2025/11/29 16:46 1/4 简况

# 简况

AC 5题,属实菜逼。

比赛链接

# 颞解

## A. 张老师和菜哭武的游戏

#### 题意:

有\$1 \sim n\$共\$n\$个数,最开始拿走\$a, b, a \ne b\$\[]当数\$j\$能被拿走时,当且仅当\$\exists x, y\$满足\$x, y\$已经被拿走且\$x + y = j\$或\$x - y = j\$\[]判断能拿走的数的个数的奇偶性。

#### 题解:

可以看出,能被拿走的数一定能用x \* a + y \* b表示,也就是说这个数一定是gcd(a, b)的倍数。

那么判断\$n/gcd(a, b)\$的奇偶性即可。

## B. 伤害计算

略,用python很好写。

## C. 张老师的旅行

#### 题意:

一条直线上有\$n, n \leq 10^3\$个点,最开始张老师在点\$x\$[]

第\$i\$个景点在位置\$p\_i\$□必须在\$t\_i\$之前到达才能打卡。

求张老师想要打卡所有景点的最短时间, 无解输出-1。

### 题解:

张老师打卡过的景点一定是连续的一段,于是令\$dp[i][j][0/1]\$表示已经打卡区间\$[i,j]\$[]\$k = 0\$表示此时在\$i\$[]为1表示此时在\$j\$[]

转移时考虑从\$dp[i+1][j][0/1], dp[i][j-1][0/1]\$过来即可,需要判断一下到当前点时能否打卡这个景点,不能则无解。

### D. 车辆调度

#### 题意:

\$w \times h\$的广场上有若干目标点,障碍和最多4辆小车。

每次可以选择一辆小车让它往上、下、左、右4个方向一直走直到遇见障碍、边界、小车停下。

问操作\$k, k \leq 5\$次后,能否使得有一辆小车停在目标点。

#### 题解:

暴力搜索即可。

### E. 弦

#### 总结:

做的时候合法方案数没去重,结果合法方案数算都算不出来。

算总方案数的时候没有考虑顺序问题,算合法方案数的却考虑了顺序问题导致白给。

#### 题意:

给定一个圆,圆上有\$2N, N \leq 10^7\$个互不重叠的点。每次操作随机选择两个先前未选择过的点连一条 弦, 共连成\$N\$条弦, 求所有弦不交的概率。

#### 题解:

总方案数为\$\frac{(2n)!}{n! \times 2^n}\$[

合法方案数为 $f(n) = \sum_{i=1}^{n} f(i) * f(n - i - 1) = \frac{C(2n, n)}{n + 1}$  也就是卡特兰数。

相除即可得到答案。

## F. 排列计算

### 题意:

给出一些询问,每次询问的是\$[I, r]\$这段数的和,并将它加入总分。

需要找到一个排列使得总分最大。

#### 颞解:

统计每个位置被取了多少次,按照被取次数从小到大填入\$1 \sim n\$即可。

https://wiki.cvbbacm.com/ Printed on 2025/11/29 16:46 2025/11/29 16:46 3/4 简况

### G. 硬币游戏Ⅲ

### H. 时空栈

#### 题意:

有三种操作:

\$0, t, v\$□在时刻\$t\$把\$v\$入栈

\$1, t\$在时刻\$t\$把栈顶元素出栈

\$2. t\$询问时刻\$t\$的栈顶元素

保证\$t\$互不相同,且出栈和询问时栈不空。

#### 题解:

首先将\$t\$离散化,维护每个时刻栈内元素的个数。

那么对于查询操作,设\$t\$时刻之前首次栈内元素个数小于现在\$t\$时刻栈内元素个数的时间为\$x\$□那么在\$x + 1\$时刻一定是一个插入操作且是\$t\$时刻的栈顶。

想的时候没有注意到\$t\$都不同,但是好像也可以做,大致方法和上述一致,需要处理的是在\$t\$时刻先删除后入栈的情况,这两种应该分开维护,因为先删除是删除的前面入栈的元素,后删除删除的是时刻\$t\$入栈的元素。

### 1. 纸牌

### 题意:

桌上有一叠共\$n\$张牌,从顶向下标号为\$1\sim n\$□

这一叠牌做k,  $k \leq 10^{18}$ \$次操作,其中第i\$次操作会将牌堆顶的牌放在牌堆中的某个位置,从而满足这张牌成为自顶向下第i(i-1)% (i-1) + 2i8张牌。求i8次操作以后这叠牌自顶向下的编号情况。

#### 题解:

假设\$k \leq (n - 1)\$□那么我们可以这样模拟:

观察到如果此时顶部纸牌插入到了\$x\$这张牌的后面,那么下次要插入的纸牌位置就是nxt[nxt[x]]□用链表模拟即可。

当\$k\$很大时,假设\$n-1\$操作后第\$i\$张牌编号为\$p[i]\$□那么\$2(n-1)\$次操作后,第\$i\$张牌编号为\$p[p[i]]\$□部分可以倍增或者把排列\$p\$拆成循环来处理。

剩下的\$k % (n - 1)\$次直接模拟即可。

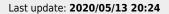
Last update: 2020/05/13 2020-2021:teams:alchemist:mountvoom:training1 https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:alchemist:mountvoom:training1&rev=1589372668 20:24

# J. 斐波那契和

From:

https://wiki.cvbbacm.com/ - CVBB ACM Team

Permanent link: https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:alchemist:mountvoom:training1&rev=1589372668





Printed on 2025/11/29 16:46 https://wiki.cvbbacm.com/