University of Arkansas – Fort Smith
5210 Grand Avenue
P.O. Box 3649
Fort Smith, AR 72913
479-788-7000

General Syllabus

CS 3103 Algorithm Design

Credit Hours: 3 Lecture Hours: 3 Laboratory Hours: 0

Prerequisite: CS 2003 Data Structures and MATH 1903 Introductory Discrete Mathematics

Effective Catalog: 2018- 2019

I. Course Information

A. Catalog Description

Examines the theoretical foundations and practical applications of algorithm analysis and design. Builds upon the data abstractions introduced in CS 2003, while introducing various algorithm strategies and techniques.

B. Additional Course Information

This course is used to satisfy the requirements for the Programming Concentration in the Information Technology degree. It may also be used as an upper level elective in all other concentrations of the IT degree.

II. Student Learning Outcomes

A. Subject Matter

Upon successful completion of this course, the student will be able to:

1. Evaluate and assess the time and space complexity of an algorithm.
2. Utilize graphing algorithms to efficiently solve applicable problems.
3. Implement and assess various programming algorithm techniques (brute force, greedy, divide and conquer, backtracking, branch and bound, heuristic and dynamic) to solve applicable problems.
4. Design and utilize hash tables for applicable problems.
5. Design and utilize the algorithm techniques for randomized algorithms.

B. University Learning Outcomes

This course enhances student abilities in the following areas:

Communication Skills (written and oral)
Students will apply programming documentation and technical explanations to concepts and code execution.

Analytical Skills
Critical Thinking Skills - Students will design algorithm requirements and properties to implement the appropriate programming code solution in the most efficient method.

III. Major Course Topics

A. Algorithm complexity
B. Algorithmic strategies
C. Sorting and searching
D. Graph algorithms
E. Advanced data structures
F. Heaps
G. Priority queues
H. Height-balanced binary search trees
I. Minimum spanning trees
J. Graphs
K. Algorithm design
L. Divide and Conquer
M. Dynamic Programming
N. Greedy
O. Backtracking
P. Basic computability