杂题选讲

A_zjzj

浙江省衢州第二中学

2025 年 3 月 12 日

The 3rd Ucup Stage 21: Ōokayama E. ReTravel

题目描述（Hint：2 min）

给定二维平面中 N 个点 (X_i, Y_i)，你需要按照 $1, 2, \cdots, N$ 的顺序访问这些点。初始你在 $(0, 0)$，同时还有一个字符串初始为空，有以下四种操作：

1. 你向右移动一个单位，在 S 的末尾加入一个字母 `X`；
2. 你向上移动一个单位，在 S 的末尾加入一个字母 `Y`；
3. 你向左移动一个单位，在 S 的末尾删除一个字母 `X`，要求 S 非空且末尾为 `X`；
4. 你向下移动一个单位，在 S 的末尾删除一个字母 `Y`，要求 S 非空且末尾为 `Y`。

求最少要操作多少次 1 或 2 操作才能依次访问所有点？

$1 \le N \le 500$，$0 \le X_i, Y_i \le 10^9$。

Hint （Solution： 2 min）

考虑最终答案的形态，尝试对此设计 dp。

Solution

设 $g_X(l,r) = \min_{i=l}^{r}\{X_i\}, g_Y(l,r) = \min_{i=l}^{r}\{Y_i\}$。

设 $f_{l,r}$ 表示从 $(g_X(l,r), g_Y(l,r))$ 开始，最少需要多少步访问完 $[l,r]$ 区间内的所有点。

转移如下：

$$
\begin{aligned}
f_{l,r} \leftarrow & f_{l,i} + f_{i+1,r} + \\
& (g_X(l,i) - g_X(l,r)) + (g_Y(l,i) - g_Y(l,r)) + \\
& (g_X(i+1,r) - g_X(l,r)) + (g_Y(i+1,r) - g_Y(l,r))
\end{aligned}
$$

边界条件：$f_{i,i} = 0$，答案即为：$f_{1,n} + g_X(1,n) + g_Y(1,n)$。

The 3rd Ucup Stage 25: Hangzhou I. Identify Chord

题目描述（Hint 1：2min）

交互题，有一个 n 元环，边为 $i \leftrightarrow (i \bmod n) + 1$，$i = 1, 2, \cdots, n$。交互库新连接了一条边 $x \leftrightarrow y$，保证 x, y 在环上不相邻。

你需要通过至多 40 次询问找到 x, y。一次询问你可以给出 $u, v \in [1, n] \cap \mathbb{N}$，交互库会返回新图中 u, v 间最短路径的边数。

$4 \leq n \leq 10^9$，每组测试点包含不超过 1000 组数据。

Hint 1（Hint 2：1 min）

设 $f(x,y)$ 表示原图中 x,y 的最短路径长度，$g(x,y)$ 表示新图中的最短路径长度。

若询问得到的 $g(u,v)=f(u,v)$，得到的信息较少。

考虑如何找到一组 (u,v) 使得 $g(u,v)\neq f(u,v)$，已经找到了之后应该如何处理?

Hint 2（Solution：2 min）

设 $f(x,y)$ 表示原图中 x,y 的最短路径长度，$g(x,y)$ 表示新图中的最短路径长度。

若询问得到的 $g(u,v)=f(u,v)$，得到的信息较少。

考虑如何找到一组 (u,v) 使得 $g(u,v)\neq f(u,v)$，已经找到了之后应该如何处理?

询问什么样的一对点，使 $g(u,v)\neq f(u,v)$ 的概率尽可能大?

Solution：找到 u, v

每次询问 $f(u,v)=\lfloor \frac{n}{2} \rfloor$ 的一对 u, v。

对于 n 是偶数：询问 $(1, \frac{n}{2}+1), (2, \frac{n}{2}+2), (3, \frac{n}{2}+3), (4, \frac{n}{2}+4)$。

对于 n 是奇数：询问
$(1, \frac{n+1}{2}), (1, \frac{n+1}{2}+1), (2, \frac{n+1}{2}+1), (2, \frac{n+1}{2}+2), (3, \frac{n+1}{2}+2)$。

容易证明，这几组询问中，一定存在一组满足 $g(u,v) \neq f(u,v)$。

Solution：找答案

若已经找到 u, v 满足 $g(u, v) \neq f(u, v)$。

则考虑 u 到 v 的最短路径，从 u 出发，先用两次询问得出 u 是答案的一端或 u 出发到 v 的最短路径应先向哪边走。

若 u 不为答案的一端，则进行二分，得到答案的一端。

最后通过距离，可得答案的另一端只可能在 $v-d$ 或 $v+d$ 上，再多查询一次即可。

注意一些交互细节。

The 3rd Ucup Stage 22: Zhengzhou A. $A + B = C$ Problem

题目描述（Hint 1：2 min）

给定 p_A, p_B, p_C，你需要找到三个无限 01 字符串 A, B, C，满足 $A \oplus B = C$，且 A, B, C 的**最小**循环节长度分别为 p_A, p_B, p_C。

$1 \leq p_A, p_B, p_C \leq 10^6$。

Hint 1（Hint 2：1 min）

若 $\gcd(p_A, p_B, p_C) > 1$，能否先除去它们的公约数?

