Course Syllabus **EECS 2510 – Nonlinear Data Structures**

Credits & Contact hours 3 credit hours & 3 50-minute lecture contact hours per week.

Coordinator Dr. Jackson Carvalho

Textbook Introduction to Algorithms, 3rd Ed, Thomas H. Cormen et al. 2009, MIT Press.

Course Information The data structures introduced in EECS 2500 are extended to include trees (binary, balanced, and n-ary), graphs, and advanced sorting techniques. In addition, the C++ language is used as the main vehicle and is introduced in the course. Students are expected to have a strong background in Java prior to this course.

Prerequisites: EECS 2500 and EECS 2520 (co-requisite)

Required course for CSE program

Specific Goals- Students Learning Objectives (SLOs)

The students will be able to

1. Use Microsoft's Visual Studio as an IDE.
2. Learn how to construct and apply a binary (and n-ary) tree to the storage, organization, and lookup of data using C++.
3. Learn how to implement special cases of binary trees, such as heaps, priority queues, and height-balanced trees, to solve problems.
4. Learn how to apply advanced sorting methods to data (Quicksort, Heapsort, Mergesort, and Radix Sort).
5. Learn the fundamentals of input and output using streams in C++.
6. Learn how to use pointers in C++ effectively.
7. Learn how to solve a limited class of recurrence relations.
8. Learn how to apply graphs and their related algorithms to solve practical problems.

Topics

1. Algorithm Analysis
   - i. Big- and Little-O, Omega, and Theta notation
   - ii. Divide-and-conquor approaches
   - iii. Recurrence Relations
2. C++ for the Java Programmer
3. Binary Search Trees (BSTs)

    i. Generic BST's
    ii. Height-Balanced BST's (AVL and Red-Black Trees)

4. Non-Search Binary Trees - Huffman Trees
5. Advanced Searching and Sorting
    i. Inversions vis a vis sorting
    ii. Quicksort
    iii. Heaps and Heapsort
    iv. Shell sort
    v. Sorting in linear time
    vi. Decision Trees as the basis for all O(n lg n) Sorts
    vii. Counting Sort
    viii. Radix Sort
    ix. Bucket Sort
    x. Stable vs. Unstable sorting algorithms
    xi. B-Trees
6. Medians and Nth Order statistics
7. Graphs and Elementary Graph Algorithms:
    i. Introduction to Graphs and Terminology
    ii. Vertices, Edges, Directed, Undirected Gaphs, Graph Representations
    iii. Breadth-First Search
    iv. Depth-First Search
    v. Topological Sorting (partial order planning)
    vi. Strongly-Connected Components of a Graph
    vii. Spanning Trees (Kruskal's and Prim's algorithms)
    viii. Shortest Paths (Traveling Salesman Problem)