程序设计艺术与方法学

第三讲　计算几何

合肥工业大学计算机与信息学院　2014-10

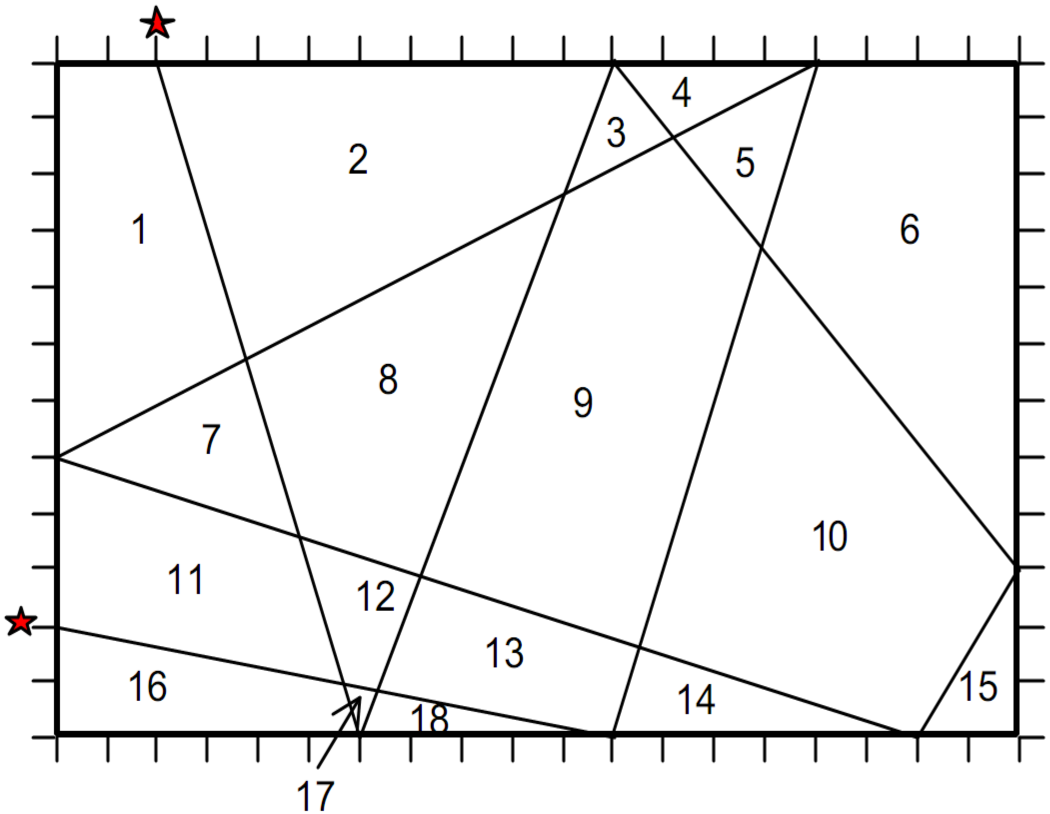
3.1 线段的性质

3.2 点集的性质

题目：地主分田

地主有块矩形土地，临死前向分给几个孩子，他在农场地图上划上一些直线，这些直线总是从矩形边界上的某一点到另一边接上的点，一条直线的终点是下一条直线的起点。不会有三线共点。他划分完后想数一下分出多少块土地，但这个地主老眼昏花，容易数错，因而向你求助。

线段个数L<=1000，请你编程求解

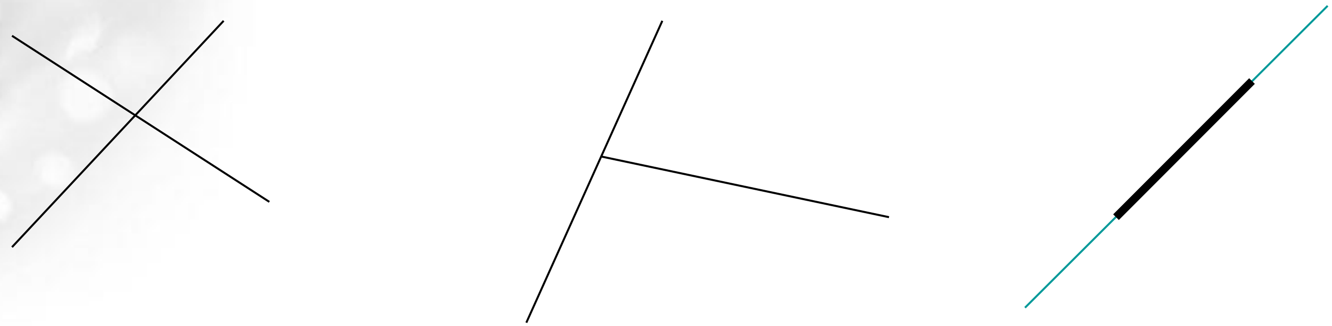


3.1 线段的性质

- 线段是几何中除去点后最基本的几何元素
- 是解决几何问题用到的最常用的几何元素
- 拟解决的两个问题：
  - **1 方向性问题**

    给出两个有公共起点的向量p_0p_1和p_0p_2，判断以p_0为中心，从p_0p_1转到到p_0p_2是顺时针方向还是逆时针方向
  - **2 两个线段是否相交**

    判断平面的两条线段是否相交

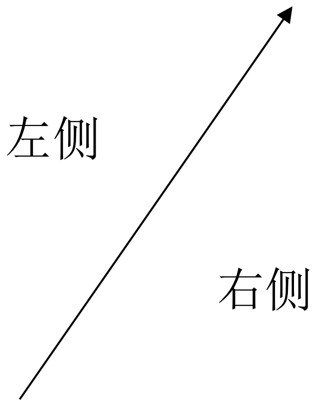


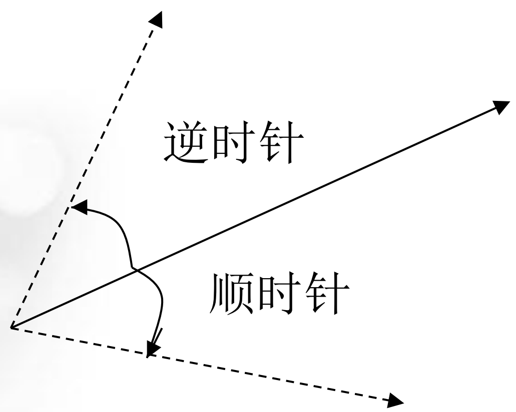
解析几何解法

- 两点式直线方程：$ax + by + c = 0$
- 解方程:
  - 无解则表示平行
  - 无穷多解表示共线
  - 唯一解，再判断交点是否在两个线段内部
- 缺点:
  - 浮点误差，用到浮点除法
  - 计算复杂

计算几何解法

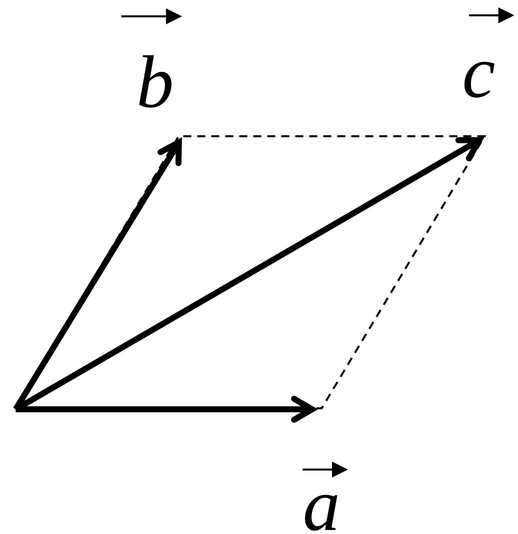
- 有向线段:
- 侧、左侧、右侧:
- 逆时针、顺时针:





