最小公倍数(lcm)

由于有 $(1+\sqrt{2})^n=(3+2\sqrt{2})(1+\sqrt{2})^{n-2}=2(1+\sqrt{2})^{n-1}+(1+\sqrt{2})^{n-2}$，所以 $f_n=2f_{n-1}+f_{n-2}$，且 $f_0=0$，$f_1=1$。

容易发现有 $f_{n+2}=f_{n+1}f_2+f_nf_1$，通过数学归纳法进而有：$f_{n+m}=f_{n+1}f_m+f_nf_{m-1}$。

因此 $\gcd(f_{n+m},f_m)=\gcd(f_{n+1}f_m+f_nf_{m-1},f_m)=\gcd(f_nf_{m-1},f_m)$。

同时由于 $\gcd(f_1,f_2)=1$，通过数学归纳法可得 $\gcd(f_n,f_{n-1})=1$，所以有：
$\gcd(f_{n+m},f_m)=\gcd(f_nf_{m-1},f_m)=\gcd(f_n,f_m)$。

发现就是在对下表进行辗转相减，所以有 $\gcd(f_n,f_m)=f_{\gcd(n,m)}$。

上面的证明对于所有 $f_n=af_{n-1}+bf_{n-2},f_0=0,f_1=1$ 且 $\gcd(a,b)=1$ 适用。

和式要求 lcm，尝试将其转成 gcd 处理，记 $S=\{1,2,\dots n\}$。

$$\begin{aligned}\operatorname*{lcm}_{i\in S} f_i&=\prod_{T\subseteq S}\gcd_{i\in T} f_i^{(-1)^{|T|-1}}\\&=\prod_{T\in S} f_{\gcd\limits_{i\in T} i}^{(-1)^{|T-1|}}\\&=\prod_{i=1}^{n} f_i^{\sum\limits_{T\subseteq S}[\gcd(T)=i](-1)^{|T|-1}}\end{aligned}$$

记 $a(i)=\sum\limits_{T\subseteq S}[\gcd(T)=i](-1)^{|T|-1}$，则：

$$\begin{aligned}a(i)&=\sum_{i|d}\mu(d)\sum_{T\subseteq S}[d|\gcd(T)](-1)^{|T|-1}\\&=\sum_{i|d}\mu(d)\sum_{j=1}^{\lfloor\frac{n}{i}\rfloor}\binom{\lfloor\frac{n}{i}\rfloor}{j}(-1)^{j-1}\\&=\sum_{i|d}\mu(d)\end{aligned}$$

因此：

$$\begin{aligned}g_n&=\prod_{i=1}^{n} f_i^{\sum\limits_{T\subseteq S}[\gcd(T)=i](-1)^{|T|-1}}\\&=\prod_{i=1}^{n}\prod_{i|j\wedge j\le n} f_i^{\mu(\frac{j}{i})}\\&=\prod_{pq\le n} f_p^{\mu(q)}\end{aligned}$$

那么 g_{n-1} 相对于 g_n，增加的就是 $\prod\limits_{pq=n} f_p^{\mu(q)}$，可以提前暴力处理出来，时间复杂度 $O(n\ln n)$

总时间复杂度 $O(\sum n\ln n)$。

白兔迷宫(rabbit)

发现带环的期望问题，尝试列出方程高斯消元。先去掉某一些奇奇怪怪的特殊情况，例如以 T 为入点的边。

方差就是平方的期望减去期望的平方。现在的问题转化成求期望和平方的期望。

记 f_i 表示从 S 走到 i 点的期望次数（也就是期望分数的零次方，规定 $0^0=1$），g_i 表示从 S 走到 i 时的期望分数，h_i 表示从 S 走到 i 时的分数的平方的期望。

记 d_i 表示 i 点的出边数量。有 $f_v=\sum\limits_{(u,v,o)\in E}\frac{1}{d_u}f_u$，$g_v=\sum\limits_{(u,v,1)\in E}\frac{1}{d_u}(g_u+f_u)$，
$h_v=\sum\limits_{(u,v,1)\in E}\frac{1}{d_u}(h_u+2g_u+f_u)$。

发现这个一个 $3n$ 元一次方程组，可以直接高斯消元，时间复杂度 $O(n^3)$。

算数(calc)

考虑对于某一个 k，如果它满足条件，就有 $b^{k+1}\equiv -1\pmod P$。

显然 b,P 不互质的时候无解。

考虑先找到 $b^{k'}\equiv 1\pmod P$ 的最小的 k'。由于 $b^{\varphi(P)}\equiv 1\pmod P$，所以有 $k'|\varphi(P)$。

所以我们可以枚举 $\varphi(P)$ 的因子 g，看除掉 g 之后是否仍然满足 $b^{k'}\equiv 1\pmod P$。

找到 k' 之后，我们检验是否有 $2|k'$ 且 $b^{\frac{k'}{2}}\equiv -1\pmod P$，如果有，那么就有 $k=\frac{k'}{2}-1$。

