概率与期望

liuhengxi

February 21, 2024

1. 基础
   - 概率的定义及性质
   - 期望的定义及性质
2. 计算一个结构的信息
   - 可拆
   - 不可拆
3. 状态之间转移
   - DAG 上转移
   - 双向链上转移
   - 树上转移
   - 缩小状态集合
   - 终止状态之间有对称性
4. 带有最优化和自依赖的问题

1. 基础
   - 概率的定义及性质
   - 期望的定义及性质
2. 计算一个结构的信息
   - 可拆
   - 不可拆
3. 状态之间转移
   - DAG 上转移
   - 双向链上转移
   - 树上转移
   - 缩小状态集合
   - 终止状态之间有对称性
4. 带有最优化和自依赖的问题

Definition 1 (样本空间)

样本空间 Ω 是由所有结果构成的集合。

Definition 2 (事件)

事件是 Ω 的子集。

Definition 1 (样本空间)

样本空间 Ω 是由所有结果构成的集合。

Definition 2 (事件)

事件是 Ω 的子集。

Definition 3 (概率空间)

概率空间 $(\Omega, \mathcal{F}, P)$ 是由

- 样本空间 Ω。
- 由事件构成的集合 $\mathcal{F}$。
- 概率测度 $P : \mathcal{F} \to \mathbb{R}$。

构成的三元组。

Definition 1 (样本空间)

样本空间 Ω 是由所有结果构成的集合。

Definition 2 (事件)

事件是 Ω 的子集。

Definition 3 (概率空间)

概率空间 $(\Omega, \mathcal{F}, P)$ 是由

- 样本空间 Ω。
- 由事件构成的集合 $\mathcal{F}$。
- 概率测度 $P : \mathcal{F} \to \mathbb{R}$。

构成的三元组。

通常认为 $\mathcal{F}$ 必须是 σ-代数。

Axiom 1 (非负性)

$$\forall A \in \mathcal{F}, P(A) \geq 0$$

Axiom 1 (非负性)

$$\forall A \in \mathcal{F}, P(A) \geq 0$$

Axiom 2 (归一化)

$$P(\Omega) = 1$$

Axiom 1 (非负性)

$$\forall A \in \mathcal{F}, P(A) \geq 0$$

Axiom 2 (归一化)

$$P(\Omega) = 1$$

Axiom 3 (可加性)

对于任意两两不相交的事件列 $A_1, A_2, \cdots$，有

$$P\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} P(A_i)$$

Definition 4 (事件的独立性)

事件 A, B 独立当且仅当 $P(A \cap B) = P(A)P(B)$。[a]

Definition 4 (事件的独立性)

事件 A, B 独立当且仅当 $P(A \cap B) = P(A)P(B)$。[a]
事件 $A_1, A_2, \cdots A_n$ 相互独立当且仅当

$$\forall S \subseteq \{1, 2, \cdots, n\}, P\left(\bigcap_{i \in S} A_i\right) = \prod_{i \in S} P(A_i)$$

[a] $A \cap B$ 表示 A 和 B 都发生，$A \cup B$ 表示 A 和 B 至少有一个发生。

Definition 4 (事件的独立性)

事件 A, B 独立当且仅当 $P(A \cap B) = P(A)P(B)$。[a]
事件 $A_1, A_2, \cdots A_n$ 相互独立当且仅当

$$\forall S \subseteq \{1, 2, \cdots, n\}, P\left(\bigcap_{i \in S} A_i\right) = \prod_{i \in S} P(A_i)$$

[a] $A \cap B$ 表示 A 和 B 都发生，$A \cup B$ 表示 A 和 B 至少有一个发生。

Example 5 (多个事件两两独立不一定相互独立)

令 $\Omega = \{\omega_0, \omega_1, \omega_2, \omega_3\}, P(\omega_0) = P(\omega_1) = P(\omega_2) = P(\omega_3) = \frac{1}{4}$,
$A_1 = \{\omega_2, \omega_3\}, A_2 = \{\omega_3, \omega_1\}, A_3 = \{\omega_1, \omega_2\}$。
则 A_1, A_2, A_3 两两独立而不相互独立。

Property 6

事件 A, B 独立

$$
\begin{aligned}
&\iff P(A \cap B) = P(A)P(B) \\
&\iff P(\overline{A} \cap B) = P(\overline{A})P(B) \\
&\iff P(A \cap \overline{B}) = P(A)P(\overline{B}) \\
&\iff P(\overline{A} \cap \overline{B}) = P(\overline{A})P(\overline{B})
\end{aligned}
$$

Property 6

事件 A, B 独立

$$
\begin{aligned}
&\iff P(A\cap B) = P(A)P(B) \\
&\iff P(\overline{A}\cap B) = P(\overline{A})P(B) \\
&\iff P(A\cap \overline{B}) = P(A)P(\overline{B}) \\
&\iff P(\overline{A}\cap \overline{B}) = P(\overline{A})P(\overline{B})
\end{aligned}
$$

Property 7

事件 $A_1, A_2, \cdots A_n$ 相互独立当且仅当

$$\forall S \subseteq \{1, 2, \cdots, n\}, P\left(\bigcap_i \begin{cases} A_i & i \in S \\ \overline{A_i} & i \notin S \end{cases}\right) = \prod_{i\in S} P\left(\begin{cases} A_i & i \in S \\ \overline{A_i} & i \notin S \end{cases}\right)$$

Property 6

事件 A, B 独立

$$
\begin{aligned}
&\iff P(A \cap B) = P(A)P(B) \\
&\iff P(\overline{A} \cap B) = P(\overline{A})P(B) \\
&\iff P(A \cap \overline{B}) = P(A)P(\overline{B}) \\
&\iff P(\overline{A} \cap \overline{B}) = P(\overline{A})P(\overline{B})
\end{aligned}
$$

Property 7

事件 $A_1, A_2, \cdots A_n$ 相互独立当且仅当

$$\forall S \subseteq \{1, 2, \cdots, n\}, P\left(\bigcap_i \begin{cases} A_i & i \in S \\ \overline{A_i} & i \notin S \end{cases}\right) = \prod_{i \in S} P\left(\begin{cases} A_i & i \in S \\ \overline{A_i} & i \notin S \end{cases}\right)$$

容易发现这是定义做了 *FWT* 的结果。

Definition 8 (条件概率)

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \ (P(B) \neq 0)$$

Definition 8 (条件概率)

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \ (P(B) \neq 0)$$

Property 9

若事件 A, B 独立，且 $P(B) \neq 0$，则 $P(A|B) = P(A)$。

Theorem 10

$$
\begin{aligned}
&P(A_1 \cap A_2 \cap \cdots \cap A_n) \\
=&P(A_1)P(A_2|A_1)P(A_3|A_1 \cap A_2)\cdots P(A_n|A_1 \cap A_2 \cap \cdots \cap A_{n-1})
\end{aligned}
$$

