## I YEAR I SEMESTER

<table>
<thead>
<tr>
<th>Code</th>
<th>Group</th>
<th>Subject</th>
<th>L</th>
<th>P</th>
<th>Credit</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Advanced Problem Solving</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Computer System Design</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Embedded Systems</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Java and Web Technologies</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Elective -I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Object Oriented Modeling</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software Quality Assurance and Testing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software Architecture and Design Patterns</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Elective -II</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Software Design and Engineering</td>
<td>3</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Advanced Compiler Design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Image Processing and Pattern Recognition</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Lab</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Java and Web Technologies Lab</td>
<td>0</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Seminar</td>
<td>-</td>
<td>-</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Total Credits (6 Theory + 1 Lab.)</td>
<td></td>
<td></td>
<td>22</td>
</tr>
</tbody>
</table>
ADVANCED PROBLEM SOLVING

Unit I
OOP Using Java - Class and Objects, Variables, Operators, Expressions, Methods, Decision statements, Loops, Arrays, OOP concepts- Encapsulation, Inheritance, Polymorphism, Abstraction, Modularity, Exception handling, Input and Output, Java and Pointers, Interfaces, Packages, Abstract classes, Casting in Inheritance hierarchy, Casting with Interfaces, Vectors in java.util, Data Structures and OOP, Writing a java program- Design, coding, testing and debugging.
Basic concepts (Review) - Abstract Data Types, Data structures, Algorithms- Characteristics of Algorithms, Performance analysis- Time complexity and Space complexity, Asymptotic Analysis- Big O, Omega and Theta notations.

Unit II
Linear data structures - The List ADT, Array and Linked Implementations, Singly Linked Lists- Operations- Insertion, Deletion, Traversals, Doubly Linked Lists- Operations- Insertion, Deletion, Stack ADT, definitions, operations, Array and Linked implementations, applications- infix to postfix conversion, recursion implementation, tail recursion, nontail recursion, indirect recursion, Queue ADT, definitions and operations, Array and Linked Implementations, Priority Queue ADT, Deque ADT, Implementation using doubly linked lists, Stacks and Queues in java.util.

Unit III
Non Linear data structures-Trees-Basic Terminology, Binary tree ADT, array and linked representations, iterative traversals, threaded binary trees, Applications- Disjoint-Sets, Union and Find algorithms, Huffman coding, General tree to binary tree conversion, Realizing a Priority Queue using Heap.

Unit IV
Searching- Linear Search, Binary Search, Hashing- Hash functions, Collision- Handling schemes, Hashing in java.util, Dictionary ADT, Linear list representation, Skip list representation, Hash table representation, Comparison of Searching methods.
Sorting- Bubble Sort, Insertion Sort, Shell sort, Heap Sort, Radix Sort, Quick sort, Merge sort, Comparison of Sorting methods, Sorting in java.util.

Unit V
Graphs– Basic Terminology, Graph Representations- Adjacency matrix, Adjacency lists, Adjacency multilists, Graph traversals- DFS and BFS, Spanning trees- Minimum cost spanning trees, Kruskal’s Algorithm for Minimum cost Spanning trees, Shortest paths- Single Source Shortest Path Problem, All Pairs Shortest Path Problem.

TEXT BOOKS :
REFERENCE BOOKS:
1. Data structures and algorithms in Java, Robert Lafore, Pearson Education.
3. Classic Data structures in Java, T. Budd, Pearson Education.
5. An Introduction to Data structures and Algorithms, J.A. Storer, Springer.
7. Data structures with Java, J.R. Hubbard and A. Huray, PHI.
9. Data structures with Java, J.R. Hubbard, Schaum’s Outlines, TMH.
UNIT I
Computer structure – hardware, software, system software, Von-neumann architecture – case study. IA-32 Pentium: registers and addressing, instructions, assembly language, program flow control, logic and shift/rotate instructions, multiply, divide MMX, SIMD instructions, I/O operations, subroutines.
Input/Output organization, interrupts, DMA, Buses, Interface circuits, I/O interfaces, device drivers in windows, interrupt handlers

UNIT II
Processing Unit: Execution of a complete instruction, multiple bus organization, hardwired control, micro programmed control.
Pipelining: data hazards, instruction hazards, influence on instruction sets, data path & control consideration, RISC architecture introduction.