3.1 线段的性质

叉积



$$|\vec{a}\times\vec{b}|=|\vec{a}||\vec{b}|\sin\alpha$$

$$p_1\times p_2=\det\begin{pmatrix}x_1 & x_2\\ y_1 & y_2\end{pmatrix}=x_1y_2-x_2y_1=-p_2\times p_1$$

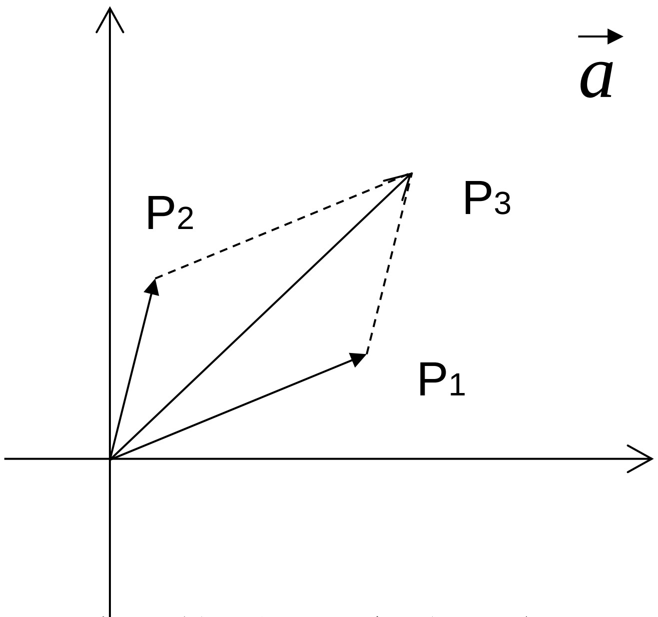
两个有公共起点的向量的叉积，

如果为正则由转到为逆时针方向，

如果为负则为顺时针方向。

特别的如果为零，则两向量共线，

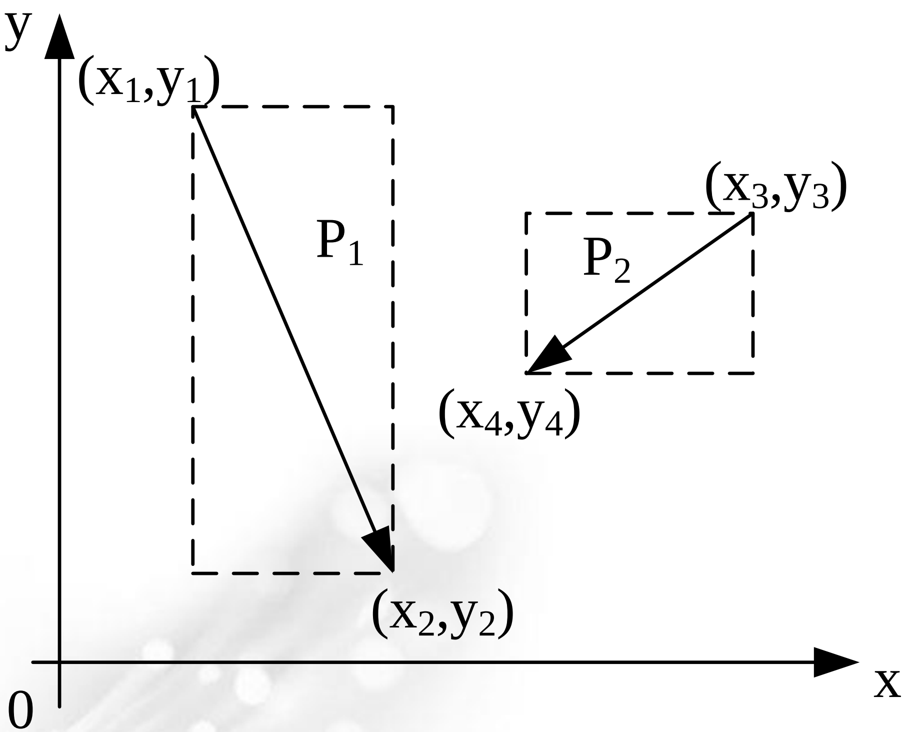
可以进一步去判断共向的还是反向的

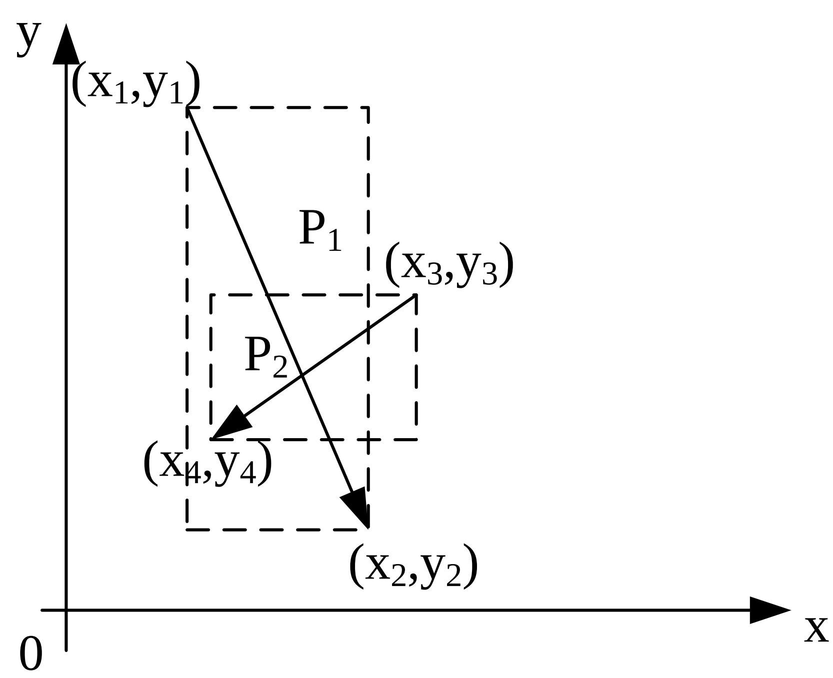


叉积和有向面积：叉积的绝对值等于平行四边形的面积，保留正负号即为有向面积，按逆时针为正，反之为负

线段是否相交

- 快速排斥试验：





线段是否相交

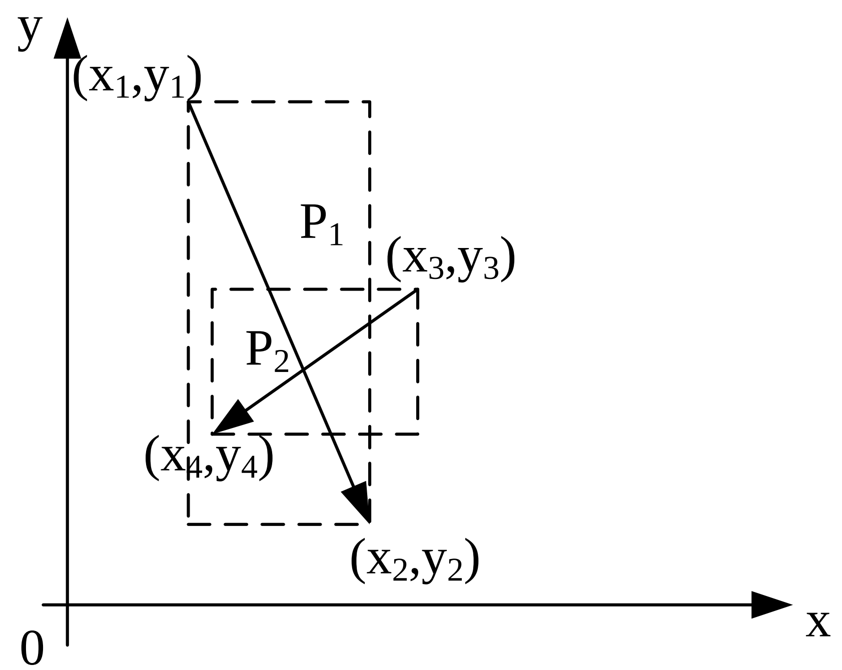
- 快速排斥试验：
- 对于第一个线段P_1
  - 左下角的点为：$(\min(x_1,x_2),\min(y_1,y_2))$；
  - 右上角的点为：$(\max(x_1,x_2),\max(y_1,y_2))$；
- 对于第二个线段P_2