Hint 2（Hint 3：1 min）

显然，$A \oplus B$ 的最小循环节一定为 $\operatorname{lcm}(p_A, p_B)$ 的因数。

即 $p_C | \operatorname{lcm}(p_A, p_B)$。

观察 p_A, p_B, p_C 满足什么条件?

Hint 3（Solution：3 min）

当 $\gcd(p_A, p_B, p_C) = 1$ 时，有解的 (p_A, p_B, p_C)，一定存在 $a, b, c \in \mathbb{N}^*$ 满足 a, b, c 两两互质且 $p_A = bc, p_B = ca, p_C = ab$。

Solution

由于 $A_i = A_{i \bmod p_A}$，发现 $i \bmod bc$ 保留了 $i \bmod b$ 和 $i \bmod c$ 的性质。

故我们构造 $A_i = [i \bmod b = 0] \oplus [i \bmod c = 0]$，$B_i, C_i$ 同理。

容易证明，这是符合条件的方案。

The 2nd Ucup Stage 22: Hangzhou L. Master of Both V

题目描述（Hint 1：2 min）

有 n 次操作，每次操作有两种：

- 加入一条线段，端点为 $(px, py), (qx, qy)$，$(px, py) \neq (qx, qy)$；
- 删除第 i 次操作加入的线段，保证第 i 次操作为加入操作，且第 i 次操作加入的线段还未删除。

你需要在每次操作过后，判断是否存在一个凸包，使得当前所有的线段都在凸包的边界上。

$1 \leq n \leq 5 \times 10^5$，坐标绝对值 $\leq 10^9$。

Hint 1（Hint 2：3 min）

如果直接使用线段树分治维护，则时间复杂度至少是 $O(n\log^2 n)$ 的，无法接受。

考虑通过某种方式优化。

Hint 2（Solution：3 min）

首先，对于某个时刻没有线段的情况，是容易处理的。

接下来考虑时刻都有线段的情况，考虑定期重构，每次拿出当前时刻最晚删除的一个线段作为基准线段。

当基准线段被删除时，重新找一个基准线段并直接重构。

容易发现，一个线段至多只会被重构一次，时间复杂度不会改变，仅仅是多了两倍的常数。

Solution

有了基准线段，就能够判断出每个线段应该是在上凸壳还是下凸壳中。

每次插入删除时判断一下前驱后继之间是否符合条件，使用 set 即可维护。

即便如此，实现起来也并不容易，具体细节需要同学们针对自己的写法考虑清楚。

The 3rd Ucup Stage 21: Ōokayama F. Origami Warp

题目描述（Hint：2 min）

二维平面中给定 N 条直线，每条直线给出其经过的两个**不同**的**整点** (a_i, b_i) 和 (c_i, d_i)。一次操作你可以选择一条直线，将你此时的位置对称到该直线的另一边。Q 次询问，每次询问给出你的初始位置 (X_i, Y_i) 和目标位置 (Z_i, W_i)，你需要判断你是否能够在有限次操作内到达目标位置。

更精确地说，我们称 S 能够在有限次操作内到达 P，当且仅当 $\forall \varepsilon > 0$，存在 Q 满足 S 能够在有限次操作内到达 Q，且 $|PQ| \le \varepsilon$。

$2 \le N \le 2000$，$1 \le Q \le 2000$。

$(a_1, b_1, c_1, d_1) = (0, 0, 1, 0)$，$(a_2, b_2, c_2, d_2) = (0, 0, 0, 1)$。

每条直线互不相同，$-10^8 \le a_i, b_i, c_i, d_i \le 10^8$。

Hint（Property：4 min）

如果只有两条相交的直线，一个点可以在有限步内到达哪些点?

Property

如果只有两条相交的直线，一个点可以在有限步内到达哪些点?

