

Due Wednesday, October 10, 2001.

1. There are n trading posts along a river. At any of the posts you can rent a canoe to be returned at any other post downstream. (It is next to impossible to paddle against the current.) For each possible departure point i and each possible arrival point j the cost of a rental from i to j is known. However, it can happen that the cost of renting from i to j is higher than the total cost of a series of shorter rentals. In this case you can return the first canoe at some post k between i and j and continue your journey in a second canoe. There is no extra charge for changing canoes in this way.

Give an efficient algorithm to determine the minimum cost of a trip by canoe from each possible departure point i to each possible arrival point j . In terms of n , how much time is needed by your algorithm?

2. The *Euclidean Traveling-Salesperson Problem* is the problem of determining the shortest closed tour that connects a given set of n points in the plane. The general problem is NP-complete, and its solution is therefore believed to require more than polynomial time (see Chapter 36 in your textbook).

Jon L. Bentley has suggested that we simplify the problem the problem by restricting our attention to *bitonic tours*, that is, tours that start at the leftmost point, go strictly left to right to the rightmost point, and then go strictly right to left back to the starting point. In this case, a polynomial-time algorithm is possible.

Describe an $O(n^2)$ -time algorithm for determining an optimal bitonic tour. You may assume that no two points have the same x -coordinate. (*Hint*: Scan left to right, maintaining optimal possibilities for the two parts of the tour.)

3. A magnetic tape holds n programs of lengths l_1, l_2, \dots, l_n . We know how often each program is used: a fraction p_i of requests to load a program concern program i , $1 \leq i \leq n$. (This of course implies that $\sum_{i=1}^n p_i = 1$.) Information is recored along the tape at constant density, and the speed of the tape drive is also constant. Each time a program has been loaded, the tape is rewound to the beginning.

If the programs are held in the order i_1, i_2, \dots, i_n , the average time required to load a program is $\bar{T} = c \sum_{j=1}^n \left[p_{i_j} \sum_{k=1}^j l_{i_k} \right]$, where the constant c depends on the recording density and the speed of the drive. We want to minimize \bar{T} .

- (a) Prove by giving an explicit example that it is not necessarily optimal to hold the programs in order of increasing values of l_i .
 - (b) Prove by giving an explicit example that it is not necessarily optimal to hold the programs in order of decreasing values of p_i .
 - (c) Prove that \bar{T} is minimized if the programs are held in order of decreasing $\frac{p_i}{l_i}$.
4. Read Cormen, Leiserson, Rivest, and Stein, Section 16.3 **Huffman Codes**, and do exercises
- 16.3-1
 - 16.3-2
 - 16.3-3
 - 16.3-4