  - 左下角的点为：$(\min(x_3,x_4),\min(y_3,y_4))$；
  - 右上角的点为：$(\max(x_3,x_4),\max(y_3,y_4))$；

线段是否相交

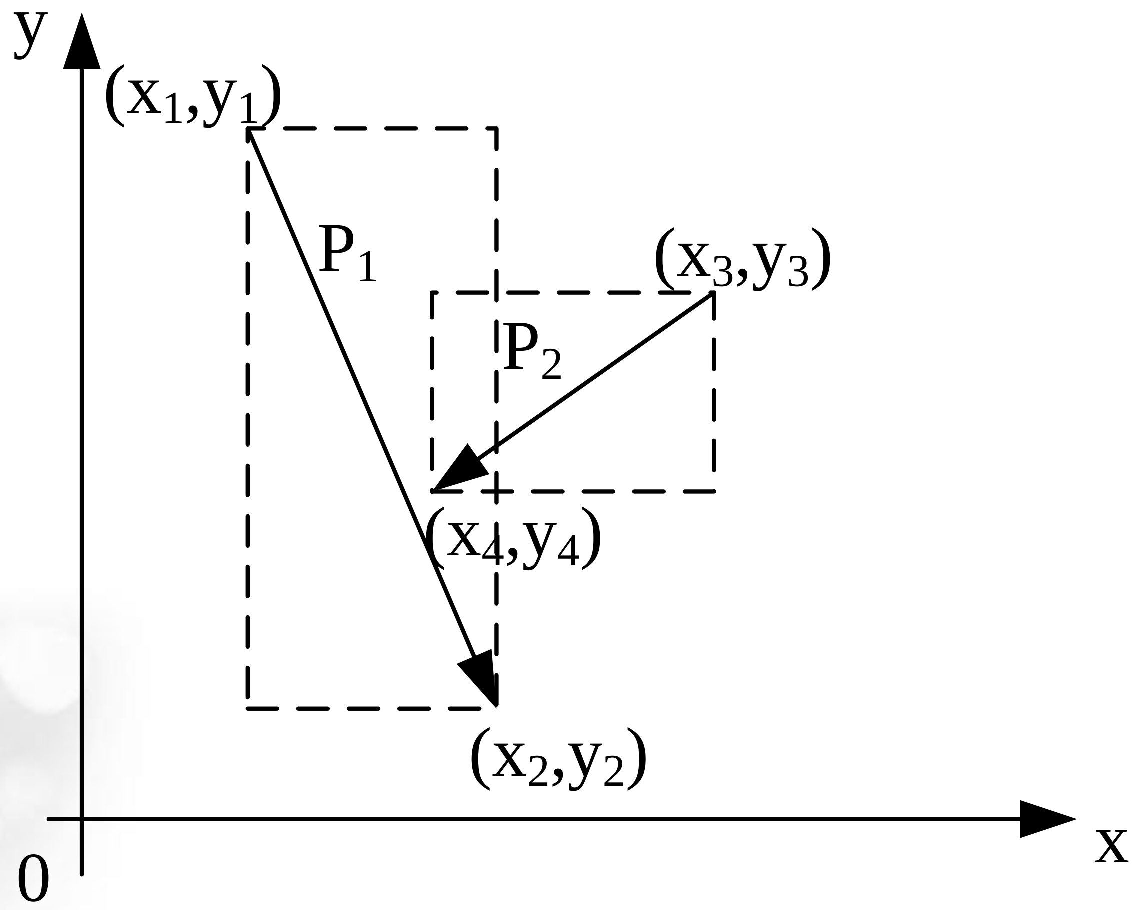
- 快速排斥试验：
- P_1右上角的x坐标大于等于P_2左下角的x坐标并且P_1左下角的x坐标小于等于P2右上角的x坐标；

  $(\min(x_1, x_2) <= \max(x_3, x_4))$ &&
  $(\min(x_3, x_4) <= \max(x_1, x_2))$
- P_1右上角的y坐标大于等于P_2左下角的y坐标并且P_1左下角的y坐标小于等于P_2右上角的y坐标；

  $(\min(y_1, y_2) <= \max(y_3, y_4))$ &&
  $(\min(y_3, y_4) <= \max(y_1, y_2))$



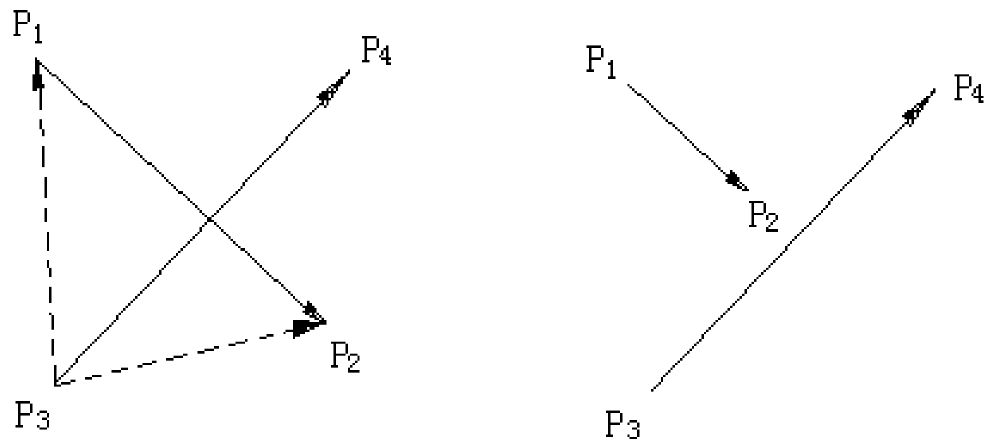
线段是否相交



线段是否相交

- 跨立：如果一条线段的一个端点在一条直线的一边，另一个端点在这条直线的另一端，我们就说这条线段跨立在这条直线上；
- 如果两条线段互相跨立，它们一定是相交的；
- 线段相交满足且只需满足如下两个条件就可以了：
  1. 两条线段相互跨立；
  2. 一条线段的一个端点在另一条线段上（针对边值情况）。

