

Practice Final Exam for Spring 2012

These problems are four of the seven problems from the final exam given in spring 2011, seven out of ten true or false questions, along with the full instructions given with the exam so that you have a sense of what the final exam will be like.

- Do not open this exam booklet until you are directed to do so. Read all the instructions first.
- The exam contains 7 problems, several with multiple parts. You have 180 minutes to earn 180 points.
- This exam booklet contains 10 pages, including this one, and two sheets of scratch paper which can be detached.
- This exam is closed book. You may use two double sided Letter ($8\frac{1}{2}'' \times 11''$) or A4 crib sheets. No calculators or programmable devices are permitted. Cell phones must be put away.
- Write your solutions in the space provided. If you run out of space, continue your answer on the back of the same sheet and make a notation on the front of the sheet.
- Do not waste time deriving facts that we have studied. It is sufficient to cite known results.
- Do not spend too much time on any one problem. Generally, a problem's point value is an indication of how many minutes to spend on it.
- Show your work, as partial credit will be given. You will be graded not only on the correctness of your answer, but also on the clarity with which you express it. Please be neat.
- Good luck!

Problem	Title	Points	Parts	Grade	Initials
0	Name	1	10		
1	True or False	40	10		
2	Substitution Method	14	1		
3	Updating a Flow Network	25	3		
4	Candy Land	20	1		
5	Combined-Signal Problem	25	4		
6	Eleanor's Problem Revisited	25	4		
7	Majority Rules	30	5		
Total		180			

Name: _____

Problem 0. Name. [1 point] (10 parts)

Write your name on every page of this exam booklet! Don't forget the cover.

Problem 1. True or False. [40 points] (10 parts)

Circle **T** or **F** for each of the following statements to indicate whether the statement is true or false, respectively, and briefly explain why (one or two sentences). Your justification is worth more points than your true-or-false designation. *Careful*: Some problems are straightforward, but some are tricky!

- (a) **T F** Suppose that every operation on a data structure runs in $O(1)$ amortized time. Then the running time for performing a sequence of n operations on an initially empty data structure is $O(n)$ in the worst case.

- (b) **T F** Suppose that a Las Vegas algorithm has expected running time $\Theta(n)$ on inputs of size n . Then there may still be an input on which it always runs in time $\Omega(n \lg n)$.

- (c) **T F** If there is a randomized algorithm that solves a decision problem in time t and outputs the correct answer with probability 0.5, then there is a randomized algorithm for the problem that runs in time $\Theta(t)$ and outputs the correct answer with probability at least 0.99.
- (d) **T F** Let $\mathcal{H} : \{0, 1\}^n \rightarrow \{1, 2, \dots, k\}$ be a universal family of hash functions, and let $S \subseteq \{0, 1\}^n$ be a set of $|S| = k$ elements. For h chosen at random from \mathcal{H} , let E be the event that for all $y \in \{1, 2, \dots, k\}$, the number of elements in S hashed to y is at most 100, that is, $|h^{-1}(y) \cap S| \leq 100$. Then we have $\Pr \{E\} \geq 3/4$.
- (e) **T F** Let $\Sigma = \{a, b, c, \dots, z\}$ be a 26-letter alphabet, and let $s \in \Sigma^n$ and $p \in \Sigma^m$ be strings of length n and $m < n$ respectively. Then there is a $\Theta(n)$ -time algorithm to check whether p is a substring of s .

- (f) **T F** If an iteration of the Ford-Fulkerson algorithm on a network places flow 1 through an edge (u, v) , then in every later iteration, the flow through (u, v) is at least 1.
- (g) **T F** There exists a minimization problem such that (i) assuming $P \neq NP$, there is no polynomial-time 1-approximation algorithm for the problem; and (ii) for any constant $\epsilon > 0$, there is a polynomial-time $(1 + \epsilon)$ -approximation algorithm for the problem.

Problem 2. Substitution Method. [14 points]

Use the substitution method to show that the recurrence

$$T(n) = \sqrt{n}T(\sqrt{n}) + n$$

has solution $T(n) = O(n \lg \lg n)$.

Problem 4. Eleanor's Problem Revisited. [25 points] (4 parts)

Recall the following problem abstracted from the take-home quiz, which we shall refer to as *Eleanor's optimization problem*. Consider a directed graph $G = (V, E)$ in which every edge $e \in E$ has a length $\ell(e) \in \mathbb{Z}$ and a cost $c(e) \in \mathbb{Z}$. Given a source $s \in V$, a destination $t \in V$, and a cost x , find the shortest path in G from s to t whose total cost is at most x .

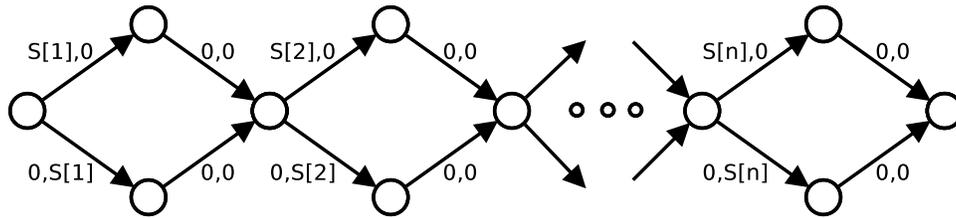
We can reformulate Eleanor's optimization problem as a decision problem. *Eleanor's decision problem* has the same inputs as Eleanor's optimization problem, as well as a distance d . The problem is to determine if there exists a path in G from s to t whose total cost is at most x and whose length is at most d .

- (a) Argue that Eleanor's decision problem has a polynomial-time algorithm if and only if Eleanor's optimization problem has a polynomial-time algorithm.

Recall the NP-complete PARTITION problem: Given an array $S[1..n]$ of natural numbers and a value $r \in \mathbb{N}$, determine whether there exists a set $A \subseteq \{1, 2, \dots, n\}$ of indices such that

$$\max \left\{ \sum_{i \in A} S[i], \sum_{i \notin A} S[i] \right\} \leq r .$$

(b) Prove that the Eleanor’s decision problem is NP-hard. (*Hint:* Consider the graph illustrated below, where the label of each edge e is “ $c(e), \ell(e)$ ”.)



(c) Argue on the basis of part (b) that Eleanor's decision problem is NP-complete.

(d) The solutions to the take-home quiz showed that there is an efficient $O(xE + xV \lg V)$ algorithm for Eleanor's optimization problem. Why doesn't this fact contradict the NP-completeness of Eleanor's decision problem?

SCRATCH PAPER

SCRATCH PAPER

MIT OpenCourseWare
<http://ocw.mit.edu>

6.046J / 18.410J Design and Analysis of Algorithms
Spring 2012

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.