## V-Semester B. E. (Computer Technology)

**Course Code:** CT501  
**Title of the Course:** Advanced Data Structure

<table>
<thead>
<tr>
<th>Course Scheme</th>
<th>Evaluation Scheme (Theory)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture</td>
<td>Tutorial</td>
</tr>
<tr>
<td>03</td>
<td>01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Overview of data structures arrays, stacks, queues, link list, trees, graphs. Sparse matrix. Priority queues – definition, ADT, realizing a priority queue using heaps, definition, insertion, deletion, application-heap sort, external sorting- model for external sorting, multiway merge, polyphase merge.</td>
<td>09</td>
</tr>
<tr>
<td>II</td>
<td>Dictionaries, linear list representation, skip list representation, operations-insertion, deletion and searching, hash table representation, hash functions, collision resolution-separate chaining, open addressing-linear probing, quadratic probing, double hashing, rehashing extendible hashing, comparison of hashing and skip lists.</td>
<td>09</td>
</tr>
<tr>
<td>III</td>
<td>Search trees (part i) : binary search trees, definition, adt implementation, operations-searching insertion and deletion, balanced search trees- AVL trees, definition, height of an AVL tree, representation, operations - insertion, deletion and searching.</td>
<td>09</td>
</tr>
<tr>
<td>IV</td>
<td>Search trees (part ii) : introduction to red-black trees and splay trees, B-trees-B-tree of order-m, height of a B-tree, insertion, deletion and searching, comparison of search trees. Definitions &amp; operations on weight balanced trees (Huffman trees), 2-3 trees. Augmenting red-black trees to dynamic order statistics and interval tree applications. Operations on disjoint sets and its union-find problem implementing sets. Dictionaries, priority queues and concatenable queues using 2-3 trees.</td>
<td>09</td>
</tr>
<tr>
<td>V</td>
<td>Mergeable heaps: Mergeable heap operations, binomial trees implementing binomial heaps and its operations, 2-3-4. Trees and 2-3-4 heaps. Fibonacci heap. Graph theory definitions: definitions of isomorphic components. Circuits, fundamental circuits, cut-sets. Cut-vertices planar and dual graphs, spanning trees, kuratovski's two graphs.</td>
<td>09</td>
</tr>
</tbody>
</table>

**Text Book/s:**  
V-Semester B. E. (Computer Technology)

Course Code: CT502
Title of the Course: Design and Analysis of Algorithms

<table>
<thead>
<tr>
<th>Course Scheme</th>
<th>Evaluation Scheme (Theory)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture</td>
<td>Tutorial</td>
</tr>
<tr>
<td>03</td>
<td>01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td><strong>Introduction:</strong> Analyzing and Designing algorithm, Asymptotic notations: Big Oh, Omega, Theta notation, Average, Best and Worst case analysis of algorithms for Time and Space complexity, Amortized Analysis, Solving Recurrence Equations using Substitution method, Recursion-Tree Method and Master method.</td>
<td>09</td>
</tr>
<tr>
<td>II</td>
<td><strong>Divide-and-Conquer and Greedy Strategy:</strong> Binary Search, Merge and Quick Sort, The maximum-subarray problem, Strassen’s algorithm for matrix multiplication. Greedy Method-General Strategy, Knapsack problem, Job sequencing with deadlines problem, minimum cost spanning trees: Prim’s algorithm, Kruskal’s algorithm, Single source shortest path: Bellman-Ford algorithm, Dijkstra’s algorithm, Difference constraints and shortest paths, Huffman Coding etc.</td>
<td>09</td>
</tr>
<tr>
<td>III</td>
<td><strong>Dynamic Programming:</strong> Basic strategy, all pair shortest path: Shortest paths and matrix multiplication, Floyd-Warshall algorithm, Single source shortest paths, optimal binary search trees, Matrix-chain Multiplication, Elements of dynamic programming, traveling salesman problem.</td>
<td>09</td>
</tr>
<tr>
<td>IV</td>
<td><strong>Backtracking and Hash tables:</strong> The general method, 8-Queen’s problem, Sum of subsets, Graph Coloring, Hamiltonian Cycle, Graph Coloring, Hash tables, Hash functions, Open addressing.</td>
<td>09</td>
</tr>
</tbody>
</table>

