CS 2302 Data Structures
Fall 2016

1. General Information

Instructor:

Diego Aguirre
Email: daguirre6@miners.utep.edu
Web: www.aguirrediego.com
Office hours: Tuesdays and Thursdays 12:10-13:30, or by appointment.
Office location: Lockheed Martin Storefront
Chat: diego4.aguirre@gmail.com

Teaching Assistants (TA):

Anjon Basak

Peer Leader:
TBA

Lectures

MW 12:00 – 13:20 in LART 323

Textbook

Data Structures and Algorithm Analysis in Java, by Mark Allen Weiss. Addison Wesley, Third Edition, 2011.

2. Objectives and Outcomes

This is the third and final course in the fundamental computer science sequence. Students will learn about fundamental data structures and analysis and design of algorithms.

Level 3: Synthesis and Evaluation:
Level 3 outcomes are those in which the student can apply the material in new situations. This is the highest level of mastery. On successful completion of this course, students will be able to

1. Given a problem, judge which data structures are required to solve it efficiently and justify the selection.
2. Given a non-recursive algorithm, examine its loop structure, assess its asymptotic running time, and express it using big-O notation.
3. Given a recursive algorithm, examine its structure, formulate and solve a recurrence equation defining its running time, and express it using big-O notation.
4. Design and implement solutions to computational problems based on iteration and recursion.
5. Trace the behavior of non-trivial methods and algorithms.

Level 2: Application and Analysis:
Level 2 outcomes are those in which the student can apply the material in familiar situations, e.g., can work a problem of familiar structure with minor changes in the details. Upon successful completion of this course, students will be able to:

1. Describe, implement, and use the following data structures:
    a. Heaps
    b. Hash tables
    c. Balanced trees
    d. Graphs
    e. Disjoint set forests
2. Describe, implement, and apply the following graph algorithms:
    a. Connected components
    b. Breadth-first search
    c. Depth-first search
    d. Topological sorting
    e. Minimum spanning trees (Kruskal's and Primm's)
    f. Single-source shortest paths
3. Trace the behavior of recursive programs using activation records.
4. Reason about the running times of algorithms in relation to the size of their inputs.

Level 1: Knowledge and Comprehension
Level 1 outcomes are those in which the student has been exposed to the terms and concepts at a basic level and can supply basic definitions. On successful completion of this course, students will be able to:

1. Explain the concept of polymorphism Identify and explain the following algorithm design techniques:
    a. Greedy algorithms
    b. Divide and conquer
    c. Dynamic programming
    d. Backtracking
2. Explain the concept of NP completeness.
3. Explain the utility of randomized algorithms.

3. Policies and Other Information

Prerequisites: Minimum "C" grade in CS2401 and MATH 2300.

Textbook: Reading and laboratory assignments will be drawn from Data Structures and Algorithm Analysis in Java by Mark Allen Weiss. You are required to obtain this book for use in this course.

Grading: Final grades will be based on a combination of lab projects, homework assignments, in-class attendance and performance, three partial exams, and a final exam. The approximate weights are as follows:

- 25% - Lab projects
- 10% - Homework assignments, in-class exercises, and quizzes
- 39% - Partial Exams (3 exams, 13% each)
- 26% - Final Comprehensive Exam

The nominal percentage-score-to-letter-grade conversion is as follows: