# 动态规划Assignment\_1

|  |  |
| --- | --- |
| 学号 | BY1706126 |
| 姓名 | 刘云飞 |

## 投资问题的dp手工求解

### 问题描述

有 8 万元的投资可以投给3 个项目，每个项目在不同投资数额下（以万元为单位）的利

润如下表。

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 投资额 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 项目1 | 0 | 5 | 15 | 40 | 80 | 90 | 95 | 98 | 100 |
| 项目2 | 0 | 5 | 15 | 40 | 60 | 70 | 73 | 74 | 75 |
| 项目3 | 0 | 4 | 26 | 40 | 45 | 50 | 51 | 52 | 53 |

写出你所设的状态变量、决策变量、状态转移方程与递推关系式，和手工求解的详细步

骤及结果。

### 问题求解

## 凸8边形的三角割分

### 问题描述

一个凸 8 边形P 的顶点顺时针为{，，… ，}，任意两顶点间的线段的权重由矩阵D 给出。若与与是P上不相邻的两个顶点，则线段称为P的一条弦。求P的一个弦的集合T，使得T中所有的弦恰好将P分割成互不重叠的三角形，且各三角形的权重之和为最小（一个三角形的权重是其各边的权重之和）。

要求：写出递推关系式、伪代码和程序相关说明，并分析时间复杂性。

### 问题分析

将凸多边形的三角割分问题可以看作每完成一次三角形割分，剩下的部分，可以看作两个多边形的三角割分的子问题（所分割的三角形不是相邻的三个顶点，否则可以看作为剩下的一个凸多边形的三角割的分子问题），如下图所示：



所以如果取得的T区域的三角形为凸8边形的一个最优的三角割分，那么当且仅当剩下的这两个割分的区域D1与D2求得的凸多边形割分的结果也是最优的。所以当选取其中两个顶点和来求解凸多边形的最优三角割时，有如下推导式：

其中表示D1的最优解，表示D2的最优解，本案例中是求解三角形的权重之和最小，考虑到迭代的边界条件可以得到下面的递推关系式：

### 伪代码

伪代码的展示如下：

定义一个N×N的二维方阵T，此处N=8

1. 初始化矩阵T，使全部元素都为0
2. 初始化与矩阵T同样形状的矩阵S用于记录最终相连的边
3. 子问题的规模为N-1，进行如下循环：  
   **For** r = [1: 1: N] **do** // r表示当前子问题的长度，即点的个数  
    **For** i = [1: 1: N-r] **do** // N-r表示最后一个子问题的前边界，三角形的第一个点  
    j ← i + r // 前边界为r，链长度为r的后边界，三角形的第二个点  
    T[i, j] ← T[i+1, j] + weight(i-1, i, j) // 初始化T[i, j]，此时k==i  
    S[i, j] ← i // 初始化S[i, j]，暂时记录为i  
    **For** k = [i + 1: 1 : j] **do** // 在前边界i到后边界j之间搜索更新最优解  
    tmp ← T[i+1, j] + T[k+1, j] + weight(i-1, k, j)  
    **if** tmp ＜ T[i, j] **then** // 更新最小值和顶点T[i, j] ← tmp  
    S[i, j] ← k  
    **EndIf  
    EndFor  
    EndFor  
   EndFor**
4. 最后得到的T[1, N-1]就是累加得到的凸多边形的三角割分的最小权重

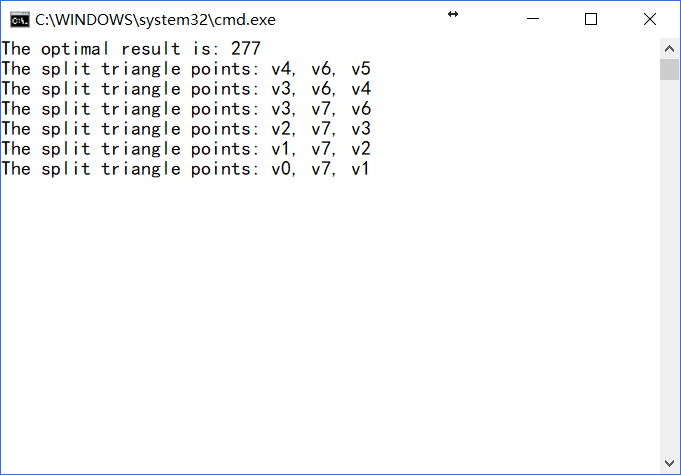
注：其中weight(i, j, k)是求解以，，为定点的三角形三条边的权重和

### 流程图



### 处理结果

通过运行附件2的算法对应的可执行程序可以得到对应结果如下：



对应的定点序号从0~7，共八个顶点。分割的结果如下图：