Total 45

**Text Book/s:**

**References:**
V-Semester B. E. (Computer Technology)

Course Code: CT503
Title of the Course: Java Programming

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
</tr>
</thead>
</table>

Text Book/s:

Reference Book/s:
V-Semester B. E. (Computer Technology)

Course Code: CT504
Title of the Course: System Programming

<table>
<thead>
<tr>
<th>Course Scheme</th>
<th>Evaluation Scheme (Theory)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture</td>
<td>Tutorial</td>
</tr>
<tr>
<td>03</td>
<td>01</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>II</td>
<td>Machine Structure, Machine Language, And Assembly Language General Machine Structure, General Approach to a New Machine,</td>
<td>10</td>
</tr>
<tr>
<td>III</td>
<td>Assemblers General Design Procedure, Design of Assembler. Macro Language and the Macro processor Macro instructions, features of Macro Facility, Macro Instruction Arguments, Conditional Macro Expansion, Macro Calls within Macros, Macros Instructions Defining Macros, Implementation, Implementation of a restricted Facility A Two pass Algorithm, A Single pass Algorithm, Implementation of Macro Calls within Macros, Implementation within an Assembler.</td>
<td>9</td>
</tr>
<tr>
<td>IV</td>
<td>Loaders Loader schemes, &quot;Compile and go&quot; Loaders, general Loader scheme, absolute loaders, subroutine linkages, relocating loaders, direct linking loaders, other loader schemes-Binders, linking loaders; Overlays, Dynamic Binders, Design of and absolute Loaders, Design of a Direct-Linking loaders</td>
<td>09</td>
</tr>
<tr>
<td>V</td>
<td>Introduction to Device Drivers. Device drivers for Windows, Linux/Unix. Lexical Analysis in Compiler Design. Role of lexical analysis, recognition of tokens.</td>
<td>08</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>45</td>
</tr>
</tbody>
</table>

Text Books:
1. System Programming by John J. Donovan, Mc Graw Hill
2. System Programming by Leland Beck, Pearson Ed.
3. Assembly & Assemblers by Gorshine, Prentice Hall.

Reference Book/s:
1. Unix device drives by George Pajani, Pearson Ed.
2. Device Drives for Windows by Norton, Add Wesley
V-Semester B. E. (Computer Technology)

Course Code: CT505
Title of the Course: Design Principles of Programming Languages

<table>
<thead>
<tr>
<th>Unit</th>
<th>Course Scheme</th>
<th>Evaluation Scheme (Theory)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Lecture</td>
<td>Tutorial</td>
</tr>
<tr>
<td>I</td>
<td>03</td>
<td>01</td>
</tr>
<tr>
<td>II</td>
<td>Elementary Data types: Properties of Types &amp; objects, Scalar data types, Composite Data types.</td>
<td></td>
</tr>
<tr>
<td>III</td>
<td>Encapsulation: Structured data types, Abstract data types, Encapsulation by subprograms, Type definition, Inheritance, Polymorphism.</td>
<td></td>
</tr>
<tr>
<td>IV</td>
<td>Sequence Control: Implicit and Explicit Sequence Control, Sequencing with arithmetic expressions, Sequence control between statements, sequencing with non-arithmetic statements, Subprogram Control: Subprogram Sequence Control, Attributes of data control, parameters transmission.</td>
<td></td>
</tr>
<tr>
<td>V</td>
<td>Storage Management: Elements requiring storage, programs &amp; system controlled storage, static storage management, Distributed Processing :- Variations in subprogram control, Parallel programming.</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Text Book/s:
1. Programming Languages: Design and Implementation by Terrance W. Pratt, Marvin V. Zelkowitz & T.V. Gopal (Pearson Education)
V-Semester B. E. (Computer Technology)

Course Code: CT506
Title of the Course: Advanced Data Structure

<table>
<thead>
<tr>
<th>Course Scheme</th>
<th>Evaluation Scheme (Laboratory)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture</td>
<td>Tutorial</td>
</tr>
<tr>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

