

# Solution

ExfJoe

福建省长乐第一中学

July 25, 2017

# Outline

- 1 摆棋子
- 2 卡牌配对
- 3 最优价值

# 摆棋子

# 摆棋子

- 可以把题意转为去掉尽量多的棋子

# 摆棋子

- 可以把题意转为去掉尽量多的棋子
- 行建一排点，列建一排点，源向行连容量为  $n - X_i$  的边，列向汇连容量为  $m - Y_i$  的边

# 摆棋子

- 可以把题意转为去掉尽量多的棋子
- 行建一排点，列建一排点，源向行连容量为  $n - X_i$  的边，列向汇连容量为  $m - Y_i$  的边
- 若  $(x, y)$  未损坏则  $x$  行向  $y$  列连一条容量为 1 的边

# 摆棋子

- 可以把题意转为去掉尽量多的棋子
- 行建一排点，列建一排点，源向行连容量为  $n - X_i$  的边，列向汇连容量为  $m - Y_i$  的边
- 若  $(x, y)$  未损坏则  $x$  行向  $y$  列连一条容量为 1 的边
- 总棋子数减去最大流即是答案

# 摆棋子

- 可以把题意转为去掉尽量多的棋子
- 行建一排点，列建一排点，源向行连容量为  $n - X_i$  的边，列向汇连容量为  $m - Y_i$  的边
- 若  $(x, y)$  未损坏则  $x$  行向  $y$  列连一条容量为 1 的边
- 总棋子数减去最大流即是答案
- 也可以直接正着做用带下界的最小费用流



# Outline

- 1 摆棋子
- 2 卡牌配对
- 3 最优价值

# 卡牌配对

# 卡牌配对

- 60 分做法：明显的匹配问题，暴力构建二分图即可

# 卡牌配对

- 60 分做法：明显的匹配问题，暴力构建二分图即可
- 暴力做法边数过多，考虑将边分类进行优化

# 卡牌配对

- 60 分做法：明显的匹配问题，暴力构建二分图即可
- 暴力做法边数过多，考虑将边分类进行优化
- $A$  属性能被  $x$  整除且  $B$  属性能被  $y$  整除的所有点，只要在两侧则一定能够匹配，所以我们在网络流中增加一排这样的点

# 卡牌配对

- 60 分做法：明显的匹配问题，暴力构建二分图即可
- 暴力做法边数过多，考虑将边分类进行优化
- $A$  属性能被  $x$  整除且  $B$  属性能被  $y$  整除的所有点，只要在两侧则一定能够匹配，所以我们在网络流中增加一排这样的点
- 满足要求的左右点分别与它相连，边权为正无穷

# 卡牌配对

- 60 分做法：明显的匹配问题，暴力构建二分图即可
- 暴力做法边数过多，考虑将边分类进行优化
- $A$  属性能被  $x$  整除且  $B$  属性能被  $y$  整除的所有点，只要在两侧则一定能够匹配，所以我们在网络流中增加一排这样的点
- 满足要求的左右点分别与它相连，边权为正无穷
- 由于  $x$  与  $y$  只需是质数，这样的点一共  $3 * 46 * 46$  个 (200 内 46 个质数)，而  $200 < 2 * 3 * 5 * 7$ ，所以两侧点至多连出  $3 * 3 * 3$  条边

# 卡牌配对

- 60 分做法：明显的匹配问题，暴力构建二分图即可
- 暴力做法边数过多，考虑将边分类进行优化
- $A$  属性能被  $x$  整除且  $B$  属性能被  $y$  整除的所有点，只要在两侧则一定能够匹配，所以我们在网络流中增加一排这样的点
- 满足要求的左右点分别与它相连，边权为正无穷
- 由于  $x$  与  $y$  只需是质数，这样的点一共  $3 * 46 * 46$  个 (200 内 46 个质数)，而  $200 < 2 * 3 * 5 * 7$ ，所以两侧点至多连出  $3 * 3 * 3$  条边
- 70000 个点， $2 \times 10^6$  边的网络流，由于是分层图，所以 Dinic 仍然可以快速通过



# Outline

- 1 摆棋子
- 2 卡牌配对
- 3 最优价值

# 最优价值

# 最优价值

- 对于  $w_{i,j} + w_{j,i}$  建一个点,

# 最优价值

- 对于  $w_{i,j} + w_{j,i}$  建一个点,
- 对于每个位置建一个点, 代价为  $-a$

# 最优价值

- 对于  $w_{i,j} + w_{j,i}$  建一个点,
- 对于每个位置建一个点, 代价为  $-a$
- 对于每个字符建一个点, 代价为  $-b + a$

# 最优价值

- 对于  $w_{i,j} + w_{j,i}$  建一个点,
- 对于每个位置建一个点, 代价为  $-a$
- 对于每个字符建一个点, 代价为  $-b + a$
- 一类点连向二类点, 二类点连向三类点

# 最优价值

- 对于  $w_{i,j} + w_{j,i}$  建一个点,
- 对于每个位置建一个点, 代价为  $-a$
- 对于每个字符建一个点, 代价为  $-b + a$
- 一类点连向二类点, 二类点连向三类点
- 最大权闭合图