2025 ICPC 网络赛第二场部分题目讲评

isaunoya

Hangzhou Dianzi University

目录

C. Jiaxun!

D. Arcane Behemoths

J. Reconstruct the tree

L. Xor Mirror

附录

目录

C. Jiaxun!

D. Arcane Behemoths

J. Reconstruct the tree

L. Xor Mirror

附录

C：Jiaxun!

正式题目 C。三个学生分配 S 道题，每题必须交给一个会做它的人。非空掩码 $m \in [1,7]$ 从低位起的第 j 位表示学生 $j+1$ 会做（$j=0,1,2$），恰好属于该类型的题有 F_m 道。

最大化三人完成题数的最小值。$T \le 10^5$，$0 \le F_m \le 10^8$，$S = \sum_m F_m \le 7 \times 10^8$。

C：Jiaxun!

正式题目 C。三个学生分配 S 道题，每题必须交给一个会做它的人。非空掩码 $m \in [1, 7]$ 从低位起的第 j 位表示学生 $j+1$ 会做（$j = 0, 1, 2$），恰好属于该类型的题有 F_m 道。

最大化三人完成题数的最小值。$T \le 10^5$，$0 \le F_m \le 10^8$，$S = \sum_m F_m \le 7 \times 10^8$。

只检查每个人会做的题数还不够；一组学生可能在争夺同一批题。要对每个学生子集检查资源是否足够。

C：七种资源上界

假设每人至少分到 x 题。取非空学生集合 U，其中的人共需 $|U|x$ 题。能够供给他们的题，是至少有一个 U 中学生会做的题。因此必须有

$$|U|x \le \sum_{m:m \text{ AND } U\neq 0} F_m.$$

枚举七个非空集合，得到上界

$$A = \min_{U\neq\varnothing} \left\lfloor \frac{\sum_{m:m \text{ AND } U\neq 0} F_m}{|U|} \right\rfloor.$$

