博弈论选讲

zhou888

雅礼中学

May 30, 2018

Preface

博弈论是OI中一个比较有趣的内容

近年来这方面的题目也比较多

由于本人比较菜，如果有错误，请多多包涵。

Nim游戏

Nim游戏是博弈论中最经典的模型

Nim游戏

Nim游戏是博弈论中最经典的模型

通常的Nim游戏的定义是这样的：有若干堆石子，每堆石子的数量都是有限的，合法的移动是“选择一堆石子并拿走若干颗（不能不拿）”，如果轮到某个人时所有的石子堆都已经被拿空了，则判负（因为他此刻没有任何合法的移动）。

这里需要介绍一个新东西SG函数，这个东西同时可以解决NIM游戏。

先将游戏中的每个局面抽象成一个有向图中的一个点,游戏中的一次决策看成一条边,那么每次游戏都可以看成一条从入度为 0 的点走到一个出度为 0 的点的路径。

SG 函数是对图中每个结点的评估函数,定义如下:

$$SG(x) = mex\{SG(y) |\ \exists(x, y)\ in\ graph\}$$

其中 mex 是定义在整数集合上的操作:

$$mex(S) = min\{k | k\ \ \in S, k \in N\}$$

在 Nim 游戏中,一个 n 个石头的堆的 SG 值就是为 n。

SG 函数有三个性质:

1. 若 $SG(x) = 0$,那么 x 所有后继点的 SG 值都不为 0。

SG 函数有三个性质:

1. 若 $SG(x) = 0$,那么 x 所有后继点的 SG 值都不为 0。

2. 若 $SG(x) != 0$,那么 x 的后继点中至少有一个点的 SG 值为 0。

SG 函数有三个性质:

1. 若 $SG(x) = 0$,那么 x 所有后继点的 SG 值都不为 0。
2. 若 $SG(x) != 0$,那么 x 的后继点中至少有一个点的 SG 值为 0。
3. 结束状态的 SG 值都为 0。

SG 函数有三个性质:

1. 若 SG(x) = 0,那么 x 所有后继点的 SG 值都不为 0。

2. 若 SG(x) != 0,那么 x 的后继点中至少有一个点的 SG 值为 0。

3. 结束状态的 SG 值都为 0。

这样的性质可以帮助我们判定一个局面是否有必胜策略。如果$SG(x) = 0$,那么无论接下来走哪一步 y,$SG(y) > 0$,另一个人都可以将其走到一个 z 满足 SG(z) = 0,结合最终的失败状态的 SG 值都为 0,$SG(x) = 0$ 的时候是必败的,相反 $SG(x) > 0$ 的时候是必胜的。

现在考虑怎么样将多个石堆并起来考虑。

现在考虑怎么样将多个石堆并起来考虑。

我们定义游戏的和:有若干个同时进行的游戏,玩家每轮可以选择任意一个游戏进行决策,最终无法操作的玩家输。那么 Nim 游戏相当于若干个单堆石头的和。

现在考虑怎么样将多个石堆并起来考虑。

我们定义游戏的和:有若干个同时进行的游戏,玩家每轮可以选择任意一个游戏进行决策,最终无法操作的玩家输。那么 Nim 游戏相当于若干个单堆石头的和。

SG定理：对于若干个游戏图 $G1, G2, \cdots, Gn$,有:

$$SG(G1 + G2 + \cdots + Gn) = SG(G1)\ xor\ SG(G2)\ xor \cdots xor\ SG(Gn)$$

所以之前Nim游戏的结论有：

当且仅当a1 xor a2 xor ... xor an=0时先手必败，否则先手必胜。

更广泛的有：对于一个问题，如果某些子问题是相互独立的，我们就可以用sg定理，总问题的sg等于各个子问题的异或和。

一般解决SG函数有两种方法：打表 or 暴力

但对于一些特殊的题有特殊的解法

ARC091F

题意：有n堆石子，每堆石子有a_i个，有 一个数k_i现在两个人博弈，每个人可以拿掉一堆石子里$1 \rightarrow \lfloor \frac{x}{k_i} \rfloor$数量的石子，$x$为当前这一堆石子的数量，谁不能拿就输了，求谁赢。