Theorem 10

$$
\begin{aligned}
&P(A_1 \cap A_2 \cap \cdots \cap A_n) \\
=&P(A_1)P(A_2|A_1)P(A_3|A_1 \cap A_2) \cdots P(A_n|A_1 \cap A_2 \cap \cdots \cap A_{n-1})
\end{aligned}
$$

Theorem 11

若 $\forall i \neq j, B_i \cap B_j = \varnothing$ 且 $\bigcup_i B_i = \Omega$，则

$$P(A) = \sum_i P(A|B_i)P(B_i)$$

Theorem 10

$$
\begin{aligned}
&P(A_1 \cap A_2 \cap \cdots \cap A_n) \\
=&P(A_1)P(A_2|A_1)P(A_3|A_1 \cap A_2) \cdots P(A_n|A_1 \cap A_2 \cap \cdots \cap A_{n-1})
\end{aligned}
$$

Theorem 11

若 $\forall i \neq j, B_i \cap B_j = \varnothing$ 且 $\bigcup_i B_i = \Omega$，则

$$P(A) = \sum_i P(A|B_i)P(B_i)$$

Theorem 12

$$P(A|B) = \frac{P(A)P(B|A)}{P(B)}$$

Definition 13 (随机变量)

随机变量 $X:\Omega \to \mathbb{R}$ 是定义在样本空间上的实函数。

Definition 13 (随机变量)

随机变量 $X:\Omega\to\mathbb{R}$ 是定义在样本空间上的实函数。

Definition 14 (期望)

$$\mathrm{E}\,X=\int_\Omega X\,\mathrm{d}\,P$$

Definition 13 (随机变量)

随机变量 $X:\Omega\to\mathbb{R}$ 是定义在样本空间上的实函数。

Definition 14 (期望)

$$\mathrm{E}\,X=\int_{\Omega}X\mathrm{d}\,P$$

对于 Ω 有限（可列）的情况：

$$\mathrm{E}\,X=\sum_{\omega\in\Omega}X(\omega)P(\omega)$$

Definition 13 (随机变量)

随机变量 $X:\Omega\to\mathbb{R}$ 是定义在样本空间上的实函数。

Definition 14 (期望)

$$\mathrm{E}\,X=\int_\Omega X\mathrm{d}\,P$$

对于 Ω 有限（可列）的情况：

$$\mathrm{E}\,X=\sum_{\omega\in\Omega}X(\omega)P(\omega)$$

对于 X 的值有限（可列）的情况：

$$\mathrm{E}\,X=\sum_{x}x\cdot P(X=x)$$

Definition 13 (随机变量)

随机变量 $X:\Omega\to\mathbb{R}$ 是定义在样本空间上的实函数。

Definition 14 (期望)

$$\mathrm{E}\,X=\int_\Omega X\mathrm{d}\,P$$

对于 Ω 有限（可列）的情况：

$$\mathrm{E}\,X=\sum_{\omega\in\Omega}X(\omega)P(\omega)$$

对于 X 的值有限（可列）的情况：

$$\mathrm{E}\,X=\sum_x x\cdot P(X=x)$$

Example 15

抛掷一个骰子出现 $1,2,3,4,5,6$ 的概率分别是 $\frac{1}{2211},\frac{10}{2211},\frac{100}{2211},\frac{1000}{2211},\frac{100}{2211},\frac{1000}{2211}$。则点数的 φ 函数的期望是 $\frac{1}{2211}\times1+\frac{10}{2211}\times1+\frac{100}{2211}\times2+\frac{1000}{2211}\times2+\frac{100}{2211}\times4+\frac{1000}{2211}\times2=\frac{1537}{737}$。

Definition 16 (随机变量的独立性)

$X_1, \cdots, X_n$ 独立当且仅当
$\forall x_1, \cdots, x_n, F_{X_1,\cdots,X_n}(x_1, \cdots, x_n) = F_{X_1}(x_1) \cdot \cdots \cdot F_{X_n}(x_n)$。[a]

[a] $F_X(x) = P(X \leq x)$ 称为 X 的（累积）分布函数。
$F_{X_1,\cdots,X_n}(x_1, \cdots, x_n) = P(X_1 \leq x_1 \wedge \cdots \wedge X_n \leq x_n)$

Property 17 (线性性)

$$\mathrm{E}(X+Y) = \mathrm{E}\,X + \mathrm{E}\,Y$$

$$\mathrm{E}(aX) = a\,\mathrm{E}\,X$$

一般地,

$$\mathrm{E}\left(\sum_{i=1}^{n} a_i X_i\right) = \sum_{i=1}^{n} a_i\,\mathrm{E}\,X_i$$

Property 17 (线性性)

$$\mathrm{E}(X+Y)=\mathrm{E}\,X+\mathrm{E}\,Y$$

$$\mathrm{E}(aX)=a\,\mathrm{E}\,X$$

一般地，

$$\mathrm{E}\left(\sum_{i=1}^{n}a_iX_i\right)=\sum_{i=1}^{n}a_i\,\mathrm{E}\,X_i$$

Property 18

若 X,Y 独立，$\mathrm{E}(XY)=\mathrm{E}\,X\cdot\mathrm{E}\,Y$。
一般地，若 $X_1,\cdots,X_n$ 相互独立，$\mathrm{E}(\bigcap X_i)=\prod\mathrm{E}\,X_i$。

Property 17 (线性性)

$$\mathrm{E}(X+Y)=\mathrm{E}\,X+\mathrm{E}\,Y$$

$$\mathrm{E}(aX)=a\,\mathrm{E}\,X$$

一般地，

$$\mathrm{E}\left(\sum_{i=1}^{n}a_iX_i\right)=\sum_{i=1}^{n}a_i\,\mathrm{E}\,X_i$$

Property 18

若 X,Y 独立，$\mathrm{E}(XY)=\mathrm{E}\,X\cdot\mathrm{E}\,Y$。
一般地，若 $X_1,\cdots,X_n$ 相互独立，$\mathrm{E}(\bigcap X_i)=\prod\mathrm{E}\,X_i$。

Property 19

对于值一定在 $\mathbb{N}$ 中的随机变量 X，有 $\mathrm{E}\,X=\sum_{i=1}^{\infty}P(X\geq i)$。

1 基础
- 概率的定义及性质
- 期望的定义及性质

2 计算一个结构的信息
- 可拆
- 不可拆

3 状态之间转移
- DAG 上转移
- 双向链上转移
- 树上转移
- 缩小状态集合
- 终止状态之间有对称性

4 带有最优化和自依赖的问题

Problem 20 (LOJ 6901. 「THUPC 2024 初赛」前缀和)

小兰很喜欢随机数。
TA 首先选定了一个实数 $0 < p < 1$，然后生成了 n 个随机数 $x_1, \cdots, x_n$，每个数是独立按照如下方式生成的：

- x_i 有 p 的概率是 1，有 $(1-p)p$ 的概率是 2，有 $(1-p)^2p$ 的概率是 3，以此类推。

生成完这些随机数之后，小艾对这个数列求了前缀和，得到了数列 $y_1, \cdots, y_n$。
给定 $1 \le l \le r \le n$，小兰想知道，期望有多少 y_i 落在 $[l, r]$ 内？
$1 \le l \le r \le n \le 10^9$，$p$ 的位数不超过 6。
1s, 512MiB。

