networks, Stochastic Networks &amp; Stimulated Annealing, Boltzman machine Multilayer Perceptrons, Analysis of pattern storage Networks.</td>
<td>6</td>
</tr>
<tr>
<td><strong>Unit V</strong></td>
<td>Unsupervised Learning, Adaptive Resonance Theory, Unsupervised Learning Networks Competitive learning, learning vector quantization, Principal component analysis of Hebbian Learning, Adaptive Resonance Theory, Kohonen self-organizing maps.</td>
<td>8</td>
</tr>
<tr>
<td><strong>Unit VI</strong></td>
<td>Applications of Neural Networks, Pattern classification, Pattern classification, Associative memories, Optimization, Applications in Image Processing, Applications in decision making, Applications in Image Processing, Applications in decision making</td>
<td>6</td>
</tr>
</tbody>
</table>

**Text Books**  
1. B. Yegnanarayana, “Artificial Neural Networks”, PHI  

Reference Books
1. Haykin, “Neural Network a comprehensive Foundation”, PHI

Neural Networks Laboratory Assignment
Teachers are instructed to frame suitable assignments on the following topics
1. Implementation of Perceptron Learning algorithm and Adaline algorithm for two class / three class problems.
3. Implementation of BAM simulator.
5. Implementation of ART simulator
### 410450: Elective III: Advanced Databases

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit I</td>
<td>Introduction, Parallel database architecture, speedup, scale-up I/O parallelism, Inter-query and Intra-query parallelism, Inter-operational and Intra-operational parallelism, parallel query evaluation, Design of parallel systems, Implementation issues of Parallel query evaluation, Design of parallel systems, Comparison of Inter-query and Intra-query parallelism.</td>
<td>08</td>
</tr>
<tr>
<td>Unit II</td>
<td><strong>Distributed Databases</strong>, Study of DDBMS architectures, Comparison of Homogeneous and Heterogeneous Databases, Analysis of Concurrency control in distributed databases, Implementation of Distributed query processing. Distributed data storage, Distributed transactions, Commit protocols, Availability, Distributed query processing, Directory systems-Idap, Distributed data storage and transactions.</td>
<td>08</td>
</tr>
<tr>
<td>Unit IV</td>
<td>Introduction to Decision Support, Data Warehousing, Creating and maintaining a warehouse. Introduction to Data warehouse and OLAP, Multidimensional data model, Data Warehouse architecture, OLAP and data cubes, Operations on cubes, Data preprocessing need for preprocessing, Multidimensional data model, OLAP and data cubes, Data warehousing Concepts, Study of Data preprocessing need for preprocessing, Simulating and maintaining a Warehouse, Analysis of Data preprocessing.</td>
<td>06</td>
</tr>
<tr>
<td>Unit V</td>
<td>Introduction to data mining, Data mining functionalities, clustering - k means algorithm, classification - decision tree, Baysian classifiers, Outlier analysis, association rules - apriori algorithm, Introduction to text mining, Implementing Clustering - k means algorithm, Analysis of Decision tree.</td>
<td>06</td>
</tr>
</tbody>
</table>
## Unit VI


### Text Books:

2. Jiawei han, Micheline Kamber, "Data Mining: Concepts and systems", Morgan Kaufmann Publishers

### Reference Books:


### List of Laboratory Assignment Topics:

1. Implementation of partitioning techniques and handling of skew
2. Directory systems using LDAP (Latest version)
3. Building cubes and OLAP analysis and construction of data mart
4. Association rule (Apriori algorithm) or Clustering algorithm (K-means)
5. IR assignment: TF, IDF, Stop words
6. Case Study : Anyone DBMS from open source: mysql, postgres SQL
   Anyone DBMS from: ORACLE, SQL server, DB2, CauchDB
# 410451 Elective IV: VLSI and Digital Systems

