THE AMERICAN UNIVERSITY IN CAIRO
100 YEARS | SCHOOL OF SCIENCES AND ENGINEERING
Department of Computer Science and Engineering

CSCE 2211
Applied Data Structures

Prof. Amr Goneid

Spring 2024

CSCE 2211
Applied Data Structures

Prof. Amr Goneid

Course Resources

Instructor: Prof. Amr Goneid
E-mail: goneid@aucegypt.edu
Office: Rm 2152 SSE

Text book
"Data Structures and Algorithm Analysis in C++", by Mark A. Weiss, 4th edition, Pearson, 2013, ISBN-13: 978-0132847377

Other supplemental materials: Course Slides

Course Website: **http://www1.aucegypt.edu/faculty/cse/goneid/csce2211**

Language: **C++**

Course Goals

- Demonstrate knowledge and understanding of Data Models, Data Abstraction and ADTs and their role in problem solving and S/W development.
- Choose the appropriate data structure for modeling a given problem.
- Design and implement various ADTs in a high level language (C++) using Object Oriented Concepts. Topics include but are not limited to: trees, binary search trees, dictionaries, self-balancing trees, B-trees, red black trees, heaps, priority queues, sets, and graphs.
- Compare alternative implementations of data structures with respect to performance.
- Demonstrate experience in the design of algorithms for solving problem that use the above data structures.
- Demonstrate knowledge of common applications for each data structure in the topic list.
- Practice basic algorithm analysis using complexity bounds (Big-Oh, Big-Theta and Big-Omega).

Course Contents

1. Data Modeling and ADTs **(2 Lecture)**
   a. Data Modeling
   b. A Classification of Abstract Structures
   c. Another Classification
   d. Special Data Structures
   e. Examples on Modeling
2. Revision of Linear Data Structures **(1 Lecture)**
   a. Tables, Lists, vectors
   b. Stacks and Queues
3. Introduction to the Analysis of Algorithms **(4 Lectures)**
   a. Algorithms
   b. Analysis of Algorithms
   c. Time Complexity
   d. Bounds and the Big-O
   e. Types of Complexities
   f. Rules for Big-O
   g. Examples of Algorithm Analysis
   h. Comparing Complexities
   i. Polynomial and Intractable Algorithms
4. Trees (General) **(1 Lectures)**
   a. General
   b. Binary Trees
   c. Tree Traversal
5. Dictionaries and Binary Search Trees **(3 Lectures)**
   a. The Dictionary Data Structure
   b. The Binary Search Tree (BST)
   c. Search, Insertion and Traversal of BST
   d. Removal of nodes from a BST
   e. Binary Search Tree ADT
   f. Tries (Prefix Trees), Suffix Trees
6. Self-Organizing Trees **(3 Lectures)**
   a. AVL Trees
   b. Red-Black Trees
   c. Splay Trees
7. B- Trees **(2 Lectures)**
   a. B-Trees as M-ary Search
   b. Insertion in and removal from B-Trees
8. Heaps and Priority Queues **(3 Lectures)**
   a. The Binary Heap
   b. Insertion and Removal

c. A Priority Queue Class
d. Analysis of PQ Operations
e. Heapify: A Modified Insertion Algorithm
f. Heap Sort
g. Fibonacci Heaps and Applications

9. The Set Data Structure: Disjoint Sets **(1 Lectures)**
a. Tree Representation
b. Basic Operations
c. Parent Array Representation
d. Simple Find and Simple Union
e. Disjoint Sets Class
f. Multisets
g. Applications

10. Graphs **(5 Lectures)**
a. Basic Definitions
b. Paths and Cycles
c. Connectivity
d. Other Properties
e. Representation
f. Examples of Graph Algorithms:
- Graph Traversal
- Shortest Paths
- Minimum Cost Spanning Trees
- Maximum Flow

Grading Criteria:

- Assignments: 30%
- Midterms: 20%
- Term Project: 20%
- Term Paper: 10%
- Final: 20%