Day 2 Solution & Analysis

♪² 赛后总结

赛时预计的大众分 $50\text{pt} + 12\text{pt} + 35\text{pt} = 97\text{pts}$, 实际上最高分才 75pt, 令人感慨.

T2 为了防止赛时被搜到原, 进行了大量描述修改, 导致题面出现了大量问题, 谢罪.

T1 严重低于预期, 在后台看榜发现前 2h 一直有人在尝试 polylog, 有点难绷.

♪² 胖头鱼和冰淇淋 (icecream)

> 下设 n, q 同阶.

♪³ Subtask 1

用 `set` 暴力维护, $O(q \log q)$.

♪³ Subtask 2

数据随机, 因此每个集合中的元素不会太多, 照搬 Subtask 1 的做法即可通过.

♪³ Subtask 3

经典根号分治.

考虑对每个集合最后的大小进行分治. 我们称大小 $> B$ 的为大集合, $\leq B$ 的为小集合.

每次询问 x, y 两个集合并集大小时:

- 若两个集合中有一个小集合:
  - 可以暴力插入计算贡献, $O(nB)$.
- 若两个集合都是大集合:
  - 由于大集合只有 $O(\frac{n}{B})$ 个, 可以维护 $f(x, y)$ 表示大集合 x, y 的交集大小. 有效的 f 值有 $O(\frac{n^2}{B^2})$ 个.
  - 如何维护? 考虑在修改大集合时, 暴力更新所有与它有关的 f 值, 需要更新 $O(\frac{n}{B})$ 个值, 复杂度 $O(\frac{n^2}{B})$.
  - 于是我们可以 $O(1)$ 查询答案.

取 $B = \sqrt{n}$, 不难做到复杂度 $O(n\sqrt{n})$. 可能可以通过精密分析做到更优.

在赛前预估中, Subtask $1 \sim 3$ 是大家都应该拿到的分数.

♪³ Subtask 4

iznomia 告诉我这是诈骗 Subtask, 在我的强烈谴责下降低分值至 10pts.

♪³ Subtask 5

♪⁴ Approach α

只存 01 太亏了, 考虑把 $O(\omega)$ 个元素压到一位计算.

设一个集合的**计算大小**为最终这个集合中**有值的块数**.

我们考虑对集合的计算大小分治, 小集合可以 $O(B)$, 大集合可以暴力维护答案.

乍一看还是 $O(q\sqrt{n})$ 的.

不过感性认识一下可以发现, 很难构造数据把处理大集合的部分复杂度卡满.

可以把元素按照出现次数从大到小重新映射, 这样全局有值的块数期望是 $O(\frac{n}{\omega})$ 个, 大集合期望有 $O(\frac{n}{\omega B})$ 个.

取 $B = \sqrt{\frac{n}{\omega}}$, 期望可以做到 $O(n\sqrt{\frac{n}{\omega}})$. 确实没有构造出足够强的数据, 因此这个做法可以通过.

♪⁴ Approach β

对元素的出现次数分治.

同样设 $f(x, y)$ 表示集合 x, y 的交集大小, 不过注意这里已经没有大小集合的概念了.

对于出现次数 $< B$ 的元素:

- 设当前 包含它的集合 的编号 的集合为 T.
- 每次把它插入到某个集合时, 可以更新 $O(|T|)$ 个 f 值.
- 这样复杂度为什么是对的?
  - 设每个元素的出现次数分别为 $t_1, t_2, \ldots, t_m$.
  - $\sum t_i = O(n)$.
  - $\forall t_i < B$.
  - 我们关心 $O(\sum t_i^2)$ 的值.
  - 幂平均不等式告诉我们, 在 t_i 取等时能去到最大值, 这个最大值是 $O(nt)$, 即 $O(nB)$.

对于出现次数 $\geq B$ 的元素:

- 只有 $O(\frac{n}{B})$ 个, 开 $O(n)$ 个长度为 $O(\frac{n}{B})$ 的 `bitset`.
- 每次查询 `bitset or` 再 `count`, 复杂度 $O(\frac{n^2}{\omega B})$.