题目与题解来自
`https://github.com/ckw20/thupc2024_pre_public`。

Solution

问题转化：
考虑有无穷多个灯，编号从 1 开始一字排开，每个灯独立有 p 的概率亮起，$1-p$ 的概率是灭的。
我们从左往右数，第一次找到一个灯的编号，这就是 x_i 的分布，所以序列 $\{y_i\}$ 就是第 i 个亮起的灯的位置。

Solution

问题转化：
考虑有无穷多个灯，编号从 1 开始一字排开，每个灯独立有 p 的概率亮起，$1-p$ 的概率是灭的。
我们从左往右数，第一次找到一个灯的编号，这就是 x_i 的分布，所以序列 $\{y_i\}$ 就是第 i 个亮起的灯的位置。
所以问题就是在问：$[l,r]$ 之间，期望有多少亮起的灯？
答案就是 $(r-l+1)p$。

Solution (不需要观察的方法)

容易发现 $n \geq r$ 时，可以视为 n 足够大。
设数列 y 中出现 i 的概率为 f_i，令 $f_0 = 1$。

Solution (不需要观察的方法)

容易发现 $n \geq r$ 时，可以视为 n 足够大。
设数列 y 中出现 i 的概率为 f_i，令 $f_0 = 1$。
对于 $i \geq 1$，有 $f_i = \sum_{j=0}^{i-1} f_j \cdot p(1-p)^{i-j-1}$。
计算前几项可以发现 $f_i = p$。
容易用数学归纳法证明。

Solution (不需要观察的方法)

容易发现 $n \geq r$ 时，可以视为 n 足够大。
设数列 y 中出现 i 的概率为 f_i，令 $f_0 = 1$。
对于 $i \geq 1$，有 $f_i = \sum_{j=0}^{i-1} f_j \cdot p(1-p)^{i-j-1}$。
计算前几项可以发现 $f_i = p$。
容易用数学归纳法证明。
答案是 $\sum_{i=l}^{r} f_i = (r-l+1)p$。

Problem 21 (Codeforces 559D. Randomizer)

给定一个包含 n 个点的凸多边形，等概率选取顶点集中大于等于三的子集形成一个新凸多边形，求新凸多边形内部（不包含边界）整点个数的期望值。
绝对或相对误差不超过 10^{-9}。
$3 \le n \le 10^5$。
$2.5\mathrm{s}, 256\mathtt{MiB}$。

Solution

凸多边形内的整点个数可以用 *Pick* 定理计算。

Solution

凸多边形内的整点个数可以用 *Pick* 定理计算。
每个新凸多边形内的整点个数用原凸多边形内的整点个数减去“收缩”少的整点个数。

Solution

凸多边形内的整点个数可以用 *Pick* 定理计算。
每个新凸多边形内的整点个数用原凸多边形内的整点个数减去“收缩”少的整点个数。
“收缩”(i,j) 代表点 i,j 都被选中，但 $i+1,\cdots,j-1$ 都未被选中的情况。

Solution

凸多边形内的整点个数可以用 *Pick* 定理计算。
每个新凸多边形内的整点个数用原凸多边形内的整点个数减去“收缩”少的整点个数。
“收缩” (i,j) 代表点 i,j 都被选中，但 $i+1,\cdots,j-1$ 都未被选中的情况。
根据期望的线性性，答案就是原凸多边形内的整点个数减去每种“收缩”少的整点个数乘上其出现概率之和。

Solution

凸多边形内的整点个数可以用 *Pick* 定理计算。
每个新凸多边形内的整点个数用原凸多边形内的整点个数减去“收缩”少的整点个数。
“收缩”(i,j) 代表点 i,j 都被选中，但 $i+1,\cdots,j-1$ 都未被选中的情况。
根据期望的线性性，答案就是原凸多边形内的整点个数减去每种“收缩”少的整点个数乘上其出现概率之和。
注意到允许 10^{-9} 的误差，对于 $j-i\geq 60$ 的“收缩”(i,j)，由于其出现概率极低，可以忽略其贡献。
时间复杂度 $O(n\log V\log\frac{1}{\varepsilon})$。

Problem 22 (Codeforces 494C. Helping People)

有一个长为 n 的数列，初始时为 $a_{1\ldots n}$。
给你 q 个操作，第 i 个操作将 $[l_i, r_i]$ 内的数全部加一，有 p_i 的概率被执行（相互独立）。
保证每对区间不交或包含。
求操作完成后数列的最大值的期望。
$1 \le n \le 10^5, 1 \le q \le 5000, 0 \le a_i \le 10^9$。
$2\mathtt{s}, 512\mathtt{MiB}$。

Solution

显然，数列的最大值 $\in \mathbb{N}$，可以利用 *Property 19*，转化为计算数列的最大值 $\geq x$ 的概率。

Solution

显然，数列的最大值 $\in \mathbb{N}$，可以利用 *Property 19*，转化为计算数列的最大值 $\geq x$ 的概率。
将所有 a_i 减去 $\max a_i - q$，数列的最大值 $\in [0, 2q] \cap \mathbb{N}$，只需对于 $[0, 2q]$ 中的 x 计算。

Solution

显然，数列的最大值 $\in \mathbb{N}$，可以利用 *Property 19*，转化为计算数列的最大值 $\geq x$ 的概率。
将所有 a_i 减去 $\max a_i - q$，数列的最大值 $\in [0, 2q] \cap \mathbb{N}$，只需对于 $[0, 2q]$ 中的 x 计算。
区间构成树形结构，使用树形 *dp* 不难解决。
时间复杂度 $\mathcal{O}(q^2)$。

Problem 23 (SOJ 1618. 期望)

给出 n 个点，以及任意两个点 i, j 之间存在一条无向边的概率 $p_{i,j}$（相互独立），求图中连通块个数的期望。

$n \le 18$。

$2\mathrm{s}, 256\mathtt{MiB}$。

Solution

利用期望的线性性，转化为计算每个子集是一个连通块的概率。
设 f_i 为子集 i 连通的概率，$g_{i,j}(i\&j=0)$ 为点集 i 与 j 之间没有连边的概率。

Solution

利用期望的线性性，转化为计算每个子集是一个连通块的概率。
设 f_i 为子集 i[a]连通的概率，$g_{i,j}(i\&j=0)$ 为点集 i 与 j 之间没有连边的概率。
有 $f_i = 1 - \sum_{\text{lowbit}(i)\subset j\subsetneq i} f_j \cdot g_{j,i\oplus j}$。（因为每个不连通的方案对应一种 $\text{lowbit}(i)$ 所在的连通块与其他点不连通的方案）
答案为 $\sum_{i=1}^{2^n-1} f_i g_{i,2^n-1-i}$。

