CMSC 641 Design & Analysis of Algorithms, Spring 2008

Course Syllabus

Updated March 2, 2008.

This syllabus has been revised to reflect some slippage in the schedule. The original syllabus is still available here.

We will follow the textbook *Introduction to Algorithms*, second edition, by Cormen, Leiserson, Rivest and Stein. The following schedule outlines the material to be covered during the semester and specifies the corresponding sections of the textbook. Selected topics not in the textbook will require reading from handouts.

				Homework	
Date	**Topic**	**Quizzes**	**Reading**	**Assign**	**Due**
Tue 01/29	Review: Greedy Algorithms		16.1-16.4		
Thu 01/31	Review: Dynamic Programming		15.1-15.5	HW1	
Tue 02/05	Review: Greedy vs Dynamic Programming				
Thu 02/07	Amortized Analysis		17.1-17.4	HW2	HW1
Tue 02/12	Binomial Heaps		19.1-19.2		
Thu 02/14	Fibonacci Heaps		20.1-20.4	HW3	HW2
Tue 02/19	Fibonacci Heaps	Quiz 1			
Thu 02/21	Maximum Flow			HW4	HW3
Tue 02/26	Maximum Flow				
Thu 02/28	Maximum Flow		handout	HW5	HW4

Tue 03/04	Maximum Flow	Quiz 2			
Thu 03/06	Linear Programming		29.1 – 29.2	HW6	HW5
Tue 03/11	Linear Programming		29.3		
Thu 03/13	NP-completeness		34.1-34.5	HW7	HW6
Tue 03/18	*Spring Break*				
Thu 03/20	*Spring Break*				
Tue 03/25	NP-completeness				
Thu 03/27	NP-completeness			HW8	HW7
Tue 04/01	Approximation Algorithms		35.1 – 35.5		
Thu 04/03	Approximation Algorithms			HW9	HW8
Tue 04/08	Approximation Algorithms	Quiz 3			
Thu 04/10	Randomized Algorithms		handout	HW10	HW9
Tue 04/15	Randomized Algorithms				
Thu 04/17	Randomized Algorithms			HW11	HW10
Tue 04/22	Sorting Networks	Quiz 4	27.1-27.5		
Thu 04/24	Sorting Networks			HW12	HW11
Tue 04/29	Parallel Merge Sort		handout		
	Computational Geometry		33.1-33.2	HW13	HW12

Thu 05/01					
Tue 05/06	Computational Geometry	Quiz 5	33.3-33.4		
Thu 05/08	Computational Geometry		handout		HW13
Tue 05/13	Review				
Thu 05/15	**Final Exam 1pm - 3pm**				

Last Modified: 16 Apr 2008 19:48:55 EDT by Richard Chang

to Spring 2008 CMSC 641 Homepage

CMSC 641 Design & Analysis of Algorithms, Spring 2008

Homework Assignments

Homework 1, Due Thursday 02/07

1. Exercise 16.2-3, page 384.
   Note 1: You must prove your algorithm is correct by showing that no other packing of the knapsack with total weight less than W yields greater value.
   Note 2: "Describe an algorithm" means that you should give an *English* description of your algorithm. Do not use pseudo-code (or real code).
   Note 3: You should state and briefly justify the running time of your algorithm.

2. Problem 15-1, page 364.
   Note: Include in the description of your dynamic programming algorithm the formulation of a recursive function that solves this problem.
   Hint: If p_1 and p_2 are the two leftmost points in the set, then any bitonic tour must have a segment from p_1 to p_2. This is a useful observation when you set up your recursive optimization function.

3. Problem 15-7, page 369.
   Note 1: Include in the description of your dynamic programming algorithm the formulation of a recursive function that solves this problem.
   Note 2: It is tempting to take a greedy approach to this problem similar to the greedy algorithm for Unit Job Scheduling. However, you *must* use dynamic programming for this problem.
   Hint: Concentrate on finding a schedule for those jobs that will be processed before their deadlines (the *early* jobs). Argue that the schedule for the early jobs can always be in order of increasing deadlines.

Homework 2, Due Thursday 02/14

1. Exercise 17.2-3, page 412.
2. Exercise 17.3-7, page 416.
3. Problem 17-3, page 427.

Homework 3, Due Thursday 02/21

1. Problem 17-2, page 426.

2. Exercise 21.2-3, page 505.
   Note: The question is asking you to use the accounting method or the potential method to prove an amortized running time of O(1) for MakeSet and FindSet and O(log n) for Union. The temptation is to make MakeSet O(log n) and Union O(1), but that is not correct.

3. Problem 21-1, page 518.

Homework 4, Due Thursday 02/28

1. Exercise 20.2-5, page 489.

2. Problem 20-1, parts a-d, page 496.
   Note: The potential function mentioned in part c is on page 479, Equation 20.1:

   $$\Phi = t(H) + 2\, m(H)$$

   where $t(H)$ is the number of trees in the root list of H and $m(H)$ is the number of marked nodes in H.

3. Problem 20-2, part a, page 497.

Homework 5, Due Thursday 03/06

1. Exercise 26.2-8, page 664.

2. Problem 26-1, page 692.

3. Problem 26-2, page 692.
   (Previously given page number was incorrect.)

Homework 6, Due Thursday 03/13

Note: [DPV] = *Algorithms* by Dasgupta, Papadimitriou and Vazirani. An online version is available here. The page numbers of the printed and online versions are not the same. The page numbers are given as printed/online.

