**CS430 Lecture 13 Activities**

Opening Questions

1. Briefly explain what two properties a problem must have so that a dynamic programming algorithm approach will work.

2. Previously we have learned that divide-and-conquer algorithms partition a problem into independent sub-problems, solve each sub-problem recursively, and then combine their solutions to solve the original problem. Briefly, how are dynamic programming algorithms similar and how are they different from divide-and-conquer algorithms?

3. Why does it matter how we parenthesize a chain of matrix multiplications? We get the right answer any way we associate the matrices for multiplication. i.e. If A, B and C are matrices of correct dimensions for multiplication

(A B) C = A (B C)

Dynamic Programming

Dynamic Programming Steps

1. Define structure of optimal solution, including what are the largest sub-problems.
2. Recursively define optimal solution
3. Compute solution using table bottom up
4. Construct Optimal solution

Optimal Matrix Chain Multiplication (optimal parenthesization)

1. How many ways are there to parenthesize (two at a time multiplication) 4 matrices A\*B\*C\*D?

2. Step 1: Generically define the structure of the optimal solution to the Matrix Chain Multiplication problem.

The optimal way to multiple n matrices A1 through An is:

3. Step 2: Recursively define the optimal solution. Assume P(1,n) is the optimal cost answer. Make sure you include the base case.

4. Use proof by contradiction to show that Matrix Chain Multiplication problem has optimal substructure, i.e. the optimal answer to problem must contain optimal answers to sub-problems.

5. Step 3: Compute solution using a table bottom up for the Matrix Chain Multiplication problem. Use your answer to question 3 above. Note the overlapping sub-problems as you go. Step 4: Construct Optimal solution

A B C D

2\*4 4\*6 6\*3 3\*7