

问题1：奶牛跨栏 (cowjump)

题意

二维平面上有 N 条线段，去除一条后就没有线段相交，求应该去掉的线段。

思路

使用叉乘可以判断线段相交，通过枚举任意一对线段并判断是否相交，找到相交次数最多的线段，就是应该删除的，但这样做效率较低。

实际上，我们只需要找到任意一对相交的线段，其中必然有一条是应该被删除的。

可以采用扫描线的思想来寻找这对线段，考虑有一条垂直的扫描线从左往右进行扫描，对扫描线按顺序穿过的线段进行观察，如果线段没有相交，则随着扫描线的移动，这些被穿过的线段的顺序是不会发生变化的，但每经过一个交点，相交的这对线段之间的顺序就会发生交换，并且这对线段一定是相邻的。所以，只需要检查每一对相邻过的线段，就能找出这对相交的线段。

我们可以先将每条线段分成左右两个端点（垂直线段随意划分），然后将所有端点按照 x 坐标排序，然后按顺序检查这些端点，并维护一个端点的集合。如果遇到一个左端点，则将它插入集合中，并检查插入位置上下两个相邻端点对应的线段和当前线段是否有交。如果遇到一个右端点，则将它对应的左端点从集合中删除，并检查删除位置上下两个相邻端点对应的线段是否有交。

可以使用 STL 中的 `set` 来有序的维护端点的集合。

复杂度

时间

排序 $O(N \log N)$ 。

扫描端点 $O(N)$ ，维护 $O(\log N)$ ，总共 $O(N \log N)$ 。

统计交点 $O(N)$ 。

总时间复杂度为 $O(N \log N)$ 。

空间

记录线段 $O(N)$ 。

问题2：疯狂的栅栏 (crazy)

题意

给定平面中多个不相交的简单多边形和一些点，求分割出的区域中包含的最多点数。

思路

由于给出的是一些无序线段，所以要先找到多边形。考虑到线段数量很少，所以可以每次从一个没有访问过的线段开始，不断暴力找相邻的线段，直到得到完整的多边形。这样就能找到所有的多边形。

然后需要知道多边形中包含了哪些点，可以使用射线法，让点发出的射线和多边形的每个线段求交，根据交点的奇偶性就能判断是否在多边形中。

但是我们不能知道找到包含点数最多的多边形，因为多边形中还有可能包含多边形，导致区域被进一步划分。对于两个点，只有包含这两个点的多边形集合完全相同，才能确保两个点在一个区域中。为此，可以将点按照包含点的集合排序，再进行校验即可。

复杂度

时间

寻找多边形，边数 $O(N)$ ，寻找邻边 $O(N)$ ，总共 $O(N^2)$ 。

判断点和多边形线段交 $O(NC)$ 。

点比较 $O(N)$ ，排序 $O(NC \log C)$ 。

总时间复杂度为 $O(N^2 + NC \log C)$ 。

空间

记录点线段交的情况 $O(NC)$ 。

问题3：三角形计数（tricount）

题意

平面上给定一些点，求能构成多少个包含原点的三角形。

思路

由于要包含原点，所以可以考虑从原点观察这些三角形。

对于一个包含了原点的三角形，原点与任意点构成的直线都会将另外两点分开。对于一个没有包含原点的三角形，只有一个点与原点构成的直线会将另外两个点分开。由此可以发现，统计不包含原点的三角形数量更加简单，然后用三角形总数减去它就得到了包含原点的三角形数量。

要寻找不包含原点的三角形，只需要枚举三角形中的一个点，它与原点构成的直线会将平面分成两半，然后在相同半边中任选两个点即可。因此，通过统计两个平面中的点数，就可以直接计算出三角形的数量。

为了快速的枚举与统计，可以先将所有点进行极角排序，然后按照极角序来枚举点，并维护半平面的分界点，分界点可以用双指针进行维护。

注意每个三角形会被统计两次，最后的结果要除以 2。

复杂度

时间

极角排序 $O(N \log N)$ 。

枚举点 $O(N)$ ，维护半平面分界点总共 $O(N)$ ，计算 $O(1)$ ，总共 $O(N)$ 。

总时间复杂度为 $O(N \log N)$ 。

空间

记录点 $O(N)$ 。