数据范围：

$1 \leqslant N \leqslant 200$

$1 \leqslant A_i, K_i \leqslant 10^9$

ARC091F

k=2的情况是石子游戏。规律也有些相似

ARC091F

k=2的情况是石子游戏。规律也有些相似

通过打表发现

当 $x\%k = 0$ 时 $sg(x) = x/k$

$sg(x) = sg(x - (\lfloor \frac{x}{k} \rfloor + 1))$

所以我们可以递归求sg函数，但是这样会T

ARC091F

k=2的情况是石子游戏。规律也有些相似

通过打表发现

当 $x\%k=0$ 时 $sg(x)=x/k$

$sg(x)=sg(x-(\lfloor\frac{x}{k}\rfloor+1))$

所以我们可以递归求sg函数，但是这样会T

我们$\lfloor\frac{x}{k}\rfloor+1$相同的放在一次一起处理就可以了

AGC002E

有 N 堆糖果,第 i 堆有 A_i 个糖。

Snuke 和 Ciel 在玩游戏,Snuke 先手。

两人人轮流进行,每次必须挑选做两个操作之一:

1.将当前最大的那堆糖果全部吃完

2.将当前存在每堆糖果各吃一个

吃完最后一个的人输,问先手是否必胜。

$N \leqslant 10^5, A_i \leqslant 10^9$

AGC002E

把所有的A_i排序,把这张图画成一个不规则轮廓的图形

AGC002E

把所有的A_i排序,把这张图画成一个不规则轮廓的图形

问题可以转化成这样:

假设一开始(0,0)处有一个棋子

两个人轮流移动,每个人把棋子向右或向上走,走到边缘上的人输

AGC002E

把所有的A_i排序,把这张图画成一个不规则轮廓的图形

问题可以转化成这样:

假设一开始(0,0)处有一个棋子

两个人轮流移动,每个人把棋子向右或向上走,走到边缘上的人输

打表可得$SG(i,j) = SG(i+1,j+1)$

AGC002E

这样可以变成计算不在边缘上的最大的(i,i)的SG值

AGC002E

这样可以变成计算不在边缘上的最大的(i,i)的SG值

此时的SG的只能向右或向上一直走，特判即可

AGC002E

这样可以变成计算不在边缘上的最大的(i,i)的SG值

此时的SG的只能向右或向上一直走，特判即可

时间复杂度$O(n)$

BZOJ1188

聪聪和睿睿最近迷上了一款叫做分裂的游戏。该游戏的规则试：共有n个瓶子，标号为0,1,2.....n-1,第i个瓶子中装有p[i]颗巧克力豆，两个人轮流取豆子，每一轮每人选择3个瓶子。标号为i,j,k,并要保证$i < j, j \leqslant k$且第i个瓶子中至少要有1颗巧克力豆，随后这个人从第i个瓶子中拿走一颗豆子并在j,k中各放入一粒豆子（j可能等于k）。如果轮到某人而他无法按规则取豆子，那么他将输掉比赛。胜利者可以拿走所有的巧克力豆！两人最后决定由聪聪先取豆子，为了能够得到最终的巧克力豆，聪聪自然希望赢得比赛。

BZOJ1188

他思考了一下，发现在有的情况下，先拿的人一定有办法取胜，但是他不知道对于其他情况是否有必胜策略，更不知道第一步该如何取。他决定偷偷请教聪明的你，希望你能告诉他，在给定每个瓶子中的最初豆子数后是否能让自己得到所有巧克力豆，他还希望你告诉他第一步该如何取，并且为了必胜，第一步有多少种取法?