UNIT III
Memory: types and hierarchy, model level organization, cache memory, performance considerations, mapping, virtual memory, swapping, paging, segmentation, replacement policies.

UNIT IV
Processes and Threads: processes, threads, inter process communication, classical IPC problems, Deadlocks.

UNIT V
File system: Files, directories, Implementation, Unix file system
Security: Threats, intruders, accident data loss, basics of cryptography, user authentication.

TEXT BOOKS:

REFERENCE BOOKS:
EMBEDDED SYSTEMS

UNIT I

UNIT II

UNIT III
**Embedded Programming Concepts:** Software programming in Assembly language and High Level Language, Data types, Structures, Modifiers, Loops and Pointers, Macros and Functions, object oriented Programming, Embedded Programming in C++ & JAVA

UNIT IV
**Real – Time Operating Systems:** OS Services, Process and Memory Management, Real – Time Operating Systems, Basic Design Using an RTOS, Task Scheduling Models, Interrupt Latency, Response of Task as Performance Metrics - **RTOS Programming:** Basic functions and Types of RTOSES, RTOS VxWorks, Windows CE

UNIT V

**TEXT BOOKS:**

**REFERENCE BOOKS :**
4. An Embedded Software Primer, David E. Simon, Pearson Education.
5. Micro Controllers, Ajay V Deshmukhi, TMH.
7. Introduction to Embedded Systems, Shibu K.V, TMH.
JAVA AND WEB TECHNOLOGIES

Unit I:
HTML Common tags- List, Tables, images, forms, Frames; Cascading Style sheets;
Introduction to Java Scripts, Objects in Java Script, Dynamic HTML with Java Script, CSS

Unit II:
XML: Document type definition, XML Schemas, Document Object model, Presenting XML,
Using XML Processors: DOM and SAX
Review of Applets, Class, Event Handling, AWT Programming.
Introduction to Swing: JApplet, Handling Swing Controls like Icons – Labels – Buttons – Text
AWT Controls & Swing Controls Developing a Home page using Applet & Swing.

Unit III:
Java Beans: Introduction to Java Beans, Advantages of Java Beans, BDK Introspection, Using
Bound properties, Bean Info Interface, Constrained properties Persistence, Customizes, Java
Beans API.
Web servers: Tomcat Server installation & Testing.
Introduction to Servelets: Lifecycle of a Serverlet, JSDK The Servelet API, The javax.servelet
Package, Reading Servelet parameters, Reading Initialization parameters.

Unit IV:
More on Servelets: The javax.servelet HTTP package, Handling Http Request & Responses,
Using Cookies-Session Tracking, Security Issues.
JSP Application Design with MVC architecture. AJAX.

Unit V:
JSP Application Development: Generating Dynamic Content, Using Scripting Elements
Implicit JSP Objects, Conditional Processing – Displaying Values Using an Expression to Set
an Attribute, Declaring Variables and Methods Error Handling and Debugging Sharing Data
Between JSP pages, Requests, and Users Passing Control and Date between Pages – Sharing
Session and Application Data – Memory Usage Considerations
Database Access Database Programming using JDBC Studying javax.sql.* package Accessing
a Database from a JSP Page Application – Specific Database Actions Deploying JAVA Beans
in a JSP Page

TEXT BOOKS:
1. Web Programming, building internet applications, Chris Bates 2nd edition,
   WILEY Dreamtech (UNIT 1,2)
2. The complete Reference Java 2 Fifth Edition,Patrick Naughton and Herbert Schildt., TMH
   (Chapters: 25) (UNIT 2,3)
3. Java Server Pages –Hans Bergsten, SPD O’Reilly (UNITs 3,4,5)
REFERENCE BOOKS:

1. Programming world wide web-Sebesta,Pearson
2. Core SERVLETS AND JAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES, Marty Hall and Larry Brown Pearson
3. Internet and World Wide Web – How to program, Dietel and Nieto PHI/Pearson.
5. Murach’s beginning JAVA JDK 5, Murach, SPD
6. An Introduction to web Design and Programming – Wang-Thomson
8. Java Server Programming, Ivan Bayross and others, The X Team, SPD
10. Beginning Web Programming-Jon Duckett WROX.
12. Java Script, D. Flanagan, O’Reilly, SPD.
UNIT I
Introduction to UML: The meaning of Object Orientation, object identity, Encapsulation, information hiding, polymorphism, generosity, importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture.

Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams.

Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams.


Sequence Diagrams: Terms, concepts, depicting asynchronous messages with/without priority, callback mechanism, broadcast messages.

UNIT II
Basic Behavioral Modeling: Use cases, Use case Diagrams, Activity Diagrams.

Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT III
The Unified process: use case driven, architecture centric, iterative, and incremental

The Four Ps: people, project, product, and process

Use case driven process: why use case, capturing use cases, analysis, design, and implementation to realize the use cases, testing the use cases

Architecture-centric process: architecture in brief, why we need architecture, use cases and architecture, the steps to architecture, an architecture description.

UNIT IV
Iterative incremental process: iterative incremental in brief, why iterative incremental development?
The iterative approach is risk driven, the generic iteration.

The Generic Iteration workflow: phases are the first division workflow, planning proceeds doing, risks affect project planning, use case prioritization, resource needed, assess the iteration and phases

Inception phase: early in the inception phase, the archetypal inception iteration workflow, execute the core workflows, requirements to test.

UNIT V
Elaboration Phase: elaboration phase in brief, early in the elaboration phase, the architectural elaboration iteration workflow, execute the core workflows-Requirements to test.

Construction phase: early in the construction phase, the archetypal construction iteration workflow, execute the core workflow.

Transition phase: early in the transition phase, activities in transition phase

Case Studies: Automation of a Library, Software Simulator application (2-floor elevator simulator)

TEXT BOOKS:
2. UML 2 Toolkit By Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado WILEY-Dreamtech India Pvt. Ltd.
3. The Unified Software Development Process By Ivar Jacobson, Grady Booch, James Rumbaugh, Pearson Education

REFERENCE BOOKS:
1. Fundamentals of Object Oriented Design in UML By Meilir Page-Jones, Pearson Education
3. Practical Object-Oriented Design with UML By Mark Priestley, TATA McGraw-Hill
4. Object Oriented Analysis & Design By Brett D McLaughlin, Gary Pollice and David West, O’REILY.
7. UML and C++, R.C.Lee, and W.M.Tepfenhart, PHI.
UNIT I

UNIT II

UNIT III
Software Testing Strategy and Environment: Establishing testing policy, structured approach to testing, test factors, Economics of System Development Life Cycle (SDLC) Testing
Software Testing Methodology
Defects hard to find, verification and validation, functional and structural testing, workbench concept, eight considerations in developing testing methodologies, testing tactics checklist

UNIT IV
Software Testing Techniques
Black-Box, Boundary value, Bottom-up, Branch coverage, Cause-Effect graphing, CRUD, Database, Exception, Gray-Box, Histograms, Inspections, JADs, Pareto Analysis, Prototyping, Random Testing, Risk-based Testing, Regression Testing, Structured Walkthroughs, Thread Testing, Performance Testing, White-Box Testing
Software Testing Tools
Taxonomy of Testing tools, Methodology to evaluate automated testing tools, Load Runner, Win runner and Rational Testing Tools, Silk test, Java Testing Tools, JMetra, JUNIT and Cactus.

UNIT V
Testing Process
Testing Specialized Systems and Applications
Testing Client/Server – Web applications, Testing off the Shelf Components, Testing Security, Testing a Data Warehouse
TEXT BOOKS:

REFERENCE BOOKS:
UNIT I
Envisioning Architecture

Creating an Architecture
Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture.

UNIT II
Analyzing Architectures
Architecture Evaluation, Architecture design decision making, ATAM, CBAM.

UNIT III
Moving from one system to many
Software Product Lines, Building systems from off the shelf components, Software architecture in future.

UNIT IV
Patterns
Pattern Description, Organizing catalogs, role in solving design problems ,Selection and usage.

Creational and Structural patterns
Abstract factory, builder, factory method, prototype, singleton, adapter, bridge, composite, façade, flyweight, Proxy.

UNIT V
Behavioral patterns
Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

Case Studies
A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development

TEXT BOOKS:

REFERENCE BOOKS:
2. Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR, 2001
5. Design Patterns in Java, Steven John Metsker & William C. Wake, Pearson education, 2006
UNIT I

**Introduction to Software Engineering:** The evolving role of software, Changing Nature of Software, legacy software, Software myths.

**A Generic view of process:** Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models.

**Process models:** The waterfall model, Incremental process models, Evolutionary process models, Specialized process models, The Unified process.

