22C:31 Homework 5

Due in class on Thursday, Nov 19th

---

1. Last Thursday (10-29), I almost completed describing the dynamic programming solution to the *matrix-chain multiplication* problem. The one piece that was left was writing the actual code. Recall that the problem is given as input a sequence of positive integers $p_1, p_2, \ldots, p_{n+1}$. These integers correspond to dimensions of matrices; specifically, matrix M_i has dimensions $p_i \times p_{i+1}$. Assuming that cost of multiplying a matrix A of dimensions $p \times q$ and a matrix B of dimensions $q \times r$ is $p \cdot q \cdot r$, the problem is to find a parenthesization of the matrix chain $M_1, M_2, \ldots, M_n$ that yields lowest cost if the matrices are multiplied according to this parenthesization.

   The recurrence we derived for this problem was

   $$OPT(i,j) = \min_{i \le k < j} OPT(i,k) + OPT(k+1,j) + p_i \cdot p_{k+1} \cdot p_{j+1}.$$

   I have deliberately avoided writing base cases.

   (a) Write pseudocode for the dynamic programming solution to the matrix chain multiplication problem.

   (b) Enhance this pseudocode to make it not just return the cost of an optimal parenthesization, but one optimal parenthesization.

   (c) Suppose that $p_1 = 8$, $p_2 = 72$, $p_3 = 50$, $p_4 = 100$, $p_5 = 700$, and $p_6 = 40$. Show how the 2-dimensional table gets filled by your pseudocode and the specific values that get placed in each slot. Using this table, find an optimal parenthesization of the matrix chain $M_1 M_2 M_3 M_4 M_5$ and report the cost of such a parenthesization. You can either do this by hand or if that is too tedious, you could turn your pseudocode into a simple program and execute it.

2. Problem 9, Pages 320-321.

3. Problem 10, Pages 321-323.

4. Problem 14, Pages 324-325.

5. Problem 24, Pages 331-332.

6. Problem 28, Page 334.
