

《童话》

题目大意

模拟以下程序，答案对 998244353 取模。

```
read(n),read(a);
for(int i=0;i<=n;i++)read(f[i]);
for(int t=1;t<=n;t++){
    for(int i=1;i<=n;i++)f[i]=f[i]+a*f[i-1];
    ans[t]=f[t];
}
```

3s/2048MB。

数据范围与约定

对于 100% 的数据，保证 $2 \leq n \leq 10^6, 0 \leq f_i < 998244353, 1 \leq a < 998244353$ 。

子任务编号	$n \leq$	特殊性质	分值
1	2000		5
2	10^5	A	5
3	10^5	BC	5
4	10^5	BD	10
5	10^5	C	10
6	5×10^4		10
7	10^5		10
8	2×10^5		15
9	5×10^5		15
10	10^6		15

特殊性质 A：保证 $f_i \neq 0$ 的 i 数量不超过 100。

特殊性质 B：保证 $a = 1$ 。

特殊性质 C：保证对于所有 $i \in [0, n]$ ，都满足 $f_i = 1$ 。

特殊性质 D：保证对于所有 $i \in [0, n]$ ，都满足 $f_i = \binom{i+2}{2} = \frac{(i+2)(i+1)}{2}$ 。

解题过程

子任务 1

把题面里的代码加个头文件交上去，5 分到手！

子任务 2

对于 $i \leq t$ ，考虑位置 i 对 ans_t 的贡献：我们可以将向后移动的 $t - i$ 步分给 t 个时刻，而贡献乘上的系数显然为 a^{t-i} ，于是：

$$ans_t = \sum_{i=0}^t f_i a^{t-i} \binom{t-i+t-1}{t-1}$$

对有值的位置求和即可。

子任务 3

即求 $\sum_{i=0}^t \binom{i+t-1}{t-1}$ ，这是经典的上指标求和，可以在预处理阶乘及其逆元后 $O(1)$ 计算。

子任务 4

即求 $\sum_{i=0}^t \binom{i+t-1}{t-1} \binom{t-i+2}{2}$ ，其显然可以被 $\sum_{i=0}^t \binom{i+t-1}{t-1} \binom{i}{2}$ ， $\sum_{i=0}^t \binom{i+t-1}{t-1} i$ ， $\sum_{i=0}^t \binom{i+t-1}{t-1}$ 线性组合得到（视 t 为常数），而上面三式均可使用吸收公式后上指标求和计算（也能使用组合意义直接得到结果）。

子任务 5

即求 $\sum_{i=0}^t \binom{i+t-1}{t-1} a^i$ ，我们可以使用递推：

$$\begin{aligned} ans_t &= \sum_{i=0}^t \binom{i+t-1}{t-1} a^i = \sum_{i=0}^t \binom{i+t-1-1}{t-1} a^i + \sum_{i=0}^t \binom{i+t-1-1}{t-2} a^i \\ &= a(ans_t - \binom{t+t-1}{t-1} a^t) + (ans_{t-1} + \binom{t+t-2}{t-2} a^t) \end{aligned}$$

递推时需要特殊处理 $a = 1$ 的情况，复杂度 $O(n)$ 。

子任务 6,7

容易将 a 的影响消除，将 f_i 除以 a^i 并令第 i 项答案乘上 a^i 即可。

记 $v_{i,j}$ 为第 i 次操作后第 j 个位置的值，那么问题可以被刻画为二维平面的格路计数—— $v_{i,j}$ 表示任选一个第 0 排的起点 $(0, i)$ ，进行任意下步和右步（第 0 排不能进行右步），到达 (i, j) 的方案数乘上 f_i 之和，我们即求 $v_{1,1}, v_{2,2}, \dots, v_{n,n}$ 。