3.1 线段的性质



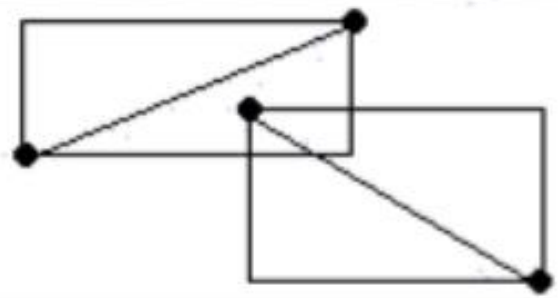
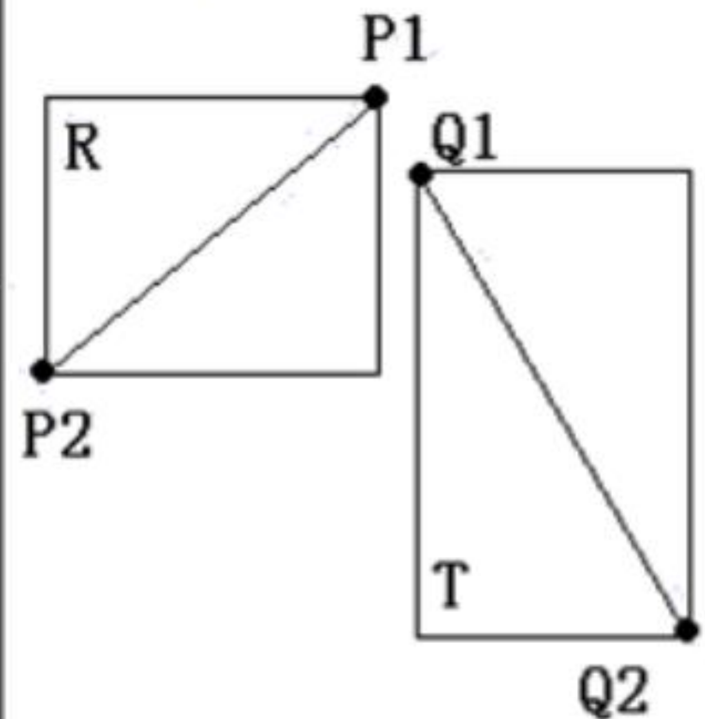
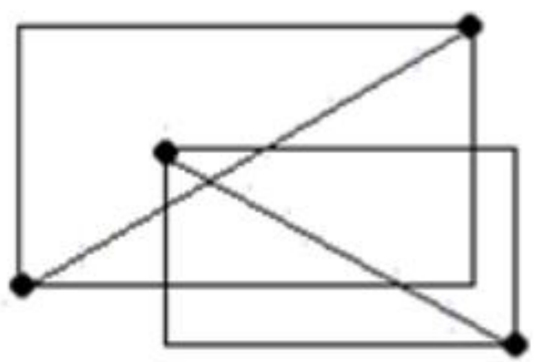
-线段向量P_1P_2是跨立在P_3P_4上的

-取端点P_3为虚线所示向量的起点，P_1和P_2为终点，则向量组P_3P_1、P_3P_4 和向量组P_3P_2、P_3P_4的叉积符号是相反的

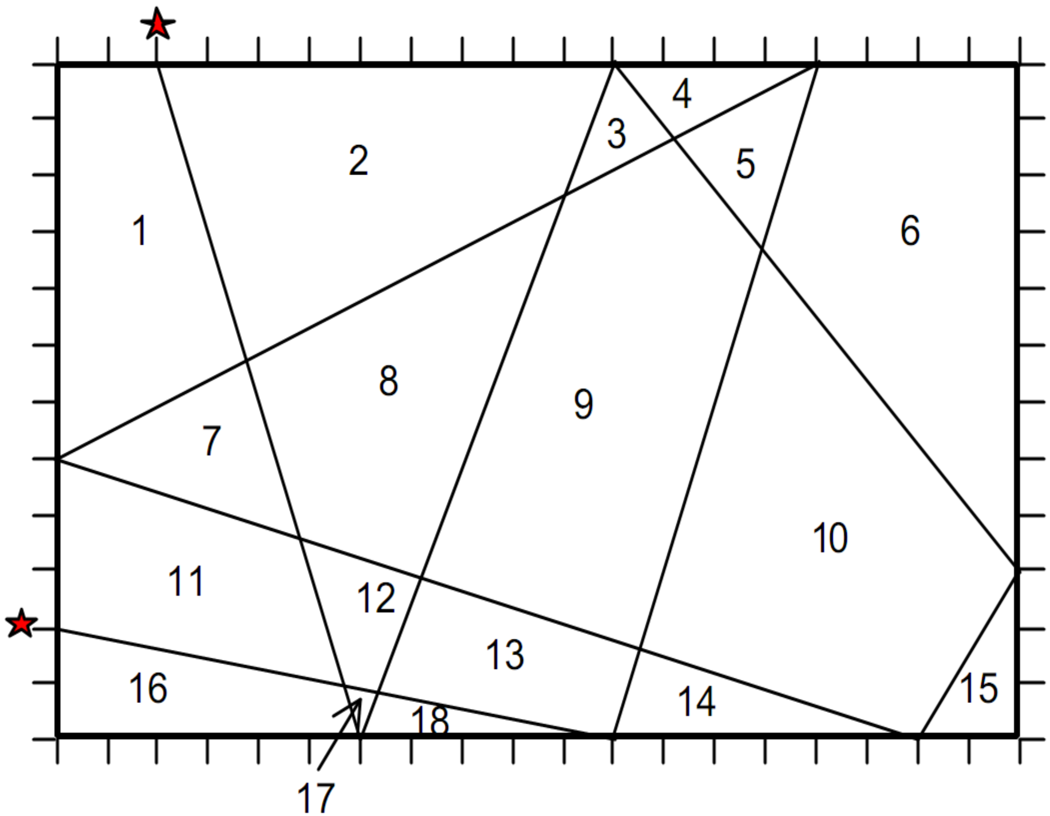
-同样可以判断出P_3P_4是跨立在P_1P_2上的

一些说明：特殊情况可以用点积辅助判断，叉积说明左右情况，点积说明前后情况，参见《算法艺术与信息学竞赛》

3.1 线段相交总结

<table>
<tr><th></th><th>通过快速排斥实验</th><th>未通过快速排斥实验</th></tr>
<tr><td>未通过跨立实验</td><td></td><td rowspan="2">
</td></tr>
<tr><td>通过跨立实验</td><td></td></tr>
</table>

题目：地主分田



输入： 第1行输入地图的宽度w和高度h(1<=w,h<=1000)，第2行输入线段L(1<=L<=1000)。以下L+1行每行一个坐标(Xi,Yi)，庄主画的线段为(Xi,Yi)(Xi+1,Yi+1)，i=1,2,…,L。(Xi,Yi)必定在矩形的边界上。

