

NOIP 欢乐%你赛题解

小澳的方阵 (matrix)

因为放兵种和染色十分相似，所以后面就称为染色好了。

【algorithm1】

模拟。每次涂色对矩阵的一行或者一列操作。最后输出时输出矩阵即可。

时间复杂度 $O(k*n)$ ，空间复杂度 $(n*m)$ 。可以通过 123 的数据点。

【algorithm2】

注意到每次只染色一行或者一列，那么我们最后输出第 i 行第 j 列的数字是多少的时候只需要看一下最后一次 i 行和第 j 行被染了什么颜色，所以我们需要对每一行和一列记录最后一次染色的颜色。

但是我們也需要比较是行比列后染色还是列比行后染色，所以我们还需要记录一下每行每列染色的最后时间。

综上，我们只需要对每一行和每一列开两个记录的值：一个记录最后一次染色的时间，一个记录最后一次染色的颜色，然后比较 i 行和 j 列的时间大小即可确定这个位置的颜色了。

时间复杂度 $O(k+m*n)$ ，空间复杂度 $(n+m)$ 。可以通过所有数据点。

原题来自 codeforces:

<http://codeforces.com/problemset/problem/631/B>

小澳的坐标系 (coordinate)

【algorithm1】

搜索，枚举所有走法，同时判断其合法性。

时间复杂度 $O(3^n)$ ，出题人十分善良，这种算法可以通过前两个测试点。

【algorithm2】

考虑在直线 $y=k$ 上走的情况：要么直接向上走到直线 $y=k+1$ 上，或者连续向左走 x 步再走到 $y=k+1$ 上，或者连续向右走 x 步再走到 $y=k+1$ 上。

由此可以列出递推方程， $f[i][j]$ 表示即将离开直线 $y=i$ 上，已经走了 j 步的方案数，递推方程：

$$f[i][j] = f[i-1][j-1] + 2 * (f[i-1][j-2] + f[i-1][j-3] + \dots + f[i-1][0])$$

时间复杂度 $O(n^3)$ ，这种算法通过前四个测试点。

【algorithm3】

对 algorithm2 进行前缀和优化，可以将复杂度降低到 $O(n^2)$ ，可以通过前六个测试点。

【algorithm4】

考虑合法路径的特点，如果第 $i-1$ 步向上走，那么第 i 步可以向上、左、右走；如果第 $i-1$ 步向左走，那么第 i 步可以向上或者向左走；如果第 $i-1$ 步向右走，

那么第 i 步可以向上或者向右走。

用 $f[i][0]$ 表示走了 i 步，第 i 步向上走的方案数； $f[i][1]$ 表示走了 i 步，第 i 步向左走的方案数； $f[i][2]$ 表示走了 i 步，第 i 步向右走的方案数，递推方程：

$$f[i][0]=f[i-1][0]+f[i-1][1]+f[i-1][2];$$

$$f[i][1]=f[i-1][0]+f[i-1][1];$$

$$f[i][2]=f[i-1][0]+f[i-1][2];$$

时间复杂度 $O(n)$ ，可以通过前八个测试点。

【algorithm5】

对 algorithm4 进行矩阵乘法优化，可以将时间复杂度降低到 $O(\log n)$ ，可以通过所有测试点。

加强自 openjudge 原题：

<http://noi.openjudge.cn/ch0206/9271/>

小澳的葫芦 (calabash)

【algorithm1】

第一个测试点只有一条边，输出 $w/2$ 就可以啦。

可以通过第 1 个测试点。

【algorithm2】

注意到“除 1 外，所有葫芦的入度均为 1”，也就是说，从 1 到 n 的路径只有一条，输出这一条路径的长度与这条路径上的点数的比值就可以了。

可以通过第 1、2、6 个测试点。

【algorithm3】

对于这样一类特殊数据，“所有从 1 到 n 的路径经过的葫芦数相等”，也就是说 1~ n 的最短路就是最优路径，最短路的长度与路径上的点数的比值就是答案。

可以通过第 1、2、3、6、7 个测试点。

【algorithm4】

另建一个起点 0，连接一条 0 到 1 长度为 0 的边，就此将问题转化为长度和边数最小比值。这个问题的求解需要分数规划。

假设答案为 ans ，对于任意一条由 k 条边组成的路径，有：

$$(w_1+w_2+w_3+\dots+w_k)/k \geq ans;$$

转化一下：

$$(w_1+w_2+w_3+\dots+w_k) \geq ans * k;$$

$$\text{即 } (w_1-ans) + (w_2-ans) + (w_3-ans) + \dots + (w_k-ans) \geq 0.$$

于是就得到了这样一个算法：

二分答案 x ，每次将每一条边的权值减去 x 求最短路，判断 1~ n 的最短路是否大于 0：若大于 0，则说明答案 $ans > x$ ；否则说明 $ans < x$ 。

这样可以通过所有测试点。

[题目思考来源：经典题：最小比例生成树]