

Illinois Institute of Technology  
Department of Computer Science

## Second Examination

CS 430 Introduction to Algorithms  
Spring, 2017

Wednesday, March 8, 2017  
10am–11:15am, 002 Herman Hall  
11:25am–12:40pm, 104 Rettaliata Engineering Center

Print your name and student ID, *neatly* in the space provided below; print your name at the upper right corner of *every* page. Please print legibly.

Name:
Student ID:

This is an *open book* exam. You are permitted to use the textbook, any class handouts, anything posted on the web page, any of your own assignments, and anything in your own handwriting. Foreign students may use a dictionary. *Nothing else is permitted:* No calculators, laptops, cell phones, Ipods, Ipads, etc.!

Do all four problems in this booklet. *All problems are equally weighted, so do not spend too much time on any one question.*

*Show your work!* You will not get partial credit if the grader cannot figure out how you arrived at your answer.

Question	Points	Score	Grader
1	25		
2	25		
3	25		
4	25		
Total	100		

**1. Dynamic Programming**

We want to move a checker from the top left corner to the bottom right corner of an  $n \times n$  checker board moving only right or down. Each square of the checkerboard contains a cost  $c_{ij}$ ; we want to find the *cheapest* route: the cost of a route is the sum of the costs of the squares entered, including the starting and the ending squares.

- (a) Give a recursive dynamic programming recurrence for the cheapest route.
- (b) Analyze, as a function of  $n$ , the time required to evaluate directly your recurrence in (a).
- (c) Give memoized code for (a).
- (d) Analyze, as a function of  $n$ , the time required by your code in (c).
- (e) Explain how to modify your memoized code so that you can recover the optimal path, not just compute its cost.

**1. Dynamic Programming, continued.**

**2. Greedy Heuristics**

Professor Reingold is planning to have  $n$  problems on Exam 3. There happen to be exactly  $n$  students in the course and each has discovered a different one of the exam problems. The students want to share their information by sending email, so that every student knows every problem. Assume that a student includes all the problems she/he knows at the time a message is sent and that email can go only to one recipient.

- (a) Give a greedy algorithm organizing the email communication, trying to minimize the total number of email messages for the students to fully share their knowledge.
- (b) Prove that your greedy algorithm results in the fewest messages possible, or give an example where it does not.

**3. Amortized Analysis**

We saw in class (February 27) and in CLRS3 (Chapter 17) that a  $k$ -bit binary counter can be incremented in  $O(1)$  amortized time (measured in bit changes). Suppose we also want an operation RESET which sets the counter to zero. Use the potential function method to show that these two operations can be done in  $O(1)$  amortized time.

**4. Union-Find**

We have students  $1, 2, \dots, n$  who need to be assigned to dormitories at a university that has an arbitrarily large number of dormitories. There are  $m$  “same dormitory” requests  $(s_1, t_1), (s_2, t_2), \dots, (s_m, t_m)$  meaning that students  $s_i$  and  $t_i$  must be assigned to the same dormitory. There are also  $k$  “different dormitory” constraints  $(u_1, v_1), (u_2, v_2), \dots, (u_k, v_k)$  meaning that students  $u_i$  and  $v_i$  must be assigned to different dormitories.

- (a) Give an algorithm using the union-find data structure of section 21.3 of CLRS3 to determine whether it is possible to assign students to dormitories so that all constraints are satisfied.
- (b) Analyze the total running time of your algorithm using the amortized analysis of Theorem 21.14 on page 581 of CLRS3.