CSE 202 Final Examination
March 2009

Maximum points: 50
Extra credit: 20 Points

Due March 19, 3:00 PM. Return the solutions at EBU3B 4246. No late submissions.

Instructions:

1. Although this exam is an open book exam, you are only allowed to consult standard textbooks or lecture notes that we used in the course. You cannot consult the WWW.
   (a) Introduction to Algorithms (Cormen, Leiserson, Rivest, Stein)
   (b) Algorithm Design by Kleinberg and Tardos
   (c) Algorithms by Dasgupta, Papadimitriou and Vazirani
2. This test is designed to give you an opportunity to demonstrate *your* understanding of principles of algorithm design, analysis and application. Consequently, you are required to work by yourself in answering the questions. Group work or consultation with others is not allowed. You can check with the instructor or the TA for any clarification.
3. Guidelines for writing a solution:
   (a) Write your solutions neatly. Proofread them before you submit.
   (b) Clearly state the main ideas and strategies used in developing the algorithm.
   (c) Write the algorithm using a high-level pseudo language providing important details.
   (d) If you are using a modified version of a well-known algorithm, show clearly what your modifications are.
   (e) If you are using a well-known algorithm without any modifications, it is enough to specify how you plan to apply the algorithm (for example, by specifying the input).
   (f) Provide a detailed proof of correctness.
   (g) Analyze the time complexity of your algorithm.
4. Well-written solutions will be rewarded. Characteristics of good solutions: correct solution, logical coherence, an appropriate balance between completeness and succinctness, and an orderly presentation of ideas. Excessively long solutions will be penalized.
5. There are five problems each carrying *ten* points towards regular credit and two problems each carrying *ten* points towards extra credit. In addition, those who scored no more than five points on the dynamic programming problem in the mid-term can attempt the extra credit problem for dynamic programming to score a maximum of six points.

Problems:

1. A mission-critical production system has n stages that have to be performed sequentially; stage i is performed by machine M_i. Each machine M_i has a probability r_i of functioning reliably and a probability $(1 - r_i)$ of failing (and the failures are independent). Therefore, if we implement each stage with the single machine, the probability that the whole system works is $r_1 \times r_2 \times \cdots \times r_n$. To improve this probability we add redundancy, by having m_i copies of the machine M_i so that stage i can be performed by m_i independent copies. The probability that all m_i copies fail simultaneously is only $(1 - r_i)^{m_i}$ the probability that the whole system works is $\prod_{i=1}^{n}(1-(1-r_i)^{m_i})$. Each machine has a nonnegative cost c_i, and there is a total budget B to buy machines.

   Given the probabilities $r_1, \cdots, r_n$, the costs $c_1, \cdots, c_n$, and the budget B, find the redundancies $m_1, \cdots, m_n$ that are within the available budget and that maximize the probability that the system works correctly.

   You can assume the costs c_i and the budget B are integers.

2. A hotel manager must make reservations for the bridal suite for the coming n days. The hotel has received a variety of reservation requests (a total of m requests) for various combinations of arrival and departure days. Assume that the arrival date is always strictly less than the departure date. Note that as a party departs on a day, another party can arrive on the same day. Each reservation would earn a different amount of revenue (always non-negative) for the hotel due to a variety of rates and requirements. Design an efficient algorithm to schedule the bridal suite with maximum profit for the hotel.

3. Consider the problem of selecting sites for an electronic message transmission system. Any number of sites can be chosen from a finite set of potential locations. We know the cost c_i of establishing site i and the revenue r_{ij} generated between sites i and j, if they are both selected. Find an efficient algorithm to determine the subset of vertices such that the sum of the edge revenues less the vertex costs is as large as possible.

4. Problem 19(b), Chapter 7, Page 425 (Kleinberg and Tardos)

5. Problem 3, Chapter 11, Page 652 (Kleinberg and Tardos)

Extra Credit Problems:

1. Problem 8, Chapter 11, Page 656 (Kleinberg and Tardos)

2. Suppose you have one machine and a set of n jobs $a_1, a_2, \cdots, a_n$ to process on that machine. Each job a_j has a processing time t_j, a profit p_j, and a deadline d_j. The machine can process only one job at a time, and job a_j must run uninterruptedly for t_j consecutive time units. If job a_j is completed by its deadline d_j, you receive a profit p_j, but if it is completed after its deadline, you receive a profit of 0. Give an efficient algorithm to find a schedule that obtains the maximum amount of profit, assuming that all processing times are integers between 1 and n. Your algorithm must in time polynomial in n.