单人集合约束个人，两人集合约束共享资源，三人集合给出 $\left\lfloor \frac{S}{3} \right\rfloor$。

C：为什么上界一定可达

把每个学生复制成 x 个需求槽位，每道题仍只有一个顶点。槽位与该学生会做的题连边。

给所有槽位找到互不相同的题，等价于让每人先拿到 x 题。二分图的 Hall 定理指出：这种匹配存在，当且仅当任意槽位集合的邻居数不少于其大小。

任取槽位集合 X，设它涉及的学生为 U。同一学生的槽位具有相同邻居，所以

$$|N(X)| = \sum_{m:m \text{ AND } U\neq 0} F_m \geq |U|x \geq |X|.$$

七个条件保证所有 Hall 条件；剩余题任意分配，不会降低最小完成数。

增广路解释

C：求值不必真的展开题目

实现只需枚举学生掩码 $U = 1, ..., 7$，累加与它按位与非零的 F_m，除以 U 的置位数，取最小值。

每组只有 7×7 次检查，时间和空间均为 $O(1)$。$S = 0$ 时答案自然为零。

如果还要输出分类分配，可以建立七个类型点与三个学生点的容量网络：源到类型容量为 F_m，类型到会做的学生容量为 S，学生到汇点容量为 A。

最大流值为 $3A$，其余题任意分配。这提供构造方式，但本题只求最优值，不需要运行最大流。

目录

C. Jiaxun!

D. Arcane Behemoths

J. Reconstruct the tree

L. Xor Mirror

附录

D：Arcane Behemoths

正式题目 D。选取非空子序列，可以任意安排出售顺序。卖掉一个元素时，它当前的攻击力会加到所有尚未卖掉的已选元素上。

子序列的价值为最优顺序下最后剩余元素的攻击力。求所有非空子序列的价值之和，模 998244353。

$\sum n \le 2 \times 10^5$，$0 \le a_i < 998244353$；相同数值的不同位置仍是不同选法。

D：Arcane Behemoths

正式题目 D。选取非空子序列，可以任意安排出售顺序。卖掉一个元素时，它当前的攻击力会加到所有尚未卖掉的已选元素上。

子序列的价值为最优顺序下最后剩余元素的攻击力。求所有非空子序列的价值之和，模 998244353。

$\sum n \le 2 \times 10^5$，$0 \le a_i < 998244353$；相同数值的不同位置仍是不同选法。

先确定一个固定选集的最优出售顺序，再计算每个元素在全部选集中的系数。

D：出售使已有增量翻倍

固定出售顺序 $b_1, \ldots, b_m$，令 s 为此前出售给剩余元素带来的累计增量。

卖掉 b_i 时，实际攻击力为 $b_i + s$，于是

$$s \leftarrow s + (b_i + s) = 2s + b_i.$$

对 $m \geq 2$，最后剩余的攻击力为

$$2^{m-2}b_1 + 2^{m-3}b_2 + \ldots + 2b_{m-2} + b_{m-1} + b_m.$$

$m = 1$ 时价值就是 b_1。最后两项的系数相同，都为一。

D：交换论证确定最优顺序

设两个位置的系数为 $u \geq v$，所放的攻击力为 $x \geq y$。将较大数放在较大系数处，比反过来多

$$ux + vy - (uy + vx) = (u - v)(x - y) \geq 0.$$

因此按攻击力从大到小出售最优。最后两个系数相同，其顺序不影响价值。

选择子序列等价于选择原位置的子集；出售顺序可以任意安排，故可以先对全部元素排序，再对选集进行计数。

相等元素用任意固定的原下标打破平局，每个位置仍被单独统计。

D：固定一个元素看它的系数

排序为 $a_0 \le \ldots \le a_{n-1}$。固定 a_i，若在前 i 个位置中选了 j 个，它在最优出售顺序后的元素数量就是 j。

它的系数为

$$c(j) = \begin{cases} 1 & j = 0 \\ 2^{j-1} & j \ge 1 \end{cases}.$$

后面 $n-1-i$ 个位置各自选或不选，共 2^{n-1-i} 种，与该系数无关。

前面选择恰好 j 个有 $\binom{i}{j}$ 种。因此 a_i 的总系数为

$$2^{n-1-i} \sum_{j=0}^{i} \binom{i}{j} c(j).$$

D：二项式求和消去枚举

单独处理 $j=0$：

$$\sum_{j=0}^{i}\binom{i}{j}c(j)=1+\frac{1}{2}\sum_{j=1}^{i}\binom{i}{j}2^j.$$

由二项式定理，$\sum_{j=0}^{i}\binom{i}{j}2^j=(1+2)^i=3^i$，所以结果为 $\frac{3^i+1}{2}$。

最终答案为

$$\sum_{i=0}^{n-1}a_i\times 2^{n-1-i}\times\frac{3^i+1}{2}.$$

每个非空选集的每个元素贡献恰好被计入一次，线性求和可直接交换。

D：实现只需排序与幂数组

线性预处理 $2^i, 3^i$。在模 998244353 下，除以二使用逆元 499122177；每步乘法及时取模。

排序 $O(n \log n)$，逐项贡献求和 $O(n)$，空间 $O(n)$。也可复用所有测试的数据规模上界预处理幂。

$n = 1$ 时系数为一，无需访问第二个数组元素；零攻击力不产生贡献。相等数的任意固定排序都得到相同答案。

整个过程先在整数意义下确定最大值，最后计算其模值；不能先将攻击力的中间大小关系取模再比较。

目录

C. Jiaxun!

D. Arcane Behemoths

J. Reconstruct the tree

L. Xor Mirror

附录

J：Reconstruct the tree

正式题目 J。给定 N 个标号点和 M 对互不重复的无序点对，构造一棵树，使所有直径端点对**恰好**等于给定集合，或报告无解。

直径指树中最长简单路径，长度按边数计算。$2 \le N \le 2 \times 10^5$，$1 \le M \le 2 \times 10^5$；多组的 N, M 之和分别不超过 2×10^5。