当然可能要判断一些 $P=2$ 或者 $k'=2$ 的 corner case。

时间复杂度 $O(\sqrt{P}+T\log^2 P)$。

染色球(color)

解决这道题，你需要了解一个东西：**鞅与停时定理**。

关于它的具体内容，笔者的数学水平还不足以详细讲述，具体可见：[概率论科技：鞅与停时定理 - littleZ_meow 的小窝]。

在这里给出一个比较 OI 的解释：就是我们想要设计一个关于当前局面 S 的势能函数 $F(S)$，使得在 S 为非终止态的情况下，进行一次操作之后的势能函数 $F(S')$ 期望为 $F(S)+1$。如果我们得到了这个函数，我们就可以使用终止态的势能函数减去初始态的势能函数，就可以得到期望的操作次数了。

注：更为常用的写法是 $E(F(S'))=F(S)-1$ 以及初始态减终止态，但是我个人习惯是 $+1$ 和终止态减初始态，所以就这样写了。

对于这道题目，需要使得这个 F 要能够普适所有的局面（例如这道题目中，颜色之间没有本质区别），我们自然会想到使用 $F(S)=\sum\limits_{i=1}^{n}f(x_i)$ 这样的势能函数，其中 x_i 为 S 局面中，颜色为 A_i 的球的数量。

我们核心的等式就是 $F(S)+1=E(F(S'))$，我们现在分别使用 $f(x)$ 来表示左右的式子。

对于左式，$F(S)+1=\sum\limits_{i=1}^{n}f(x_i)+1$。

对于右式，我们先要考虑如何刻画一次操作。

第一步，随机选出若干个球。第二步，选出两两不同的和第一步选出的球数量相同的颜色，随机分配给这些球。

对于第 i 个颜色，在第一步中，我们有 $\frac{\binom{x_i}{j}}{2^{x_i}}$ 的概率选中这个颜色的 j 个球，这样第 i 种颜色的球会减少 j 个；除开这 x_i 个球之外，我们有 $\frac{\binom{n-x_i}{k}}{2^{A-x_i}}$ 的概率选出 k 个球。在第二步中，我们需要从 n 个颜色中选出 $j+k$ 中颜色，则我们有 $\frac{j+k}{n}$ 的概率选出第 i 种颜色，有 $\frac{n-j-k}{n}$ 的概率没有选出第 i 种颜色，其中第一种情况会让第 i 种颜色的球增加 1 个。

由此，我们可以得到：

$$E(F(S'))=\sum_{i=1}^{n}\sum_{j=0}^{x_i}\sum_{k=0}^{n-x_i}\frac{\binom{x_i}{j}}{2^{x_i}}\frac{\binom{n-x_i}{k}}{2^{n-x_i}}\left(\frac{j+k}{n}f(x_i-j+1)+\frac{n-j+k}{n}f(x_i-j)\right)$$

因为我们要求 $F(S)+1=E(F(S'))$ 恒成立，所以需要在满足 $\sum\limits_{i=1}^{n}x_i=n$ 的情况下，x_i 取任意值的时候都成立，也就是要对于每一个 i，上式都要成立。

我们发现右式和左式中的 $\sum\limits_{i=1}^{n} f(x_i)$ 都有着枚举 i 求和的结构，只有一个 1 比较突兀，所以我们也需要将其拆成枚举 i 求和的结构，也就是 $\sum\limits_{i=1}^{n} \frac{x_i}{n}$。

因为我们需要对于每一个单独的 i 成立，所以我们有：

$$f(x_i) + \frac{x_i}{n} = \sum_{j=0}^{x_i} \sum_{k=0}^{n-x_i} \frac{\binom{x_i}{j}}{2^{x_i}} \frac{\binom{n-x_i}{k}}{2^{n-x_i}} \left(\frac{j+k}{n} f(x_i - j + 1) + \frac{n-j+k}{n} f(x_i - j) \right)$$

将 x_i 替换成 x，也即是我们有：

$$f(x) + \frac{x}{n} = \sum_{j=0}^{x} \sum_{k=0}^{n-x} \frac{\binom{x}{j}}{2^{x}} \frac{\binom{n-x}{k}}{2^{n-x}} \left(\frac{j+k}{n} f(x - j + 1) + \frac{n-j+k}{n} f(x - j) \right)$$