**Teaching Scheme:** 4Hrs/Week  
**Evaluation Scheme:** Theory 100 Marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit I</td>
<td>VLSI concepts and technology, Semiconductors, Semiconductor Devices, Integrated circuits and the Planar Process, Semiconductor Technology Families, Computer Simulation of VLSI</td>
<td>6</td>
</tr>
</tbody>
</table>
| Unit II | Modern CMOS Technology  
Introduction, Choosing a Substrate, Active Region Formation, Process Option for Device Isolation – Shallow Trench Isolation, N and P Well Formation, Process Options for Active Region and Well Formation, Gate Formation, Tip or Extension(LDD) Formation, Source/Drain Formation, Contact and Local Interconnect Formation, Multilevel Metal Formation, CMOS Process Flow | 8 |
| Unit III | Historical Development and Basic Concepts  
Crystal Structure, Basic Properties of Silicon Wafers  
Crystal Growth Wafer Fabrication Wafer Preparation and specification, Light sources, wafer exposure systems, Photoresists, Defects in crystals, Raw materials and Purification, Czochralski and Float-Zone Crystal Growth Methods, Wafer Preparation and specification, Lithography: Light sources, Mask engineering: Basic concepts of Thermal oxidation and the si/sio2 Interface, Basic concepts of dopant diffusion, implants in real silicon, basic concepts of chemical vapor deposition, physical vapor deposition, basic concepts of wet etching and plasma etching, back-end technology, contacts, interconnects and vias, dielectrics, Contamination reduction: Clean factories, wafer Cleaning, gettering, Wafer Development and Dopants. | 6 |
| Unit IV | What is VHDL? Introduction: Modeling Digital Systems: describing systems, events, propagation delays, and concurrency, waveforms and timing, signal values, shared signals, simulation vs. synthesis: the simulation model, the synthesis model, field programmable gate arrays(FPGAs), basic language concepts: signals, entity-architecture, concurrent statements, constructing VHDL models using CSAs, understanding delays, The role of hardware description languages, VHDL Programming | 6 |
| Unit V | Introduction to Digital Design, Analog versus Digital, Software aspects of digital design, application-specific IC’s, Digital devices, electronic aspects of digital design, integrated circuits, programmable logic devices, number systems and codes, digital circuits, digital-design levels, Software aspects of digital design | 8 |
| Unit VI | Combinational logic design principles, sequential logic design principles, Memories, CPLDS, and FPGAs | 6 |
Text Books:

410451 Elective IV: Operations Research

Teaching Scheme: 4Hrs/Week  
Evaluation Scheme: Theory 100 Marks

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit I</td>
<td>Operation Research Models, Solving the OR model, Queuing and Simulation Models, Linear Programming (LP) Applications, Modeling with Linear Programming: Two-Variable LP model, Simplex method and sensitivity analysis, Advanced Linear Programming, Graphical LP Solutions, Duality, Goal Programming.</td>
<td>6</td>
</tr>
<tr>
<td>Unit II</td>
<td>Review of basic probability, Random variables and Probability distributions, Random Processes, Transition Probability and Transition Matrix, Variations in expected value criteria, Four Common distributions, Solutions of mixed strategy games. Decision making under certainty, Decision making under risk, Decision making under uncertainty, Markov Process and Markov chain, Game theory: Introduction and definitions, Characteristics of Game theory, Maxima-minima, Saddle point, Optimal Strategies and value of game, solution of game with saddle point, Continuous review models, Multi-period models.</td>
<td>8</td>
</tr>
<tr>
<td>Unit III</td>
<td>Queuing system, queuing problem, Transient and steady States, List of symbols, Utilization factor, Elements of Queuing models, Generalized poisson queuing model, Specialized Poisson Queues, M/G/1 formula, Role of exponential distribution in queuing systems, Birth and Death Models, Queuing decision models, Review of recent advances in the subject.</td>
<td>6</td>
</tr>
<tr>
<td>Unit IV</td>
<td>Job sequencing: Introduction, Terminology and Notions, Principal Assumptions, Solution of sequencing problem, Project Management: Applications and basic steps in PERT/CPM techniques, Network diagrams with time estimates and analysis, resource allocation, Processing $n$ jobs with 2 machines, processing 2 jobs with $m$ machines, Processing $n$ jobs with $m$ machines, Problem of dimensionality, Inventory Models</td>
<td>6</td>
</tr>
<tr>
<td>Unit V</td>
<td>Non-Linear Programming (NLP) Introduction, Practical solutions of NLP, General from of NLP, Canonical form of NLP, Introduction to Separable functions, Piece-wise linear approximation of NLP, Geometric programming problem, Constrained and unconstrained problems of Maxima and Minima, Lagrangian method, To derive necessary condition of optimality, “Normality” and “Orthogonality” conditions, Problems with inequality constrains. Computational procedures of Fractional Algorithms</td>
<td>8</td>
</tr>
<tr>
<td>Unit VI</td>
<td>Concept of dynamic programming, Deterministic dynamic programming, Forward and Backward recursion, Minimum path problem, models of single additive constraint, Multiplicatively Separable return, Applications of dynamic programming in</td>
<td>6</td>
</tr>
</tbody>
</table>
production, linear programming and reliability, Decision trees and Bellman’s principal of optimality, Recursive nature of computations in dynamic programming, Equipment replacement problem, Investment models, Expectation of a random variable. Problem of dimensionality, Inventory Models,