输出： 一个整数，为划分出的土地块数。

解题思路

A） 增加一条线，则增加一块地
B） 增加一个点，则增加一块地

因而需要求取所有交点个数（不在边界上的）

两线跨立成立

题目：机器蛇

在未来的某次战争中，我军计划了一次军事行动，目的是劫持敌人的航母。计划中要将数百条机器蛇投放到航母的各个角落里。由于航母内部舱室、管线错综复杂，且大部分由金属构成，因此屏蔽效应十分强烈，况且还要考虑敌人的大强度电子干扰，如何保持机器蛇间的联系，成了一大难题。每条机器蛇的战斗位置由作战计划部门制定，将会及时通知你。每条机器蛇上都带有接收、发射系统，可以同时与多条机器蛇通讯。由于整个系统承载的数据量庞大，需要一个固定的通讯网络。情报部门提供了极其详尽的敌方航母图纸，使你对什么地方有屏蔽了如指掌。

请你设计一个程序，根据以上信息构造通讯网络，要求信息可以在任意两条机器蛇间传递，同时为了避免干扰，通讯网络的总长度要尽可能的短。

题目：机器蛇

【输入】第一行是一个整数n（$n \le 200$）表示参战的机器蛇总数。以下n行，每行两个整数x_i，y_i，为第i支机器蛇的战斗位置。接下来一行是一个整数m（$m \le 100$）表示航母内部可能产生屏蔽的位置。最后m行，每行四个整数a_i，b_i，c_i，d_i，表示线段$(a_i，b_i)$-$(c_i，d_i)$处可能有屏蔽，也就是说通讯网络不能跨越这条线段。

【输出】输出数据应仅包括一个实数，表示建立的通讯网的最短长度，保留3位小数。如果不能成功建立通讯网，请输出-1.000。

解题思路

题目中要求信息可以在任意两条机器蛇间传递，通信网络的总长度要尽可能的短，显然这是一个求图的最小生成树问题。但问题的关键是构图，因为如果以机器蛇为顶点，以通信线路为边，则每一条边都不能与任一条屏蔽线相交。

其他和线段相关的计算

- 计算两条相交线段的交点
- 判断任意一组线段中是否存在相交的情况
- 计算线段的中垂线方程
- 计算凸多边形的重心位置和面积
- 寻找最近点对
- ……

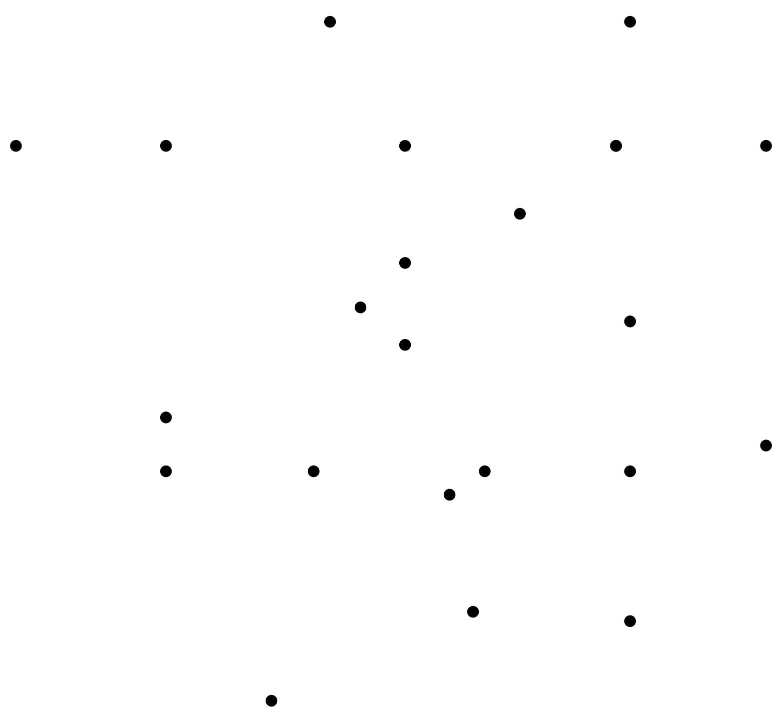
3.1 线段的性质

3.2 点集的性质

题目：地主圈地

地主有一片不规则的树林，经常遭到一些小动物的破坏，于是地主就想给这片树林围一圈篱笆墙以保护他的树林，但这个地主比较吝啬，想用最短的篱笆墙围起这片树林，聪明的你给他计算一下最短的篱笆墙的长度吧。

