CS 4120 : DESIGN AND ANALYSIS OF ALGORITHMS

<table>
<thead>
<tr>
<th>Semester Hours:</th>
<th>3.0</th>
<th>Contact Hours:</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Coordinator:</td>
<td>Tianyi Song</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Text:</td>
<td>Introduction to Algorithms</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Author(s):</td>
<td>CORMEN, LEISERSON, RIVEST, AND STEIN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Year:</td>
<td>2009</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

SPECIFIC COURSE INFORMATION

Catalog Description:

Algorithms for solving problems that occur frequently in computer applications. Basic principles and techniques for designing and analyzing algorithms. Introduction to computational complexity, divide-and-conquer, dynamic programming, greedy approach, and graph algorithms. Prerequisites: MATH 2220 or MATH 3220 or equivalents and grade of C or better in CS 3350.

Course type: REQUIRED

SPECIFIC COURSE GOALS

- I can determine the complexity of an algorithm.
- I can explain and implement different types of algorithms (e.g., Divide-and-Conquer, Dynamic Programming, Greedy Algorithms).
- I can explain and implement different graph algorithms.
- I understand the classes of algorithms (P, NP, and NP-complete) and the role of polynomial-reduction in establishing NP-completeness.
- I understand the implications of algorithm design in real-world applications.

COMPUTER SCIENCE STUDENT OUTCOMES ADDRESSED BY THIS COURSE

- CS 1 Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions
- CS 6 Apply computer science theory and software development fundamentals to produce computing-based solutions

LIST OF TOPICS COVERED

- Introduction (1 week)
- Algorithmic Complexity (1 week)
- Divide-and-Conquer Strategy (2 weeks)
- Binary Search Trees (1 week)
- Dynamic Programming (3 weeks)
- Greedy Algorithms (1 week)
- Graph Algorithms (3 weeks)
- NP-Complete Problems (3 weeks)