Master Course Syllabus

School of Engineering and Computer Science
Washington State University Vancouver

CS 450

Design and Analysis of Algorithms

3 Semester Hours
(3 lecture hours)

Catalog Description

Analysis of data structures and algorithms; computational complexity and design of efficient data-handling procedures.

Prerequisite Courses

- CS 223 with a C or better
- STAT 360 with a C or better
- Certified major in Computer Science

Prerequisite Topics

- Elementary calculus and statistics
- Mathematical induction
- Fundamental data structures: hash tables, binary trees, linked lists, heaps and queues
- Proficiency in at least one programming language

Measured Course Outcomes

Students taking this course will:

1. Design new data structures and algorithms to meet particular performance requirements. (Contributes to performance criterion 6-b.)
2. Analyze asymptomatic complexity of algorithms and analyze the relationships of the complexity classes, P, NP, and NP Complete. (Contributes to performance criterion 6-a.)
3. Compare use of algorithms and data structures with regards to solution complexity. (Contributes to performance criterion 1-b.)

Required Textbooks

Introduction to Algorithms, Cormen, Leiserson, Rivest & Stein. McGraw-Hill.

Reference Material

Major Topics Covered in the Course

1. Formal definitions of big-O, big-Omega, big-Theta, little-o and little-omega notations
2. Bounding the Growth of Summations and Recurrences
3. Analysis of Sorting Algorithms
4. Analysis of Fundamental Data Structures
5. Design Techniques (e.g., dynamic programming, greedy, search, divide & conquer)
6. Complexity Classes (P, NP, NP-Complete)

Projects
Programming projects are to be developed by students individually.

Programming Project Area	Weeks
Advanced Algorithms	2

Students implement a self-balancing binary search tree called a scapegoat tree. It provides worst-case O(log n) lookup time, and O(log n) amortized insertion and deletion time. Their implementation is based on the explanation of Scapegoat, given in the original publication describing scapegoat trees, and also on Igal Galperin's thesis: On Consulting a Set of Experts and Searching.

Design, Implementation and Analysis
This course requires students to design and analyze both algorithms and data structures. Specifically, desired algorithmic characteristics, such as, efficient run-time are examined. Design may be at either the conceptual or implementation level. Analysis in this course centers on mathematical models of algorithms and data structures. Successful analysis is then used to inform potential solutions. Design and analysis tasks are examined in class; they also form the basis for homework exercises, exam questions and programming projects.

CS2013
This course provides coverage of CS2013 knowledge areas. Values listed are minimum course hours dedicated to the topic, percentages indicate the fraction of CS2013 knowledge area topics covered (acceptable values are: <25%, 25-75%, >75%, or 100%).

Area	Tier 1	Tier 2	Elective
AL/Basic Analysis	2 (100%)	3 (25-75%)	N
AL/Algorithmic Strategies	5 (100%)	2 (25-75%)	N
AL/Fundamental Data Structures and Algorithms	10 (>75%)	3 (>75%)	N
AL/Basic Automata, Computability and Complexity		2 (25-75%)	
AL/Advanced Computational Complexity			Y 2 (25%)
AL/Advanced Data Structures, Algorithms, and Analysis			Y 2 (25%)

Course Coordinator:	Scott Wallace
Last Updated:	October 14, 2020
Syllabus Version Number:	2.0