

Art in Digital Age 解题报告

吉大附中实验学校 王天懿

November 10, 2015

1 题目描述

给定一张 $n * n$ 的矩阵，每个位置上有一个正整数。

你有一张 $n * n$ 的全0矩阵，每次可以选择一个子矩形区域并将区域中所有位置都修改为同一个数。

要求用尽量少的操作使这个矩阵变成给定的矩阵。

2 数据规模与约定

$1 \leq n \leq 1000$

$0 \leq \text{矩阵中的每个数} \leq 100$

数据保证随机生成。

3 题目解法

这是一道使用启发式算法的问题。

首先观察题目性质，我们发现随机化算法在此题上并没有什么作为。

由于数据范围非常大，我们可以考虑强化限定，然后贪心/DP的思路。

考虑到数据随机生成，因此几乎不会有将一个子矩形涂成0的情况，故在下列所有算法中我们默认不涂所有为0的格子。

3.1 解法1

考虑强化限定，我们让一个格子只能被涂一次。

这个问题仍然不是很好做，因此我们可以放弃最优解，采用贪心的方法获取一个较优解。

按顺序枚举每一个没有被覆盖的格子，以这个格子为左上角，贪心地向右枚举使得 $R - L + 1$ 最大，然后向下枚举使得 $U - D + 1$ 最大。

3.1.1 时空复杂度

时间复杂度： $O(n^2)$

空间复杂度： $O(n^2)$

3.2 解法2

考虑让一个格子可以被涂多次，但是每次都只能被涂成它最终的颜色。

我们仍然枚举每个没有被覆盖的格子，然后分别向右和向下二分计算出极大子矩形。

理论上会比算法1优一些。

3.2.1 时空复杂度

时间复杂度： $O(n^2 \log n)$

空间复杂度： $O(n^2)$

3.3 解法3

考虑到一个格子最终的颜色等于最后一次被涂成的颜色，因此我们可以考虑再度弱化上一做法的限定。我们可以令一个格子可以被涂上任何颜色，但是一旦被涂上其应有的颜色后就不能再被涂色。做法上我们可以枚举每一个格子，然后暴力向右枚举能涂的格子，再暴力向下枚举能涂的格子，接着将选定的矩形区域内所有与左上角颜色相同的格子都标记为不可涂。顺带一提CC上目前的rank1写的就是这一算法，但是我交他的程序只能得到0.231pts，不知道是因为换数据了还是什么原因，不明白CC的评测机制。

3.3.1 时空复杂度

暴力实现：

时间复杂度： $O(ans * n^2)$

空间复杂度： $O(n^2)$

其中ans大概在100左右。

数据结构实现：

时间复杂度： $O(n^2 \log^2 n)$

空间复杂度： $O(n^2)$

3.4 解法4

考虑再一次强化限定，我们让一个格子自从被涂上其应有的颜色后就只能被涂上这种颜色。那么我们只需要改造一下解法3的代码，在枚举的时候特判一下颜色就可以了。这个解法要优于解法3，在CC上能得到0.243pts，已经优于rank1的程序。

3.4.1 时空复杂度

时间复杂度： $O(ans * n^2)$

空间复杂度： $O(n^2)$

4 其他思路

由于官方题解上写了一大堆没用的废话，提供的参考程序又过于的冗长，因此对这道题暂时没有什么其他的思路。

由于这是个二维问题，所以我考虑过一维问题怎么搞，但是感觉并不太会做……

总之这种强化限定+贪心/DP的思路应该是这道题的主流，不知道还有什么其他的做法。