

动态规划：矩阵连乘求最优解问题

- 从整体往局部看，求解 $A_{i...j}$ 的矩阵连乘问题，可以考虑在某一个位置将这些矩阵分开，即 $A_{i...d}$ 和 $A_{d+1...j}$ ，划分开来后，再求子问题的最优加括弧方法
- 从局部往整体看，计算 $A_{i...j}$ 的代价=计算 $A_{i...d}$ 的代价+计算 $A_{d+1...j}$ 的代价+计算 $A_{i...d} * A_{d+1...j}$ 的代价

动态规划迭代实现问题求解：

可以先通过迭代方法将计算子问题的代价 $B[i...j]$ 和最优切割位置 $S[i...j]$ ，计算出来后保存到备忘录中以便求解过程中直接调用。

- 矩阵规模存储在一维数组 $p[1...n]$ 中
- 最优代价存储在二维数组 $B[i,j]$ 中 ($i,j \in [1...n]$)，表示 $A_{i...j}$ 连乘的最小结果
- 最优分割点存储在二维数组 $S[i,j]$ 中 ($i,j \in [1...n]$)，表示 $A_{i...j}$ 的最优分割点

时间复杂度：三层迭代，时间复杂度为 $O(n^3)$

空间复杂度： $O(n)+O(n^2)+O(n^2)=O(n^2)$

实验结果截图



```
Microsoft Visual Studio 调试控制台
INPUT:
6
30 35
35 15
15 5
5 10
10 20
20 25
OUTPUT:
((A1(A2A3))((A4A5)A6))

INPUT:
8
10 100
100 5
5 50
OUTPUT:
(A1(A2A3))

INPUT:
8
10 45
45 15
15 10
10 10
10 200
200 25
25 5
5 10
OUTPUT:
((A1(A2A3))(((A4A5)A6)A7)A8))

F:\c++\算法分析与设计\矩阵连乘\Debug\矩阵连乘.exe (进程 20016) 已退出，返回代码为: 0。
若要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口...
```