点的个数n<=10000，请你编程求解

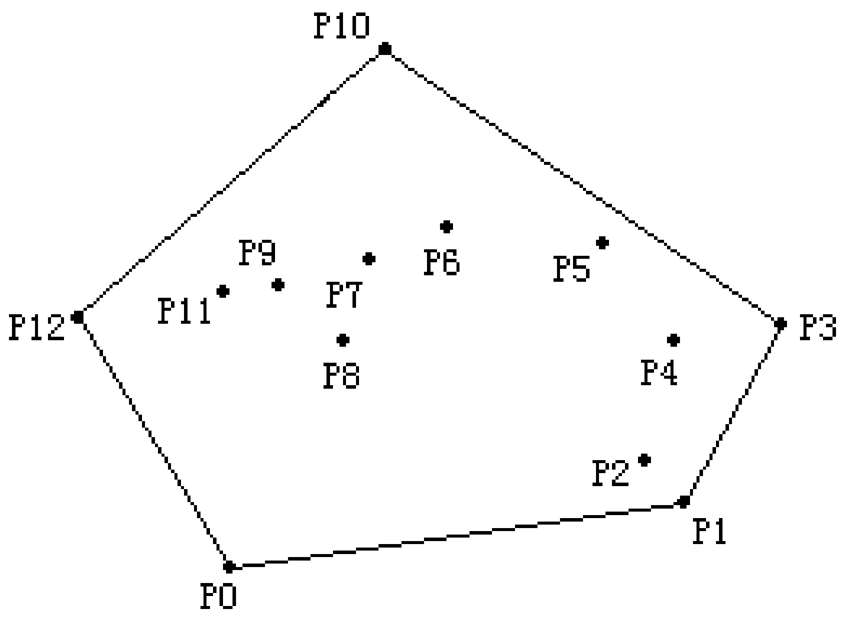


凸包

❖凸包是一组点集中的子集，这一子集形成的凸多边形可以将点集中所有的点都围住，并且这一凸多边形的面积是最小的。

如图所示：凸包为点P10、P3、P1、P0和P12

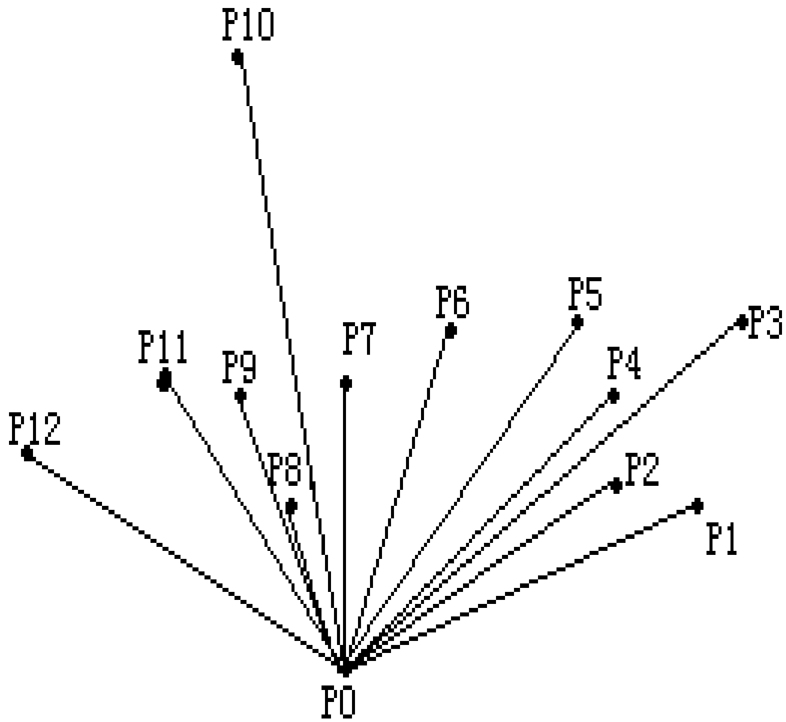
CH（Q）={P10、P3、P1、P0，P12}

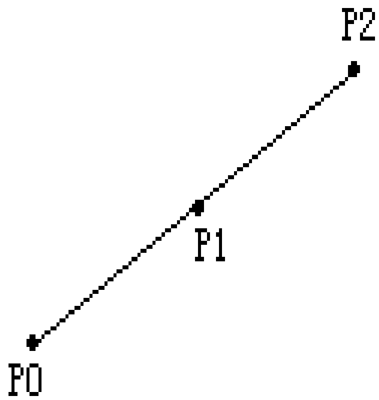


寻找凸包的方法

- 首先，找出点集中最下方的点，如果这样的点不止一个，就选用最左边的点（如P0）
- 显然，点（P0）是凸包子集中的一个点
- 设想在P0处拴了一根皮筋 ，放在和P0成水平位置的右侧
- 将皮筋，沿逆时针方向转动，首先会碰到P1
- 以P1为中心，做和P0一样的事，将碰到P3
- 一直这样做下去，直到再一次遇到P0，凸包就被找出来了

寻找凸包的方法





—— 找到P0点之后，以P0 为每个矢量的起点，其它的点为矢量的终点，来比较任意两个矢量的转角，就可以对余下的点进行按极角排序

——先看向量P0P1和P0P2，由于从P0P1转向P0P2为逆时针，于是认为点P1大于点P2

——对于右图中的特殊情况，比较的结果应该是P2大于P1

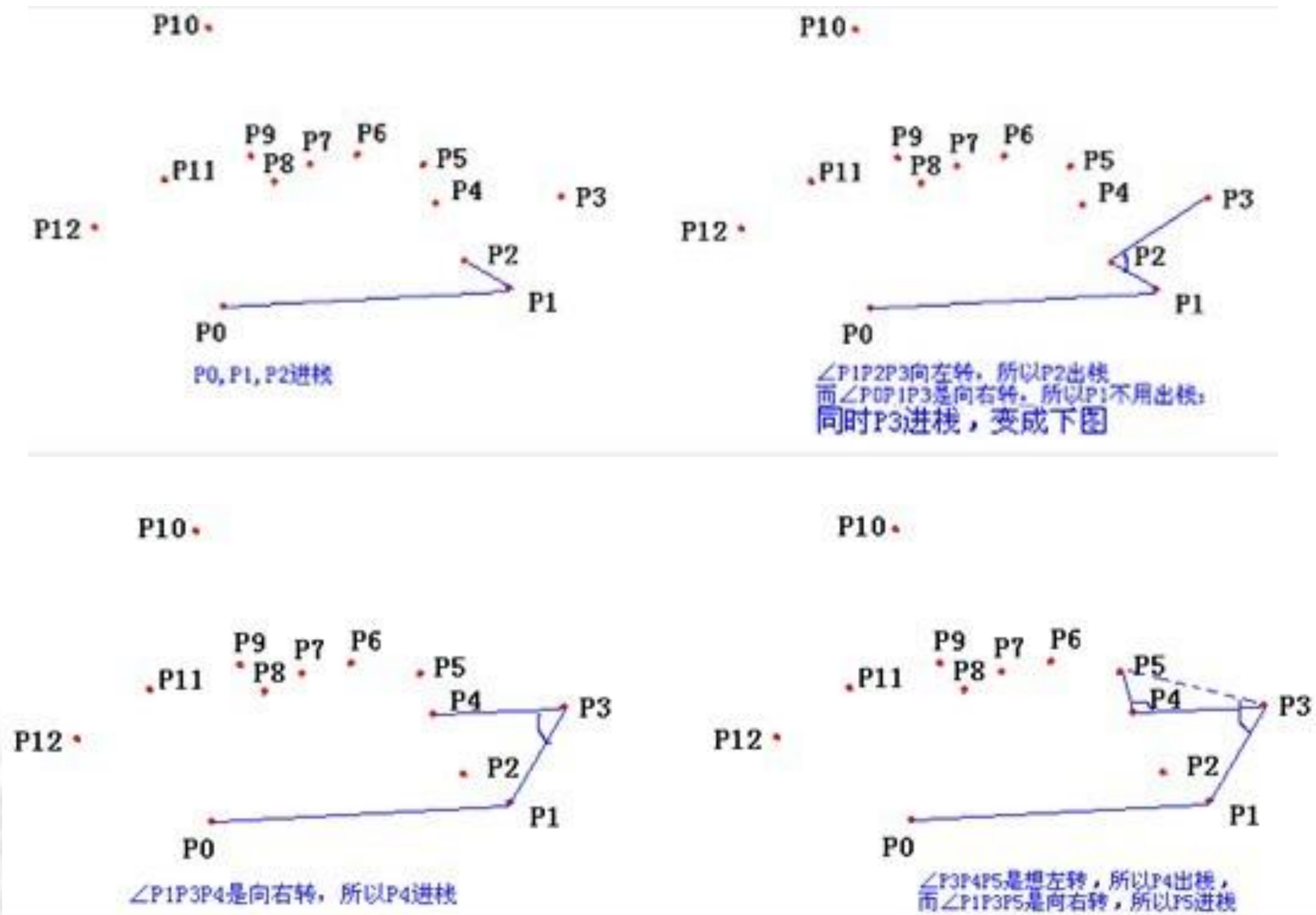
Graham扫描法

- 令p0为Q中Y-X坐标排序下最小的点
- 设<p1,p2,...pm>为对其余点按以p0为中心的极角逆时针排序所得的点集（如果有多个点有相同的极角，除了距p0最远的点外全部移除）
- 压p0进栈S
- 压p1进栈S
- 压p2进栈S

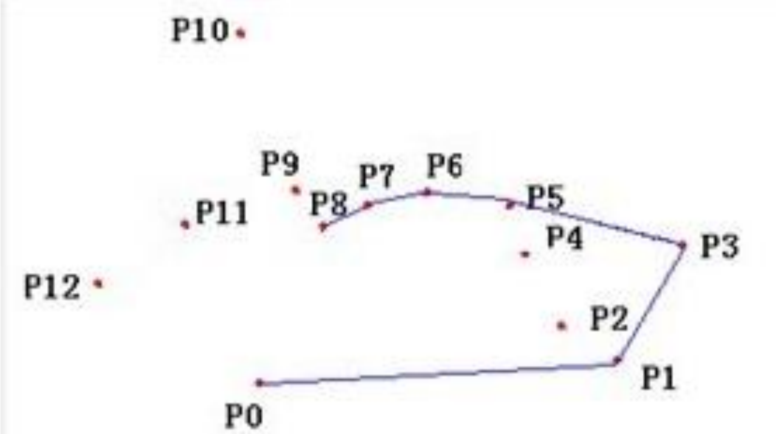
```
for i ← 3 to m
   do while （由S的栈顶元素的下一个元素、S的栈顶元素
            以及pi构成的折线段不拐向左侧）
        对S弹栈
      压pi进栈S
return S;
```

