CSE 101 Homework 2
Dynamic Programming and Greedy Algorithms
Due November 17, 1998 at the beginning of lecture
100 points total

Note that "Exercises" and "Problems" are numbered separately in this textbook.

1. (5 pts.) Exercise 16.1-1. Note that the "sequence of dimensions" indicates that the matrices are (in order) 5x10, 10x3, 3x12, 12x5, 5x50, and 50x6. Follow the execution of MATRIX-CHAIN-ORDER as in Figure 16.1 in the textbook, showing the resulting m and s tables, and then give the optimal parenthesization.
2. (25 pts.) Problem 16-3, except that you do not need to analyze the space requirements of your algorithm. Prove that the problem has an optimal substructure property (see the proof of Theorem 16.1 in the textbook for clues as to how to do this); your algorithm should exploit this property.

   (Algorithms similar to this are used in the UNIX `diff` utility, and are also used in computational biology to measure similarity between DNA sequences.)
3. (15 pts.) Exercise 17.1-3.
4. (25 pts.) Problem 17-1 (a) and (c). In part (a), the running time of your algorithm should be $O(1)$ if we assume that the arithmetic operations used each take only constant time.
5. (15 pts.) Exercise 26.1-8. No correctness proof or running time analysis necessary - but your solution shouldn't slow down FASTER-ALL-PAIRS-SHORTEST-PATHS too much.
6. (15 pts.) Exercise 24.2-6. No correctness proof necessary, but give a brief explanation of how your algorithm works, and give an upper bound on the running time - make your algorithm as efficient as you can.