C3-B-进厂填括号

21371160 王旭阳

首先可以证明，对于计算n个矩阵链乘，可以通过计算前p个矩阵的乘积乘以后q个矩阵的乘积（满足p+q=n）。
因此，可以将矩阵链乘转为为加强版的“截钢管”问题。即，对于钢管而言，确定的截取长度则意味着确定的价格，而矩阵则随着截取位置不同而“计算价值”不同。
为了计算n个矩阵的最小相乘计算次数，则需要计算出每个切割方式所带来的不同的相乘计算次数。
如果设f[i][j]表示第i个矩阵到第j个矩阵的最小相乘计算次数。首先继续进行初始化工作，将f[i][i]全设为零，其余均设为＋∞。其次，通过动态规划思想，我们先计算任意两个连续矩阵的最小计算次数，然后由此计算任意连续三个，直至计算出任意连续n个，此即为最终答案。
公式为

$$
f[i][j]=\begin{cases} 0 & (i=j) \\ f[i][k]+f[k+1][j]+a[i]*a[k+1]*a[j+1] & (i<=k<j) \end{cases}
$$

```c
#include<stdio.h>
#include<string.h>
long long int a[1005],f[1005][1005];
int main(){
    long long int n,min=1,i,j,l,t,k;
    min=min<<40;
    scanf("%lld",&n);
    for(i=1;i<=(n+1);i++) scanf("%lld",&a[i]);
    for(i=1;i<=n;i++) f[i][i]=0;
    for(i=1;i<=n;i++)
        for(j=1;j<=n;j++) if(i!=j) f[i][j]=min;
    for(l=1;l<n;l++)
        for(i=1;i+l<=n;i++){
            j=i+l;
            for(k=i;k<j;k++) {
                t=f[i][k]+f[k+1][j]+a[i]*a[k+1]*a[j+1];
                if(t<f[i][j]) f[i][j]=t;
            }
        }
    printf("%lld",f[1][n]);
    return 0;
}
```