

牛客多校5

比赛时间	比赛名称	赛中过题	总计过题	题目总数	罚时	Dirt	校内排名
25.07.29	牛客多校5	3	5	13	539	5/8	17/19

赛时

I 00:09 +0

Ender_hz: 直接考虑面积 $S=1+2+\cdots+n=\frac{n(n+1)}{2}$ 可以证明这样拼成的矩形周长最小（最长边长度 $\geq n$ ）

J 02:37 +5

Ender_hz: 一开始想着答案可能的范围，最后发现二分的时候好像用不到。

+1: 二分模板没改完全（边界）；

+2: 瞎改一通，避免了编译时的 warning

+3: 重构代码，把用数组维护边界改成了在 check 内部找边界；

+4: 发现了 tm 没有赋初值导致的二分越界和全部为 1 的情况，但是没有考虑全部为 0 的情况；

+5: 发现了全部为 0 的情况以及判定边界能否覆盖时的错误。

E 04:32 +0

istina: 最幽默的一集，封榜后才成功签到。定义【神秘异或】的运算结果是把异或从低位数起第偶数个 1 删去。给定 n 个正整数，求所有无序对神秘异或和。考虑按位考虑，当前对 (n, m) 在第 i 位对答案有贡献当且仅当 $(n \oplus m)_i = 1$ 且 $\sum_{j=0}^{i-1} (n \oplus m)_j$ 是偶数。注意到 $\sum_{j=0}^{i-1} (n \oplus m)_j = \sum_{j=0}^{i-1} n_j + \sum_{j=0}^{i-1} m_j - 2 \sum_{j=0}^{i-1} (n \& m)_j$ 而最后一项是偶数，因此只需要分别考虑 n 和 m 各自前 i 位和的奇偶性。开四个变量分别记录当前位为 0/1，低位和为 奇/偶 的数量即可，将各位贡献累加即可。

赛后

H _istina_ 补)

这题的难度相当一部分来自于冗长的题面，我费了老大劲才把原题面中 **Civilization VI** 的元素换成了数学语言。赛时也是被这题面吓到以为是大模拟。

首先显然可以二分答案，将问题转化成给定 p 判定方案存在性。

考虑 dp 来做。令 $dp_{i,j,l}$ 表示前 i 次操作中，已经使 $a_j=0$ 选择了 l 次 $b_{j+1} \dots$

$n\}$ 中元素的情况下 $\$cnt\$$ 的最大值。

设 $\$sum_i = m + \sum_{j=1}^i k_j\$$ 有如下三种转移：

- 第 $\$i\$$ 次操作选择增加 $\$cnt\$$

$$dp_{i,j,l} + sum_j \rightarrow dp_{i+1,j,l+1}$$

- 通过若干次操作使 $a_{j+1} \rightarrow 0$ 未触发 $a_i \rightarrow \max\{a_i - c_i, 0\}$

$$dp_{i,j,l} \rightarrow dp_{i+\lceil \frac{a_{j+1}}{sum_j} \rceil, j+1, l+\lceil \frac{a_{j+1}}{sum_j} \rceil}$$

- 通过若干次操作使 $a_{j+1} \rightarrow 0$ 触发 $a_i \rightarrow \max\{a_i - c_i, 0\}$

$$dp_{i,j,l} \rightarrow dp_{i+\lceil \frac{a_{j+1} - c_{j+1}}{sum_j} \rceil, j+1, l+\lceil \frac{b_{j+1}}{p} \rceil + \lceil \frac{a_{j+1} - c_{j+1}}{sum_j} \rceil}$$

存在某 $dp_{i,j,l} \geq s$ 则方案存在。

注意写好转移时的边界条件，并特判 $p=0$ 的情况即可。

时间复杂度 $O(t^2 n \log b)$ 可以通过本题。

Listina_补)

将总期望尝试次数拆成 $b > 0$ 和 $b = 0$ 两部分（可能在第一部分中已经达成目标，那么第二部分就没有被经过）

先考虑 $b > 0$ 的情形。考虑二维状态 (i, j) 表示当前 $a = i, b = z - j$ 初始状态为 $(0, 0)$ ，目标状态为 $(n, ?)$ 状态转移为：有 p 的概率 $(i, j) \rightarrow (i+1, j)$ 的概率 $(1-p)$ 的概率 $(i, j) \rightarrow (i, j+1)$ 由于不会走回头路，第一部分的期望步数就是所有状态的概率之和。

假设 $P_{i,j}$ 表示经过某状态的概率，可得方程
$$P_{i,j} = \begin{cases} 1 & \text{if } i, j = 0 \\ p P_{i-1,j} & \text{if } j = 0 \\ (1-p) P_{i,j-1} & \text{if } i = 0 \\ p P_{i-1,j} + (1-p) P_{i,j-1} & \text{if } i, j \neq 0 \end{cases}$$
 设 $S_j = \sum_{i=0}^{n-1} P_{i,j}$ 由于 $P_{i,j} = p P_{i-1,j} + (1-p) P_{i,j-1}$ 累加可得
$$\sum_{i=0}^{n-1} P_{i,j} = p \sum_{i=0}^{n-1} P_{i-1,j} + (1-p) \sum_{i=0}^{n-1} P_{i,j-1}$$
 整理之后可得
$$S_j = p(S_j - P_{n-1,j}) + P_{-1,j} + (1-p) S_{j-1} \implies S_j = S_{j-1} - \frac{p}{1-p} P_{n-1,j} \implies S_0 = \frac{1-p^n}{1-p}$$
 再结合 $P_{n-1,j}$ 的递推式
$$P_{n-1,j} = C_{n-1+j}^j p^{n-1} (1-p)^j = \frac{(n-1+j)!}{j! (n-1)!} p^{n-1} (1-p)^j$$
 就可以通过递推得到 S_j 而对于每个 z 这一部分的答案就是 $\sum_{j=0}^z S_j$

再考虑 $b = 0$ 的情形。这一部分的期望尝试次数是好计算的。

设从 a 从 0 到 i 期望尝试次数为 D_i 从 $i-1$ 到 i 的期望尝试次数为 d_i 则可以得到以下式子

$$\begin{aligned} \begin{cases} d_i = p + (1-p)(D_{i+1}) \\ d_i = D_i - D_{i-1} \end{cases} \\ \end{aligned}$$

$$p(D_i + \frac{1}{1-p}) = D_{i-1} + \frac{1}{1-p}$$

$D_i = \frac{1-p^i}{(1-p)p^i}$ 而进入第二部分的概率就是简单的 $S_z(1-p)$ 将概率和期望尝试次数相乘即为第二部分的答案。

总体上时间复杂度为 $O(m \log m)$ 可通过预处理逆元优化为线性。

总结

Ender_hz:

istina: 在宿舍里打的究极坐牢场，到都签不出。

MeowScore:

From:

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

Permanent link:

https://wiki.cvbbacm.com/doku.php?id=2025-2026:teams:the_server_is_busy_please_try_again_later:20250729



Last update: 2025/08/03 14:27