$1 \leqslant n \leqslant 21, p[i] \leqslant 10000$

BZOJ1188

由于每一堆会影响后面的堆，所以不能把每一堆看成独立。

BZOJ1188

由于每一堆会影响后面的堆，所以不能把每一堆看成独立。

我们发现实际上问题只和每一堆的奇偶性有关。

BZOJ1188

由于每一堆会影响后面的堆，所以不能把每一堆看成独立。

我们发现实际上问题只和每一堆的奇偶性有关。

所以我们可以把第i堆的一粒石子是独立的

BZOJ1188

由于每一堆会影响后面的堆，所以不能把每一堆看成独立。

我们发现实际上问题只和每一堆的奇偶性有关。

所以我们可以把第i堆的一粒石子是独立的

这样我们就可以算sg函数了。

BZOJ1188

由于每一堆会影响后面的堆，所以不能把每一堆看成独立。

我们发现实际上问题只和每一堆的奇偶性有关。

所以我们可以把第i堆的一粒石子是独立的

这样我们就可以算sg函数了。

需要注意的是因为n不同所以预处理要反过来记录

BZOJ1188

由于每一堆会影响后面的堆，所以不能把每一堆看成独立。

我们发现实际上问题只和每一堆的奇偶性有关。

所以我们可以把第i堆的一粒石子是独立的

这样我们就可以算sg函数了。

需要注意的是因为n不同所以预处理要反过来记录

求方案数直接枚举第一次的操作去算就行了。

BZOJ4035

有一个长度为N的数组，甲乙两人在上面进行这样一个游戏：首先，数组上有一些格子是白的，有一些是黑的。然后两人轮流进行操作。每次操作选择一个白色的格子，假设它的下标为x。接着，选择一个大小在$1 \to n/x$之间的整数k，然后将下标为x、2x、...、kx的格子都进行颜色翻转。不能操作的人输。现在甲（先手）有一些询问。每次他会给你一个数组的初始状态，你要求出对于这种初始状态他是否有必胜策略。

$N \leqslant 1000000000, K,$白色格子数量$\leqslant 100$

BZOJ4035

考虑可以把问题转化为既可以选白点，又可以选黑点。因为如果选黑点，一定不能一次胜利，而反倒对对方有利，且对方可以选一样的将状态重置。

BZOJ4035

考虑可以把问题转化为既可以选白点，又可以选黑点。因为如果选黑点，一定不能一次胜利，而反倒对对方有利，且对方可以选一样的将状态重置。

这样$sg(x) = mex\{sg(2*x), sg(2*x) \text{ xor } sg(3*x) \cdots\}$

BZOJ4035

考虑可以把问题转化为既可以选白点，又可以选黑点。因为如果选黑点，一定不能一次胜利，而反倒对对方有利，且对方可以选一样的将状态重置。

这样$sg(x) = mex\{sg(2*x), sg(2*x) \text{ xor } sg(3*x) \cdots\}$

这样时间复杂度还是不能过。

BZOJ4035

考虑可以把问题转化为既可以选白点，又可以选黑点。因为如果选黑点，一定不能一次胜利，而反倒对对方有利，且对方可以选一样的将状态重置。

这样$sg(x) = mex\{sg(2*x), sg(2*x) \text{ xor } sg(3*x)\cdots\}$

这样时间复杂度还是不能过。

BZOJ4035

我们可以发现一个性质：

BZOJ4035

我们可以发现一个性质：

当 $\lfloor \frac{n}{i} \rfloor = \lfloor \frac{n}{j} \rfloor$ 时 $sg(i) = sg(j)$，可以用归纳法证明

BZOJ4035

我们可以发现一个性质：

当 $\lfloor \frac{n}{i} \rfloor = \lfloor \frac{n}{j} \rfloor$ 时 $sg(i) = sg(j)$，可以用归纳法证明

之后我们就可以分块，相同的块放在一起。

BZOJ4035

我们可以发现一个性质：

当 $\lfloor \frac{n}{i} \rfloor = \lfloor \frac{n}{j} \rfloor$ 时 $sg(i) = sg(j)$，可以用归纳法证明

之后我们就可以分块，相同的块放在一起。

记录答案时，大于$\sqrt{n}$的直接记录在$\lfloor \frac{n}{i} \rfloor$上就可以了

Take-Away

现在有一个石堆,包含 n 个石子,每次可以拿走 $1 \to k$ 个石子,两个玩家轮流操作,最后无法操作的玩家输。

$n \leqslant 20$

Take-Away Solution

很显然 $SG(n) \equiv n \ mod \ (k+1)$

BZOJ3759

给 n 个箱子,每个箱子内有一些石子,两个人轮流操作,每个玩家可以进行以下操作之一:

