CSC 342 - ALGORITHM ANALYSIS

CREDIT HOURS: 3
PREREQUISITES: CSC 302 and 333
GRADE REMINDER: Must have a grade of C or better in each prerequisite course.

CATALOG DESCRIPTION

Study of algorithm design, analysis tools and techniques for selected problems, including sorting, searching, graphs, branch and bound strategies, dynamic programming, algebraic methods, string matching, and sets. An introduction to order notation, timing routines and complexity classes.

PURPOSE OF COURSE

The purpose of this course is to provide the student with tools and techniques for analyzing problem solutions. Complexity theory and computability issues are introduced. Evaluation of algorithms used in solving representative problems will be emphasized.

EDUCATIONAL OBJECTIVES

This course will provide students an opportunity to do the following:

1. To develop the concept of an algorithm, and thereby distinguish between solvable and unsolvable problems.

2. To present various complexity-levels of algorithms, and illustrate the concept with examples of algorithms that run in polynomial time as well as some that require exponential time.

3. To apply formal analysis techniques, based on algorithm time and space requirements, to algorithms involving iteration and recursion.

4. To develop the use of mathematical techniques, such as recurrence relations, as tools for analyzing the complexity of algorithms.

5. To study, implement, and analyze the performance of algorithms for sorting, generalized searching, string matching, pattern matching, and data compression.

6. To develop and implement branch-and-bound algorithms for solving selected NP-complete problems, and present efficient heuristic methods for finding sub-optimal but practical solutions to such problems.

7. To discuss emerging trends in algorithm developments, including parallel and distributed processing.

COURSE CALENDAR

This course meets for a minimum of 37.5 lecture contact hours during the semester, including the final exam. Students have significant weekly reading assignments. Students are expected to complete weekly homework/programming assignments, and 2-3 periodic exams in addition to the final exam. Students are expected to prepare for any class assignments or quizzes over the material covered in class or in the reading material. Successful completion of these activities requires at a minimum six additional hours of outside of classroom work each week.

CONTENT **Hours**

Introduction and Math Preliminaries 5
Notation, recursion, recurrence, induction
Order notation

Analysis tools and timing routines
Overview of complexity classes (P, NP, exponential)

Sorting and Searching Algorithms 5
Selection, insertion, exchange, special, binary, tree, hashing

String Algorithms 3
Matching (Knuth-Morris-Pratt, Boyer-Moore), parsing, compression

Graph Algorithms 9
Representation, connectivity, reachability, traversal, shortest path, minimum spanning tree, transitive closure, topological sort, Steiner trees, networks, depth first search, breadth first search.

Set Algorithms 3
Union-find, dictionary, Boolean matrices

Algebraic Methods 3
Polynomials, matrix operations, random numbers

Complexity Classes 4
Definition, classes, examples

Branch and Bound Algorithms 3
Backtracking, greedy methods, traveling salesman problem

Dynamic Programming 3
Optimal search trees, all pairs shortest path

Advanced Topics 4
Computation models - computability, decidability, finite state machines, grammars, pushdown automata
Iterative Refinement
Parallel and Distributed algorithms
Geometric algorithms
Heuristics (including genetic and neural techniques)
Probabilistic and approximation algorithms

Exams 3

TOTAL 45

REFERENCES

Aho, A. V., Hopcroft, J. E. and Ullman, J. D. The Design and Analysis of Computer Algorithms, Addison-Wesley, 1974.

Baase, S. Van Gelder, A., Computer Algorithms, 3rd. Ed., Addison-Wesley, 1999.

Cormen, T., Leiserson, C., Rivest, R., Stein, C., Introduction to Algorithms, 3rd. Ed., MIT Press/McGraw Hill, 2009.

Goodrich, M., Tamassia, R., Algorithm Design, Wiley, 2002.

Harel, D., Algorithmics, Addison-Wesley, 1987.

Kleinberg, J., Tardos, E., Algorithm Design, Addison Wesley, 2006.

Knuth, D. E., The Art of Computer Programming, Addison-Wesley, Vol I (2nd Ed.) 1973, Vol II (2nd Ed.) 1981, Vol III 1973.

Levitin, A., The Design & Analysis of Algorithms, 3rd. Ed., Addison-Wesley, 2012.

Manber, U., Introduction to Algorithms, Addison-Wesley, 1989.

Sedgewick, R., Wayne, K., Algorithms, 4th Ed., Addison-Wesley, 2011.

Wirth, N., Algorithms + Data Structures = Programs, Prentice-Hall, 1976.