如果这两条相交的直线的夹角不为 45° 的倍数，那么一个点可以到达所有以两直线交点为圆心的圆上所有点。

因此，只要有不经过原点且仰角不为 45° 倍数的直线，那么所有询问都是 `Yes`。

Solution

只要有不经过原点且仰角不为 45° 倍数的直线，那么所有询问都是 `Yes`。

否则，所有直线要么经过原点，要么仰角为 45° 的倍数。

接下来就是一顿分类讨论。

若所有直线都经过原点，当存在仰角不为 45° 倍数的直线时，只需判断到原点的距离是否相等即可；否则需要翻转到 x 轴正半轴和 $y = x$ 在第一象限的部分围成的区域，检查是否重合。

否则，若不存在仰角为 45° 或 135° 的直线，那么横纵坐标就独立了，分别求一下 $\gcd$，检查是否等价即可。

否则，求出所有直线截距的 $\gcd$，先对 $2\gcd$ 取模，在进行翻转的等价操作，判断是否重合。

此题细节较多，可以参考我的代码：https://qoj.ac/submission/813617。

The 3rd Ucup Stage 25: Hangzhou D. Dividing Sequence

题目描述（Solution 1： 5 min）

给定一个长度为 n 的序列 A，你需要将 A 恰好划分为两个子序列 B, C，使得 B 的字典序不超过 C，求出字典序最小的 C。

$1 \le n \le 5000$。

Solution 1

考虑逐位确定答案，若当前答案为 ans。

每次维护 f_i 表示原串的前 i 个字符，能否划分成 ans 和 ans 的前缀。

每次判断下一个字符能否接在 ans 的前缀的下一个字符上进行转移。

再判断是否存在一个字符小于 ans 的前缀的下一个字符，如果有，答案就为 ans。

否则找到所有可能的后继字符的最小的那个，转移所有最小的那个转移。

Hint（Solution 2：5 min）

接下来考虑线性算法，那就需要用到更多的字符串性质——Lyndon 分解。

Solution 2

若原串 Lyndon 分解的结果为 $w_1 \geq w_2 \geq \cdots \geq w_k$。

每次取出 w_1, w_2，若 $w_1 = w_2$，则 B, C 分别在末尾加入 w_1，进入下一轮；否则 $w_1 > w_2$，则将 w_1 接在 C 末尾，剩余的都接在 B 末尾，此时的 C 即为答案。

说明

首先需要一个重要的引理：对于 Lyndon 串 s，不可能分成两个子序列 a, b 使得 $a < s$ 且 $b < s$。

这个引理的证明并不容易，经过和方心童同学的讨论，我们认为官方题解的证明是假的，并得出了正确的证明方式。

引理的证明

下面是简要证明：

- 考虑一个 Lyndon 串 s，假设 $< |s|$ 的 Lyndon 串都满足条件了。
- 将 s 做一遍 Duval 分解，发现 s 可以拆分为 $u^k vw$，其中 u 是 Lyndon 串，v 是 u 的非空真前缀，w 是一个字符且 $vw > u$。
- 接着拿出来一个 u，考虑可以怎么分配，发现只能放到一起，继续拿出剩下的 u 也是这样；
- 最后的 vw 同样可以证明是 Lyndon 串，发现也只能放到一起，所以 s 也符合，归纳即可。

有了这个引理，就不难说明该做法的正确性了。

The 2nd Ucup Stage 22: Hangzhou K. Card Game

题目描述（Hint： 2 min）

给定一个长为 n 的序列 $a_1, a_2, \cdots, a_n$。q 次询问，每次询问给出 l, r，求出 $a_l, a_{l+1}, \cdots, a_r$ 进行接龙游戏后序列中剩余的元素个数。

接龙游戏：对于序列 $b_1, b_2, \cdots, b_m$，维护一个初始为空的序列。第 i 轮将 b_i 加入序列末尾，若此时序列中存在两个 b_i，则删除它们和它们中间的所有元素。m 轮结束后，序列中剩余的元素个数即为答案。

$1 \le n, q \le 3 \times 10^5$，$1 \le a_i \le n$。

Hint（Solution：3 min）

设 p_i 表示 i 右边第一个和 i 相同的元素，若没有，设为 $n+1$。

$$ans(l,r)=\begin{cases}0 & l>r\\ ans(p_l+1,r) & p_l\le r\\ ans(l+1,r)+1 & p_l>r\end{cases}$$

考虑这个东西应该怎么维护?

