<table>
<thead>
<tr>
<th>Dept Number</th>
<th>CS 330</th>
<th>Course Title</th>
<th>Introduction to the Design and Analysis of Algorithms</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semester Hours</td>
<td>3</td>
<td>Course Coordinator</td>
<td>Qiang Cheng</td>
</tr>
</tbody>
</table>

**Catalog Description**

A detailed treatment of the design, analysis, and complexity of algorithms, including greedy algorithms, divide and conquer, dynamic programming, and limitations of algorithms as problems get larger or more complex.

### Textbooks


### References


### Course Learning Outcomes

- To understand the advance data structures in-depth.
- To learn the basic concepts of algorithm design.
- To learn how to determine complexity of algorithms.

### Assessment of the Contribution to Student Outcomes

<table>
<thead>
<tr>
<th>Outcome</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assessed</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Prerequisites by Topic

CS 220 with a grade of C or better.
## Major Topics Covered in the Course

1. Mathematical Foundation: formal treatment of analysis and design of algorithms, growth of functions, summations, recurrences, recursive vs. iterative algorithms, worst cast and average case analysis of algorithms, lower bounds {8 classes}

2. Trees: B-Trees and other balanced trees {8 classes}

3. Hashing: hash functions, collisions and resolutions {6 classes}

4. Heaps: implementations, applications, and variations {3 classes}

5. Sorting: variations of quick sort, merge sort, heap sort {4 classes}

6. Graph algorithms: DFS, BFS, topological sort, minimum spanning trees algorithm, and shortest path algorithm {3 classes}

7. Advanced algorithm design techniques: divide and conquer, greedy and backtracking {4 classes}

8. Introduction to parallel algorithms {4 classes}