- 效率提高到O(nlogn)

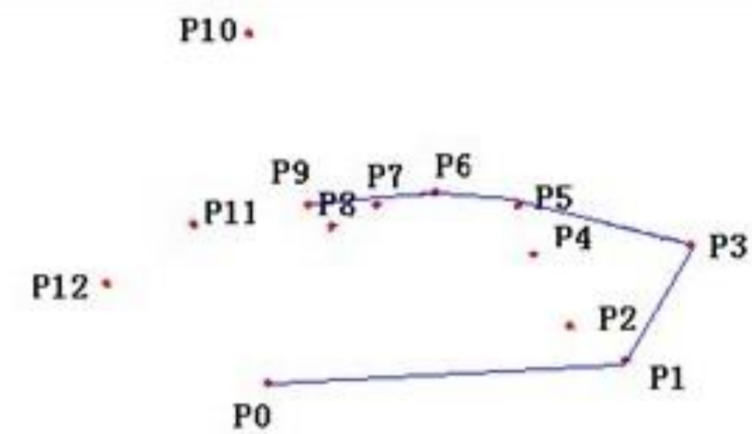
Graham扫描法


P0,P1,P2进栈
∠P1P2P3向左转，所以P2出栈
而∠P0P1P3是向右转，所以P1不用出栈；
同时P3进栈，变成下图
∠P1P3P4是向右转，所以P4进栈
∠P3P4P5是想左转，所以P4出栈，
而∠P1P3P5是向右转，所以P5进栈

