

使用动态规划算法求解矩阵连乘问题的最优计算次序及最优值：

M1: 5×10 , M2: 10×5 , M3: 5×25 , M4: 25×8 , M5: 8×8

使用动态规划算法求解矩阵连乘问题的最优计算次序及最优值：

M1:5×10, M2:10×5, M3:5×25, M4:25×8, M5:8×8

$$M_1 \cdot M_2 \cdot M_3 \cdots M_n$$

$$r_1, r_2, r_3, \cdots, r_n, r_{n+1}$$

$$(M_i)_{r_i \times r_{i+1}} \quad 1 \leq i \leq n$$

$$M_{i,j} = M_i \cdot M_{i+1} \cdots M_{j-1} \cdot M_j$$

$C[i, j]$ ：计算 $M_{i,j}$ 所需的最小乘法次数。

$i = 1, j = n$ 时，原问题得解。

$$\underbrace{M_{i,j}}_{C[i,j]} = \underbrace{(M_i \cdot M_{i+1} \cdots M_{k-1})}_{C[i,k-1]} \underbrace{(M_k \cdot M_{k+1} \cdots M_{j-1} \cdot M_j)}_{C[k,j]}$$

$C[i, j]$

$C[i, k-1]$

$C[k, j]$

$\leftarrow k \rightarrow$

$$C[i, j] = C[i, k-1] + C[k, j] + r_i \cdot r_k \cdot r_{j+1}$$



$$C[i, j] = \min_{i < k \leq j} \{C[i, k-1] + C[k, j] + r_i \cdot r_k \cdot r_{j+1}\}$$

使用动态规划算法求解矩阵连乘问题的最优计算次序及最优值：

M1:5×10, M2:10×5, M3:5×25, M4:25×8, M5:8×8

	1	2	3	4	5
1	0	250(2)			
2	-	0	1250(3)		
3	-	-	0	1000(4)	
4	-	-	-	0	1600(5)
5	-	-	-	-	0

使用动态规划算法求解矩阵连乘问题的最优计算次序及最优值：

M1:5×10, M2:10×5, M3:5×25, M4:25×8, M5:8×8

	1	2	3	4	5
1	0	250(2)	875(3)		
2	-	0	1250(3)	1400(3)	
3	-	-	0	1000(4)	1320(5)
4	-	-	-	0	1600(5)
5	-	-	-	-	0

使用动态规划算法求解矩阵连乘问题的最优计算次序及最优值：

M1:5×10, M2:10×5, M3:5×25, M4:25×8, M5:8×8

	1	2	3	4	5
1	0	250(2)	875(3)	1450(3)	
2	-	0	1250(3)	1400(3)	1720(3)
3	-	-	0	1000(4)	1320(5)
4	-	-	-	0	1600(5)
5	-	-	-	-	0

使用动态规划算法求解矩阵连乘问题的最优计算次序及最优值：

M1:5×10, M2:10×5, M3:5×25, M4:25×8, M5:8×8

	1	2	3	4	5
1	0	250(2)	875(3)	1450(3)	1770(3)
2	-	0	1250(3)	1400(3)	1720(3)
3	-	-	0	1000(4)	1320(5)
4	-	-	-	0	1600(5)
5	-	-	-	-	0

最优计算次序：((M1 ×M2)×(M3 ×M4 ×M5))

最优值：1770