Solution

利用期望的线性性，转化为计算每个子集是一个连通块的概率。
设 f_i 为子集 i[a]连通的概率，$g_{i,j}(i\&j=0)$ 为点集 i 与 j 之间没有连边的概率。
有 $f_i = 1 - \sum_{\text{lowbit}(i)\subset j\subsetneq i} f_j \cdot g_{j,i\oplus j}$。（因为每个不连通的方案对应一种 $\text{lowbit}(i)$ 所在的连通块与其他点不连通的方案）
答案为 $\sum_{i=1}^{2^n-1} f_i g_{i,2^n-1-i}$。
改变枚举顺序——计算 $g_{i,j}$ 的实用方法：

- 预处理 j 是单元素集的情况。
- *dp* 转移中，外层枚举 j，内层枚举 i。
- 记忆化当前的 $g_{j,x}$，每次根据预处理的结果 $g_{j,\text{lowbit}(x)}$ 和 $g_{j,x\oplus\text{lowbit}(x)}$ 计算新的 $g_{j,x}$。

时间复杂度 $O(3^n)$。

[a]不区分子集和 bitmask。

Problem 24 (Codeforces 1842H. Tenzing and Random Real Numbers)

有 n 个 $[0,1]$ 范围内的均匀随机变量 $x_{1\dots n}$ 和 m 条限制，每条限制形如 $x_i + x_j \le 1$ 或 $x_i + x_j \ge 1$。请你求出所有限制均满足的概率。对 998244353 取模。

$1 \le n \le 20$，$0 \le m \le n^2 + n$。

$1\mathtt{s}, 1024\mathtt{MiB}$。

Solution

考虑枚举按照 $|x_i - 0.5|$ 从小到大排列 x 的顺序，及 x_i 的正负号（共 $2^n n!$ 种）。

Solution

考虑枚举按照 $|x_i - 0.5|$ 从小到大排列 x 的顺序，及 x_i 的正负号（共 $2^n n!$ 种）。
发现容易确定是否合法，并且 $2^n n!$ 种情况等概率出现。

Solution

考虑枚举按照 $|x_i - 0.5|$ 从小到大排列 x 的顺序，及 x_i 的正负号（共 $2^n n!$ 种）。
发现容易确定是否合法，并且 $2^n n!$ 种情况等概率出现。
所以只需要计算合法的情况数。

Solution

考虑枚举按照 $|x_i - 0.5|$ 从小到大排列 x 的顺序，及 x_i 的正负号（共 $2^n n!$ 种）。
发现容易确定是否合法，并且 $2^n n!$ 种情况等概率出现。
所以只需要计算合法的情况数。
使用状压 *dp*，按照 $|x_i - 0.5|$ 从小到大的顺序加入，*dp* 状态记录每个 x_i 是否已加入。

Solution

考虑枚举按照 $|x_i - 0.5|$ 从小到大排列 x 的顺序，及 x_i 的正负号（共 $2^n n!$ 种）。
发现容易确定是否合法，并且 $2^n n!$ 种情况等概率出现。
所以只需要计算合法的情况数。
使用状压 *dp*，按照 $|x_i - 0.5|$ 从小到大的顺序加入，*dp* 状态记录每个 x_i 是否已加入。
时间复杂度 $\mathcal{O}(2^n n)$。

将在节点（状态）u 时所求的值设为 f_u，若 f_u 只与 u 的出点有关（是 $f_v\ (uv \in E), 1$ 的线性组合），容易按照拓扑序计算。

Problem 25 (Codeforces 1156F. Card Bag)

你有一个装有 n 张卡牌的卡包，卡包中第 i 张卡牌上写有数字 a_i。在接下来的每一个回合，你会从卡包中等概率随机抽出一张卡牌，每一回合抽出的卡牌不会重新放回卡包中。

从第二回合开始，每一回合，你需要对这一回合抽出的卡牌的点数 x 和上一次抽出的卡牌的点数 y 进行比较：

- 如果 $x < y$，游戏失败并结束；
- 如果 $x = y$，游戏胜利并结束；
- 如果 $x > y$，游戏继续。

如果某一次抽牌的时候卡包中没有牌，则游戏失败。

你需要求出游戏胜利的概率，对 998244353 取模。

$2 \le n \le 5000$，$1 \le a_i \le n$。

2s, 512MiB。

Solution

加入一张卡牌 0，第一回合强制抽 0，容易发现答案不变。

Solution

加入一张卡牌 0，第一回合强制抽 0，容易发现答案不变。
容易发现只需记录上一次抽到的卡牌 i，和已经抽过的卡牌个数 j。

Solution

加入一张卡牌 0，第一回合强制抽 0，容易发现答案不变。
容易发现只需记录上一次抽到的卡牌 i，和已经抽过的卡牌个数 j。
以 (i,j) 为状态，转移构成 *DAG*。

Solution

加入一张卡牌 0，第一回合强制抽 0，容易发现答案不变。
容易发现只需记录上一次抽到的卡牌 i，和已经抽过的卡牌个数 j。
以 (i, j) 为状态，转移构成 *DAG*。
按照拓扑序进行 *dp*，时间复杂度 $\mathcal{O}(n^3)$。
使用前缀和优化，容易将时间复杂度减小至 $\mathcal{O}(n^2)$。

Problem 26 (Codeforces 1874D. Jellyfish and Miku)

有 $n+1$ 个点，编号为 $0,1,\cdots,n$，n 条道路，第 i 条连接编号为 $i-1$ 和 i 的点。道路是双向的。第 i 条道路的美丽值为 a_i。*Jellyfish* 通过一条道路所花费的时间为 1。
初始时，*Jellyfish* 在 0 号点，她要去 n 号点拿她的 *Miku fufu*。
Jellyfish 无法辨别方向。假设当前她在 u 号点，连接 u 号点的道路的美丽值总和为 s，那么对于任意一条美丽值为 x 且连接 u 号点的道路，*Jellyfish* 下一步通过这条道路的概率为 $\frac{x}{s}$。
道路的美丽值由你来决定，但是美丽值必须为正整数，且总和不得超过 m。
求 *Jellyfish* 第一次到达 n 号点的最小期望时间。相对误差不得超过 10^{-9}。
$1 \le n \le m \le 3000$。
$\texttt{2s}, \texttt{512MiB}$。

Solution

先考虑 a_i 确定时如何计算期望时间。

Solution

先考虑 a_i 确定时如何计算期望时间。

如果直接设每个点到达 n 期望时间，需要用到带状矩阵高斯消元，较为复杂。

Solution

先考虑 a_i 确定时如何计算期望时间。
如果直接设每个点到达 n 期望时间，需要用到带状矩阵高斯消元，较为复杂。
因此，可以设 f_i 表示从 i 出发下一次到达 $i+1$ 的期望时间，这样 f_i 的计算只要用到 f_{i-1}。

