1. **Canoe Rental Problem.** There are \( n \) trading posts numbered 1 to \( n \) as you travel downstream. At any trading post \( i \) you can rent a canoe to be returned at any of the downstream trading posts \( j > i \). You are given an array \( R[i, j] \) which gives the cost of a canoe which is picked up at post \( i \) and dropped off at post \( j \), for \( 1 \leq i < j \leq n \). Assume that \( R[i, i] = 0 \) and that you can’t take a canoe upriver (so perhaps \( R[i, j] = \infty \) when \( i > j \)). Your problem is to determine a sequence of rentals which start at post 1 and end at post \( n \), and which has a minimum total cost. As usual there are really two problems: determine the cost of a cheapest sequence, and determine the sequence itself.

Design a dynamic programming algorithm for both problems. What are the dimensions of your table and what are its entries? How is it initialized and what is the recurrence which determines table entries? Why is the principle of optimality satisfied in this problem? Write algorithms which fill the table given the cost array \( R \), and which determine an optimal sequence given the filled table. Determine the asymptotic run time of your algorithms.

2. **Activity Scheduling Problem:** Consider \( n \) activities with fixed start times \( s_1, \ldots, s_n \) and fixed finish times \( f_1, \ldots, f_n \), which must use the same resource (such as lectures in a lecture hall, or jobs on a machine.) At any time only one activity can be scheduled. Two activities are incompatible if they overlap. Your objective is to schedule activities so as to maximize the number of activities which can be completed, while respecting the compatibility constraint. In other words determine a set of mutually compatible activities of maximum size.

Consider the following greedy strategies:

a. Order the activities in increasing order of total duration. Schedule the activities with the shortest duration first, satisfying the compatibility constraint. If there is a tie, choose the one which starts first.

b. Order the activities in increasing order of start time. Schedule the activities with the earliest start times first, subject to the compatibility constraint. If there is a tie, choose the one which is of shortest duration.

c. Order the activities in increasing order of finish times. Schedule the activities with the earliest finish times first, subject to the compatibility constraint. If there is a tie, pick one arbitrarily.

Which, if any, of these strategies provide a correct solution to all instances of the problem? If your answer is yes, state and prove a theorem which establishes the correctness of the proposed strategy. If your answer is no, provide a counterexample (i.e. specific start and end times) showing that the strategy can fail, and compute the solution given by the proposed strategy as well as the true optimal solution.