# DP

qwqUwU

gdsy

1782151568@qq.com

2024 年 3 月 31 日

# 吐槽一下

原来要求讲的是概率 DP，但我寻思又有概率又有 DP 的就那么一小撮东西，干脆讲 DP 好了。
本人水平有限，做题很少，今天挑的都是大原题。如果碰到做过的题可以自行改为十分钟优质睡眠 qwq 。

# 概率

还是讲一下吧。

# 概率

还是讲一下吧。

## 古典概率

$\frac{\text{目标空间大小}}{\text{样本空间大小}}$ 。
OI 中最常用的概率，因为样本空间有限，所以本质就是计数。

## 无穷概率

样本空间无穷大的情况，分为连续和不连续两种
比如几何图形上选一点，一直扔硬币直到投出正面的次数等等。
由于样本空间无穷大，所以任何一个非无穷大的目标空间的概率都是 $0$ 。

## 期望

$\sum$ 事件的贡献 $\times$ 事件的概率 。
两种概率都有定义。
一般用 $E(X)$ 表示 $X$ 这个随机量的期望。

## 一些性质

$E(aX+bY)=aE(X)+bE(Y)$
$E(XY)=E(X)E(Y)$ 这里要求 $X,Y$ 是相互独立的变量。

# 概率

## [THUPC 2024 初赛] 前缀和

一个序列 $x_i$ ，$x_i$ 有 $(1-p)^{j-1}p$ 的概率为 $j(j \geq 1)$ 问期望有多少个点的前缀和在 $[l,r]$ 中。

# 概率

## [THUPC 2024 初赛] 前缀和

一个序列 $x_i$ ， $x_i$ 有 $(1-p)^{j-1}p$ 的概率为 $j(j \geq 1)$ 问期望有多少个点的前缀和在 $[l, r]$ 中。

$p(r-l+1)$
考虑一个无穷序列，每个数有 $p$ 的概率为 $1$ 。则相邻两个 $1$ 的距离期望就是 $x$ 。于是 $i$ 处前缀和就是第 $i$ 个 $1$ 的位置。相当于问 $[l, r]$ 有多少个 $1$ 。

# 概率

**[HAOI2015] 按位或**

刚开始你有一个数字 $0$。每一秒钟你会随机选择一个 $[0, 2^n - 1]$ 的数字，与你手上的数字进行或。选择数字 $i$ 的概率是 $p_i$。保证 $0 \le p_i \le 1$，$\sum p_i = 1$ 。问期望多少秒后，你手上的数字变成 $2^n - 1$。$n \le 20$

# 概率

## [HAOI2015] 按位或

刚开始你有一个数字 $0$。每一秒钟你会随机选择一个 $[0, 2^n - 1]$ 的数字，与你手上的数字进行或。选择数字 $i$ 的概率是 $p_i$。保证 $0 \le p_i \le 1$，$\sum p_i = 1$ 。问期望多少秒后，你手上的数字变成 $2^n - 1$。$n \le 20$

经典结论：min-max 容斥在期望意义下也成立。每一位拆开，相当于求每位变成 $1$ 的时间的 $\max$ ，转成对每个非空子集求时间 $\min$ 。发现 $\min$ 是好求的，相当于选到这些位中任意一位是 $1$，前缀和一下。

# 概率

## ARC132E Paw

$n$ 个位置排一排，每个位置上有 $<,>$ 或者空。每次等概率随机一个空点，随一个方向，往那个方向走直到边界或空点。将走过的点附上这次的方向（起点覆盖，终点还是空）。没有空点时停止。问 $<$ 的个数的期望。$O(n)$

# 概率

## ARC132E Paw

$n$ 个位置排一排，每个位置上有 $<,>$ 或者空。每次等概率随机一个空点，随一个方向，往那个方向走直到边界或空点。将走过的点附上这次的方向（起点覆盖，终点还是空)。没有空点时停止。问 $<$ 的个数的期望。

考虑最终状态，一定是相邻两个空点，中间不变，左边全左，右边全右。于是 DP 全左的概率。考虑确定操作顺序，对于后缀 $\max$ 只能选左，其余随便。加入最左边的点，$f_i = (\frac{i-1}{i} + \frac{1}{2i})f_{i-1}$ 。

# 概率

### ARC154E Reverse and Inversion

长为 $n$ 的排列 $P$, 重复 $m$ 次操作。每次选 $1 \le i \le j \le n$ ，翻转 $P_i \sim P_j$ 。求所有方案的 $\sum_{i<j}[P_i > P_j](j-i)$ 的和。

# 概率

## ARC154E Reverse and Inversion

长为 $n$ 的排列 $P$, 重复 $m$ 次操作。每次选 $1 \le i \le j \le n$ ，翻转 $P_i \sim P_j$ 。求所有方案的 $\sum_{i<j}[P_i > P_j](j-i)$ 的和。