**Software Requirements:** Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

**Requirements engineering process:** Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

UNIT II

**Role of Software Design**
The nature of the design process, transferring design knowledge, constraints upon the design process and product, recording design decisions, designing with others, context for design, economic factors, assessing design qualities, quality attributes of the design product, assessing the design process.

**Transferring Design Knowledge:** Representing abstract ideas, design viewpoints, the architecture concept, design methods, design patterns, Design representations, rationale for design methods.

**Design Processes and Strategies:** The role of strategy in design methods, describing the design process – The D – Matrix, design by top-down decomposition, design by composition, organizational influences upon design.

UNIT III

**Designing with objects and components**

**Designing with objects:** design practices for object-oriented paradigm, Object- oriented frameworks, Hierarchial object oriented design process and heuristics, the fusion method, the unified process.

**Component – based design:** The component concept, designing with components, designing components, COTS.

**Performing User interface design:** The Golden rules, Interface analysis and design models, user and task analysis, analysis of display content and work environment, applying interface design steps, user interface design issues, design evaluation.

UNIT IV

**Project Management and Metrics**

**Project Management:** The management spectrum: people, product, process and project, W5HH principle, critical practices.

**Metrics for Process and Projects:** Process metrics, project metrics, size-oriented metrics, function-oriented metrics, Object-oriented and use-case metrics, metrics for software quality, integrating metrics with in the software process.

UNIT V

**Project Scheduling and Risk Management**

**Project Scheduling:** Basic concepts, project scheduling, defining a task set and task network, timeline charts, tracking the schedule, tracking the progress for an OO project, Earned value analysis.

**Risk Management:** Reactive Vs. Proactive risk strategies, software risks, risk identification, risk projection, risk refinement, risk mitigation and monitoring, the RMMM plan.
TEXT BOOKS:

REFERENCE BOOKS:
UNIT I: Overview of Compilation: Phases of Compilation – Lexical Analysis, Regular Grammar and regular expression for common programming language features, pass and Phases of translation, interpretation, bootstrapping, data structures in compilation – LEX lexical analyzer generator.


TEXT BOOKS:
1. Principles of compiler design - A.V. Aho . J.D.Ullman; Pearson Education

REFERENCE BOOKS:
3. lex &yacc – John R. Levine, Tony Mason, Doug Brown, O’reilly
5. Engineering a Compiler-Cooper & Linda, Elsevier.
6. Compiler Construction, Louden, Thomson..
UNIT – I
Fundamental steps of image processing, components of an image processing of system. The image model and image acquisition, sampling and quantization, relationship between pixels, distance functions, scanner.

UNIT – II
Statistical and spatial operations, Intensity functions transformations, histogram processing, smoothing & sharpening – spatial filters Frequency domain filters, homomorphic filtering, image filtering & restoration. Inverse and weiner filtering, FIR weiner filter, Filtering using image transforms, smoothing splines and interpolation.

Morphological and other area operations, basic morphological operations, opening and closing operations, dilation erosion, Hit or Miss transform, morphological algorithms, extension to grey scale images.

UNIT- III
Segmentation and Edge detection region operations, basic edge detection, second order detection, crack edge detection, gradient operators, compass and laplace operators, edge linking and boundary detection, thresholding, region-based segmentation, segmentation by morphological watersheds.

Image compression: Types and requirements, statistical compression, spatial compression, contour coding, quantizing compression, image data compression-predictive technique, pixel coding, transfer coding theory, lossy and lossless predictive type coding, Digital Image Water marking.

UNIT –IV
Representation and Description
Chain codes, Ploygonal approximation, Signature Boundary Segments, Skeltons, Boundary Descriptors, Regional Descriptors, Relational Descriptors, Principal components for Description, Relational Descriptors

UNIT V
Pattern Recognition Fundamentals: Basic Concepts of pattern recognition, Fundamental problems in pattern recognition system, design concepts and methodologies, example of automatic pattern recognition systems, a simple automatic pattern recognition model

Pattern classification:
Pattern classification by distance function: Measures of similarity, Clustering criteria, K-means algorithm, Pattern classification by likelihood function: Pattern classification as a Statistical decision problem, Bayes classifier for normal patterns.

TEXT BOOKS :

REFERENCE BOOKS :
6. Pattern Recognition, R.Shinghal, Oxford University Press.