

Final Examination

CS 362 Data Structures and Algorithms
Fall, 2004

Name:
Email:

-
- Print your name and email, *neatly* in the space provided above; print your name at the upper right corner of *every* page. Please print legibly.
 - This is an *closed book* exam. You are permitted to use *only* two pages of “cheat sheets” that you have brought to the exam and a calculator. *Nothing else is permitted.*
 - Do all the problems in this booklet. *Show your work!* You will not get partial credit if we cannot figure out how you arrived at your answer.
 - Write your answers in the space provided for the corresponding problem. Let us know if you need more paper.
 - Don't spend too much time on any single problem. The questions are weighted equally. If you get stuck, move on to something else and come back later.
 - If any question is unclear, ask us for clarification.
-

Question	Points	Score	Grader
1	20		
2	20		
3	20		
4	20		
5	20		
6	20		
Total	120		

1. True or False

True or False: (circle one, 2 points each)

- (a) **True or False:** $4^{\log_2 n}$ is $o(n^2)$.
- (b) **True or False:** For some graph G , a BFS tree of G can be the same as a DFS tree of G .
- (c) **True or False:** The number of possible subsets of n elements is $O(n^2)$.
- (d) **True or False:** There is a greedy algorithm for 0–1 knapsack which runs in $O(n \log n)$ time.
- (e) **True or False:** If there is an activity with finish time earlier than all other activities, then the greedy algorithm for activity selection will always choose this activity.
- (f) **True or False:** Consider a graph $G = (V, E)$, with negative weight edges. The fastest time to solve the single source shortest paths problem on G is $O(|V|^3)$.
- (g) **True or False:** Consider a graph $G = (V, E)$, with negative weight edges. The fastest time to determine if there is a negative cycle in G is $O(|V||E|)$.
- (h) **True or False:** Consider a graph $G = (V, E)$, with negative weight edges. We can solve all pairs shortest paths on G in $O(|V|^3)$ time.
- (i) **True or False:** If there is a polynomial time algorithm for some problem in NP, then all problems in NP can be solved in polynomial time.
- (j) **True or False:** We can always edge-color a graph G with maximum degree Δ using no more than $3\lceil \Delta/2 \rceil$ colors.

2. Short Answer (4 points each) Where appropriate, circle your final answer.

(a) Solve the following recurrence using the master method: $T(n) = T(n/2) + n$

(b) Solve the following recurrence using the master method: $T(n) = 2T(n/4) + 1$

(c) Solve the following recurrence using annihilators: $T(n) = 3T(n - 1) - 2T(n - 2)$. Give the solution in general form i.e. do not solve for the constants

(d) In your own words, define what it means for a problem A to be in the class NP.

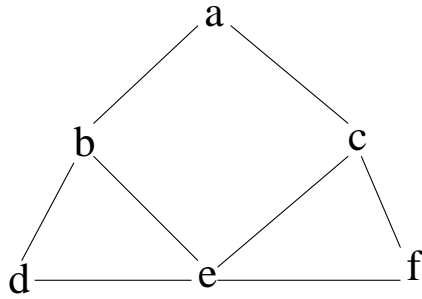
(e) In your own words, define what it means for a problem A to be NP-Hard.

3. Asymptotic Notation

Prove that $n^2 = \Omega(n \log n)$.

4. Spanning Trees

- (a) Give BFS and DFS trees for the following graph. Assume that BFS and DFS are initially called with the vertex a and that the edges are stored in the adjacency lists in alphabetical order. Make sure that you label which tree is a BFS tree and which is a DFS tree.



- (b) Assume you are given an undirected, connected graph G with n nodes and n edges. Give an algorithm to delete an edge in G which will not disconnect the graph.

5. Graph Theory

Your first assignment in your first job after college is wiring up the office computers. With fond memories of CS362, you decide to model the office as a weighted, undirected, connected graph $G = (V, E)$. Each computer in the office is a vertex in V and the distance between any pair of computers x and y is given by the weight of the edge (x, y) in E . A *wiring* is a subset of the edges in E that connects all the computers. The *cost* of the wiring is just the sum of the weights of all the edges in the wiring.

Your boss is both cheap and self-centered. He demands that you find a wiring with the properties that: 1) it costs no more than any other wiring and 2) his own computer, represented by the special node b in V , is connected to the network by the shortest possible edge. In other words, one of the lightest-weight edges incident to b in G must be in the wiring.

- (a) Give an algorithm which meets both demands of your boss. Show that your algorithm is correct.

- (b) Now imagine that you have two bosses with computers represented by the nodes b_1 and b_2 . They demand that you find a wiring with the properties that : 1) it costs no more than any other wiring and 2) each of them has their computer connected to the network with the shortest edge possible. Can you always meet both demands? If so, prove it. If not, give a weighted graph G and nodes b_1 and b_2 for which both demands can not be met.

6. Approximation Algorithms

Recall that a k -coloring of an undirected graph G is an assignment of one of k colors to each of the vertices so that no two adjacent vertices have the same color. In class we showed that the problem of determining whether a graph has a k -coloring is NP-Hard. In this problem, you will be designing an approximation algorithm for the optimization version of coloring.

Consider a graph G with maximum degree Δ . Give an algorithm to color G with $\Delta + 1$ colors. Hint: Use recursion (you can reduce to a smaller problem by removing a single vertex). Show that your algorithm is correct.