考虑除一下变期望。记一个 $g(i) = \sum_{j<i}[p_j > p_i] - \sum j > i[p_j < p_i]$ ，贡献是 $ig(i)$ 。不难发现 $g(i) = i - p_i$ 。发现对于一个 $i$ ，到 $j$ 和 $n-j+1$ 的概率一样（因为方案数一样）。于是变成操作过变成 $\dfrac{n+1}{2}$ 没操作不变。

# 概率

**[CTSC2006] 歌唱王国**

猴子打字。问打出给出字符串的时间的期望。

## [CTSC2006] 歌唱王国

猴子打字。问打出给出字符串的时间的期望。$n \le 10^5$

结论是所有 border 的 $|\Sigma|^{len}$ 之和。
有一个很有意思的证明。

# 概率

考虑一个赌场，庄家对这只猴子设庄。
赌徒可以在每个时刻下注猴子下个字符是什么。猜中了获得 $|\Sigma|$ 的倍率，猜错不返还。
考虑每个时刻有一个赌徒带一元进入赌场，第 $i$ 次赌赌目标串的第 $i$ 位。
当猴子打出目标字符串时赌博结束。
考虑这个游戏的期望资金。
首先游戏是公平的，所以所有赌徒手中的前就是初始的钱。
每个时刻一个赌徒带一元，因此初始前就是期望时间。
对于目标串的所有 border，赌了 $|len|$ 次的赌徒还存在，手中有 $|\Sigma|^{len}|$ 元。
其他赌徒手上都没钱。
于是钱总数就是 $|\Sigma|^{len}$ 之和。

### CF1349D Slime and Biscuits

$n$ 个人，$i$ 有 $a_i$ 个饼干。每次随机一个饼干，随机给除原所有者的 $n-1$ 个人。求使得一个人获得所有饼干的期望步数。$n \le 10^5$

### CF1349D Slime and Biscuits

$n$ 个人，$i$ 有 $a_i$ 个饼干。每次随机一个饼干，随机给除原所有者的 $n-1$ 个人。求使得一个人获得所有饼干的期望步数。$n \le 10^5$

鞅-停时定理。

考虑能否设计一个势能函数 $f(i)$ 满足 $\sum f(a_i)$ 每一次操作后期望增加 $1$ 。

$$\sum_{i=1}^{n} f(a_i) + 1 = \sum_{i=1}^{n} \frac{a_i}{s} (f(a_i - 1) + \sum_{j \neq i} \frac{1}{n-1} f(a_j + 1) + \frac{n-2}{n-1} f(a_j))$$

注意我们只要构造，于是尝试对每个 $i$ 独立开。
整理可得：

$$f(i+1) = \frac{(n-2)i + s}{s-i} f(i) + \frac{i(n-1)}{s-i} (1 - f(i-1))$$

于是算出始状态和末状态的势能，作差即可。

# DP

也没啥好讲的，主要是题。

# DP

也没啥好讲的，主要是题。严格意义上讲，转移式子里没 $\min, \max$ 的都不算 DP。
这里就分成最优化和计数两种 DP 来讲了。

# DP

也没啥好讲的，主要是题。严格意义上讲，转移式子里没 $\min, \max$ 的都不算 DP。
这里就分成最优化和计数两种 DP 来讲了。
一个记号：记 xD/yD 表示 $x$ 维状态，$y$ 维转移的 DP。

# 决策单调性

最优化 DP 经常用的一个方法。

# 四边形不等式 & Monge 矩阵

## Monge 矩阵

对于所有 $i1 < i2$ 和 $j1 < j2$ 都满足四边形不等式
$A_{i1,j1} + A_{i2,j2} \leq A_{i2,j1} + A_{i1,j2}$ ，即交叉优于包含的矩阵 $A$ 称为 Monge 矩阵。

## 判定

只要对于所有 $i, j$ 有 $A_{i,j} + Ai + 1, j + 1 \leq A_{i,j+1} + A_{i+1,j}$ 即可。

## Monge 矩阵

对于所有 $i1 < i2$ 和 $j1 < j2$ 都满足四边形不等式 $A_{i1,j1} + A_{i2,j2} \le A_{i2,j1} + A_{i1,j2}$ ，即交叉优于包含的矩阵 $A$ 称为 Monge 矩阵。

## 判定

只要对于所有 $i, j$ 有 $A_{i,j} + Ai + 1, j + 1 \le A_{i,j+1} + A_{i+1,j}$ 即可。

## 决策单调性

存在 $j_1 \le j_2 \le \cdots \le j_m$ 使得 $A_{i,j_i}$ 是 $A_i$ 这一行的最小值。

## 交叉偏序性

对于 $i, j$ 若有 $k$ 满足 $A_{k,i} \le A_{k,j}$ ，则 $\forall l < k$ 有 $A_{l,i} \le A_{l,j}$ 。