Text Books:

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit I</td>
<td>Introduction to Cloud Computing, Definition, Characteristics, Components, Cloud provider, SAAS, PAAS, IAAS and Others, Organizational scenarios of clouds, Administering &amp; Monitoring cloud services, benefits and limitations, Deploy application over cloud, Comparison among SAAS, PAAS, IAAS Cloud computing platforms: Infrastructure as service: Amazon EC2, Platform as Service: Google App Engine, Microsoft Azure, Utility Computing, Elastic Computing</td>
<td>8</td>
</tr>
<tr>
<td>Unit II</td>
<td>Introduction to Cloud Technologies, Study of Hypervisors Compare SOAP and REST Web services, AJAX and mashups - Web services: SOAP and REST, SOAP versus REST, AJAX: asynchronous 'rich' interfaces, Mashups: user interface services Virtualization Technology: Virtual machine technology, virtualization applications in enterprises, Pitfalls of virtualization Multitenant software: Multi-entity support, Multi-schema approach, Multitenance using cloud data stores, Data access control for enterprise applications,</td>
<td>8</td>
</tr>
<tr>
<td>Unit III</td>
<td>Data in the cloud: Relational databases, Cloud file systems: GFS and HDFS, BigTable, HBase and Dynamo. Map-Reduce and extensions: Parallel computing, The map-Reduce model, Parallel efficiency of Map-Reduce, Relational operations using Map-Reduce, Enterprise batch processing using Map-Reduce, Introduction to cloud development, Example/Application of Mapreduce, Features and comparisons among GFS, HDFS etc, Map-Reduce model</td>
<td>6</td>
</tr>
</tbody>
</table>
**Unit VI**

Cloud computing platforms, Installing cloud platforms and performance evaluation

Features and functions of cloud platforms:
Xen Cloud Platform, Eucalyptus, OpenNebula, Nimbus, TPlatform, Apache Virtual Computing Lab (VCL), Enomaly Elastic Computing Platform,

**Text Book:**
2. Enterprise Cloud Computing by Gautam Shroff, Cambridge
3. Cloud Security by Ronald Krutz and Russell Dean Vines, Wiley-India

**Reference Book:**
1. Google Apps by Scott Granneman, Pearson
2. Cloud Security & Privacy by Tim Malhar, S.Kumaraswammy, S.Latif (SPD, O’REILLY)
4. Cloud Computing Bible by Barrie Sosinsky, Wiley India
### 410451: Elective IV: Information Security

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Teaching Scheme</th>
</tr>
</thead>
</table>
| Unit I | **Need of information security** - Legal, Ethical and Professional Issues  
Attributes of security - authentication, access control, confidentiality, authorization, integrity, non-reproduction  
OSI security architecture - attacks, services and mechanisms  
Information security management - security policy, standards, guidelines and procedures, security lifecycle. Introduction to cryptography - Classical cryptography. Case-study: Information security management used in organizations, Problems on classical ciphers, Security architecture, Different standards related to IS. | 8               |
| Unit II | **Introduction to Secret key and cryptography** - Encrypt given messages using DES, AES, IDEA, Problems on cryptography algorithms, Principles, finite fields, stream cipher, block cipher modes of operation, DES, Triple DES, AES, IDEA, RC5, key distribution. | 6               |
| Unit III | **Introduction to Public key and Cryptography** - Encrypt given messages using ECC, Problems on key generation, cryptography algorithms Principles, Introduction to number theory, RSA- algorithm, security of RSA, **Key management** - Diffie-Hellman key exchange, man-in-the-middle attack, Elliptical curve cryptography, | 6               |
| Unit IV | **Study of applications and comparison of authentication functions, MAC, hash functions etc.** Implementation and analysis of authentication functions, MAC, hash functions etc. Requirements of message authentication, authentication functions, MAC, hash functions, MD5, SHA-512, HMAC, digital signatures, DSA, PKI, Authentication applications-, X.509, Kerberos. | 8               |
IPSEC, TLS, and SSL, Firewalls-design principles, trusted systems, Intrusion Detection System, Intrusion Prevention System. Introduction to cryptography - Classical cryptography. | 6               |
| Unit VI | **Study of Security services, web and email security considerations,** Electronic commerce security. Implementation and analysis of Security services, web security considerations, Electronic commerce security, Security services, web security considerations, PEM, PGP, S/MIME, | 6               |
Electronic commerce security- SET, Smart cards, strengths of Security services.

