Problem 1 (70 pts) Implement the methods *minMult*, *maxMult*, and *printOrder*. I have provided the template Java file ***CMM_YOURNAME.java*** and the *main* method, as well as the input file *cmm_input.txt*. These methods are in Section 3.4. The *maxMult* method finds the largest possible number of multiplications and the worst order. For each matrix, the ratio ~~*minCost / maxCost*~~ ***maxCost / minCost*** is printed.

Add one more example to the input file. Give the number of matrices *n*, and then the dimensions *d*. The goal is to maximize the ratio above. DO NOT WORK ON THIS PART TOGETHER. You may use any value of *n* up to 40, and the dimensions can be any number between 1 and 100. The one with the largest ratio will get **+5 extra credit points** on this assignment. If there are ties, the points will be divided.

If your program does NOT compile, you get 0 points for this question. There will be no redoing this question. You must also save your output to a text file (or PDF for MAC users) and turn that in as well.

Problem 2 (10 pts) You are almost done tracing through the MinMult function for the Chained Matrix Multiplication problem, with $n = 6$, and the dimensions of the 6 matrices as follows:

A_1	A_2	A_3	A_4	A_5	A_6
6×5	5×4	4×5	5×4	4×10	10×5

Compute the values of M[1][6] and P[1][6]. ***Show clearly the work that done by the algorithm***

Answers: M[1][6] = _____ P[1][6] = _____

M	1	2	3	4	5	6
1	0	120	240	280	520	
2		0	100	160	360	460
3			0	80	240	360
4				0	200	300
5					0	200
6						0

P	1	2	3	4	5	6
1		1	2	1	4	
2			2	2	4	2
3				3	3	4
4					4	4
5						5
6						

Show the optimum order for multiplying these matrices 1 – 6: _____

Show the optimum order for multiplying these matrices 1 – 5: _____

Show the optimum order for multiplying these matrices 2 – 6: _____

Problem 3 (20 pts) Do the tracing for tabs *sa_2, sa_3, sa_4, and sa_5* the tracing in the "sequence_alignment_handout". For all of these, assume a mismatch penalty of 1 and a gap penalty of 2.