List of Practicals

- Practical no .1 & 2 should be based on Abstract data type
- Practical no .3 should be based on priority queue
- Practical no .4 should be based on dictionary
- Practical no .5 & 6 should be based on binary search tree and AVL tree
- Practical no .7 should be based on Search trees, red black tree, B-tree
- Practical no .8 should be based on Huffman tree
- Practical no .9 & 10 should be based on mergeable heaps
### V-Semester B. E. (Computer Technology)

#### Course Code: CT507
#### Title of the Course: Design and Analysis of Algorithms

<table>
<thead>
<tr>
<th>Lecture</th>
<th>Tutorial</th>
<th>Practical</th>
<th>Periods/week</th>
<th>Credits</th>
<th>TW</th>
<th>POE</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>03</td>
<td>01</td>
<td>02</td>
<td>25</td>
<td>25</td>
<td></td>
<td>50</td>
</tr>
</tbody>
</table>

#### Practicals
- Practical No. 1 & 2 should be based on Unit No. 1
- Practical No. 3 & 4 should be based on Unit No. 2
- Practical No. 5 & 6 should be based on Unit No. 3
- Practical No. 7 & 8 should be based on Unit No. 4
- Practical No. 9 & 10 should be based on Unit No. 5
V-Semester B. E. (Computer Technology)

Course Code: CT508
Title of the Course: Java Programming

<table>
<thead>
<tr>
<th>Course Scheme</th>
<th>Evaluation Scheme (Laboratory)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture</td>
<td>Tutorial</td>
</tr>
<tr>
<td>--</td>
<td>--</td>
</tr>
</tbody>
</table>

List of Practicals

Students are expected to perform 10 practicals from the given list

1. Programs illustrating overloading and overriding method in JAVA. (Use any application)
2. Programs illustrating the implementation of various forms of inheritance (Ex. Single, Hierarchical, Multilevel inheritance etc.).
3. Programs which illustrate the implementation of multiple inheritance using interfaces in JAVA.
4. Programs which illustrates the Implementation of Inheritance by Method overriding, Super constructor and super keyword, abstract class (Use any application)
5. Programs which illustrate the manipulation of strings: 1) Sorting an array of strings in ascending order. 2) Frequency count of words and characters in a text file. Etc.,
6. Programs for sorting and searching a list of elements.
7. Programs for addition and multiplication of matrices.
8. Programs to create packages in JAVA.
9. Programs to create multiple threads in JAVA.
10. Programs to write applets to draw the various shapes: a) Cylinder b) Cube c) Square inside a circle d) Circle inside a square e) Polygons etc.,
11. Create and manipulate labels, lists, text fields, text areas and panels.
12. Understand and handle mouse events and keyboard events.
13. Client/Server interaction with stream socket connections (Use NET packages).
14. Exception Handling for – (a) Divide by zero error (b) Null values (c) Data entry
15. Program to read the data from user and save it to two different files, display the contents and exchange the contents of those two files using IO package.
17. Program to scroll the banner using applet.
18. Design 8-digit calculator using AWT package and layout managers.
20. Design Database program for Employee details and implement INSERT, SELECT, DELETE, UPDATE queries using JDBC
21. Design concurrent server that will handle multiple clients using multithreading.
V-Semester B. E. (Computer Technology)

Course Code: CT509
Title of the Course: Programming-II

<table>
<thead>
<tr>
<th>Course Scheme</th>
<th>Evaluation Scheme (Laboratory)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecture</td>
<td>TW</td>
</tr>
<tr>
<td>Tutorial</td>
<td>01</td>
</tr>
<tr>
<td>Practical</td>
<td>03</td>
</tr>
<tr>
<td>Periods/week</td>
<td>02</td>
</tr>
<tr>
<td>Credits</td>
<td>25</td>
</tr>
</tbody>
</table>

Practicals

1st, 2nd and 3rd practical should be on program development using netbeans, eclipse and Jcreator.

4th practical should be on GUI designing using dreamweaver.

5th and 6th practical should be on developing java server pages.

7th practical should be on use of apache tomcat server.

8th and 9th practical should be on introduction to android application development.