

CS2223
HW#6

DUE: Friday, December 10

1. (8 points) Suppose you want to compute the product $A_1 * A_2 * \dots * A_n$, where each A_i , $1 \leq i \leq n$, is a $d_{i-1} \times d_i$ matrix. Consider the following greedy algorithm to compute the product:

repeat

Choose the largest remaining d_i (among the internal dimensions) and multiply the two adjacent matrices that share that dimension

until only 1 matrix remains

So, for $d_0 \times d_1 \times d_2 \times d_3 \times d_4 = 10 \times 20 \times 50 \times 1 \times 100$, the algorithm produces the optimal parenthesization $(A_1 * (A_2 * A_3)) * A_4$. Either prove that this greedy algorithm always produces an optimal parenthesization or provide an example for which it fails.

2. (10 points) Do **Problem 22.2**, parts *e* and *f* on pages 558→559 of our text.

CS2223
HW#6 SOLUTIONS

1. For $d_0 \times d_1 \times d_2 \times d_3 \times d_4 \times d_5 \times d_6 = 1 \times 40 \times 50 \times 60 \times 30 \times 1 \times 1$, the algorithm yields

$(A_1 * ((A_2 * (A_3 * A_4)) * A_5)) * A_6$, with a cost of 16,231, whereas the optimal

$(A_1 * (A_2 * (A_3 * (A_4 * A_5)))) * A_6$ has a cost of 6841, so the algorithm is not guaranteed to provide an optimal answer.

2. *e* If an edge lies on a simple cycle, then its removal can't disconnect the component to which it belongs, and it can't be a bridge. If an edge, say (u,v) , is a bridge, then its removal disconnects the component to which it belongs. Since there does not exist a path between u and v which does not contain edge (u,v) , the edge can not belong to a cycle.

f An edge (u,v) is a bridge if and only if each of u and v is an articulation point or has degree 1 (it has at most one edge incident with it). To solve the problem, we do a depth first search of the graph in $\Theta(|E|)$ time to identify the articulation points, and then in $\Theta(|E|)$ we examine every edge. If each endpoint of an edge is an articulation point or has degree 1, then it is a bridge, otherwise it isn't.