J：Reconstruct the tree

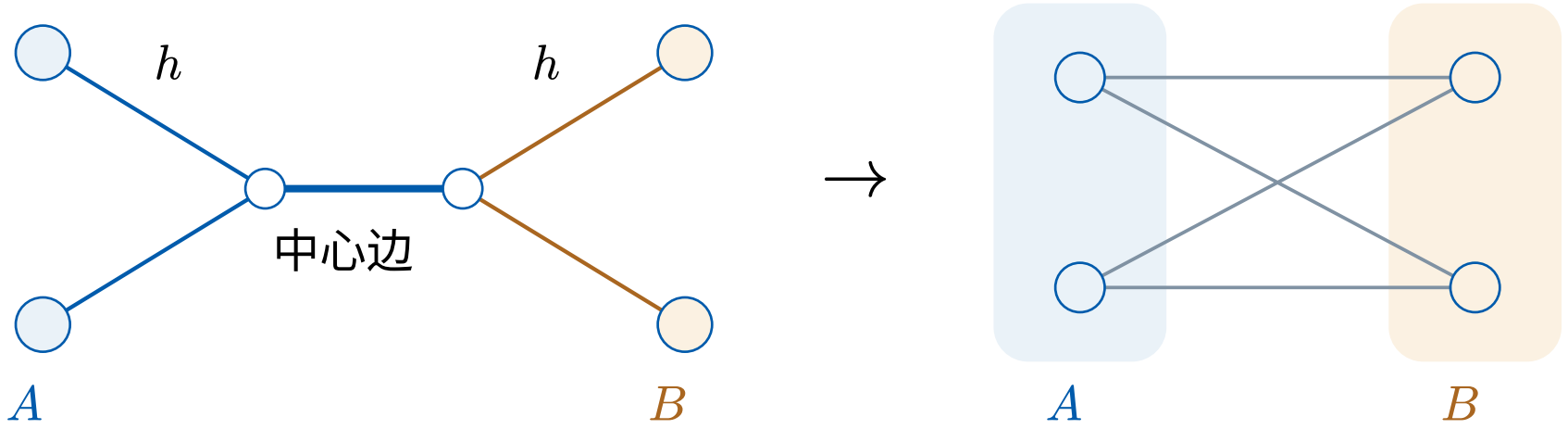
正式题目 J。给定 N 个标号点和 M 对互不重复的无序点对，构造一棵树，使所有直径端点对**恰好**等于给定集合，或报告无解。

直径指树中最长简单路径，长度按边数计算。$2 \le N \le 2 \times 10^5$，$1 \le M \le 2 \times 10^5$；多组的 N, M 之和分别不超过 2×10^5。

把输入点对当作简单图 G 的边。先从树的中心分析：哪些顶点能够同时成为一条直径的两端?