直接使用主席树维护一个 l 对应的所有 r 的答案。

那么只需要支持前后缀合并（实际上就是线段树分裂的逆过程）和区间 $+1$ 即可。

当然，也存在其他的做法，但不如该做法简洁优美。

The 2nd Ucup Stage 22: Hangzhou C. Yet Another Shortest Path Query

题目描述（Hint： 4 min）

给定一个 n 个点的无向图，每条边有边权，保证该图是平面图。q 次询问，每次询问给出 s_i, t_i，求出 s_i 到 t_i 至多经过三条边的最短路，或判断不存在这样的路径。

$2 \le n \le 10^6$，$1 \le q \le 10^6$，$12s$。

Hint（Solution：3 min）

由于平面图至多只有 $< 3n$ 条边，故总度数 $< 6n$，所以一定存在一个点度数 ≤ 5。每次找到这么一个点，删去这个点和相邻的所有边（平面图删掉一些点之后仍然是平面图)。

接着，我们像拓扑序一样给点一个删除顺序。然后，将无向边拆成两个有向边。对于一条边 $u \to v$，若 u 先删除，则称为 R 边；否则称为 L 边。

容易发现，从每个点出发的 R 边、到达每个点的 L 边均至多只有 5 条。

Solution

接着，对于一组询问，不妨设 s 先于 t 删除，则有以下几种情况：

- 第一条边为 R 边：转化为至多 5 个 $k \le 2$ 的问题；
- 最后一条边为 L 边：转化为至多 5 个 $k \le 2$ 的问题；
- LR/LRR：离线询问，固定 s，枚举 L 边，再向右枚举 R 边；
- LLR：和上一种同理，固定 t 即可。

至此，我们就在 $O(n+q)$ 的时间复杂度内解决了该问题，虽然常数较大。

The 3rd Ucup Stage 18: Southeastern Europe E. Shrooks

题目描述（Hint：3 min）

有一个 $n \times n$ 的网格图，你需要在其中放置 n 个石子，满足：

- 每行每列恰好有一个石子；
- 任意两个石子的曼哈顿距离不超过 n。

给定 $a_1, a_2, \cdots, a_n$，若 $a_i = -1$，表示第 i 行仍未放置石子；否则表示第 i 行的石子放置在 a_i 列。求出在当前局面下有多少种符合条件的继续放置剩余的石子的方案。对 998244353 取模。

$2 \le n \le 2 \times 10^5$。

Hint（Solution： 4 min）

首先，考虑将曼哈顿距离转化为切比雪夫距离。

这样，只需保证两个方向的坐标极差不超过 n 即可。

Solution

对于 n 的奇偶性分类讨论，这部分需要不少画图，用数学语言表示也没有画图清晰直接，~~这里就不写啦~~。

参考代码：https://qoj.ac/submission/775241。

The 3rd Ucup Stage 14: Harbin F. 1D Galaxy

题目描述（Hint 1： 1 min)

在数轴上有 n 个行星，第 i 个行星质量为 w_i，0 时刻的位置为 x_i。每一单位时间，所有行星将会**同时**进行如下操作：

- 设第 i 个行星在 t 时刻的位置为 $x_{i,t}$；
- 设 $w^l_{i,t} = \sum_{x_{j,t}<x_{i,t}} w_j$，$w^r_{i,t} = \sum_{x_{j,t}>x_{i,t}} w_j$。
- 则 i 在 $t+1$ 时刻的位置 $x_{i,t+1}$ 满足：

$$x_{i,t+1} = \begin{cases} x_{i,t} - 1 & w^l_{i,t} > w^r_{i,t} \\ x_{i,t} & w^l_{i,t} = w^r_{i,t} \\ x_{i,t} + 1 & w^l_{i,t} < w^r_{i,t} \end{cases}$$

q 次询问，每次询问给出 t, i，求出 $x_{i,t}$。$n, q \leq 10^5$，值域 10^9。

Hint 1 （Hint 2： 2 min）

如果两个行星在某一时刻到达了同一位置，那么它们之后不可能再分开。

于是，就可以把这两个行星合并成一个。

Hint 2（Solution：3 min）

假设其他行星均在 $-\infty$ 或 $+\infty$ 的位置，考虑两个行星互相靠近，最后一步走到了对方的位置上，接下来会发生什么?