大概吧，我贺的，错了轻点骂 qwq。

### 分治

处理离线的。
$solve(l,r,L,R)$ 表示 $[l,r]$ 的最优决策点在 $[L,R]$ 之中。
求出 $mid$ 的最优决策点，递归两边。
复杂度 $O(n\log nf(n))$ 。

# 四边形不等式 & Monge 矩阵

## 分治

处理离线的。
$solve(l, r, L, R)$ 表示 $[l, r]$ 的最优决策点在 $[L, R]$ 之中。
求出 $mid$ 的最优决策点，递归两边。
复杂度 $O(n \log n f(n))$ 。

## 二分栈

可以在线。
维护一个栈，每个元素是 $(i, l, r)$ 表示当前 $i$ 是 $[l, r]$ 的最优决策点。
每次加入 $i + 1$ ，如果栈顶不可能比 $i + 1$ 优就弹栈。如果能就二分更新最优区间。
复杂度 $O(n \log n f(n))$ 。

## CDQ

在线的。
$CDQ(l,r)$ 表示在求 $[l,r]$ 的 DP 值。
流程是 $CDQ(l,mid), solve(mid+1,r,l,mid), CDQ(mid+1,r)$。

**[NOI2009] 诗人小 G**

$n$ 个数，划分，一段的代价是 $|S-L|^P$ 。最小化代价和。

**[NOI2009] 诗人小 G**

$n$ 个数，划分，一段的代价是 $|S-L|^P$ 。最小化代价和。

显然次幂满足四边形不等式。二分栈一下。

**[POI2011 R1] 避雷针**

对每个 $i$ 求 $h_j - h_i + \sqrt{|i-j|}$ 的最小值。

# 四边形不等式 & Monge 矩阵

**[POI2011 R1] 避雷针**

对每个 $i$ 求 $h_j - h_i + \sqrt{|i-j|}$ 的最小值。

拆成左右，分别单调，离线转移。

# 四边形不等式 & Monge 矩阵

**smawk 算法**

可以 $O(nf(n))$ 的做离线。

**smawk 算法**

可以 $O(nf(n))$ 的做离线。

考虑将一个 Monge 矩阵无用列删去。逐列比较这列是否存在能成为最小值。
用交叉偏序性可以 $O(n)$ 做到这个删。
求所有最优决策点，可以将偶数行拿出来，做一次删，然后递归。
然后奇数行就会被卡住上下，总共 $O(n)$ 可以求出。
于是 $T(n) = T(\frac{n}{2}) + O(n) = O(n)$ 。

# 凸优化

对于较为简单的转移，可能会发现具有凸性，可以套用凸优化。

# 凸优化

## [APIO2010] 特别行动队

$f_i = \max_j f_j + A(s_i - s_j)^2 + B(s_i - s_j) + C$

平移系数，就一项 $-2As_is_j$ 是乘积。
每个 $j$ 都是一项一次函数。李超树可以直接维护。
但显然斜率和查询点都单调，单调队列维护凸包即可。

**[IOI2000] 邮局加强版**

数轴上 $n$ 个白点，要求点 $m$ 个黑点，使得每个白点到最近的黑点距离和最小。
$n \leq 5 \times 10^5$

# 凸优化

## [IOI2000] 邮局加强版

数轴上 $n$ 个白点，要求点 $m$ 个黑点，使得每个白点到最近的黑点距离和最小。
$n \le 5 \times 10^5$

考虑暴力。$dp(i,j)$ 表示前 $i$ 个点白点 $j$ 个黑点的最小距离和。
枚举 $dp(k,j-1)$ 转移。
这是 2D/1D 的，但发现既可以 WQS 二分又可以斜率优化，于是 $O(n\log n)$ 。

## [NOI2019] 回家路线

图。$(u,v,p,q)$ 表示 $u \to v$ 的单项路径，$p$ 时刻出发，$q$ 时刻到。
从 $1$ 到 $n$ 。
可以在一个点等待，等待 $t(t \geq 0)$ 代价 $At^2 + B^t + C$ 。
若 $z$ 时刻到，再支付 $z$ 的代价。
最小代价。
$n \leq 10^5, m \leq 2 \times 10^5$

# 凸优化

## [NOI2019] 回家路线

图。$(u, v, p, q)$ 表示 $u \to v$ 的单项路径，$p$ 时刻出发，$q$ 时刻到。
从 $1$ 到 $n$ 。
可以在一个点等待，等待 $t(t \geq 0)$ 代价 $At^2 + B^t + C$ 。
若 $z$ 时刻到，再支付 $z$ 的代价。
最小代价。
$n \leq 10^5, m \leq 2 \times 10^5$