记 $r_i(l, r)$ 表示第 i 行的 $[l, r]$ 列上面的值， $c_i(l, r)$ 表示第 i 列的第 $[l, r]$ 行上面的值。

考虑使用分治求解，我们设计函数 `solve(l, r, F, G)`，表示已知 $F = r_{l-1}(l, r)$ ， $G = c_{l-1}(l, r)$ ，需通过 F, G 计算出 $v_{l,l}, v_{l+1,l+1}, \dots, v_{r,r}$ 。

取 $[l, r]$ 中点 mid ，尝试计算第 mid 行，第 mid 列的所有值。具体地，我们先使用 $r_{l-1}(l, mid), c_{l-1}(l, mid)$ 计算出 $r_{mid}(l, mid), c_{mid}(l, mid)$ ，再用 $r_{l-1}(mid+1, r), c_{mid}(l, mid)$ 计算出 $r_{mid}(mid+1, r)$ ，类似地使用 $c_{l-1}(mid+1, r), r_{mid}(l, mid)$ 计算出 $c_{mid}(mid+1, r)$ 。于是我们不仅计算出了 $v_{mid,mid}$ ，还得到了递归 `solve(mid+1, r, u, v)` 所需的

$U = r_{mid}(mid + 1, r), V = c_{mid}(mid + 1, r)$, 可以直接递归求解剩余部分。

由于转移系数均为组合数, 可以表示为卷积形式, 使用 NTT 优化卷积即可做到 $O(n \log^2 n)$ 的时间复杂度。

精细实现可以通过子任务 8。

子任务 8,9,10

定理 (扩展拉格朗日反演) :

$$[x^n]H(F(x)) = [x^{n-1}]H'(x)\left(\frac{x}{G(x)}\right)^n$$

证明可见参考资料《信息学竞赛中的生成函数计算理论框架》3.2.3 Lagrange 反演部分。

回到原问题, 我们所求的即 $[x^t]\left(\frac{1}{1-ax}\right)^t F$, 注意到它与扩展拉格朗日反演右式形式类似, 我们尝试式子化成类似的形式:

$$[x^t]\left(\frac{1}{1-ax}\right)^t F = [x^t]\left(\frac{x}{G}\right)^{t+1} \frac{FG}{x} = [x^{t+1}](t+1)H(P)$$

其中 $G(x) = -ax^2 + x, H(x) = \int\left(\frac{F(x)G(x)}{x}\right)$, 且 P 与 G 互为复合逆, 满足 $x = -aP^2 + P$ 。

解得 $P = \frac{1-\sqrt{1-4ax}}{2a}$ (仅保留常数项为 0 的解), 于是我们只需处理任意多项式右复合 $\frac{1-\sqrt{1-4ax}}{2a}$ 这一问题。

子任务 4 是为了做到这一步后不会的选手设计的, 此时的 H 每一项系数均为 1, 可以暴力计算出 P 后做多项式快速幂 (不过后来得到了更简单的做法)。

我们将复合拆解成右复合 $-\frac{1}{2a}$, 右复合 $x - 1$, 右复合 $\sqrt{1 - 4ax}$ 三步。第一步容易线性处理, 第二步需要进行一次多项式平移 (可以使用 NTT 加速卷积做到 $O(n \log n)$), 重点在复合 $\sqrt{1 - 4ax}$, 我们对于奇数次数与偶数次数系数分开处理:

对于偶数项, 我们可以先右复合 $\sqrt{x}$ (将 $2k$ 上的系数平移到 k), 再右复合 $1 - 4ax$ (同样是多项式平移, 卷积即可)。

对于奇数项, 使用牛顿二项式定理:

$$(1 - 4ax)^{k+\frac{1}{2}} = \sum_{i \geq 0} \binom{k+\frac{1}{2}}{i} (-4ax)^i = \sum_{i \geq 0} \frac{(-4a)^i (2k+1)(2k-1)\cdots(2k-2i+3)}{2^i i!} x^i$$

仍然可以卷积处理。

于是本题仅需三次多项式卷积, 复杂度 $O(n \log n)$, 常数较小。

参考资料

李白天: 《信息学竞赛中的生成函数计算理论框架》(2021 年集训队论文)