COMPSCI 342: Data Structures and Algorithms

4-credit course with 4 lecture hours per week
Course Coordinator: Jyh-haw Yeh

Textbook(s) and Supplemental Material


Catalog Description

Basic data structures (continued from COMPSCI 225), introduction to design and analysis of algorithms, fundamental algorithms for sequences, sets, graphs and combinatorial problems, introduction to complexity of problems and to parallel and distributed algorithms. Examples are drawn from various areas of computer science.

PREREQ: COMPSCI 225, MATH 170, and MATH 187.

Required

Goals for the Course

Successful students will be expected to:

- apply mathematical skills to evaluate running time of algorithms
- choose appropriate data structures to implement algorithms
- use efficient algorithms to solve problems such as searching, sorting, and scheduling
- solve optimization problems using either dynamic programming or greedy algorithms
- apply fundamental graph algorithms such as depth-first search, breadth-first search to applications such as minimum spanning trees, and shortest paths
- use Subversion to develop team work

Outcomes Addressed

a. an ability to apply knowledge of computing and mathematics appropriate to the discipline
b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution
c. an ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
d. an ability to function effectively on teams to accomplish a common goal
h. recognition of the need for and an ability to engage in continuing professional development
i. an ability to use current techniques, skills, and tools necessary for computing practice
j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the trade-offs involved in design choices
Outcomes Assessed: a, b, c, d, i

Topics Covered

Introduction to Design and Analysis of Algorithms
Growth of Functions
Divide and Conquer
Sorting Algorithms and their Applications
Medians and Order Statistics
Hash Tables
Binary Search Trees
Balanced Search Trees: AVL and B Trees
Optimization Problems: Dynamic Programming
Optimization Problems: Greedy Algorithms
Graph Algorithms
NP-Completeness

Grading

A letter grade is assigned to each student at the end of the course based on the numerical scores of these activities:

<table>
<thead>
<tr>
<th>Activity</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Homeworks and Programming Assignments</td>
<td>40%</td>
</tr>
<tr>
<td>Midterm 1</td>
<td>15%</td>
</tr>
<tr>
<td>Midterm 2</td>
<td>15%</td>
</tr>
<tr>
<td>Final</td>
<td>30%</td>
</tr>
</tbody>
</table>

Curriculum Category Content (Credits)

<table>
<thead>
<tr>
<th>Area</th>
<th>Core</th>
<th>Advanced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithms</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Software Design</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Computer Architecture</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Data Structures</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Programming Languages</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>