1. 打开任意多的箱子。

2. 从一个打开的箱子中拿走任意多的石子。

不能操作者判负,求先手是否必胜。

$n \leqslant 20$

BZOJ3759

定义必败状态:当前打开的箱子中石子异或和为 0,没打开的箱子中不存在一个子集满足异或和为 0。这个很容易证明。

BZOJ3759

定义必败状态:当前打开的箱子中石子异或和为 0,没打开的箱子中不存在一个子集满足异或和为 0。这个很容易证明。

那么只要一开始存在子集满足异或和为 0,那么先手必胜。

Misère Nim

取到最后一个石子的玩家输,其他条件和普通 Nim 一样。

先手必胜条件:

1. SG 和为 0 且不存在一个石堆中石子的数量大于 1。

2. SG 和不为 0 且存在一个石堆中石子的数量大于 1。

先手必胜条件:

1. SG 和为 0 且不存在一个石堆中石子的数量大于 1。

2. SG 和不为 0 且存在一个石堆中石子的数量大于 1。

如何证明?

Misère Nim Proof

考虑不存在石堆的大小大于 1 的情况。

Misère Nim Proof

考虑不存在石堆的大小大于 1 的情况。

这种情况下,如果 SG 和为 0,说明有偶数个 1,对方一定会取到最后一个,先手必胜。SG 不为 0 情况反之。

Misère Nim Proof

考虑存在石子数量大于 1 的情况。

Misère Nim Proof

考虑存在石子数量大于 1 的情况。

如果大于 1 的石堆有一个,那么 SG 和大于 0,我们选择取到只剩0 或 1 个,根据 1 的石堆数量决定,先手必胜。

Misère Nim Proof

考虑存在石子数量大于 1 的情况。

如果大于 1 的石堆有一个,那么 SG 和大于 0,我们选择取到只剩0 或 1 个,根据 1 的石堆数量决定,先手必胜。

如果大于 1 的石堆至少有两个,如果 SG 和大于 0,我们按照常规Nim 的最佳策略操作,直到我们的局面只剩一个石堆大于 1(为什么?),然后和上面操作一样,先手必胜。如果 SG 和等于 0,那么一次操作后 SG 大于 0,留给对面的是必胜局面。

Misère Nim Proof

考虑存在石子数量大于 1 的情况。

如果大于 1 的石堆有一个,那么 SG 和大于 0,我们选择取到只剩0 或 1 个,根据 1 的石堆数量决定,先手必胜。

如果大于 1 的石堆至少有两个,如果 SG 和大于 0,我们按照常规Nim 的最佳策略操作,直到我们的局面只剩一个石堆大于 1(为什么?),然后和上面操作一样,先手必胜。如果 SG 和等于 0,那么一次操作后 SG 大于 0,留给对面的是必胜局面。

因此这种情况下 SG 大于 0 即可保证先手必胜。

poj3710

题意：

一个树图，然后1永远是根，两人轮流删边，不能删者输。

删边限制：只能删跟1连通的边。

树图限制：它首先是一棵树，然后某些点上可能带一个环。

我们会发现，拥有奇数条边的环可简化为一条边，偶数条边的环可简化为一个节点。

我们会发现，拥有奇数条边的环可简化为一条边，偶数条边的环可简化为一个节点。

所以我们可以直接用树的删边游戏做。

极大极小搜索

局面估价函数：我们给每个局面规定一个估价函数值f，评价它对于己方的有利程度。胜利的局面的估价函数值为inf，而失败的局面的估价函数值为-inf。

极大极小搜索

局面估价函数：我们给每个局面规定一个估价函数值f，评价它对于己方的有利程度。胜利的局面的估价函数值为inf，而失败的局面的估价函数值为-inf。

Max 局面：假设这个局面轮到己方走，有多种决策可以选择，其中每种决策都导致一种子局面。由于决策权在我们手中，当然是选择估价函数值f最大的子局面，因此该局面的估价函数值等于子局面f值的最大值，把这样的局面称为 max 局面。