Solution

先考虑 a_i 确定时如何计算期望时间。
如果直接设每个点到达 n 期望时间，需要用到带状矩阵高斯消元，较为复杂。
因此，可以设 f_i 表示从 i 出发下一次到达 $i+1$ 的期望时间，这样 f_i 的计算只要用到 f_{i-1}。
有 $f_i = \frac{a_i}{a_i+a_{i+1}}(f_i + f_{i-1}) + 1$，可得

$$
\begin{aligned}
f_0 &= 1 \\
f_i &= \frac{a_i}{a_{i+1}}(f_{i-1}+1)+1 \quad (1 \le i < n)
\end{aligned}
$$

Solution

先考虑 a_i 确定时如何计算期望时间。
如果直接设每个点到达 n 期望时间，需要用到带状矩阵高斯消元，较为复杂。
因此，可以设 f_i 表示从 i 出发下一次到达 $i+1$ 的期望时间，这样 f_i 的计算只要用到 f_{i-1}。
有 $f_i = \frac{a_i}{a_i+a_{i+1}}(f_i + f_{i-1}) + 1$，可得

$$
\begin{aligned}
f_0 &= 1 \\
f_i &= \frac{a_i}{a_{i+1}}(f_{i-1} + 1) + 1 \quad (1 \le i < n)
\end{aligned}
$$

令 $f_i' = f_i + 1$。

$$
\begin{aligned}
f_0' &= 2 \\
f_i' &= \frac{a_i}{a_{i+1}} f_{i-1}' + 2 \quad (1 \le i < n)
\end{aligned}
$$

Solution

$$f_0' = 2$$

$$f_i' = \frac{a_i}{a_{i+1}} f_{i-1}' + 2 \quad (1 \le i < n)$$

$$f_i' = \frac{2 \sum_{j=1}^{i+1} a_j}{a_{i+1}}$$

Solution

$$f_0' = 2$$
$$f_i' = \frac{a_i}{a_{i+1}} f_{i-1}' + 2 \quad (1 \le i < n)$$

$$f_i' = \frac{2\sum_{j=1}^{i+1} a_j}{a_{i+1}}$$

答案为 $\sum_{i=0}^{n-1} f_i = \sum_{i=0}^{n-1} f_i' - n = 2\sum_{1\le i<j\le n} \frac{a_i}{a_j} + n$。

Solution

$$f_0' = 2$$
$$f_i' = \frac{a_i}{a_{i+1}} f_{i-1}' + 2 \quad (1 \le i < n)$$

$$f_i' = \frac{2\sum_{j=1}^{i+1} a_j}{a_{i+1}}$$

答案为 $\sum_{i=0}^{n-1} f_i = \sum_{i=0}^{n-1} f_i' - n = 2\sum_{1\le i<j\le n} \frac{a_i}{a_j} + n$。
直接 dp，记录目前确定 a 的前缀长度及元素和，最小化已确定的 $\frac{a_i}{a_j}$ 之和，状态 $\mathcal{O}(nm)$，转移 $\mathcal{O}(nm^2)$。

Solution

$$f_0' = 2$$
$$f_i' = \frac{a_i}{a_{i+1}} f_{i-1}' + 2 \quad (1 \le i < n)$$

$$f_i' = \frac{2\sum_{j=1}^{i+1} a_j}{a_{i+1}}$$

答案为 $\sum_{i=0}^{n-1} f_i = \sum_{i=0}^{n-1} f_i' - n = 2\sum_{1\le i<j\le n} \frac{a_i}{a_j} + n$。
直接 dp，记录目前确定 a 的前缀长度及元素和，最小化已确定的 $\frac{a_i}{a_j}$ 之和，状态 $\mathcal{O}(nm)$，转移 $\mathcal{O}(nm^2)$。
注意到交换 a 的逆序对，答案更优，因此 a 递增，从而 $a_i \le \frac{m}{i}$，转移降低至 $\mathcal{O}(m^2 \log n)$。

Problem 27 (Codeforces 963E. Circles of Waiting)

在平面直角坐标系上，有一个点，一开始在 $(0,0)$ 。每秒钟这个点都会随机移动：如果它在 (x,y) ，下一秒它在 $(x-1,y)$ 的概率是 p_1 ，在 $(x,y-1)$ 的概率是 p_2 ，在 $(x+1,y)$ 的概率是 p_3 ，在 $(x,y+1)$ 的概率是 p_4 。保证 $p_1+p_2+p_3+p_4=1$，各次移动互不关联。
求出这个点移动至距离原点距离为大于 R 的点的期望步数。距离为欧几里得距离。
$0 \le R \le 50, 1 \le a_1, a_2, a_3, a_4 \le 1000, p_i = \frac{a_i}{a_1+a_2+a_3+a_4}$。
$2\mathtt{s}, 256\mathtt{MiB}$。

Solution

设点 (i,j) 走出圆的期望步数为 $f_{i,j}$。

$$f_{i,j}=\begin{cases} p_1f_{i-1,j}+p_2f_{i,j-1}+p_3f_{i+1,j}+p_4f_{i,j+1}+1 & i^2+j^2\le R^2 \\ 0 & i^2+j^2>R^2 \end{cases}$$

直接高斯消元，时间复杂度 $\mathcal{O}(R^6)$，不能通过。

Solution

设点 (i,j) 走出圆的期望步数为 $f_{i,j}$。

$$f_{i,j}=\begin{cases} p_1f_{i-1,j}+p_2f_{i,j-1}+p_3f_{i+1,j}+p_4f_{i,j+1}+1 & i^2+j^2\le R^2 \\ 0 & i^2+j^2>R^2 \end{cases}$$

直接高斯消元，时间复杂度 $\mathcal{O}(R^6)$，不能通过。
容易发现 (i,j) 先按照 i，再按照 j 排序后形成的矩阵是一个带状矩阵，带宽 $\mathcal{O}(R)$。
使用带状矩阵高斯消元，时间复杂度 $\mathcal{O}(R^4)$。

Solution

设点 (i,j) 走出圆的期望步数为 $f_{i,j}$。

$$f_{i,j} = \begin{cases} p_1 f_{i-1,j} + p_2 f_{i,j-1} + p_3 f_{i+1,j} + p_4 f_{i,j+1} + 1 & i^2 + j^2 \le R^2 \\ 0 & i^2 + j^2 > R^2 \end{cases}$$

直接高斯消元，时间复杂度 $\mathcal{O}(R^6)$，不能通过。
容易发现 (i,j) 先按照 i，再按照 j 排序后形成的矩阵是一个带状矩阵，带宽 $\mathcal{O}(R)$。
使用带状矩阵高斯消元，时间复杂度 $\mathcal{O}(R^4)$。
有 $\mathcal{O}(R^3)$ 的做法，但依赖本题圆的性质。

本题可以看成在一条长为 $\mathcal{O}(R^2)$ 的链上，可以走到相邻的 $\mathcal{O}(R)$ 个点。一般地，在一条长为 $\mathcal{O}(n)$ 的链上，可以走到相邻的 $\mathcal{O}(m)$ 个点，所产生的方程可以 $\mathcal{O}(nm^2)$ 消元。

