学习报告

2018.05.01—2018.05.07

目录

小凸玩矩阵

唯一标示:180088

这道题考点是二分图匹配，也是我做的第二道二分图匹配题。

考虑在一个$n*m(n \leq m)$的矩阵中选择n个数，使得第k大值最小，其中任意两个数不能在同一行或同一列。

题目中说k大值最小，这使我们很容易联想到求“最大值最小”和“最小值最大”的二分答案法，由于题目让找到尽可能小的值，我们只需要判断猜测的是是否“过小”即可。

我们可以得出：设猜测值为ans，可以选择c个小于ans的数。如果$c \geq n-k+1$，则ans可能还需要继续缩小，反之，我们就不能选择足够多的$\leq ans$的数，则ans就不满足“第k大”的条件，我们需要把答案增大。

<题目跳转> <查看代码>

放棋子

唯一标示:180023

在一个$n*m$的棋盘上放入p枚棋子，求出恰好每两个棋子之间互不相邻的概率。

考虑到数据范围较小，可以用状态压缩直接解决。设定一个三维的状态$f[i][j][k]$:第i行摆放情况为j，前i行共摆放了k个棋子时的方案数为$f[i][j][k]$，

易得状态转移方程：$f[i][j][k]=\sum_{l=0}^{l\in U} f[i-1][l][k-num(j)]$，其中：(j&i)==0。为了让每一行的摆放是合法的，我们可以预处理出一行的所有摆放情况。

<题目跳转> <查看代码>

SAM I AM

唯一标示:180089

这是一道基于矩阵的二分图匹配题，矩阵中有一些数，要求选出尽可能少的行/列，使得每个数都至少被一行/列包含。

第一步，构图，将X轴、Y轴设为点，矩阵上的点设为边，现在的行/列都变为节点，这样可以得出一个结论，要想达到要求，任意一条边至少有一个端点被选择，也就是点覆盖。由于最小点覆盖=最大匹配，即计算最大匹配即可。

第二步，输出方案，从二分图中X集合中的每个非匹配点开始，试图求出一条可增广道路，当然这是不会有所收获的(增广路定理)，我们记录一下所走到的点，最终，最小点覆盖集 = Y集中被经过的点 + X集中未被经过的点。

附1 最小点覆盖=最大匹配 证明：

对于一个二分图，如图1，他的最大匹配(图2)数为4，任意两条匹配边都不相交，易得，要想让这些匹配边都被覆盖，至少需要选择4个节点，即最小覆盖≥最大匹配。

继续观察，如果选择这四个顶点之后，还有一些边没有被覆盖，那么这些边的两个端点一定不是匹配点，这时这条边会变为匹配边，但由于这条边不是匹配边，所以可以断定，所有边都被覆盖。

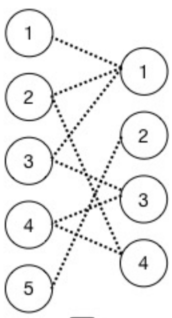

图1

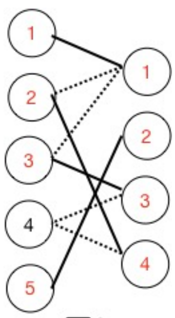

图2

附2 最小点覆盖集 = Y集中被经过的点 + X集中未被经过的点 证明：

以X集合的所有没有被匹配的点为起点开始跑匈牙利，标记经过的节点。这样扫描之后，不可能存在一条边，他在X集合的端点是有标记的，且在Y集合的顶点是没有标记的[1]。

如果这条边不是匹配边，且X集合中的端点不是匹配点，那么Y集合中的端点一定会被标记，如果Y集合中的端点不是匹配点，那么这两个端点会被一起访问/不访问。

如果这条边是匹配边，那么这两个端点会被一起访问/不访问。

由于只有[1]情况发生，才会存在一条边不被覆盖，所以任何一条边都会被覆盖。

<题目跳转> <查看代码>

Taxi Cab Scheme

唯一标示:180091

这是一道最小路径覆盖题，有n个任务需要完成，每个任务有规定的起点、终点和开始时间，求出至少需要分配多少个出租车完成这些任务。

建图，我们可以将每一个任务看做一个节点，如果完成任务i之后有足够的时间完成任务j，那么就将i和j之间连接一条有向边，完成建图之后我们可以发现，一辆车可以完成一条路径上的所有任务(一个节点也看做一条路径)，我们现在要求出最少选择多少条路径即可覆盖所有节点。

将原图的每一个点i拆成两个点i和i'，并将原图中$i-->j$的边替换为$i-->j'$，然后以原来的点和新拆出的点为X、Y集合建立二分图，由 最小路径覆盖=点数-最大匹配数 可得出最少需要的出租车数量。

<题目跳转> <查看代码>