极大极小搜索

Min 局面：假设这个局面轮到对方走，它也有多种决策可以选择，其中每种决策都导致一种子局面。但由于决策权在对方手中，在最坏的情况下，对方当然是选择估价函数值f最小的子局面，因此该局面的估价函数值等于子局面f值的最小值，把这样的局面称为min局面。

极大极小搜索

Min 局面：假设这个局面轮到对方走，它也有多种决策可以选择，其中每种决策都导致一种子局面。但由于决策权在对方手中，在最坏的情况下，对方当然是选择估价函数值f最小的子局面，因此该局面的估价函数值等于子局面f值的最小值，把这样的局面称为min局面。

终结局面：胜负已分

极大极小搜索

Min 局面：假设这个局面轮到对方走，它也有多种决策可以选择，其中每种决策都导致一种子局面。但由于决策权在对方手中，在最坏的情况下，对方当然是选择估价函数值f最小的子局面，因此该局面的估价函数值等于子局面f值的最小值，把这样的局面称为min局面。

终结局面：胜负已分

但是这样暴力时间复杂度是非常爆炸的，所以我们需要用到它的拍档alpha-beta剪枝。

alpha-beta

简单地说就是用两个参数 alpha 和 beta 表示当前局面的父局面的估价函数值f。如果当前局面为 min 局面，则需要借助父局面目前已求得的f（即 alpha）来判断是否需要继续搜索下去；如果当前局面为 max 局面，则需要借助父局面目前已求得的f（即 beta）。

纳什均衡

纳什均衡的定义：在博弈G=S1,...,Sn：u1,...，un中，如果由各个博弈方的各一个策略组成的某个策略组合（s1*,...，sn*）中，任一博弈方i的策略si*，都是对其余博弈方策略的组合（s1*,...s*i-1,s*i+1,...，sn*）的最佳对策，也即ui（s1*,...s*i-1,si*,s*i+1,...，sn*）⩾ui（s1*,...s*i-1,sij*,s*i+1,...，sn*）对任意sij∈Si都成立，则称（s1*,...，sn*）为G的一个纳什均衡。

纳什均衡

纳什均衡的定义：在博弈G=S1,...,Sn：u1,...，un中，如果由各个博弈方的各一个策略组成的某个策略组合（s1*,...，sn*）中，任一博弈方i的策略si*，都是对其余博弈方策略的组合（s1*,...s*i-1,s*i+1,...，sn*）的最佳对策，也即ui（s1*,...s*i-1,si*,s*i+1,...，sn*）⩾ui（s1*,...s*i-1,sij*,s*i+1,...，sn*）对任意sij∈Si都成立，则称（s1*,...，sn*）为G的一个纳什均衡。

纳什均衡的意思是：任何一方采取的策略都是对其余所有方采取策略组合下的最佳对策；当所有其他人都不改变策略时，为了让自己的收益最大，任何一方都不会（或者无法）改变自己的策略，这个时候的策略组合就是一个纳什均衡。

Codeforces 98E

A君有 n 张牌,B 君有 m 张牌,桌上还有一张反扣着的牌,每张牌都不一样。

每个回合可以做两件事中的一件:

1. 猜测桌上的牌是什么,猜对则胜,猜败则输。

2. 询问对方是否有某张牌,若有则需要将其示出,否则继续游戏。

A 和 B 都很聪明,问 A 的胜率。

$n, m \le 5000$

Codeforces 98E

首先如果不确定反扣着的牌的话是不会猜桌上的牌的。

假设对方如果问了一张自己没有的牌,我们可能怀疑桌上的牌就是这张。而我们可以用自己的牌去询问对方,如果对方相信的话就会输掉,我们称这种行为叫做欺骗。

设 $f(n, m)$ 表示先手胜率,则我们可以列出一个转移的表格

这个只需要求直线交即可。

Thanks