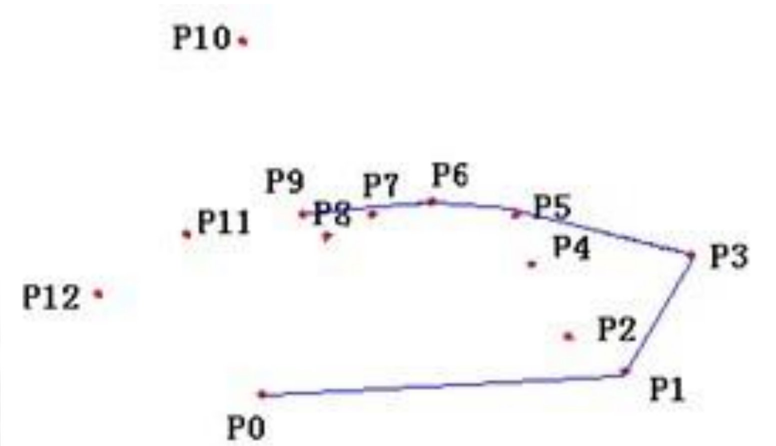
Graham扫描法



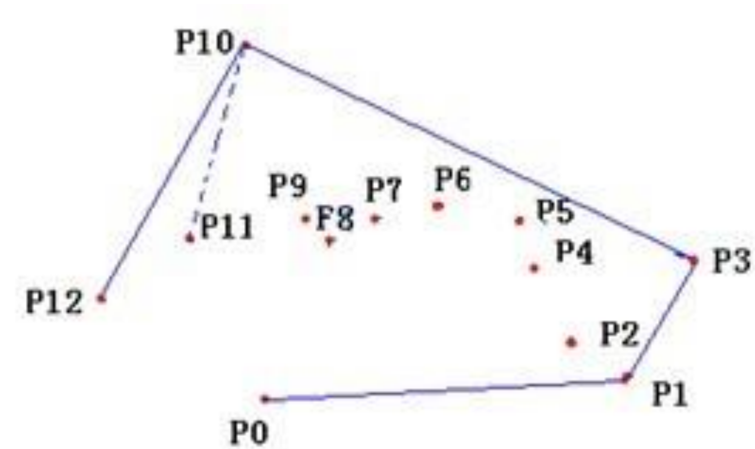
同理，会出现P6、P7、P8进栈



遇到P9，会现P8、P7出栈，而P9进栈

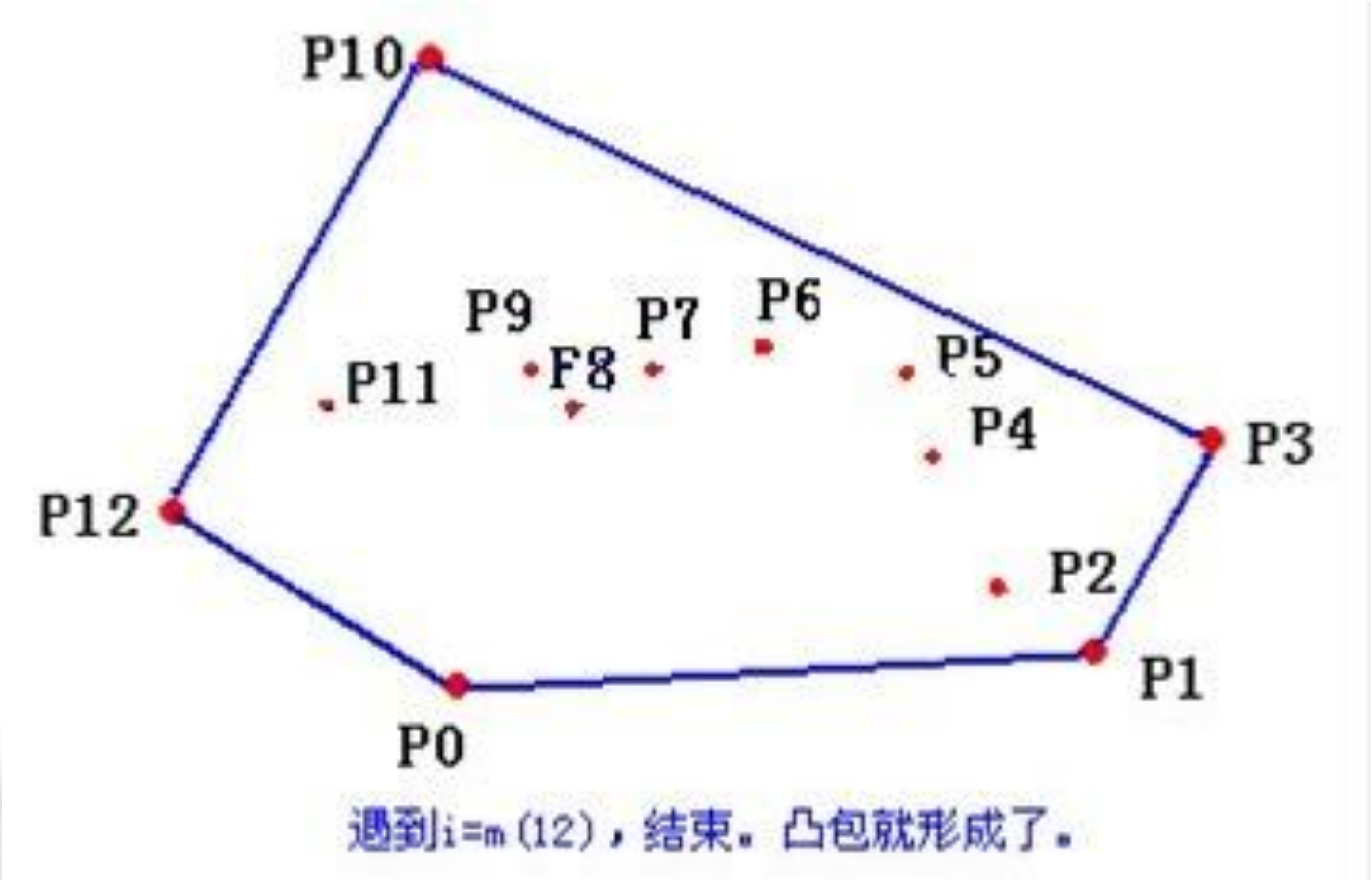


遇到P10，所以P9、P6、P5依次出栈
而P10进栈



遇到P11，P11进栈
但遇到P12后，P11又出栈，而P12进栈

Graham扫描法



巨人和魔鬼(例2)

n个巨人正与n个鬼进行战斗，每个巨人的武器是一个质子炮，它可以把一串质子流射中鬼而把鬼消灭。质子流沿直线行进，在击中鬼时就终止。巨人决定采取下述策略。他们寻找鬼配对，以形成n个巨人—鬼对。然后每个巨人同时向他选取的鬼射出一串质子流。我们知道，让质子流互相交叉是很危险的。因此巨人选择的配对方式应该使质子流都不会交叉。假定每个巨人和每个鬼的位置都是平面上的一个固定点，并且没有三个位置共线，求一种配对方案。

巨人和魔鬼

输入：第一行为n，接下来的n行为巨人信息，其中第i+1行为巨人i的坐标(xi,yi)。最后n行为魔鬼信息，其中第n+i+1行为魔鬼i的坐标(xi,yi)。

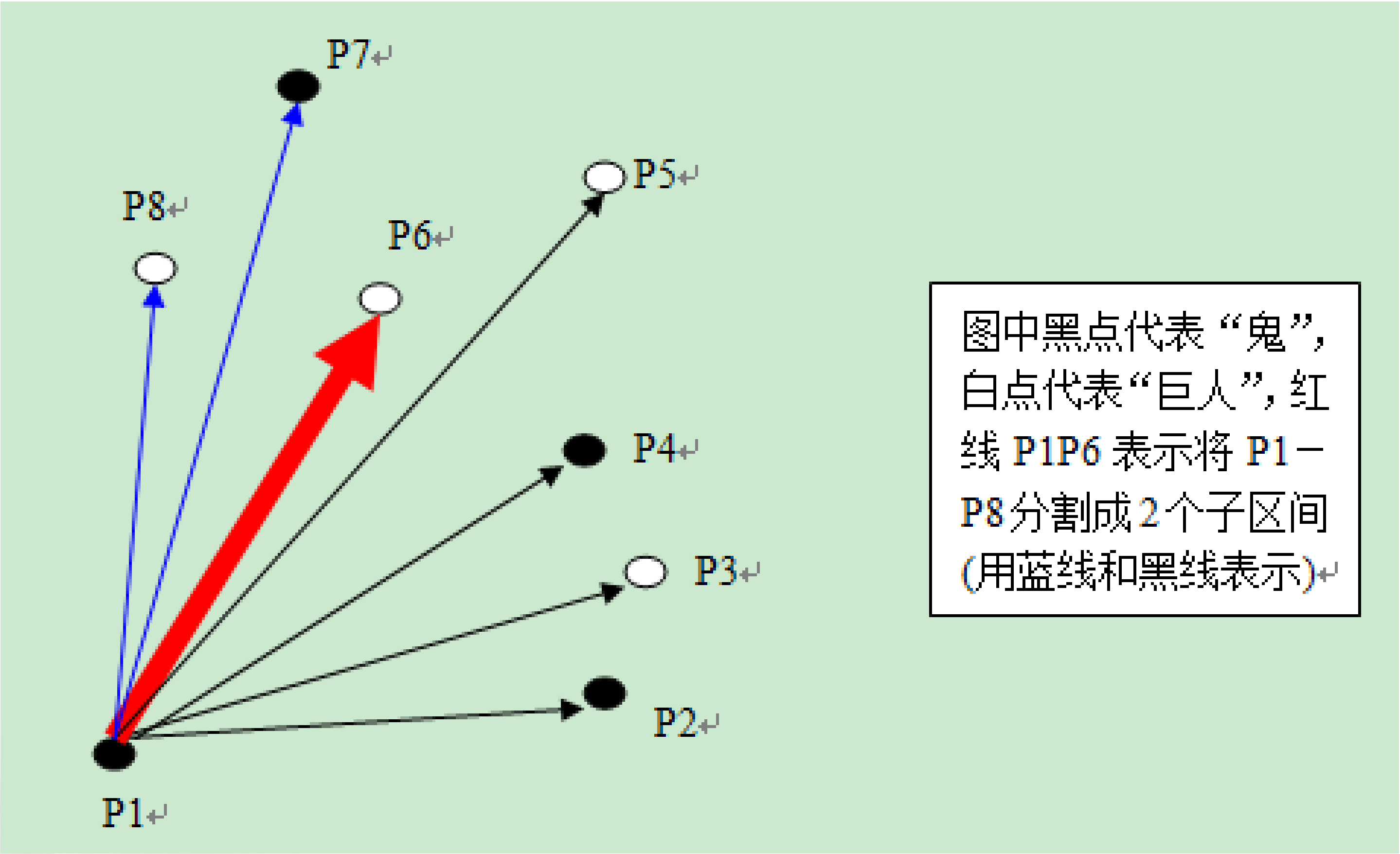
输出：n行，其中第i行的格式为“ xi,yi xi’,yi’ ”表示(xi,yi)位置的巨人和(xi’,yi’)位置的魔鬼配对。

解题思路：分治+递归

我们设P1..Pn为巨人的固定点;Pn+1..P2n为鬼的固定点。我们采取分治采取分治策略寻找序列[Pp..Pr]中的配对方案(初始时[Pp..Pr]为[P1..P2n]):

在[Pp..Pr]中找出一个最低位置(Y坐标值最小)的一个点P0,如果这样的点有多个,则选取最左边的点为P0，P0与Pp交换。然后将其余点[Pp+1..Pr]按相对 Pp的极角递增的顺序排列。显然Pp与其余点Pp+1..Pr之间的任何线段是不会交叉的。我们从Pp开始寻找一个巨人和鬼成对的最小子区间[Pp..Pi](p≤i≤r)。若该子区间仅剩一个元素,配对结束;否则巨人(鬼)Pp与鬼(巨人)Pi配对。这样使得尚未配对的巨人和鬼分布在两个子区间[Pp+1..Pi-1],[Pi+1..Pr]。继续按上述分治策略分别递归求解[Pp+1..Pi-1]和[Pi+1..Pr]。

解题思路：分治+递归


P7
P5
P8
P6
P4
P3
P2
P1
图中黑点代表“鬼”，白点代表“巨人”，红线 P1P6 表示将 P1－P8分割成2个子区间(用蓝线和黑线表示)