J：奇数直径有一条中心边

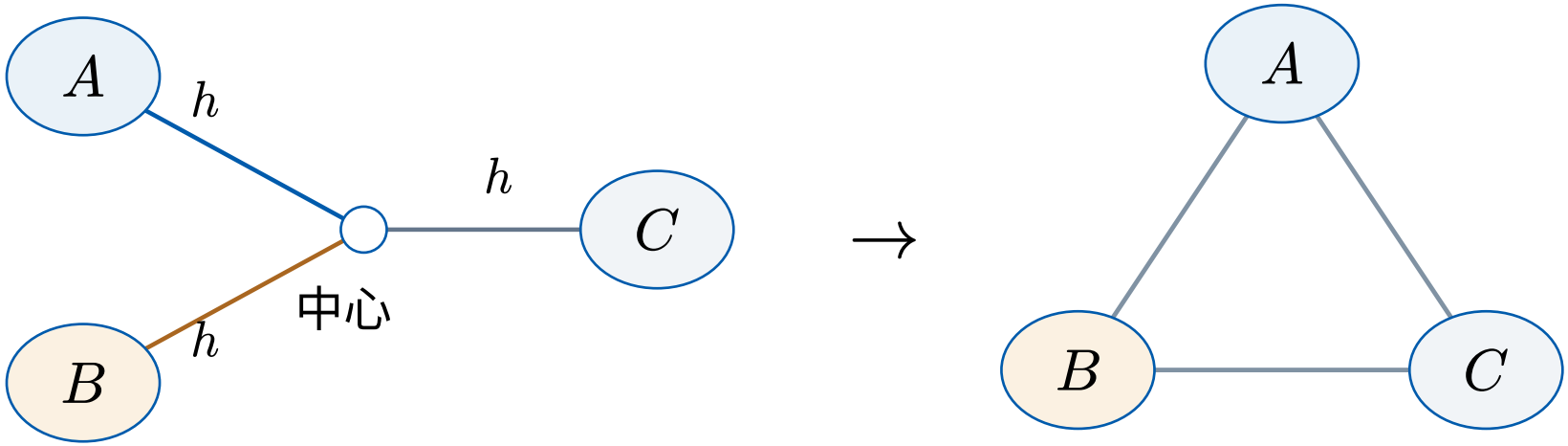
直径长为 $2h+1$。删去中心边后，两侧最深顶点到各自中心端点的距离都是 h。



跨组距离为 $2h+1$，同组距离至多 $2h$，所以端点图恰为两组之间的完全二部图。其他顶点不属于直径端点对，在输入图中度数为零。

J：偶数直径有一个中心点

直径为 $2h$；按中心分支，将深度 h 的端点分组（圆框）。



跨组距离为 $2h$，组内严格小于 $2h$，所以端点图是完全 r 部图，$r \geq 2$。组间连线表示所有端点两两相邻。两组也可能来自偶数直径，组数不能唯一确定中心类型。

