

**Final examination**  
CSC 301, “Algorithms”

1. Suppose that we have a sequence of  $n$  cylindrical rods of uniform diameter—say, one centimeter—but varying in length from, say, half a meter up to three meters. We have to weld these rods together, end to end, to form one long rod, preserving their sequence in the completed product. But the difficulty of welding two rods increases with their length and also with the difference between their lengths: The “cost”  $c(a, b)$  of welding together rods of lengths  $a$  and  $b$ , where  $a \leq b$ , is  $a + b + b^2/a$ .

So, for instance, if  $n = 4$  and the lengths of the successive components are 52, 129, 61, and 228 centimeters, then the cost of welding the first two components together is  $c(52, 129) \approx 501$ ; the cost of welding the other two components together is  $c(61, 228) \approx 1141$ ; and the cost of welding together the resulting two-component rods, which have lengths 181 and 289,  $c(181, 289) \approx 931$ , for a total cost of approximately 2572. In this case, however, it would have been better just to do the welds in order from left to right:  $c(52, 129) + c(61, 181) + c(228, 242) \approx 2007$ .

Suggest an algorithm for determining the least costly order in which to perform the  $n - 1$  welds, given the lengths of the  $n$  component rods, and describe the order of its running-time function in terms of  $n$ .

2. For any natural number  $n$  and any positive integer  $k$ , a  $k$ -way *distribution* of  $n$  is a list of  $k$  natural numbers whose sum is  $n$ . For instance, there are fifteen 3-way distributions of 4; in lexicographic order, they are:

(0 0 4)  
(0 1 3)  
(0 2 2)  
(0 3 1)  
(0 4 0)  
(1 0 3)  
(1 1 2)  
(1 2 1)  
(1 3 0)  
(2 0 2)  
(2 1 1)  
(2 2 0)  
(3 0 1)  
(3 1 0)  
(4 0 0)

The *distribution rank* of a list of natural numbers is its (zero-based) position in the lexicographic order of  $k$ -way distributions of  $n$ , where  $k$  is the length of the list and  $n$  is the sum of its elements. Propose an algorithm for computing

the distribution rank of a given list of natural numbers and describe the order of its running-time function in terms of  $n$  and  $k$ .

3. In *currency arbitrage*, a knowledgeable and quick trading agent tries to turn a profit by exploiting small, transient inconsistencies in exchange rates. For example, one might look for the opportunity to convert US dollars into euros at the rate of 0.79 euros for each dollar, euros into Japanese yen at the rate of 116 yen for each euro, and Japanese yen into US dollars at the rate of 0.011 dollars for each yen. Sending one thousand US dollars through this sequence of exchanges yields  $1000 \cdot 0.79 \cdot 116 \cdot 0.011 = 1008.04$  dollars at the end of the process.

Devise an algorithm that seeks out optimal opportunities for currency arbitrage, given an  $n \times n$  matrix  $R$  of exchange rates among  $n$  different currencies. Assume that the entries in the matrix satisfy the condition  $R_{ij} = 1/R_{ji}$  for all  $1 \leq i, j \leq n$ , so that it is impossible to profit just by converting some amount into a different currency and immediately back again. Describe the order of the running-time function for your algorithm in terms of  $n$ .

4. Suppose that two algorithms (one using a brute-force approach, the other using a divide-and-conquer strategy) are proposed for the same problem, yielding identical solutions in all instances. The brute-force approach runs in  $\Theta(n^3)$  time. The divide-and-conquer approach solves trivial instances of the problem in  $O(1)$  time, but divides any non-trivial instance of size  $n$  into  $k$  subproblems of size  $\lfloor n/4 \rfloor$ , taking  $\Theta(n)$  time to effect the division and to post-process the results. For which values of  $k$  is the divide-and-conquer approach asymptotically faster? Prove your answer.

5. Suppose that we want to determine whether, in an emergency situation, it is possible to evacuate the residents of a community from their homes to existing emergency shelters using existing systems of transportation. We can model the problem with a weighted directed graph  $G = (V, E)$  in which the vertices are the locations of the residents' homes and the locations of the shelters, along with intersections and transfer points in the transportation system and the edges are transportation routes or segments of routes. We'll also need to specify the subsets  $H$  (homes) and  $S$  (shelters) of  $V$ , which we'll assume to be disjoint. Finally, we'll need a function  $c$  that associates with each edge a capacity indicating the maximum number of residents that can use that route in an emergency, and a function  $t$  that associates with each home the number of persons that live there and with each shelter the number of persons it can accommodate.

Note that time plays no role in this model; we assume that a resident "reserves" or "occupies" the entire path from his home to a shelter throughout the evacuation. We also assume that there is no congestion at intersections and transfer points—any number of residents can pass through any such vertex, up to the sum of the capacities of its out-edges.

Suggest an algorithm for determining whether all the residents can be

successfully moved to shelters and, if so, which path each should follow.

**6.** A *Langford triple* is a list of  $3n$  integers such that, for each integer  $k$  in the range from 1 to  $n$ ,  $k$  appears at exactly three positions in the list, the positions of the first and second appearances of  $k$  differ by  $k + 1$ , and the positions of the second and third appearances of  $k$  also differ by  $k + 1$ . So, for instance, (1 8 1 9 1 5 2 6 7 2 8 5 2 9 6 4 7 5 3 8 4 6 3 9 7 4 3) is a Langford triple, with  $n = 9$ , and so is the reversal of that list. Are there any other Langford triples with  $n = 9$ ?