按照 $p$ 从小到大增广，每个点维护一个单调队列。加入直线就延迟加入一下。

# 凸优化

**Pjudge Round 12 C**

划分，长度不超过 $k$ ，代价是 $\mathrm{mex} \times S$ ，最大化代价和。

# 凸优化

## Pjudge Round 12 C

划分，长度不超过 $k$ ，代价是 $\mathrm{mex} \times S$ ，最大化代价和。

扫描线的过程，倒着是 $O(n)$ 次覆盖，正着就是 $O(n)$ 次分裂。
维护连续段，考虑一个段里的最大值，树套李超解决。
考虑询问，每个连续段的最大值再放到一个树套李超解决。
注意左端点可能在一个段中间，要单独算一下。

# 计数

前缀和，后缀和，DS 优化，状态剪枝。这些是较为通用的。
NTT。今天不讲。

# 计数

前缀和，后缀和，DS 优化，状态剪枝。这些是较为通用的。
NTT。今天不讲。
计数 DP 的难点还是在设计状态与转移。

# 例题

一些题。

### [AGC058B] Adjacent Chmax

一个排列，可以做任意次（可以不做）操作。
每次 $P_i, P_{i+1} \leftarrow \max(P_i, P_{i+1})$ 。问能得到多少种不同的结果。$n \leq 5000$

## [AGC058B] Adjacent Chmax

一个排列，可以做任意次（可以不做）操作。
每次 $P_i, P_{i+1} \leftarrow \max(P_i, P_{i+1})$ 。问能得到多少种不同的结果。$n \leq 5000$

发现最终 $i$ 能覆盖的位置是以 $p_i$ 为最大值的区间。
$f(i,j)$ 表示前 $i$ 个数，覆盖了 $1 \sim j$ 的方案数。
$f(i,j) = f(i-1,j) + [l_i \leq j \leq r_i] f(i,j-1)$ 。

# UwU

## [USACO16OPEN] 262144 P

一个 $1 \sim 40$ 的序列，每次可以合并相邻两个 $i$ 合并成 $i+1$ 。
问最大能搓出多少。
$2 \leq n \leq 2^{18}$ 。

## [USACO16OPEN] 262144 P

一个 $1 \sim 40$ 的序列，每次可以合并相邻两个 $i$ 合并成 $i+1$ 。
问最大能搓出多少。
$2 \leq n \leq 2^{18}$ 。

显然区间 DP。但 $n^2$ 。
考虑把答案拿到状态里，$f(i,s)$ 表示能搓出来的最小右端点。
转移就 $f(i,j) = f(f(i,j-1),j-1)$ 。

### [ARC146E] Simple Speed

有多少序列 $b$ ，满足相邻两项差为 $1$，并且 $i$ 出现恰好 $a_i$ 次。
$n, a_i \leq 2 \times 10^5$

## [ARC146E] Simple Speed

有多少序列 $b$ ，满足相邻两项差为 $1$，并且 $i$ 出现恰好 $a_i$ 次。
$n, a_i \leq 2 \times 10^5$

考虑插入 DP。记一个 $f(i, j, 0/1/2)$ 表示前 $i$ 个数，有 $j$ 个空位，左右能不能插。
发现这个 DP 的 $j$ 这一位如果不考虑 $0/1/2$ 只有唯一一种转移。
考虑上就只会有 $-2, 0, +2$ 几种情况。总状态是 $O(n)$ 的。开 map 记一下。

### CF1608F MEX counting

有多少个长为 $n$ 的序列 $a$ 满足：
$a_i \in [0, n]$
$|\operatorname{mex}(a_1, a_2, \cdots, a_i) - b_i| \le k$
$n \le 2000, k \le 50$

### CF1608F MEX counting

有多少个长为 $n$ 的序列 $a$ 满足:
$a_i \in [0, n]$
$|\operatorname{mex}(a_1, a_2, \cdots, a_i) - b_i| \le k$
$n \le 2000, k \le 50$

记 $dp(i, j, cnt)$ 表示前 $i$ 个数，$mex$ 为 $j$，有 $cnt$ 个大于 $j$ 的数的方案数。
注意这里考虑延后转移，大于 $cnt$ 的数只记个数不计具体值，在 $mex$ 变大时才确定具体值。
想好状态后转移是好转移的。
记得前缀和优化。

### [湖北省选模拟 2024] 沉玉谷

一个序列，每次选择一段值相同的区间，将这段删除，把后面向前平移。问删空这个序列的方案数。
$1 \le a_i \le n \le 50$ 。

## [湖北省选模拟 2024] 沉玉谷

一个序列，每次选择一段值相同的区间，将这段删除，把后面向前平移。
问删空这个序列的方案数。
$1 \le a_i \le n \le 50$ 。

区间 DP，同时记一个辅助转移 $r$ 这次的删除。
$n^5$ 和 $n^6$ 都可以过。

# The End