University of Texas at El Paso
Course Syllabus

COURSE DESCRIPTION

Dept., Number	CS2302	**Course Title**	Data Structures
Approval Date	September 2018	**Course Coordinator**	Olac Fuentes

CATALOG DESCRIPTION

Data Structures (3-3) Common Course Number: COSC 2318. Abstract data types, representation of data using sets, lists trees and graphs. Storage allocation and collection techniques.

TEXT BOOK

CS2302: Data Structures, by Zybooks, available at zybooks.zyante.com.

COURSE OUTCOMES

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. Upon successful completion of this course, students will be able to describe, at a high level:

1. 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.

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 Prim's)

f. Single-source shortest paths (Dijkstra's algorithm)

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 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. Upon successful completion of this course, students will be able to use the syntax and semantics of a higher-level language to express solutions to programming problems, including the correct use of:

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.

ABET STUDENT OUTCOMES

Course outcomes	Student Outcomes
1.2, 1.3, 3.2, 3.3, 3.4	1
1.1a-d, 1.2, 1.3, 2.1a-c, 2.2a-f, 2.4, 3.1, 3.4	2 (ABET 1)
2.1a-c, 2.2a-f, 3.4	3 (ABET 2)
None	4 (ABET 5)
None	5 (ABET 4)
None	6 (ABET 3)
None	7
None	8
None	9
2.2a-f, 2.4, 3.1, 3.2, 3.3, 3.4	10 (ABET 6)

PREREQUISITES BY TOPIC

(CS 2401 w/C or better) AND (MATH 2300 w/C or better)