

团队

2020.07.30 [XVI Open Cup named after E.V. Pankratiev. GP of Ukraine](#) pro: 11/11/13 rk: 35/409

2020.07.27 [2020牛客暑期多校训练营（第六场）](#) pro: 7/8/11 rk: 27/1019

2020.07.25 [2020牛客暑期多校训练营（第五场）](#) pro: 6/10/11 rk: 24/1116

个人

zzh

专题

无

比赛

无

题目

无

pmxm

专题

无，哦

比赛

- 2020/7/24 [SRM 788](#) problems: 1/2/3 rank: 102/190
- 2020/7/25 [ABC: M-SOLUTIONS Programming Contest 2020](#) problems: 5/6/6 rank: 58/6527

题目

无哦

jsh

比赛

- 2020/7/24 [SRM 788](#) problems: 1/2/3 rank: 112/190
- 2020/7/24 [Codeforces Round #659 \(Div. 1\)](#) problems: 2/2/6 rank: 220/1169
- 2020/7/25 [ABC: M-SOLUTIONS Programming Contest 2020](#) problems: 5/6/6 rank: 281/6527
- 2020/7/29 [Educational Codeforces Round 92 \(Rated for Div. 2\)](#) problems: 6/6/7 rank: 55/13826
- 2020/7/30 [Codeforces Round #660 \(Div. 2\)](#) problems: 5/5/5 rank: 24/13083

专题

无

题目

无

本周推荐

zzh

[XVI Open Cup named after E.V. Pankratiev. GP of Ukraine J. Joining Powers](#)

Tags□number theory, binary search

题意：见链接

题解：见链接

Comment：有点意思的数论，如果你看着指数最多 \$60\$，想着去暴搜，你就输了。

pmxm

Atcoder M-SOLUTIONS Programming Contest 2020 problem E

<https://atcoder.jp/>

Tags□search

题意：见链接

题解：见链接

Comment : 考虑清楚 $3^n, 2^n * n^2$ 等复杂度和如何优化查询

jsh

Codeforces Round #660 (Div. 2) - E. Uncle Bogdan and Projections

[题目链接](#)

Tags : 几何 Convex Hull Trick

题意 : 有 $1 \leq n \leq 2,000$ 条在 x 轴上方、平行于 x 轴的线段。现在你可以让这些线段以相同的一个方向平移到 x 轴上，就像一个平行光源打到了这些线段上，然后在 x 轴上投影。但是要求投影之间不能相交。记投影的范围为最大坐标和最小坐标的差，求投影之间不相交情况下，最小的投影范围是多大。

题解

首先，如果线段都在同一个水平线上，那投影的范围是不会变的。所以以下我们考虑存在有至少一对线段，所在高度是不同的。

记投影方向与 y 的负半轴的角度为 $\theta \in (-\pi, \pi)$ 容易得出点 (x, y) 的投影的横坐标为 $f_{x, y}(\theta) = x + y \tan\{\theta\}$

让 $u = \tan\{\theta\} \in \mathbb{R}$ 改写一下函数 $g_{x, y}(u) = x + y u$ 题目需要的就是在投影之间不相交情况下，最小化所有线段上的点的函数值的最大值和最小值的差，相当于线段端点的函数值的最大最小差。

一个显然的想法是取线段投影没有相交、但是存在几个投影恰好相切时候的 u 来对答案进行更新。因为在投影之间完全不相交也不相切的情况下，角度逐渐偏左或偏右会让投影之间变化到恰好相切，根据函数的定义容易知道在取恰好相切的时候是能够取到极值点的（极大或极小都有可能，但总有一个点会比没有相切的情况要好）。

好，接下来需要处理的是，在计算存在有投影相切的时候，如何知道其它的投影是没有相交情况的。容易发现一对高度不同的线段，会在某个特定的角度区间出现投影相交的情况，而求出来区间的左右端点刚好是这两个线段投影相切时候的角度。那我们 $\mathcal{O}(n^2)$ 处理出来所有的区间（可以用分数类），让端点为关键点，排个序 $\mathcal{O}(n^2 \log n)$ 然后在扫描线过程中不计算存在有投影相交情况下的关键点即可。

你会想说单关键点计算时间复杂度 $\mathcal{O}(n)$ 不就爆了吗。

会发现 $g_{x, y}(u)$ 都是些直线，应用一下 [Convex Hull Trick](#)，我们就能在 $\mathcal{O}(\log n)$ 的时间内计算 $\mathcal{O}(n)$ 个线性函数，在某个自变量下的最大值啦。

时间复杂度 $\mathcal{O}(n^2 \log n)$

Comment : 为数不多能现场 AK 的比赛，写一个 E 题的题解纪念一下。

From: <https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link: https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:intrepidword:2020.07.24-2020.07.30_%E5%91%A8%E6%8A%A5&rev=1596186783 

Last update: **2020/07/31 17:13**