J：同组顶点具有相同邻居

完全多部图中，同组顶点的邻居恰好是其他组的所有点，因此邻接集合相同。

不同组的邻接集合不同：取两组中的顶点 u, v，u 属于 $N(v)$，却不属于 $N(u)$。

所以可以按邻接集合分组。给每个顶点独立均匀生成 64 位权值 a_v，计算

$$b_u = \sum_{v \in N_G(u)} a_v \bmod 2^{64}.$$

扫描每条边 (u, v)，分别向 b_u, b_v 加上另一端的权值，然后按 b_u 排序或用有序映射分组。

J：分组之后做确定性核验

用真实度数识别端点，记端点总数为 s。度数为零的点进入内部点池，数量 $z = N - s$；不能用 $b_u = 0$ 代替度数判断。

对指纹得到的每个组 C，检查：

- 每个 $u \in C$ 满足 $\deg(u) = s - |C|$。
- 扫描所有输入边，两个端点必须属于不同组。

第二条保证邻居只能在组外；第一条保证组外所有点都已相邻。两条合起来，输入图就是这些组构成的完全多部图。

核验失败或组数小于二时输出无解。通过后，后面的构造不再依赖随机判等。

J：随机指纹只可能导致误拒绝

对两个固定的不同邻接集合，其指纹之差中至少有一个权值的系数为 1 或 -1。

固定其他权值后，恰有一个 64 位取值会使差为零，因此碰撞概率为 2^{-64}。对全部顶点对取并集界，单组输入的碰撞概率至多

$$\frac{\binom{s}{2}}{2^{64}}.$$

相同集合永远分在同组。没有碰撞时，分组完全正确；有碰撞时，确定性核验仍阻止错误图被接受，只可能拒绝一个有解实例。

如需更小上界，使用两套独立权值，将两个和作为联合键，上界变为 $\frac{\binom{s}{2}}{2^{128}}$。实际实现使用运行时种子。

J：两个组需要多少内部点

$N = 2$ 时，唯一合法树就是一条边。以下假设 $N > 2$。

若 $z \geq 2$，将全部内部点连成一条链；第一组端点接左端，第二组端点接右端。

跨组距离为 $z + 1 \geq 3$，同组距离至多二，内部点到其他点的距离更短，恰好得到所需端点对。

若 $z = 1$，树只能是以该内部点为中心的星形；所有叶子两两成为直径端点。故只有两组各含一个端点时可行。

若 $z = 0$ 则无解，因为 $N > 2$ 的树必有不能成为直径端点的内部顶点。

J：至少三个组时的必要条件

$r \geq 3$ 时中心只能是一个点，直径长度为偶数。

若直径为二，树就是星形，所有非中心点都是直径端点。因此必须是各组均为单点，且 $z = 1$。

若直径至少为四，每个端点组所在的中心分支，都需要至少一个非端点连接中心和这些端点。

不同组使用不同分支，还需一个中心，所以必须有

$$z \geq r + 1.$$

这两个情形覆盖所有可能的直径长度，内部点不够时确实无解。

J：满足条件时如何连边

若各组单点且 $z = 1$，直接用唯一内部点作为星形中心。

否则，在 $z \geq r + 1$ 时，取一个内部点作中心，为每组各取一个内部点作分支根。连接中心与所有分支根，再将各组端点连接到自己的分支根。

剩余内部点全部直接接中心。跨组端点距离四，同组端点距离二，额外内部点到任意端点距离至多三。

所有点连通，每个非中心点恰有一条父边，共 $N - 1$ 条边，因此确实构成树，且直径端点集合没有多出或遗漏。

J：实现流程与复杂度

先存边并统计度数、两套指纹，再对度数非零的点分组。核验组内无边和每点度数后，按组数与内部点数选择构造。

输出前只需维护一份边表；若条件不成立，输出 NO，否则输出 YES 和 $N-1$ 条边。

排序或有序映射分组为 $O(N \log N)$；扫描边、条件判断和构造为 $O(N+M)$。空间为 $O(N+M)$。

对多组数据，碰撞上界相加；复用权值处理所有比较，不意味着这些比较相互独立。确定性核验不需要额外概率假设。

目录

C. Jiaxun!

D. Arcane Behemoths

J. Reconstruct the tree

L. Xor Mirror

附录

L：Xor Mirror

正式题目 L。长度 N 为二的幂，下标从零开始。支持：

- 给出 $[l, r), k$，同时令区间内 A_i 变为修改前的 $A_{i \oplus k}$；区间外不变。
- 给出 $[l, r)$，查询普通区间和 $\sum_{i=l}^{r-1} A_i$。

$\sum N \le 2^{18}$，$\sum Q \le 2 \times 10^5$，$1 \le A_i \le 2^{20}$；$0 \le l < r \le N$，$0 \le k < N$。
空间限制为 64 MB。

L：Xor Mirror

正式题目 L。长度 N 为二的幂，下标从零开始。支持：

- 给出 $[l, r), k$，同时令区间内 A_i 变为修改前的 $A_{i \oplus k}$；区间外不变。
- 给出 $[l, r)$，查询普通区间和 $\sum_{i=l}^{r-1} A_i$。

$\sum N \le 2^{18}$，$\sum Q \le 2 \times 10^5$，$1 \le A_i \le 2^{20}$；$0 \le l < r \le N$，$0 \le k < N$。空间限制为 64 MB。

