6.4多个矩阵连乘模块设计

6.4.1问题提出

- **软件行业中客户总是在变更需求**

银行对我们公司开发的乘法模块还不满意。他们的真实想法并不是实现两个矩阵的乘法，而是是能一次够实现多个矩阵按照算法运算法则的乘法，因此要求我们进一步改进我们的系统，实现多个矩阵的连乘功能，并且需要看到我们设计的程序能够满足他们的运行效率要求时才付二期款。

关键计算问题

- 给定n个矩阵 $\{A_1, A_2, \ldots, A_n\}$ 其中 A_i 与 A_{i+1} 是可乘的，$i = 1, 2, \ldots, n-1$ 。考察这n个矩阵的连乘积

$$A_1 A_2 \ldots A_n$$

- 由于矩阵乘法满足结合律，所以计算矩阵的连乘可以有许多不同的计算次序。这种计算次序可以用加括号的方式来确定。
- 若一个矩阵连乘积的计算次序完全确定，也就是说该连乘积已完全加括号，则可以依此次序反复调用2个矩阵相乘的标准算法计算出矩阵连乘积

完全加括号的矩阵连乘积

- 设有四个矩阵 $A, B, C, D,$ 它们的维数分别是：

$$A=50\times10 \quad B=10\times40 \quad C=40\times30 \quad D=30\times5$$

- 总共有五中完全加括号的方式

$$(A((BC)D)) \qquad (A(B(CD))) \qquad ((AB)(CD))$$
$$(((AB)C)D) \qquad ((A(BC))D)$$

16000, 10500, 36000, 87500, 34500

对于 pXq 矩阵 A 和一个 qXr 矩阵 B, AB 需要 ？ 标准乘法计算.

$$c_{ij}=\sum_{k=1}^{q}a_{ik}b_{kj}$$

$$\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1r} \\ c_{21} & c_{22} & \cdots & c_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{r2} & \cdots & c_{pr} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1q} \\ a_{21} & a_{22} & \cdots & a_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1} & a_{p2} & \cdots & a_{pq} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1r} \\ b_{21} & b_{22} & \cdots & b_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ b_{q1} & b_{q2} & \cdots & b_{qr} \end{bmatrix}$$

```
matrixMultiply(int[][]a,int[][]b,int[][]c,int ra,int ca,int rb,int cb)
{if(ca!=rb)
Throw new IllegalArgumentException(“矩阵不可乘”);
for(int i=0;i<ra;i++)
for(int j=0;i<cb;j++)
{int sum=a[i][0]*b[0][j];
for(int k=1;k<rb;k++)
Sum+=a[i][k]*b[k][j];
c[i][j]=sum;
}
}
```

对于 *pXq* 矩阵 *A* 和一个 *qXr* 矩阵 *B*, *AB* 需要 ？ 标准乘法计算.

$$c_{ij}=\sum_{k=1}^{q}a_{ik}b_{kj}$$

$$\begin{bmatrix} c_{11} & c_{12} & \cdots & c_{1r} \\ c_{21} & c_{22} & \cdots & c_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n1} & c_{r2} & \cdots & c_{pr} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1q} \\ a_{21} & a_{22} & \cdots & a_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1} & a_{p2} & \cdots & a_{pq} \end{bmatrix} \times \begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1r} \\ b_{21} & b_{22} & \cdots & b_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ b_{q1} & b_{q2} & \cdots & b_{qr} \end{bmatrix}$$

```
matrixMultiply(int[][]a,int[][]b,int[][]c,int ra,int ca,int rb,int cb)
{if(ca!=rb)
Throw new IllegalArgumentException(“矩阵不可乘”);
for(int i=0;i<ra;i++)
for(int j=0;i<cb;j++)
{int sum=a[i][0]*b[0][j];
for(int k=1;k<rb;k++)
Sum+=a[i][k]*b[k][j];
c[i][j]=sum;
}
}
```

pqr次标准乘法

如果 A_1, A_2, and A_3 是 20X100, 100X10, 和 10X50 矩阵,
$A_1XA_2XA_3$乘积运算次数是多少?

$$\overbrace{\underbrace{(A_1XA_2)}_{20 \times 100 \times 10 = 20000} \times A_3}^{20 \times 10 \times 50 = 10000}$$

30000 次运算

$$(\underbrace{A_1 \times (A_2}_{20 \times 100 \times 50 = 100000} \overbrace{\times A_3)}^{100 \times 10 \times 50 = 50000})$$

150000 次运算

矩阵连乘问题

给定n个矩阵 $\{A_1,A_2,\ldots,A_n\}$ ，其中A_i与A_i+1是可乘的，i=1,2 ,…,n-1。如何确定计算矩阵连乘积的计算次序，使得依此次序计算矩阵连乘积需要的数乘次数**最少**。

穷举法求解思路

◆**穷举法**：列举出所有可能的计算次序，并计算出每一种计算次序相应需要的数乘次数，从中找出一种数乘次数最少的计算次序。

算法复杂度分析：
对于n个矩阵的连乘积，设其不同的计算次序为P(n)。
由于每种加括号方式都可以分解为两个子矩阵的加括号问题：
$((A_1\ldots A_k)(A_k+1\ldots A_n))$可以得到关于P(n)的递推式如下：

$$P(n)=\begin{cases}1 & n=1\\ \sum_{k=1}^{n-1}P(k)P(n-k) & n>1\end{cases}\Rightarrow P(n)=\frac{1}{n}C_{n-1}^{2(n-1)}=\Omega(4^n/n^{3/2})$$

Catalan 数 $P(n) = C(n-1)$

思考

能否用动态规划思想求解问题？