CSE 4081 Introduction to the Analysis of Algorithms (3 credits)

Primary instructor: Debasis Mitra **Supporting instructor:** William Shoaff

Textbook and references:

Thomas H. Cormen, Charles E. Leiserson, Ronnald L. Rivest, and Clifford Stein, Introduction to Algorithms, McGraw Hill, 2001, ISBN 0-262-03293-7, (T)

M. A. Weiss, Data Structures & Algorithm Analysis in JAVA, Addison–Wesley, 1999, ISBN 0-201-35754-2, (R)

Class notes on web page and other materials distributed in the class, (R)

Course information:

2014–2015 Catalog description: CSE 4081 Introduction to the Analysis of Algorithms (3 credits). Covers time and space complexity of algorithms. Analyzes algorithms for sorting, searching, string processing and graph problems. Presents strategies such as divide-and-conquer, greedy, and dynamic programming as problem-solution techniques. Prerequisites: CSE 2010 or ECE 2552, ECE 3541.

Prerequisites by topic: Algorithmic paradigms, efficiency measures, rates of growth, and asymptotic behavior, graph theory, recursion, basic data structures, and discrete mathematics

Place in program:

Computer Science Program: One of CSE 4081 or CSE 4083 is required. One of these courses can be replaced by another advanced CSE elective.

Software Engineering Program: Advanced elective.

Course outcomes & related student outcomes: The student will be able to

1. Design and analyze algorithms. (1: Fundamental knowledge)
2. Find algorithmic solutions to computational problems. (2: Scieutific, computing, and engineering problem solving)
3. Design space-time efficient algorithms. (4c: Trade-offs in design choices)
4. Analyze through experimention algorithms they have programmed. (2: Scientific, computing, and engineering problem solving and 4a: Skillful software construction)
5. Skillfully present their work to their peers. (7: Communicate effectively)
6. Work on a small team to complete a project. (8: Effective teamwork)

Topics covered:

1. Divide and conquer (2 hours)
2. Dynamic programming (2 hours)

3. Greedy algorithms (1 hour)
4. Backtracking (2 hours)
5. Graph algorithms (3 hours)
6. Linear programming (3 hours)
7. Complexity theory (3 hours)
8. Guest lectures on advanced topic, quizzes, and tests (4 hours)
9. Student presentations, demonstation, and test discussions (6 hours)

Approved by: Debasis Mitra, Professor & William Shoaff, Associate Professor

Signature: Debasis Mitra **Date:** 2/3/15

Signature: Will Shoaff **Date:** 2/10/15