将在节点（状态）u 时所求的值设为 f_u，若 f_u 只与 u 的邻居有关（是 f_v $(v \in N(u)), 1$ 的线性组合），可以将 f_u 表示成 $f_u = a \cdot f_{fa} + b$ 的形式，从而 $\mathcal{O}(n)$ 求出 f（忽略求逆元的时间）。

Problem 28 (LOJ 2542. 「PKUWC2018」随机游走)

给定一棵 n 个结点的树，你从点 x 出发，每次等概率随机选择一条与所在点相邻的边走过去。
有 Q 次询问，每次询问给定一个集合 S，求如果从 x 出发一直随机游走，直到点集 S 中所有点都至少经过一次的话，期望游走几步。
特别地，点 x（即起点）视为一开始就被经过了一次。
答案对 998244353 取模。
$1 \le n \le 18, 1 \le Q \le 5000, 1 \le k \le n$。
1s, 512MiB。

Problem 28 (LOJ 2542. 「PKUWC2018」随机游走)

给定一棵 n 个结点的树，你从点 x 出发，每次等概率随机选择一条与所在点相邻的边走过去。
有 Q 次询问，每次询问给定一个集合 S，求如果从 x 出发一直随机游走，直到点集 S 中所有点都至少经过一次的话，期望游走几步。
特别地，点 x（即起点）视为一开始就被经过了一次。
答案对 998244353 取模。
$1 \le n \le 18, 1 \le Q \le 5000, 1 \le k \le n$。
$1\texttt{s}, 512\texttt{MiB}$。

hint: min-max 容斥。

Solution

对于一种确定的情况，设点 u 第一次到达的时间是 t_u，要求的是 $\max_{u\in S} t_u$，不容易求。

Solution

对于一种确定的情况，设点 u 第一次到达的时间是 t_u，要求的是 $\max_{u\in S} t_u$，不容易求。
使用 *min-max* 容斥，$\max_{u\in S} t_u = \sum_{\varnothing\neq T\subset S}(-1)^{|T|-1}\min_{u\in T} t_u$。
根据期望的线性性，$\mathrm{E}\left(\max_{u\in S} t_u\right) = \sum_{\varnothing\neq T\subset S}(-1)^{|T|-1}\mathrm{E}\left(\min_{u\in T} t_u\right)$。

Solution

对于一种确定的情况，设点 u 第一次到达的时间是 t_u，要求的是 $\max_{u\in S} t_u$，不容易求。
使用 *min-max* 容斥，$\max_{u\in S} t_u = \sum_{\varnothing\neq T\subset S}(-1)^{|T|-1}\min_{u\in T} t_u$。
根据期望的线性性，$\mathrm{E}\left(\max_{u\in S} t_u\right) = \sum_{\varnothing\neq T\subset S}(-1)^{|T|-1}\mathrm{E}\left(\min_{u\in T} t_u\right)$。
$\mathrm{E}\left(\min_{u\in T} t_u\right)$ 就是第一次走到 T 中的点的期望时间。
对于每个 T，可以 $\mathcal{O}(n\log P)$ 计算：

- 设 f_u 为以节点 u 为起点，第一次走到 T 中的点的期望时间，$f_x = \mathrm{E}\left(\min_{u\in T} t_u\right)$。
- $f_u = \begin{cases} \frac{1}{deg(u)}\sum_{v\in N(u)} f_v + 1 & u\notin T \\ 0 & u\in T \end{cases}$
- 满足前文的形式。

直接实现，时间复杂度 $\mathcal{O}(2^n n\log P)$。

Solution

对于一种确定的情况，设点 u 第一次到达的时间是 t_u，要求的是 $\max_{u\in S} t_u$，不容易求。
使用 *min-max* 容斥，$\max_{u\in S} t_u = \sum_{\varnothing\neq T\subset S}(-1)^{|T|-1}\min_{u\in T} t_u$。
根据期望的线性性，$\mathrm{E}\left(\max_{u\in S} t_u\right) = \sum_{\varnothing\neq T\subset S}(-1)^{|T|-1}\,\mathrm{E}\left(\min_{u\in T} t_u\right)$。
$\mathrm{E}\left(\min_{u\in T} t_u\right)$ 就是第一次走到 T 中的点的期望时间。
对于每个 T，可以 $\mathcal{O}(n\log P)$ 计算：

- 设 f_u 为以节点 u 为起点，第一次走到 T 中的点的期望时间，$f_x = \mathrm{E}\left(\min_{u\in T} t_u\right)$。
- $f_u = \begin{cases} \frac{1}{deg(u)}\sum_{v\in N(u)} f_v + 1 & u\notin T \\ 0 & u\in T \end{cases}$
- 满足前文的形式。

直接实现，时间复杂度 $\mathcal{O}(2^n n\log P)$。
精细实现可以做到 $\mathcal{O}(2^n(n+\log P))$。

Problem 29 (LOJ 3391. 「2020-2021 集训队作业」大鱼洽水)

给定一棵边仙人掌（每条边至多在一个简单环上），从一个点出发，每次等概率选择所在点周围一条边走到另一端点。对每个点求，如果这个点作为起点，期望走多少步能走到指定点集的任意一点上？

由于某些原因，本题数据按类似对于每个点随机选一个编号比自己小的点连边的方法生成。这样的数据比较随机，很大程度上，可以忽略计算过程中，计算 0 的逆元的情况。

$2 \le n \le 10^5, 1 \le m \le 1.5 \times 10^5$。

1s, 512MiB。

Solution

只要将树推广到仙人掌即可。

Solution

只要将树推广到仙人掌即可。
切断每个简单环的一条边，并按照树的方式指定父亲 fa，如果当前节点 u 到 fa 的连边在一个简单环上，另外指定简单环中深度最小的点 $cycrt$。

Solution

只要将树推广到仙人掌即可。
切断每个简单环的一条边，并按照树的方式指定父亲 fa，如果当前节点 u 到 fa 的连边在一个简单环上，另外指定简单环中深度最小的点 $cycrt$。
将 f_u 表示成 $f_u = a \cdot f_{fa} + b \cdot f_{cycrt} + c$，可以发现这种形式是封闭的。

Problem 30 (SOJ 902. 【STR 25】第一题)

小 Y 得到了一个 1 到 n 的排列 a 并想将其排序，但他每次只会随机一个二元组 $(i,j)(1 \le i < j \le n)$ 并交换 a_i, a_j 。现在他想要知道，期望交换多少次后，这个排列会变得有序。

$1 \le n \le 20$。

$1\text{s}, 512\text{MiB}$。

Solution

把排列分解为环，状态为环长的集合。
当 $n = 20$ 时，状态数为 627。
暴力高斯消元，时间复杂度 $\mathcal{O}(P(n)^3)$。

Problem 31 (Codeforces 1349D. Slime and Biscuits)

有 n 个人，第 i 个人有 a_i 个饼干。
每次随机选择一个饼干，将其随机分配给除了它现在所有者的其他 $n-1$ 个人。
求使得一个人拥有所有饼干的期望步数，对 998244353 取模。
$2 \le n \le 10^5, 1 \le \sum a_i \le 3 \cdot 10^5$。
$2\mathtt{s}, 256\mathtt{MiB}$。

