**CS 5120 : DESIGN AND ANALYSIS OF ALGORITHMS**

<table>
<thead>
<tr>
<th><strong>Semester Hours:</strong></th>
<th>3.0</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Contact Hours:</strong></td>
<td>3</td>
</tr>
<tr>
<td><strong>Coordinator:</strong></td>
<td>Robert Dyer</td>
</tr>
<tr>
<td><strong>Text:</strong></td>
<td>Introduction to Algorithms</td>
</tr>
<tr>
<td><strong>Author(s):</strong></td>
<td>CORMEN, LEISERSON, RIVEST, AND STEIN</td>
</tr>
<tr>
<td><strong>Year:</strong></td>
<td>2009</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.
- I can analyze relevant research and communicate my findings

**STUDENT OUTCOMES ADDRESSED BY THIS COURSE**

- B.1 Analyze a given problem, and identify and define the computing requirements appropriate to its solution
- B.3 Apply mathematical foundations, algorithmic principles, and computer science theory as appropriate in modeling and solving real-world problems
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)