Dynamic Programming: Matrix Chain Multiplication

Description In this assignment you are asked to implement a dynamic programming algorithm: matrix chain multiplication (chapter 15.2), where the goal is to find the most computationally friendly matrix order when multiplying an arbitrary number of matrices in a row. You can assume that the entire input will be given as integers that can be stored using the standard C++ `int` type and that matrix sizes will be at least 1. We will use GradeMe06 to grade your code. Refer to the previous lab assignments for instructions on how to use the grading tool.

Questions and input structure The input has the following format. The first number, $n \geq 1$, in the test case will tell you how many matrices are in the sequence. Then, the first number will be followed by $n + 1$ numbers indicating the size of the dimensions of the matrices. First, you need to output the minimum number of scalar multiplications needed to multiply the given matrices. Then, print the matrix multiplications sequence via parentheses that minimizes the number of multiplications. Each matrix should be named A#, where # is the matrix number starting at 0 (zero) and ending at $n - 1$. See the examples below.

Bonus questions (worth half a lab)

1. Compute the minimum number of multiplications but using the recursive version of the algorithm without memoization (section 15.3: Recursive-Matrix-Chain).
2. Compute the minimum number of multiplications but using the recursive version of the algorithm with memoization (section 15.3: Memoized-Matrix-Chain).

Run your code for the input file 'tmemo'. How long do these two versions of the recursive algorithm take? To measure the time, for example you can use function gettimeofdays.

Examples of input and output

```
2
2 3 5
30
(A0A1)

3
10 100 5 50
7500
((A0A1)A2)

3
10 30 5 60
4500
((A0A1)A2)

6
30 35 15 5 10 20 25
15125
((A0(A1A2))((A3A4)A5))
```

Your solutions Before leaving the lab, submit a zipped tar archive of your program through the assignments page of CatCourse. Please use your UCMNetID as the filename for the zipped tar archive.