相似题目：Codeforces 850F. Rainbow Balls，Codeforces 1575F. Finding Expected Value。

Problem 32 (LOJ 6876. 「THUPC 2023 初赛」最后的活动)

玩家需要多次攻略每次随机生成的多层迷宫，每次退出迷宫时根据在迷宫中各层击杀怪物的评价独立结算本次随机迷宫的分数。每次挑战迷宫时的流程简化如下：

1. 最高难度的迷宫最深为 N 层。确定难度后，从随机生成的迷宫的第 1 层开始挑战。
2. 进行第 i 层的挑战。挑战第 i 层时，小 S 有可能挑战失败，挑战成功并获得普通评价，或者挑战成功并获得高评价。如果小 S 选择保守的挑战策略，则有 $p_{i,0}$ 的概率挑战失败，有 $p_{i,1}$ 的概率挑战成功并获得普通评价，有 $p_{i,2}$ 的概率挑战成功并获得高评价；如果小 S 选择激进的挑战策略，则有 $q_{i,0}$ 的概率挑战失败，有 $q_{i,1}$ 的概率挑战成功并获得普通评价，有 $q_{i,2}$ 的概率挑战成功并获得高评价。
   - 获得普通评价时，在当前层获得 $s_{i,1}$ 的分数；获得高评价时，在当前层获得 $s_{i,2}$ 的分数。这部分获得的分数不会直接加算到玩家的总分数中，而是在退出迷宫时结算。如果挑战成功，且当前不是最后一层（$i < N$），则跳转到第 3 步，选择是否继续挑战；否则（$i = N$），退出迷宫并跳转到第 4 步进行结算。
   - 如果挑战失败，则强制退出迷宫，跳转到第 4 步。
3. 如果当前不是最后一层，玩家可以选择是否继续挑战下一层。如果选择继续，则返回第 2 步；否则退出当前迷宫，跳转到第 4 步进行结算。
4. 本次迷宫的分数结算：如果因为失败而强制退出，则当前层不获得任何奖励，且本次迷宫中之前各层累积的分数需要乘上惩罚系数 c（为了使最终分数为整数，游戏会对惩罚后的分数先求和再下取整）；除了强制退出之外，玩家主动退出或者通关迷宫后退出都可以获得全部尚未结算的分数。

小 S 找到了你，希望你能帮忙计算当剩余分数在 1 至 M 分之间，仅按照上述的流程挑战迷宫，并采用最佳策略时，最终能够恰好达到目标分数的最大概率。

$1 \le N \le 6$，$1 \le M \le 10000$，$0 \le c' \le 100$，$1 \le s_{i,1} \le s_{i,2} \le 10000$，
$0 \le u_{i,0}, u_{i,1}, u_{i,2}, v_{i,0}, v_{i,1}, v_{i,2} \le 10000$，$u_{i,1} + u_{i,2} \ge 1$，$v_{i,1} + v_{i,2} \ge 1$。
1s, 512MiB。

Problem 32 (LOJ 6876. 「THUPC 2023 初赛」最后的活动)

玩家需要多次攻略每次随机生成的多层迷宫，每次退出迷宫时根据在迷宫中各层击杀怪物的评价独立结算本次随机迷宫的分数。每次挑战迷宫时的流程简化如下：

1. 最高难度的迷宫最深为 N 层。确定难度后，从随机生成的迷宫的第 *1* 层开始挑战。
2. 进行第 i 层的挑战。挑战第 i 层时，小 S 有可能挑战失败，挑战成功并获得普通评价，或者挑战成功并获得高评价。如果小 S 选择保守的挑战策略，则有 $p_{i,0}$ 的概率挑战失败，有 $p_{i,1}$ 的概率挑战成功并获得普通评价，有 $p_{i,2}$ 的概率挑战成功并获得高评价；如果小 S 选择激进的挑战策略，则有 $q_{i,0}$ 的概率挑战失败，有 $q_{i,1}$ 的概率挑战成功并获得普通评价，有 $q_{i,2}$ 的概率挑战成功并获得高评价。
   - 获得普通评价时，在当前层获得 $s_{i,1}$ 的分数；获得高评价时，在当前层获得 $s_{i,2}$ 的分数。这部分获得的分数不会直接加算到玩家的总分数中，而是在退出迷宫时结算。如果挑战成功，且当前不是最后一层（$i < N$），则跳转到第 *3* 步，选择是否继续挑战；否则（$i = N$），退出迷宫并跳转到第 *4* 步进行结算。
   - 如果挑战失败，则强制退出迷宫，跳转到第 *4* 步。
3. 如果当前不是最后一层，玩家可以选择是否继续挑战下一层。如果选择继续，则返回第 *2* 步；否则退出当前迷宫，跳转到第 *4* 步进行结算。
4. 本次迷宫的分数结算：如果因为失败而强制退出，则当前层不获得任何奖励，且本次迷宫中之前各层累积的分数需要乘上惩罚系数 c（为了使最终分数为整数，游戏会对惩罚后的分数先求和再下取整）；除了强制退出之外，玩家主动退出或者通关迷宫后退出都可以获得全部尚未结算的分数。

小 S 找到了你，希望你能帮忙计算当剩余分数在 1 至 M 分之间，仅按照上述的流程挑战迷宫，并采用最佳策略时，最终能够恰好达到目标分数的最大概率。

$1 \le N \le 6$，$1 \le M \le 10000$，$0 \le c' \le 100$，$1 \le s_{i,1} \le s_{i,2} \le 10000$，
$0 \le u_{i,0}, u_{i,1}, u_{i,2}, v_{i,0}, v_{i,1}, v_{i,2} \le 10000$，$u_{i,1} + u_{i,2} \ge 1$，$v_{i,1} + v_{i,2} \ge 1$。
1s, 512MiB。

hint1：考虑如果只能进入（如果出来后还没有拿够分数就失败了）一次迷宫怎么做。

Problem 32 (LOJ 6876. 「THUPC 2023 初赛」最后的活动)

