

**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 – Advanced Data Structures

CS 317 – Automata and Formal Languages

Stat 360 – Probability and Statistics

**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 (among other things):

1. Describe the meaning, use & limitations of asymptotic analysis and notation. (*Contributes to performance criterion A-2*)
2. Analyze asymptotic complexity of algorithms. (*Contributes to performance criterion J-1*)
3. Compares use of algorithms and data structures with regards to solution complexity. (*Contributes to performance criterion J-3*)
4. Design new data structures and algorithms to meet particular performance specifications. (*Contributes to performance criterion K-3*)
5. Describe the relations between complexity classes: P, NP, and NP-Complete. (*Contributes to performance criterion A-2*)

**Required Textbooks**

Introduction to Algorithms, Second Edition; Cormen, Leiserson, Rivest & Stein. McGraw-Hill. 2003.

**Reference Material**

None.

**Major Topics Covered in the Course**

1. Formal definitions of  $O$ ,  $\Omega$ ,  $\Theta$ ,  $o$  and  $\omega$  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)

**Laboratory Projects**

Programming Project Area	Weeks
Sorting & Order Statistics	1
Fundamental Data Structures	1
Dynamic Programming	1
Advanced Algorithms	1

**CSAB Category Content**

	FUNDAMENTAL	ADVANCED		FUNDAMENTAL	ADVANCED
Data Structures	0	1	Computer Organization and Architecture	0	0
Algorithm & Software Design	0	2	Concepts of Programming Languages	0	0

**Oral and Written Communications**

There are no significant oral or written communications required in this course. Virtually all assignments consist of writing computer programs.

**Social and Ethical Issues**

This course contains no significant coverage of social and ethical issues beyond the usual proscriptions against plagiarism and cheating.

**Theoretical Content**

Topic	Hours
Mathematical Analysis, Recurrences, Sums and Counting	6
Sorting	5
Data Structures / Algorithms	12
Design Techniques (Greedy, Divide & Conquer, etc)	7
Complexity Classes	2

**Problem Analysis**

Computational situations are examined in class, in assignments and in exams. Analysis in this course centers on mathematical models of algorithms and data structures. Successful analysis is then used to inform potential solutions.

**Solution Design**

This course requires students to design algorithms and data structures with desired run-time characteristics. Within this space, design may be at either the conceptual or implementation level. Design tasks are examined in class; they also form the basis for homework exercises, exam questions and programming projects.

**CC2001**

This course provides coverage of topics in the following areas (hours listed are minimums):

AL1. Basic algorithmic analysis [core]	11
AL2. Algorithmic strategies [core]	7
AL3. Fundamental computing algorithms [core]	12
AL6. The complexity classes P and NP [elective]	2
DS3. Proof techniques [core]	2
DS4. Basics of counting [core]	2
DS5. Graphs and trees [core]	3

---

Course Coordinator: Scott Wallace  
 Last Updated: April 08, 2009 (Approved)  
 Syllabus Version Number: 1.2