长度为二的幂的对齐区间，经过下标异或后仍是这样的区间。分块与 Trie 都可以利用这个性质。

L：分块中的异或映射

取块长 $B = 2^b$。把下标与异或值拆成高、低两部分：

$$i = pB + t, \quad k = hB + x, \quad 0 \le t, x < B.$$

于是

$$i \oplus k = (p \oplus h)B + (t \oplus x).$$

目的块 p 只读取源块 $p \oplus h$，块内位置统一异或 x。因此整块操作只需记录“读取哪一块、块内异或多少”，不用逐个移动元素。

块内异或是一个排列，不改变整块总和。已有的块内异或标记直接与 x 合并。

L：整块复用与边界展开

完全位于修改区间内的块，复用旧源块并合并标记；至多两个边界块，逐项读出旧值后重新生成。所有来源均取自修改前，最后统一替换受影响的块。

查询时，完整块直接累加块和，边界块逐项读取。因此修改与查询均为

$$O\left(\frac{N}{B}+B\right).$$

取接近 $\sqrt{N}$、且不超过 N 的二的幂作为块长，即得 $O\left(\sqrt{N}\right)$ 的单次操作。

共享的数据保持不变，局部修改另建块。当前至多使用 $\frac{N}{B}$ 份块数据，一次修改只新建至多两份；及时释放不再使用的块，总空间为 $O(N)$。

L：可持久化 Trie

按下标从高位到低位建立二叉 Trie，叶子存元素，内部节点维护子树总和。每棵子树对应一个长度为二的幂的对齐区间。

对子树下标异或 k，等价于：k 的某一位为一时，在对应层交换左右儿子。给子树引用附加异或标记，就能表示这个排列；总和不变。

旧节点保持不变，修改只新建必要的祖先，并共享其余子树。一个子树可以被多处引用，各自携带不同的异或标记。

这样，整段赋值就可以转为：把目的区间对应的子树，替换成旧树中对应的源子树。

L：沿区间边界拼接

保存旧根；为旧树附加异或 k 的标记，得到源树。源树在位置 i 的值，恰好是修改前的 $A_{i\oplus k}$。

按目的区间，同时沿旧树与源树的逻辑左右儿子递归：

- 与 $[l,r)$ 不相交：保留旧子树。
- 完全包含于 $[l,r)$：直接使用源子树引用。
- 部分相交：递归处理左右两半，再新建父节点拼接。

三种情况逐段实现了所需赋值。所有源节点都来自旧版本，且从不被改写；因此即使源与目的重叠，也不会读到本次修改的新值。

L：路径复制与区间和

查询沿逻辑左右儿子递归，完整覆盖直接取子树总和。修改也只在部分覆盖处继续递归，并新建节点。

连续区间在每层至多有两个部分覆盖节点，树高为 $\log_2 N$。因此一次查询或修改只访问 $O(\log N)$ 个节点，修改只新建 $O(\log N)$ 个节点。

全区间异或只改变根引用的标记。无论共享多少次，子树引用始终直接指向节点，不会形成越走越长的引用链。

若一直保留已分配的节点，空间会增长到 $O(N + Q \log N)$。原题只给 64 MB，还需要限制节点池的大小。

L：定期重建的摊还复杂度

题目不询问历史版本。节点池即将满时，先从当前根导出全部数组，再清空节点池，重新建树；两步均为 $O(N)$。

例如令池容量为 $4N + O(\log N)$。重建后只有 $2N - 1$ 个节点，下一次重建前至少新增 $\Omega(N)$ 个节点。因此每轮重建的代价，可以分摊给这批新增节点。

所有修改通过路径复制共新建 $O(Q \log N)$ 个节点，故修改均摊 $O(\log N)$，查询最坏 $O(\log N)$，空间 $O(N)$；单次触发重建的修改可能花费 $O(N)$。

重建只在两次操作之间进行，必须先读完当前值再清空旧节点。$N = 1$ 时直接维护唯一元素；其他情况均由上述流程处理。

目录

C. Jiaxun!

D. Arcane Behemoths

J. Reconstruct the tree

L. Xor Mirror

附录

C：Hall 条件的增广路解释

从一个不能再增广的匹配出发。若仍有未匹配槽位，从它开始，交替经过“未匹配边到题目、匹配边回槽位”。

如果能到达未匹配题目，就可以翻转整条路径，使匹配数加一，与不能增广矛盾。

否则，所有到达的题目都已匹配，并且它们的匹配槽位也在到达集合里。起点槽位额外未匹配，故到达槽位集合 X 满足 $|N(X)| < |X|$。

因此，只要每个集合都满足 Hall 条件，就不会在匹配全部槽位之前停下。

返回 C 的充分性证明