1. 我们对钢条切割问题进行一点修改，除了切割下的钢条段具有不同价格 p_i 外，每次切割还要付出固定的成本 c。这样，切割方案的收益就等于钢条段的价格之和减去切割的成本。设计一个动态规划算法解决修改后的钢条切割问题。

采用自底向上的动态规划来实现，先算出长度为 2 时的钢条切割的最大的销售利益 r_2 及此时的切割方案 s_2，再算出长度为 3 时的钢条切割的最大的销售利益 r_3 及此时的切割方案 s_3，依此类推，最后算出长度为 n 的钢条的切割的最大的销售利益 r_n 及此时的切割方案 s_n。依次算出长度为 i 时的钢条切割的最大的销售利益 r_i 及此时的切割方案 s_i，其中，
$r_i = max_{1\le j\le i}(p_i + r_0，max_{1\le j<i}(p_j + r_{i-j} - c))$，(r_0 = 0)，s_i 为对长度为 i 的钢条的对应的最优切割策略的最左边的第一次切割。

2. 令 R(i,j) 表示在一次调用 MATRIX-CHAIN-ORDER 过程中，计算其他表项时访问表项 m[i,j] 的次数。证明：

$$\sum_{i=1}^{n}\sum_{j=i}^{n}R(i,j) = \frac{n^3-n}{3}$$

当在计算m(k_1 , j)（k_1 = 1 , ... , i -1）、m(i，k_2)（k_2 = j + 1 , ... , n)时，才会访问表项 m(i , j)，故表项 m(i , j) 的访问次数为 i - 1 + n - j

$$\begin{aligned}
&\sum_{i=1}^{n}\sum_{j=i}^{n}R(i,j)\\
&=\sum_{i=1}^{n}\sum_{j=i}^{n}i-1+n-j\\
&=\sum_{i=1}^{n}\frac{(n+i-2)(n-i+1)}{2}\\
&=\frac{1}{2}\sum_{i=1}^{n}n^2-n-i^2+3i-2\\
&=\frac{1}{2}(n^3-n^2)+\frac{1}{2}\sum_{i=1}^{n}-i^2+3i-2\\
&=\frac{1}{2}(n^3-n^2)+\frac{1}{2}(-\frac{n(n+1)(2n+1)}{6}+3\frac{(n+1)n}{2}-2n)\\
&=\frac{n^3-n}{3}
\end{aligned}$$

3. 对输入链长度为 *n* 的矩阵链乘法问题，描述其子问题图：它包含多少个顶点？包含多少条边？这些边分别连接哪些顶点。

顶点个数

$$\sum_{i=1}^{n}i = \frac{n(n+1)}{2}$$

当顶点 (k_1 , j)、(i , k_2) 满足k_1 = i + 1 , ... , j , k_2 = i , ... , j - 1时，顶点 (k_1 , j)、(i , k_2) 与顶点 (i , j) 会有一条边相连，故与顶点 (i , j)相连的边有 2(j - i)条，故边的总个数为

$$\sum_{j=1}^{n}\sum_{i=1}^{j}2(j-i) = \sum_{j=1}^{n}j^2-j = \frac{(n(n+1)(2n+1)}{6} - \frac{n(n+1)}{2} = \frac{n^3}{3} - \frac{n}{3}$$