Text Books:


Reference Books:

3. Charlie Kaufman, Radia Perlman and mike speciner "Network security, private communication in a public world"
4. Christopher M. King, Curtis Patton and RSA press "Security architecture, design deployment and operations".
410452 Computer Lab II

Teaching Scheme
Practical: 4 Hrs/Week

Examination Scheme
Term Work: 50 Marks
Practical: 50 Marks

1. Inter-process communication using socket programming/ RPC mechanism including implementing multi-threaded echo server.

2. **Threads clock synchronization. Assignment on election algorithms.** Simulation of election algorithms (Ring and Bus Topology).
   a. a.Bully
   b. b.Ring

3. **Distributed/Hierarchical deadlock detection algorithms.** Clock synchronization: NTP / Lamports clock.

4. **Load distributing algorithms, Performance comparison.** Study of Distributed File System: NFS – CODA


410447 Project Work
(Term II)

Objectives:
To expose students with project-product development cycle using industrial experience, use of state-of-art technologies. To encourage and expose students for participation in National/ International paper presentation activities. Exposure to Learning and knowledge access techniques using Conferences, Journal papers and participation in research activities.

Term II Project Term Work
Teaching Scheme
Examination Scheme:
Practical: 6 Hours/Week
Term Work: 100 Marks
Oral: 50 Marks

1) Project workstation selection, installations and setup along with report to the guide. (recommended submission date:- 3 weeks after commencement of second term)
2) Programming of the project, GUI (if any) as per 1st Term term-work submission. (recommended submission date:- Progress report every week during laboratory)
3) Test tool selection for various testing recommended by preferably external guide and generate various testing result charts, graphs etc. including reliability testing. (7 weeks before Term II Conclusion)
4) Review of design and necessary corrective actions taking into consideration feedback report of Term I assessment, and other competitions/conferences participated like IIT, Central Universities, University Conferences or equivalent centers of excellence etc.
5) Students must submit and preferably publish atleast one technical paper in the conferences held by IITs, Central Universities or UoP Conference or International Conferences in Europe or US.
6) Final term work submissions in the prescribed format given by the guides consisting of a project report consisting of a preliminary report prepared in term-I, detailed design (all necessary UML diagrams) document, User Interface design, Laboratory assignments on test cases and test results generated by selected project testing tool, conclusions, appendix (if necessary), glossary, tools used and references at the end of Term-II.
The Term II examination is conducted by panel of examiners (preferably guide and expert from Industry having at least 5 years subject experience (or senior teacher in the subject in case of non-availability of industry expert). The project assessment shall be done using Live Project Demonstration [in existing functional condition], using necessary simulators (if required) and presentation by the students. The remarks of Term I assessment and related corrective actions must be assessed during examining the term-work.

**Term-II Project Laboratory Assignments**

1. Project workstation selection, installations along with setup and installation report preparations.
2. Programming of the project functions, interfaces and GUI (if any) as per 1st Term term-work submission using corrective actions recommended in Term-I assessment of Term-work.
3. Test tool selection and testing of various test cases for the project performed and generate various testing result charts, graphs etc. including reliability testing.
4. Review of design and necessary corrective actions taking into consideration the feedback report of Term I assessment, and other competitions/conferences participated like IIT, Central Universities, University Conferences or equivalent centers of excellence etc.