IOI2022 国家集训队作业第一部分试题准备
《序列》解题报告

南京外国语学校　张庭瑞

1　题目大意

1.1　题目描述

有一个长度为 n 的序列 $a_1, a_2, \ldots, a_n$，序列里的每个元素都是 $[1, 10^9]$ 内的正整数。

现在已知了 m 条信息，每条信息形如 i, j, k, x，表示这个序列满足 $a_i + a_j + a_k - \max(a_i, a_j, a_k) - \min(a_i, a_j, a_k) = x$。

请构造一个满足条件的序列。

1.2　输入格式

第一行两个正整数 n, m。

接下来 m 行，每行四个正整数 i, j, k, x，表示一条信息。

1.3　输出格式

如果无解，第一行输出 `NO`。

如果有解，第一行输出 `YES`。第二行输出 n 个正整数 $a_1, a_2, \ldots, a_n$，表示你构造的满足条件的序列 a，你需要保证每个 a_i 都是 $[1, 10^9]$ 内的正整数。

1.4　样例 1

1.4.1　输入

```
6 4
1 3 4 2
1 2 5 6
3 6 6 7
2 4 5 3
```

1.4.2 输出

```
YES
6 8 2 2 3 7
```

1.5 样例 2

1.5.1 输入

```
5 4
1 2 3 4
2 3 4 5
3 4 5 3
1 3 4 6
```

1.5.2 输出

```
NO
```

1.6 数据规模与约定

对于全部数据，$1 \le n, m \le 10^5, 1 \le i, j, k \le n, 1 \le x \le 10^9$。

- 子任务 1（20 分）：$n, m \le 10$。
- 子任务 2（10 分）：$m = \binom{n}{3}$，且对于任意一条信息，$i \ne j, j \ne k, k \ne i$，对于任意一个满足条件的集合 $\{i, j, k\}$，在信息中恰好出现一条。
- 子任务 3（30 分）：$n, m \le 1000$。
- 子任务 4（40 分）：无特殊限制。

时间限制： 1s
空间限制： 512MB

2 解题过程

2.1 算法 1

$a_i + a_j + a_k - \max(a_i, a_j, a_k) - \min(a_i, a_j, a_k) = x$ 相当于是序列 $\{a_i, a_j, a_k\}$ 的中位数为 x。

先 $O(n!)$ 枚举序列 $a_1, \ldots, a_n$ 的相对的大小关系，这样中位数一定就是 a_i, a_j, a_k 中在中间的那个数，条件就变成了形如 $a_i = x$ 的限制，因此直接判断是否有满足条件的序列即可。期望得分 20 分。

2.2 算法 2

把在 m 条信息中出现的所有 x 离散化，如果有解则一定存在只使用到这些 x 的解（否则可以通过调整满足它）。

考虑如何限制 $\{a_i, a_j, a_k\}$ 的中位数为 x：如果 $a_i < x$，那么 $a_j \geq x, a_k \geq x$，如果 $a_i > x$，那么 $a_j \leq x, a_k \leq x$，j, k 同理。用这些限制就可以保证 $\{a_i, a_j, a_k\}$ 的中位数为 x。因为最多只有一个小于 x 的数和一个大于 x 的数，中间的数必然为 x。

观察这个限制，可以使用 2-SAT 建图，令 $X(i, v)$ 为一个 01 变量，若 $X(i, v) = 1$ 则表示 $a_i \geq v$，若 $X(i, v) = 0$ 则表示 $a_i < v$。如果 $X(i, v) = 1$ 那么 $X(i, v-1) = 1$，如果 $X(i, v) = 0$ 那么 $X(i, v+1) = 0$，这两条限制可以保证 $X(i, v)$ 是能够合法的描述 a_i 为某个值的（只需要找到最大的满足 $X(i, v) = 1$ 或者最小的满足 $X(i, v) = 0$ 的位置，就可以确定 a_i 的值）。

将中位数的限制加入图中，“如果 $a_i < x$，那么 $a_j \geq x, a_k \geq x$”对应了“如果 $X(i, x) = 0$ 那么 $X(j, x) = X(k, x) = 1$”，“如果 $a_i > x$，那么 $a_j \leq x, a_k \leq x$”对应了“如果 $X(i, x+1) = 1$ 那么 $X(j, x+1) = X(k, x+1) = 0$”。

直接建图解 2-SAT，对于每个 a_i 都有 m 个点，点数边数都是 $O(nm)$，时间复杂度 $O(nm)$。期望得分 50 分。

2.3 算法 3

对于每个数 a_i，维护一个备选数的集合 S_i。若一条信息为 i, j, k, x，则在集合 S_i, S_j, S_k 中分别都加入元素 x。换句话说，S_i 就是所有涉及到 i 的信息的 x 组成的集合。排除掉 $S_i = \varnothing$ 的情况（因为这种情况的 a_i 可以任选），如果有解则一定存在一种解，满足 $\forall i, a_i \in S_i$（否则也可以通过调整满足）。

使用 2-SAT 建图，但是对于每个 a_i，点数只有 S_i 个，这样总点数就是 $O(m)$ 个。对于每条信息的连边也需要稍作修改，“如果 $a_i < x$，那么 $a_j \geq x, a_k \geq x$”对应了“如果 $X(i, x) = 0$ 那么 $X(j, x) = X(k, x) = 1$”，“如果 $a_i > x$，那么 $a_j \leq x, a_k \leq x$”对应了“如果 $X(i, next(S_i, x)) = 1$ 那么 $X(j, next(S_j, x)) = X(k, next(S_k, x)) = 0$”，其中 $next(S_i, x)$ 表示的是在集合 S_i 中大于 x 的最小的数。

这样点数边数都是 $O(m)$，总的时间复杂度 $O(n+m)$。期望得分 100 分。

3 参考资料

OI-Wiki 条目：2-SAT https://oi-wiki.org/graph/2-sat/