

解题报告

2014 年 5 月 12 日

1 数学题 math

1.1 题目来源

Maloyaroslavets Summer Camp 2008

1.2 解答

60 分做法较简单，枚举 λ_1 或 λ_2 ，固定一个变量后，所有可能的向量端点都在一条直线上，过原点作直线的垂线，局部最优解必然是垂足附近的两个向量端点。最多需要枚举 122 次，每次用 $O(1)$ 时间求出局部最优解。

满分做法见金斌的集训队论文《欧几里得算法的应用》。

2 挖宝藏 treasure

2.1 题目来源

WinterCamp 2008 游览计划的加强版。

2.2 $h = 1$ 的情形

$h = 1$ 时，问题与 WC2008 的游览计划完全相同。用经典的连通性状态压缩动态规划可以解决，但代码较繁琐。

注意到连通块是树状的，并且宝藏数 k 很小。可以设计另一种动态规划：设 $f[x, y, S]$ 表示树根在点 (x, y) ，树包含的宝藏集合为 S 时，权值最

小的树。状态转移方程为

$$f[x, y, S] = \begin{cases} g[x, y, S] = \min \{ f[x, y, S'] + f[x, y, S \setminus S'] - a[x, y] \mid S' \neq \Phi, S' \subset S \} \\ f[x-1, y, S] + a[x, y] \\ f[x+1, y, S] + a[x, y] \\ f[x, y-1, S] + a[x, y] \\ f[x, y+1, S] + a[x, y] \end{cases}$$

转移方程具有后效性，但可以将问题视为最短路。由小到大枚举 S ，先计算 $g[x, y, S]$ ，然后用 spfa 松弛。复杂度 $O(nm3^k)$ 。

2.3 任意情形

在上述算法的外面套一个由底层往上递推的过程即可，计算每一层时，要增加一个特殊的宝藏，为下面层的所有宝藏合并而成，即 $f[i, x, y, 1] = f[i+1, x, y, 2^{K[i+1]+1} - 1]$ 。具体实现可以参考标程。复杂度为 $O(hnm3^k)$ 。

3 理想城市

3.1 题目来源

IOI2012, Day2, Ideal City.

3.2 解答

见顾昱洲的 IOI2012 解题报告。

另一种思路：将理想城进行横向的剖分，即将处在同一行并连续的格子压缩成一个节点，如果不同行的两个连续段有公共边界，则将对应的两个节点连边。容易发现压缩后的图呈树结构，树形 DP 即可。