取 $B=\frac{\sqrt{n}}{\omega}$ 即可. 复杂度还是 $O(n\sqrt{\frac{n}{\omega}})$.

♪2 胖头鱼和草莓 (strawberry)

- 2018 Canadian Computing Olympiad Day 1 - Fun Place

> 部分分参考了 CCO 原题.
>
> 有人赛前见过这个题没得分, 我不说是谁.

♪3 Subtask 1

答案是 $\max(\left\lceil\dfrac{n+1}{2}\right\rceil, k-1)$.

不过场上只有 Terdy 一人这题有分, 十分考察选手读题水平.

♪3 Subtask 2, 3

留给 iznomia 可能存在的 $O(n^3)$ 做法.

♪3 Subtask 4

设 $f_{i,j}$ 表示前 i 个位置, 第 i 个位置最多有 j 只猫猫的答案.

转移分四种情况:

- 后面走到前面: $f_{i,j}\to f_{i+1,j+b_i}$.
- 前面走到后面: $f_{i,j}\to f_{i+1,j-a_i}$.
- 前后可以互通: $f_{i,j}\to f_{i+1,j}$.
- 前后断开: $f_{i,j}\to f_{i+1,*}$

转移的时候要考虑每种情况猫猫的最优决策, 限制 j 的取值范围.

时间复杂度 $O(nV)$.

♪2 返夏 (summer)

- Codeforces 1517 F - Reunion

> 出题的时候想的是, 把 $O(1)$ 个简单的套路融合起来, 用部分分引导选手思考.
>
> *期望差分, *正难则反, *经典树上 dp

> 虽然有 3 合 1 的嫌疑, 但不影响这题是一道很优美的数数题. 值得 *3200.
>
> 下面记不能参加的为黑点, 能参加的为白点.

♪3 Subtask 1

$O(2^n)$ 枚举状态, 没想卡任何 $O(2^n\text{poly}(n))$, 只要写的不是太离谱应该都能过.

♪3 Subtask 2

简单推推式子, 按答案是 $0,1,n$ 三种情况分讨.

♪3 Subtask 3

还是暴力, 设 $f_{i,j}$ 表示前 i 个位置, 钦定 i 为黑点, 之前能举办的最大规模为 j 的方案数.

暴力转移枚举上一个黑点位置 k, $O(n^3)$.

然而并没有人拿到 Subtask 3 的部分分, 建议多练.

♪3 Subtask 4

这里已经在引导正解了.

对于这种期望问题常见的做法是差分, 我们考虑枚举 x, 求满意度 $\geq x$ 的方案数. 这个 Trick 大家都应该见过的, 如果实在没见过, 随便找了个题 ABC295E Kth Number.

可是转化到这里还是只能做 $O(n^3)$, 因为我们根本不知道最大的满意度要在哪里取.

正难则反, 如果我们不知道在哪里取最大, 那不如限制在哪里都取不到这个最大. 这样就能转化成容易描述的限制了.

具体地, 我们设 f_i 表示前 i 个位置, 钦定 i 为黑点的方案数. 能转移到的上一个黑点位置 k 是一段区间, 可以前缀和维护. $O(n^2)$.

♪3 Subtask 5

考虑怎么把链的做法搬到树上.

链上我们考虑上一个黑点距自己的距离, 以此限定转移.

树上呢? 其实是一样的! 只不过树形结构比线性结构复杂的多, 我们要分当前这个点被子树内覆盖 / 子树外覆盖讨论.

具体的讨论方法和 [HNOI2003] 消防局的设立 完全相同, 这里不再赘述. 只不过这里的覆盖半径是 $O(n)$ 的, 所以树形 dp 部分只能做到 $O(n^2)$.

加上差分枚举的 $O(n)$, 总时间复杂度 $O(n^3)$.