

T1:

30%: $O(n^2 * m)$ 暴力判断。

100%: 很显然答案的可能性最多只有 n 种, 所以我们将所有人的答案按字典序排序后枚举将每个人的答案作为正确答案来进行判断。由于是判断题, 若当前人的答案为正确答案则零分者的答案也就确定了, 那么只需统计出这两种答案的人数判断是否满足题意即可。这一步使用字符串哈希即可解决。

T2:

30%: 枚举每个数所在的集合或者不选, 然后判定即可。复杂度 $O(n * 3^n)$ 。

60%: DP, 两个数相等就相当于两个数的 xor 为 0。设 $f[i][j][k=0..2]$ 代表 处理到第 i 个数, 如果 $k=1$ 代表 and 值为 j , 如果 $k=2$ 代表 xor 值为 j , 如果 $k=0$ 则代表一个元素都没取。所以很容易得到方程:

$$f[i][j][0] = f[i+1][j][0]$$

$$f[i][j \& a_i][1] = f[i+1][j][1] + f[i+1][j][0] + f[i+1][j \& a_i][1]$$

$$f[i][j \wedge a_i][2] = f[i+1][j][1] + f[i+1][j][2] + f[i+1][j \wedge a_i][2];$$

最后 $f[1][0][2]$ 就是答案, 复杂度为 $O(n * 1024 * 3)$

DP 还可以分开用 $f[i][j]$ 和 $g[i][j]$ 表示前 i 个 xor 值为 j , 后 i 个 and 值为 j 的方案数, 随后枚举分界点 k 来求总方案数。复杂度 $O(n * 1024 * 3)$ 。

100%: 满分数据需要高精, 答案位数较大, 需要进行压位来防止 TLE, 因为不知道答案的位数究竟多大, 压位后高精数组仍需要开的较大一些, 所以原 DP 的数组滚动即可。

T3:

30%:

当 $T \leq 10000$ 的时候, 可以设 $vis[i][j]$ 代表到达第 i 个点时间为 j 是否合法。 这样是 $O(T * m)$, 可以拿到小数据。

另外的那 30%: 各种奇怪的骗分方法。选手自行考虑。

100%:

当 T 很大的时候, 我们考虑 如果 $0 \rightarrow x \rightarrow n-1$ 路径时间为 T , 且 从 x 出发有一个时间为 d 的环, 则 一定存在一个 K 满足 $K + p * d = T$ (至少 T 满足条件), 这样我们就能绕着环走 p 次就能构成一条时间为 T 的路径。

显然要求的路径一定经过 0, 而且在合法情况下从 0 号点出发一定存在一条边, 否则 0 号点和 $n-1$ 号就是不联通的。随便取一条边时间为 d , 则能构成从 0 号点出发的一个时间为 $2d$ 的环。这样原题就化为最短路问题了, $dis[i][j]$ 代表到达 i 号点, 时间为 $j + p * 2d$, 最小的 $j + p * 2d$,

最后判断 $dis[n-1][T \% 2d]$ 是否小于等于 T 即可。

实际上就是在 30% 的基础上缩减状态。

时间复杂度为 $O(m * d)$ 。