

CS380 Algorithm Design and Analysis

Spring 2015

Catalog Description

An introduction to the formal techniques that support the design and analysis of algorithms, focusing on both the underlying mathematical theory and the practical considerations of efficiency. Topics include asymptotic complexity bounds, techniques of analysis, algorithmic strategies, advanced data structures, graph theory and other selected topics.

Topics

- Asymptotic analysis of upper and average complexity bounds
- Identifying differences among best, average, and worst case behaviors
- Big O, little o, omega, and theta notation
- Standard complexity classes
- Empirical measurements of performance
- Time and space tradeoffs in algorithms
- Using recurrence relations to analyze recursive algorithms
- Brute-force algorithms
- Greedy algorithms
- Divide-and-conquer
- Backtracking
- Branch-and-bound
- Heuristics
- Pattern matching and string/text algorithms
- Numerical approximation algorithms
- Simple numerical algorithms
- Sequential and binary search algorithms
- Quadratic sorting algorithms (selection, insertion)
- $O(N \log N)$ sorting algorithms (Quicksort, heapsort, mergesort)
- Representations of graphs (adjacency list, adjacency matrix)
- Depth- and breadth-first traversals
- Shortest-path algorithms (Dijkstra's and Floyd's algorithms)
- Transitive closure (Floyd's algorithm)
- Minimum spanning tree (Prim's and Kruskal's algorithms)
- Object Oriented programming

The above topics were copied with permission from the recommendations found at:
<http://www.acm.org/education/curricula-recommendations> (2008 and 2001)

Instructor Details

Professor:	Shereen Khoja
Email:	shereen@pacificu.edu
Office:	Strain 203C
Phone:	(503) 352-2008
Office Hours:	Mon 1:00 – 2:00pm Wed 1:00 – 2:00pm Fri 10:30 - 11:30am or by appointment

Course Details

Course Title:	CS 380 Algorithm Design and Analysis
Prerequisites:	CS 300 Data Structures with a grade of C or better and MATH 240 Discrete Mathematics with a grade of C or better
Required for:	Computer Science Major
Meeting Times:	MWF 9:15am – 10:20am
Location:	Marsh LL12
Textbooks:	Introduction to Algorithms by Cormen, Leiserson, Rivest, and Stein
Software:	Microsoft Visual Studio 2010
Course Website:	http://zeus.cs.pacificu.edu/shereen/cs380sp15/

Course Assessment

Grade Distribution:

Assignments / Homework	40%
Unscheduled Quizzes (open-note)	5%
3 Exams	35%
Final Exam	20%

Programming Projects Grading:

Successful Execution	70%
Acceptable structure, style, documentation, and efficiency. You must follow the C++ Coding Standards, version 4.0	30%

Percent Breakdown:

			92-100%	A		90-92%	A-
88-90%	B+		82-88%	B		80-82%	B-
78-80%	C+		72-78%	C		70-72%	C-
68-70%	D+		60-68%	D			
			0-60%	F			

Important Dates

Tentative dates for Exams:

Exam 1: Friday, February 20, 2015

Exam 2: Monday, March 16, 2015

Exam 3: Friday, April 17, 2015

Spring Break:

Saturday, March 21, 2015 – Sunday, March 29, 2015 (No Class)

Senior Projects Day:

Wednesday, April 22, 2015 (No Class)

Reading Day:
Wednesday, May 6, 2015

Date of Final:
Monday, May 11, 2015, 3:00pm - 5:30pm

Academic Calendar:
<http://www.pacificu.edu/current-undergraduate/academics/academic-calendar>

Course Policies

1. **Attendance:** Attendance at every class is critical to your success in this course. I expect you to be on time and ready to go once it is 9:15am and that you stay until the end of class at 10:20am. You will not be allowed into the classroom once I close the door and start teaching. Any missed lecture is your responsibility to make up; just remember that if you fall behind, it will be very difficult for you to catch up.
 - I reserve the right to raise or lower your grade based on class participation and attendance. Specifically, I may lower your grade or may officially withdraw you from the course through the tenth week of the semester for poor attendance or participation. Further, your final grade may be lowered by 1/3 of your final course grade for each day (or portion thereof) of class missed. Please notify me PRIOR to class if you must miss class for any reason.
2. **Programming Assignments:** All assignments are to be programmed in C++ in Visual Studio. Both the electronic copy and hardcopy of your assignments are due at 9:15am on the day that they are due, unless specified otherwise.
 - The hardcopy must be placed on the instructor's desk before 9:15am on the day the assignment is due. If the hardcopy uses more than one sheet, then all sheets must be stapled. The code must be printed in color. Failure to submit a hardcopy of the assignment will result in a loss of 30% of the assignment points.
 - The electronic copy must be placed in your Subversion repository by 9:15am on the day the assignment is due. Failure to submit an electronic copy will result in a loss of 70% of the assignment points.
 - A program that does not successfully compile or produces no output loses 70% of the assignment grade.
 - Assignments can be turned in up to 24 hours late with a penalty of 15% of the grade. Anything turned in later than 24 hours after of the assignment deadline will NOT be accepted.
 - All code in any form generated from this course becomes the intellectual property of Pacific University. You may not share this code with anyone without obtaining written permission from Pacific University.
 - Neither computer failure, software failure, nor lack of computer access are accepted as excuses for late programs; therefore, start work on the programs as soon as they are assigned, and don't put them off until the last minute. Further, corruption of programs due to bad disk media is also not accepted as an excuse for late programs; therefore, always keep a current backup of all programs on a separate disk. Please note that the Computer Science departmental servers are not backed up.
3. **Homework Assignments:** All homework must be turned in by 9:15am on the day it is due.
4. No early or late exams/final will be given. No incompletes will be given.
5. **Academic Dishonesty:** Pacific University has no tolerance for academic dishonesty. It is university policy

that all acts of academic dishonesty be reported to the Associate Dean. Forms of academic dishonesty include, but are not limited to, plagiarism, fabrication, cheating, tampering with grades, forging signatures, and using electronic information resources in violation of acceptable use policies. Please consult the Academic Conduct Policies in the A&S Catalog for more details.

- For programming assignments, plagiarism takes the form of, *but is not limited to* copying code from someone else, whether copying files, glancing at someone else's code, typing from someone else's notes or typing while they dictate. The source can be a classmate, former student, website, program listing found in the trash, or anything else. Furthermore, plagiarism even on a small part of the program is cheating.
- Note that in this class, you will be translating pseudo code from the book directly into source code. Performing this translation yourself with no outside help does not constitute plagiarism. However, you must cite, in comments in your source code, where each function you implement has been translated from. Further, you must not be looking at implementations of these algorithms on the Web to use in your translation.
- You should also note that aiding someone else's cheating also constitutes cheating. You should never leave your code where someone else could have access to it, such as staying logged onto a machine or placing solutions in the recycling bin where another student may take it.
- Sanctions that may be imposed for academic dishonesty are:
 - First offense for cheating: zero on the assignment/exam and 12% subtracted from your final course grade.
 - Second offense for cheating of any kind: 'F' in the course

Learning Support Services for Students with Disabilities: If you have documented challenges that will impede your learning in any way, please contact our LSS office in Scott Hall (ext.2107). The Director will meet with students, review the documentation of their disabilities, and discuss the services that Pacific offers and any appropriate ADA accommodations for specific courses.