玩家需要多次攻略每次随机生成的多层迷宫，每次退出迷宫时根据在迷宫中各层击杀怪物的评价独立结算本次随机迷宫的分数。每次挑战迷宫时的流程简化如下：

1. 最高难度的迷宫最深为 N 层。确定难度后，从随机生成的迷宫的第 *1* 层开始挑战。
2. 进行第 i 层的挑战。挑战第 i 层时，小 *S* 有可能挑战失败，挑战成功并获得普通评价，或者挑战成功并获得高评价。如果小 *S* 选择保守的挑战策略，则有 $p_{i,0}$ 的概率挑战失败，有 $p_{i,1}$ 的概率挑战成功并获得普通评价，有 $p_{i,2}$ 的概率挑战成功并获得高评价；如果小 *S* 选择激进的挑战策略，则有 $q_{i,0}$ 的概率挑战失败，有 $q_{i,1}$ 的概率挑战成功并获得普通评价，有 $q_{i,2}$ 的概率挑战成功并获得高评价。
   - 获得普通评价时，在当前层获得 $s_{i,1}$ 的分数；获得高评价时，在当前层获得 $s_{i,2}$ 的分数。这部分获得的分数不会直接加算到玩家的总分数中，而是在退出迷宫时结算。如果挑战成功，且当前不是最后一层（$i < N$），则跳转到第 *3* 步，选择是否继续挑战；否则（$i = N$），退出迷宫并跳转到第 *4* 步进行结算。
   - 如果挑战失败，则强制退出迷宫，跳转到第 *4* 步。
3. 如果当前不是最后一层，玩家可以选择是否继续挑战下一层。如果选择继续，则返回第 *2* 步；否则退出当前迷宫，跳转到第 *4* 步进行结算。
4. 本次迷宫的分数结算：如果因为失败而强制退出，则当前层不获得任何奖励，且本次迷宫中之前各层累积的分数需要乘上惩罚系数 c（为了使最终分数为整数，游戏会对惩罚后的分数先求和再下取整）；除了强制退出之外，玩家主动退出或者通关迷宫后退出都可以获得全部尚未结算的分数。

小 *S* 找到了你，希望你能帮忙计算当剩余分数在 1 至 M 分之间，仅按照上述的流程挑战迷宫，并采用最佳策略时，最终能够恰好达到目标分数的最大概率。
$1 \le N \le 6$，$1 \le M \le 10000$，$0 \le c' \le 100$，$1 \le s_{i,1} \le s_{i,2} \le 10000$，
$0 \le u_{i,0}, u_{i,1}, u_{i,2}, v_{i,0}, v_{i,1}, v_{i,2} \le 10000$，$u_{i,1} + u_{i,2} \ge 1$，$v_{i,1} + v_{i,2} \ge 1$。
1s, 512MiB。

hint1: 考虑如果只能进入（如果出来后还没有拿够分数就失败了）一次迷宫怎么做。
hint2: 考虑如果出迷宫会获得至少 1 分怎么做。

Solution

如果只能进入一次迷宫（如果出来后还没有拿够分数就失败了），可以通过简单的 *dfs*（$\mathcal{O}(2^n)$）计算剩余 i 分的最大概率（记为 f_i，定义 $f_0 = 1$）。

Solution

如果只能进入一次迷宫（如果出来后还没有拿够分数就失败了），可以通过简单的 *dfs*（$\mathcal{O}(2^n)$）计算剩余 i 分的最大概率（记为 f_i，定义 $f_0 = 1$）。如果出迷宫后过程仍会继续，但是出迷宫会获得至少 1 分，可以从小到大枚举 i，f_i 的计算只依赖 $f_0, \cdots, f_{i-1}$。

Solution

如果只能进入一次迷宫（如果出来后还没有拿够分数就失败了），可以通过简单的 *dfs*（$\mathcal{O}(2^n)$）计算剩余 i 分的最大概率（记为 f_i，定义 $f_0 = 1$）。
如果出迷宫后过程仍会继续，但是出迷宫会获得至少 1 分，可以从小到大枚举 i，f_i 的计算只依赖 $f_0, \cdots, f_{i-1}$。
去掉出迷宫会获得至少 1 分的限制，则 f_i 会受到 f_i 的影响。

Solution

如果只能进入一次迷宫（如果出来后还没有拿够分数就失败了），可以通过简单的 *dfs*（$\mathcal{O}(2^n)$）计算剩余 i 分的最大概率（记为 f_i，定义 $f_0=1$）。
如果出迷宫后过程仍会继续，但是出迷宫会获得至少 1 分，可以从小到大枚举 i，f_i 的计算只依赖 $f_0,\cdots,f_{i-1}$。
去掉出迷宫会获得至少 1 分的限制，则 f_i 会受到 f_i 的影响。
设 f_i 的真实值为 s。
如果认为 $f_i=c$，再计算一遍 f_i，得到 r。

Solution

如果只能进入一次迷宫（如果出来后还没有拿够分数就失败了），可以通过简单的 *dfs*（$\mathcal{O}(2^n)$）计算剩余 i 分的最大概率（记为 f_i，定义 $f_0 = 1$）。
如果出迷宫后过程仍会继续，但是出迷宫会获得至少 1 分，可以从小到大枚举 i，f_i 的计算只依赖 $f_0, \cdots, f_{i-1}$。
去掉出迷宫会获得至少 1 分的限制，则 f_i 会受到 f_i 的影响。
设 f_i 的真实值为 s。
如果认为 $f_i = c$，再计算一遍 f_i，得到 r。

- 若 $c < s$，则 $c < r \leq s$。
- 若 $c > s$，则 $c > r \geq s$。

Solution

如果只能进入一次迷宫（如果出来后还没有拿够分数就失败了），可以通过简单的 *dfs*（$\mathcal{O}(2^n)$）计算剩余 i 分的最大概率（记为 f_i，定义 $f_0=1$）。
如果出迷宫后过程仍会继续，但是出迷宫会获得至少 1 分，可以从小到大枚举 i，f_i 的计算只依赖 $f_0,\cdots,f_{i-1}$。
去掉出迷宫会获得至少 1 分的限制，则 f_i 会受到 f_i 的影响。
设 f_i 的真实值为 s。
如果认为 $f_i=c$，再计算一遍 f_i，得到 r。

- 若 $c<s$，则 $c<r\le s$。
- 若 $c>s$，则 $c>r\ge s$。

可以通过判断 c,r 的大小来判断 c,s 的大小，从而二分 s。
时间复杂度为 $\mathcal{O}(2^n M\log\frac{1}{\varepsilon})$。

Problem 33 (Codeforces 1912F. Fugitive Frenzy)

警察在一棵树上追捕逃犯，在 0 时刻，警察出现在编号为 s 的顶节点，而逃犯则会选择一个节点出现，之后从警察开始，他们会轮流行动。

- 每次轮到警察移动时，他可以任意选择一个与当前节点相邻的节点并移动过去。警察移动的时间为一个单位时间。此外，警察也可以决定静止不动，在这种情况下，他会在开始移动的顶点等待一个单位时间。如果在移动结束时，警察与逃犯出现在同一个节点，他会立即抓住逃犯，追捕结束。
- 每次轮到逃犯移动时，假设他位于点 b，而警察位于点 p。逃犯会选择一个 p 不在 b 到 b' 的路径上的点 b' 并立即移动到 b'，他的移动不需要花费时间，且他也可以选择停留在原地。

逃犯每时每刻都知道警官的位置（包括 s）。相反，警官对逃犯的行踪一无所知，只有在抓住逃犯的那一刻才能发现他。
警察的目标是尽快抓住逃犯，而逃犯的目标是尽可能晚地被抓住，在警察和逃犯都采取最优行动的情况下，求追捕时间的期望。$2 \leq n \leq 100$。
$5\mathtt{s}, 1024\mathtt{MiB}$。

Solution

https://codeforces.com/contest/1912/attachments/download/23436/nef-2023-tutorial.pdf