IT 218: Data Structure and Algorithm with Java

Course Objectives

This course aims to provide a systematic introduction to data structures and algorithms for constructing efficient computer programs. The course emphasizes on data abstraction issues (through ADTs) in the program development process, and on efficient implementation of chosen data structures and algorithms. Laboratory work is essential in this course.

Course Description

The course contains Complexity Analysis, Linked Lists, Stacks and Queues, Recursion, Binary Trees, Multiway Trees, Graph, Sorting, Hashing.

Course Details

Unit 1: Complexity Analysis **LH 4**

Computational and Asymptotic Complexity. Big-O Notation. Properties of Big-O Notation Ω and Q. Possible Problems. Examples of Complexities. Finding Asymptotic Complexity: Examples. The Best, Average, and Worst Cases 66. Amortized Complexity 69. NP-Completeness 73.

Unit 2: Linked Lists **LH 5**

Singly Linked Lists: Insertion, Deletion, Search. Doubly Linked Lists: Circular Lists, Skip Lists, Self-Organizing Lists. Sparse Tables. Case Study: A Library.

Unit 3: Stacks and Queues **LH 4**

Stacks, Queues, Priority Queues. Case Study: Existing a Maze.

Unit 4: Recursion **LH 4**

Recursive Definitions. Method Calls and Recursion Implementation. Anatomy of a Recursive Call. Tail Recursion. Nontail Recursion. Indirect Recursion. Nested Recursion. Excessive Recursion. Backtracking.

Unit 5: Binary Trees **LH 9**

Trees, Binary Trees, and Binary Search Trees. Implementing Binary Trees. Searching a Binary Search Tree. Tree Traversal. Breadth-First Traversal. Depth-First Traversal. Insertion, Deletion, Deletion by Merging, Deletion by Copying. Balancing a Tree. The DSW Algorithm. AVL Trees. Self-Adjusting Trees. Self-Restructuring Trees, Splaying. Heaps: Heaps as Priority Queues, Organizing Arrays as Heaps, Polish Notation and Expression Trees. Operations on Expression Trees. Case Study: Computing Word Frequencies 280.

Unit 6: Multiway Trees **LH 5**

The Family of B-Trees. B-Trees, B*-Trees, B+-Trees. Case Study: Spell Checker

Unit 7: Graphs **LH 6**

Graph Representation. Graph Traversals, Shortest Paths, All-to-All Shortest Path Problem, Cycle Detection. Spanning Trees. Connectivity. Connectivity in Undirected Graphs, Connectivity in Directed Graphs. Topological Sort, Networks.

Unit 8: Sorting **LH 6**

Elementary Sorting AlgorithmsInsertion Sort, Selection Sort, Bubble Sort. Efficient Sorting Algorithms: Heap Sort, Quicksort, Mergesort, Radix Sort. Case Study: Adding Polynomials.

Unit 9: Hashing **LH 5**

Hash Functions: Division, Folding, Mid-Square Function, Extraction. Collision Resolution: Open Addressing, Chaining, Bucket Addressing, Deletion. Case Study: Hashing with Buckets.

Textbooks:

Drozdek Adam, Data Structures and Algorithms in Java, 3rd edition

Reference:

- ***Duncan A. Buell, Data Structures Using Java***
- ***Main Michael, Data Structures and Other Objects Using Java, Prentice Hall (4th edition),***
- ***Robert Lafore, Data Structures and Algorithms in Java, Sams Publishing;***
- ***Narasimha Karumanchi, Data Structures And Algorithms Made Easy In Java, CareerMonk Publications***

)