ADVANCED DATA STRUCTURES & ALGORITHM DESIGN
21:198:435 (3 credits)

COURSE DESCRIPTION:
Advanced topics in data structures and algorithms, including mathematical induction, analysis and complexity of algorithms, and algorithms involving sequences, sets, and graphs such as searching, sorting, order statistics, sequence comparisons, and graph traversals. Optional topics include geometric, algebraic, and numeric algorithms.

PREREQUISITE:
21:198:335 (Data Structures & Algorithm Design)

TEXTBOOK:

DEPARTMENT WEB SITE:  http://www.ncas.rutgers.edu/math

THIS COURSE COVERS THE FOLLOWING TOPICS:

- Brief Review on **Elementary data structures** (Stacks, Queues, Trees, Lists, Heaps)
- Balanced Binary Search Trees (AVL Trees, Splay Trees, Red-Black Trees)
- Asymptotic Growth of Functions and Recurrence relations.
- Maps, Hashing, Hash Functions and Tables

Data structures for searching: Prefix Trees, Skip Lists

Data structures for graphs: Overview of Graph Definitions, Graph Representations
  - Edge List structure, Adjacency List Structure, Adjacency Map structure, Adjacency Matrix structure

Greedy Algorithms:
  - Minimal Cost Spanning Tree, Shortest distance in Graphs
  - Greedy Algorithm for the Knapsack Problem

Dynamic Programming:
  - Polynomials and Matrices, Chained matrix multiplication
Sorting and Searching Algorithms:

- Review on Elementary sorting algorithms (Insertion, Bubble, Selection, Quick, Merge, Heap sort, Shell sort)
- Complexity of Sorting and Distribution-based sorting (Count Sort, Radix Sort, Bucket Sort)
- Data Compression: LZW compression, Huffman coding.

Graph algorithms:

- Tree Traversal Algorithms (Depth-first search, Breadth-first search)
- Shortest path Algorithms (Dijkstra and Floyd-Warshall)
- Minimum Spanning Trees (Prim’s, Kruskal’s)
- Transitive Closure
- Directed Acyclic Graphs, Topological Sorting

String search and pattern matching Algorithms:

- The Boyer-Moore Algorithm
- NP-Completeness

Programming:

- Using Java Collection classes, Inheritance, polymorphism, iterators, Object identity and equality, immutable and unique objects, Generic methods.