Solution

假设其他行星均在 $-\infty$ 或 $+\infty$ 的位置，考虑两个行星互相靠近，最后一步走到了对方的位置上，接下来会发生什么？

此时，要么一个行星不再移动、要么两个行星重复地左右摇摆。也就是说，不存在两个行星都继续背向移动的情况。

Solution

这样，我们就可以把行星合并、行星跨越与两星摇摆当成一个事件。

把所有行星按照当前时刻的位置，用链表串起来。并使用优先级队列维护接下来会发生的事件。

当一个行星不动时，接下来就是被其他行星合并。加之合并不超过 $n-1$ 次，故事件总数是 $O(n)$ 的。

实现细节

一些实现上的细节：

- 记录所有行星的位置，当前时刻改变时，无需改变所有行星的位置。记录每个行星在之前某一时刻的位置和移动方向即可；
- 优先级队列中的事件，如何判断是否仍然有效？对于每个事件记录两个行星信息修改的时间戳，每次判断时间戳和记录的是否相等，若不相等，说明该事件无效；
- 两星摇摆的情况，需要特殊处理：当别的行星打破这种平衡时，及时取消这种绑定关系。

参考代码：https://qoj.ac/submission/709303，算是比较短的代码了。

The 3rd Ucup Stage 15: Chengdu H. Friendship is Magic

题目描述（Hint：1 min）

对于 $n \in [10, +\infty) \cap \mathbb{N}$，定义 $f(n)$ 表示：

- 将 n 写成不带前导零的十进制数 S；
- 将 S 拆分为两个非空子串（此时可以有前导零），求出两个子串代表的十进制数差的绝对值 w；
- $f(n)$ 即为所有拆分方案中 w 的最小值。

给出 l, r，求出 $\sum_{i=l}^{r} f(i) \bmod 998244353$。

$10 \le l \le r \le 10^{18}$。

Hint（Solution：6 min）

首先对于询问差分，计算 $\sum_{i=1}^{n} f(i)$。

考虑从左到右枚举每个拆分点，容易发现，左边的权值递增，右边的权值不升。

所以差是单调的，故差的绝对值会在中间某个地方取到最低点。

Solution

考虑找到左边减去右边的差恰好从 ≤ 0 变为 > 0 的分界点。

设 $n = (10A + x)10^b + B, x \in [0, 10), B \in [10^{b-1}, 10^b)$，则有

$$\begin{cases} A \leq 10^b x + B \\ 10A + x > B \end{cases}$$

即 $A - 10^b x \leq B < 10A + x$。

Solution

考虑枚举 x, b 和 A 的位数 a。

那么，讨论 $10^b x + B - A$ 和 $10A + x - B$ 的大小关系，以及 A, x 是否顶到 n 的上界，计算 B 的范围，再对于所有情况求和。

式子非常复杂，但是次数是 $O(1)$ 的，所以可以考虑使用拉格朗日插值简便计算，只需计算单点贡献，插出所有值的和即可。

参考代码：https://qoj.ac/submission/699913。

The 3rd Ucup Stage 1: St. Petersburg I. Spin & Rotate!

题目描述（Property 1：4 min）

有一个正方形 $ABCD$，初始有两根绳子分别连接 A,B 和 C,D。有如下两种操作：

- S 操作：将原来的点 A 重命名为 B，原来的点 B 重命名为 C，原来的点 C 重命名为 D，原来的点 D 重命名为 A；
- R 操作：解开 A,D 与其连接的绳子，将原来与 D 连接的绳子从下面连接到 A，将原来与 A 连接的绳子从上面连接到 D。

现在给定一个仅包含 S,R 的字符串（长度不超过 3×10^5），表示预先进行的操作序列，请你找到一个操作次数最少的操作序列，使得两根绳子回到初始状态（分别直接连接 A,B 和 C,D）。

Hint：尝试找到一些等价的操作。

Property 1（Property 2：4 min）

发现 R 和 SSR 本质相同。

所以答案中一定不会出现连续的两个 S。

换句话说，任何状态进行 SS 操作之后都是它本身。

那么，还有什么操作有也这样的性质呢?

Property 2

任何状态进行 RSRSRS 操作同样也都是它本身。

Solution

至此，我们就能够得到可行解了：$\texttt{S}$ 用 $\texttt{S}$ 抵消，$\texttt{R}$ 用 $\texttt{SRSRS}$ 抵消。

得到一个可行解之后，显然可以把子串 $\texttt{SS}, \texttt{RSRSRS}$ 都删掉。

但是，这样仍然还有缩减的空间：

- 末尾的 $\texttt{RSR}$ 可以删去；
- 末尾的 $\texttt{SR}^k\texttt{S}$ 也可以删去（如果只有 $\texttt{R}^k\texttt{S}$，则可以在之前插入一个 $\texttt{SS}$ 再删去)。

至此，剩下的串就是答案了。

至于证明，参考官方题解，一些有关拓扑学定理并没有给出证明，有兴趣的同学可以自行寻找答案。

结语

感谢聆听