发现右式十分复杂，我们还需要整理。为了方便处理，我们令 $g(x) = f(x+1) - f(x)$。

$$\begin{aligned}
&\sum_{j=0}^{x} \sum_{k=0}^{n-x} \frac{\binom{x}{j}}{2^{x}} \frac{\binom{n-x}{k}}{2^{n-x}} \left(\frac{j+k}{n} f(x - j + 1) + \frac{n-j+k}{n} f(x - j) \right) \\
=&\sum_{j=0}^{x} \sum_{k=0}^{n-x} \frac{\binom{x}{j}}{2^{x}} \frac{\binom{n-x}{k}}{2^{n-x}} \left(\frac{j+k}{n} (f(x - j) + g(x-j)) + \frac{n-j+k}{n} f(x - j) \right) \\
=&\sum_{j=0}^{x} \sum_{k=0}^{n-x} \frac{\binom{x}{j}}{2^{x}} \frac{\binom{n-x}{k}}{2^{n-x}} \left(\frac{j+k}{n} g(x-j) + f(x - j) \right) \\
=&\frac{1}{2^n} \sum_{j=0}^{x} \binom{x}{j} \left(\left(f(x-j) + \frac{j}{n} g(x-j) \right) \sum_{k=0}^{n-x} \binom{n-x}{k} + \frac{1}{n} g(x-j) \sum_{k=0}^{n-x} k \binom{n-x}{k} \right)
\end{aligned}$$

根据组合式的经典结论，我们知道 $\sum\limits_{k=0}^{n-x} \binom{n-x}{k} = 2^{n-x}$，$\sum\limits_{k=0}^{n-x} k \binom{n-x}{k} = (n-x)2^{n-x-1}$。

$$\begin{aligned}
&\frac{1}{2^n} \sum_{j=0}^{x} \binom{x}{j} \left(\left(f(x-j) + \frac{j}{n} g(x-j) \right) \sum_{k=0}^{n-x} \binom{n-x}{k} + \frac{1}{n} g(x-j) \sum_{k=0}^{n-x} k \binom{n-x}{k} \right) \\
=&\frac{1}{2^n} \sum_{j=0}^{x} \binom{x}{j} \left(\left(f(x-j) + \frac{j}{n} g(x-j) \right) 2^{n-x} + \frac{1}{n} g(x-j)(n-x)2^{n-x-1} \right) \\
=&\frac{1}{2^x} \sum_{j=0}^{x} \binom{x}{j} \left(f(x-j) + \frac{j}{n} g(x-j) + \frac{n-x}{2n} g(x-j) \right) \\
=&\frac{1}{2^x} \sum_{j=0}^{x} \binom{x}{j} \left(f(x-j) + \frac{2j+n-x}{2n} g(x-j) \right) \\
=&\frac{1}{2^x} \sum_{j=0}^{x} \binom{x}{j} \left(\frac{n+x-2j}{2n} f(x-j) + \frac{2j+n-x}{2n} f(x-j+1) \right)
\end{aligned}$$

我们将令 $j = x - j$，可以得到：

$$\frac{1}{2^x} \sum_{j=0}^{x} \binom{x}{j} \left(\frac{n+x-2j}{2n} f(j) + \frac{2j+n-x}{2n} f(j+1) \right) = \frac{1}{2^x} \sum_{j=0}^{x} \binom{x}{j} \left(\frac{n+2j-x}{2n} f(j) + \frac{n+x-2j}{2n} f(j+1) \right)$$

再将左式放回来，我们就有 $f(x) + \frac{x}{n} = \frac{1}{2^x} \sum\limits_{j=0}^{x} \binom{x}{j} \left(\frac{n+2j-x}{2n} f(j) + \frac{n+x-2j}{2n} f(j+1) \right)$。

即 $2^x \left(f(x) + \frac{x}{n} \right) = \sum\limits_{j=0}^{x} \binom{x}{j} \left(\frac{n+2j-x}{2n} f(j) + \frac{n+x-2j}{2n} f(j+1) \right)$。

我们发现，这是一个关于 $f(0), f(1), f(2) \ldots f(x+1)$ 的等式，这也就意味着，我们可以利用 $f(0), f(1) \ldots f(x)$ 来推出 $f(x+1)$。

现在如果知道了 $f(0)$ 的值，我们就可以得到任意的 $f(x)$ 的值了。

但是我们要怎么确定 $f(0)$ 的值呢？这就是鞅与停时定理的巧妙之处了，由于我们除了上式没有任何其他的限制，我们可以钦定 $f(0)$ 的取值，为了方便处理，一般取 $f(0) = 0$。

然后我们就可以以此推出 $f(1), f(2) \ldots$ 了（顺带说一句，$f(0) = 0$ 可以得到 $f(1) = 0$）。

现在考虑求解最终的答案，根据前面的说法，答案应该等于终止态的势能减去初始态的势能。

最终态的势能为 $f(n)+(n-1)f(0)=f(n)$，初始态的势能等于 $\sum_{i=1}^{n} f\left(\sum_{j=1}^{n}[A_j==i]\right)$。

单次递推是 $O(n)$ 的，所以总体复杂度为 $O(n^2)$。

感觉这是递推式很可以半在线卷积做，但是没有必要（没事谁写多项式啊）。