CSCE 500 Design and Analysis of Algorithms
Spring 2021

Class Time: Tuesday, Thursday: 11:00 am-12:15 pm; taught by Dr. Miao Jin

Link to the Zoom lectures of this course: https://ullafayette.zoom.us/j/91779340197

Instructor's Email: miao.jin@louisiana.edu

Office Hours: Tuesday, Thursday: 9:00 am-11:00 am or by appointment

Moodle Page: http://moodle.louisiana.edu (follow the link for CSCE 500 in your list of courses)

Text Book: "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, 3rd Edition, The MIT Press, 2009.

Reference Material: These will be made available in Moodle by the instructor.

Course Goals:

This course provides comprehensive coverage of modern computer algorithms, aiming at in-depth treatment of algorithmic design and analysis with elementary explanation while keeping mathematical rigor.

Upon completion of this course, students will be able to do the following:

- Explain the major algorithms and data structures.
- Analyze the asymptotic performance of algorithms.
- Write rigorous correctness proofs for algorithms.
- Apply important algorithmic design paradigms and methods of analysis.
- Design efficient algorithms to model and solve common engineering problems.

Course Outline:

Module 1 **Topic(s):**
Introduction of Algorithmic Thinking, Sorting Algorithms, and Data Structures (AVL tree, Heap, and Hashing)
Objectives:
1. Explain the major sorting algorithms and their performance analyses.
2. Analyze worst-case running times of algorithms using asymptotic analysis.
3. Compare between different data structures. Pick an appropriate data structure for a design situation.
Instructional Materials:
Textbook: Chapters I, II, and III, "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 3rd Edition, The MIT Press, 2009.

Module 2 **Topic(s):**
Advanced design and analysis techniques: Dynamic Programming and Greedy Algorithms
Objectives: 1. Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. 2. Explain algorithms that employ the dynamic-programming paradigm and analyze their performance. 3. Design dynamic-programming algorithms, and analyze them. 4. Describe the greedy paradigm and explain when an algorithmic design situation calls for it. 5. Explain algorithms that employ greedy paradigm and analyze their performance. 6. Design greedy algorithms, and analyze them.
Instructional Materials: Textbook: Chapter IV, "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 3rd Edition, The MIT Press, 2009.

Module 3 **Topic(s):**
Graph Algorithms
Objectives: 1. Explain the major graph algorithms and analyze their performance. 2. Employ graphs to model engineering problems, when appropriate. 3. Design algorithms that employ graph computations as key components, and analyze them.
Instructional Materials and Tools: Textbook: Chapter VI, "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 3rd Edition, The MIT Press, 2009.

Module 4 **Topic(s):**
NP-Completeness and Approximation Algorithms.
Objectives:

1. Explain the basic concept of NP-completeness.
2. Explain what an approximation algorithm is, and the benefit of using approximation algorithms.

2. Explain representative approximation algorithms.

3. Analyze the approximation factor of an algorithm.

Instructional Materials and Tools:

Textbook: Chapter VII, "Introduction to Algorithms", Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 3rd Edition, The MIT Press, 2009.

Grading Policy:

Quiz 1	20%
Quiz 2	20%
Quiz 3	20%
Final exam (comprehensive)	35%
Participation in Class	5%