1. Problem 26-3, page 693.

2. Exercise 7.16, "Salad", in Chapter 7 of [DPV], page 225/241.
   FYI: You can also use the "solver" module in Microsoft Excel which implements the simplex method. Look for "Solver" under "Tools". You should check "Assume Linear Model" under "Options".

3. Exercise 7.29 "Hollywood", in Chapter 7 of [DPV], page 230/245.

Homework 7, Due Thursday 03/27

1. Problem 34-3, parts a-f, pages 1019-1020.

Homework 8, Due Thursday 04/03

For this assignment, when you are asked to show that a set is NP-complete, you **must** do the following explicitly:

- Describe a reduction function *f()* from a known NP-complete set *A* to the new set *B*.
- Argue that *f()* is polynomial-time computable.
- Argue that for all *x*, *x* in *A* implies *f(x)* in B.
- Argue that for all *x*, *f(x)* in *B* implies *x* in A.

Arguments that the set is related to, similar to, a special case of, a generalization of, ... an NP-complete problem is not acceptable.

1. Exercise 34.2-3, pages 983.

2. Exercise 34.5-2, page 1017.

3. The Dominating Set decision problem is defined as follows:

   DS = $\{ (G, k) \mid G=(V, E)$ is an undirected graph such that $\exists\ V' \subseteq V$ with $|V'| \leq k$ such that $\forall\ u \in V - V'\ \exists\ v \in V'$ such that $(u, v) \in E.\}$

   Intuitively, if V' is a dominating set, then every vertex in G is either in V' or is adjacent to a vertex in V'. Show that Dominating Set is NP-complete.

Homework 9, Due Thursday 04/10

1. Problem 35-1, part a, page 1049-1050.

2. Problem 35-1, parts b-f, page 1049-1050.

3. Problem 35-2, parts a & b, page 1050.

Homework 10, Due Thursday 04/17

1. Exercise C.3-2, page 1110.

2. [Adapted from *Algorithm Design* by Kleinberg & Tardos.]
   An online auction system must maintain the highest bid seen so far. Call this value b^*. Suppose that there are n bidders who bid n distinct values $b_1, b_2, b_3, \dots b_n$. Furthermore, suppose that the ordering of the bidders is uniformly random (i.e., each of the n! permutations is equally likely).

What is the expected number of times that b^* gets updated?

3. [Adapted from *Algorithm Design* by Kleinberg & Tardos.]
A different auction system, the one-pass auction, requires the seller to accept or reject a bid as soon as it is made. This system requires the seller to make a decision without having seen all the bids. Once a bid has been accepted, the remaining bids are ignored. Similarly, once a bid has been rejected, the seller cannot change his/her mind and accept it later.

Suppose that there are n bidders and n is a value known to the seller beforehand. Furthermore, assume that each bid b_i is a distinct positive integer and that the ordering of the bidders is uniformly random (as in the previous question). Devise a randomized algorithm for the seller so that the seller has a 1/4 chance of accepting the highest bid.

Note: your algorithm cannot depend on future bids. The only information available to the algorithm in stage i is the value n and the bids $b_1, \ldots b_i$. Using this information, your algorithm must decide whether to accept or reject the i^{th} bid.

Homework 11, Due Thursday 04/24

1. [Adapted from *Algorithm Design* by Kleinberg & Tardos.]
Let $G = (V, E)$ be an undirected graph with n vertices and m edges. For $V' \subseteq V$, the subgraph G' induced by V' has V' as the set of vertices and $E' = \{ (u, v) \in E \mid u \in V' \text{ and } v \in V' \}$ as the set of edges.

Devise a randomized algorithm that selects k vertices from V so that the expected number of edges in the subgraph induced by the selected vertices is at least $m\, k\, (k - 1) / [n\, (n - 1)]$.

2. Exercise 27.3-3, page 716.
Assume that Exercise 27.3-6 is already completed and argue that your sorting network is correct when the input has just 0's and 1's.

Note: This question is harder than you might think. The difficulty is when there is an odd number of inputs. You cannot just use the Half-Cleaner in the textbook. That requires an even number of input lines. Also note that Figure 27.8 only shows half the cases, inputs of the form 1...10...01...1 are also considered bitonic.

Hint: Consider the input lines with odd index and the input lines with even index separately. Think recursively. How many more 0's can the odd inputs have compared to the even inputs? what about vice versa?

3. Exercise 27.3-4, page 716.

Homework 12, Due Thursday 05/01

1. Exercise 27.4-3, page 718.

2. Exercise 27.5-3, page 720.
3. Problem 27-2, parts a–d, page 721–722.

Homework 13, Due Thursday 05/08

1. Exercise 33.1-4, page 939.
2. Exercise 33.2-3, page 946.
3. Exercise 33.2-6, page 947.

Last Modified: 1 May 2008 08:28:19 EDT by Richard Chang
to Spring 2008 CMSC 641 Homepage

CMSC 641 Design & Analysis of Algorithms, Spring 2008

Contact Information

Instructor: Prof. Richard Chang

Office:	ITE 326 [map]
Office Hours:	Tuesday & Thursday, 10am - 11am
Telephone:	(410) 455-3093
E-mail:	chang@umbc.edu

Teaching Assistant: Suresh Purini

Office:	ITE 230
Office Hours:	Wednesday 3:30-4:30pm & Thursday 3:45-4:45pm
E-mail:	suresh1@umbc.edu

Last Modified: 5 Feb 2008 10:36:03 EST by Richard Chang

to Spring 2008 CMSC 641 Homepage