题解

根号分治。对每个出现次数 $\geq \sqrt{n}$ 的数，预处理出该数与所有其他数的答案。

查询时如果有一个数出现次数 $\geq \sqrt{n}$，直接用预处理的结果，否则暴力以出现次数的复杂度扫一遍。

时间复杂度 $O(n\sqrt{n})$。

题解

注意到如果一个区间被另外一个区间包含，那么这个区间不可能成为答案。

所有不被其他区间包含的区间一定满足排序之后左端点单调且右端点单调。那么一次序列单点加对这些区间的影响可以被看作是一个区间加。只需写一个数据结构维护区间加，全局最大值。

时间复杂度 $O(n + m\log n)$。

题解

考虑枚举第 k 大，假设为 x。把所有 $\geq x$ 的数看作 1，所有 $< x$ 的数看作 0，我们需要找到一个区间使得恰好包含 k 个 1 且为 0 位置的 a_i 之和最大。

从小到大枚举所有数，遇到一个数就把对应位置的 1 变为 0，并且考虑刚才包含这个 1 的所有区间。注意到只需考虑该 1 左侧的 k 个 1 和右侧的 k 个 1。只有 $O(k)$ 段可能的区间。每一段都可以用一个 rmq $O(1)$ 算出最优的区间。

时间复杂度 $O(nk)$。

题解

考虑建一个新图，其中每个点表示一组。如果连了一条不同组之间的人，就把这个图中这两组连一条边。注意到如果图中出现了一个环，一定可以把这个环消掉，改为组内连边，代价严格变小。所有这个新图一定是一个森林。考虑状压 dp。

$d[mask][i]$ 表示子树集合为 $mask$，其中第 i 个人连向父亲的代价最小值和方案数。

转移：枚举子树的根 u，扫组 u 内的所有人，把所有 u 的子树一个一个加入。$f[mask][i][j]$ 表示子树集合为 $mask$，扫到了第 i 个人，已经有了 j 个儿子的代价最小值和方案数。每次扫到 i 时候看 i 是否连了一个子树，可以通过 $O(2^k n^2)$ 做到 $O(1)$ 转移。注意到我们还需要考虑哪个人连向了 u 的父亲，那么就可以正着 dp 一次，倒着 dp 一次，枚举连向 u 父亲的人 x，将前缀和后缀的 dp 信息合并，得到 $d[mask][x]$ 的答案。

时间复杂度 $O(2^k n^2 + 3^k n k^2)$。