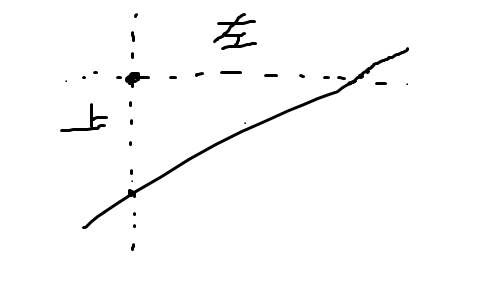
****如何判断点在线的哪边？****

将点的x，y坐标分别代入直线方程，再与原点对比



如何判断点是否在圆形内？

通过方程，对比半径

如何判断点是否在长方形内？长方形上呢？

1. 对比四条边，是不是对边相互反向

如何判断一个点是否在多边形内部？

（1）面积和判别法：判断目标点与多边形的每条边组成的三角形面积和是否等于该多边形，相等则在多边形内部。

（2）夹角和判别法：判断目标点与所有边的夹角和是否为360度，为360度则在多边形内部。

（3）引射线法：从目标点出发引一条射线，看这条射线和多边形所有边的交点数目。如果有奇数个交点，则说明在内部，如果有偶数个交点，则说明在外部。

首先要讲究速度，在运行复杂的算法之前，我们首先做一个简单的判定。在多边形的顶点中分别找出X坐标和Y坐标的最小/最大值。比如，你有点(9,1),(4,3),(2,7),(8,2),(3,6)所围成的多边形。那么Xmin为2, Xmax为9, Ymin为1 而 Ymax为7。现在我们知道你的多边形中没有一个点的X坐标比2小或者比9大，也没有一个点的Y坐标比1小或者比7大。这样你就可以快速排除很多不在多边形中的点：

// p is your point, p.x is the x coord, p.y is the y coord

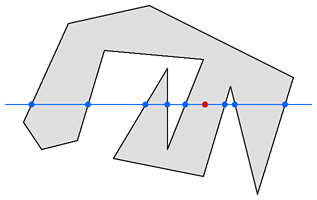
if (p.x < Xmin || p.x > Xmax || p.y < Ymin || p.y > Ymax) {

// Definitely not within the polygon!

}

这是第一个判定条件，如果在这一步已经把问题所要求的p(x,y)排除在多边形之外了(虽然这只是个非常粗糙的测试！)，那么就没必要进行下面判定了，非常"速度"吧？

如果上面的判定没排除掉p(x,y)，说明这个点是在多边形的坐标边界之内的，但并不意味着这个点就在多边形内了。我们需要一个更复杂一点的算法去判定这个点是否真在多边形内，当然我们有很多选择。以下是一些多边形的例子(凸多边形、凹多边形、"有洞"的多边形）



最简单的方法是使用**射线法**，因为它能适用于所有类型的多边形，不用考虑特殊的情况而且速度也比较快。该算法的思想很简单：在多边形外面任意一点画一条虚拟的射线到p(x,y)然后计算该射线与多边形上的边相交的次数。如果该次数是偶数，说明p(x,y)在多边形外，如果是奇数，则在多边形内。

具体做法：将测试点的Y坐标与多边形的每一个点进行比较，会得到一个测试点所在的行与多边形边的交点的列表。在下图的这个例子中有8条边与测试点所在的行相交，而有6条边没有相交。如果测试点的两边点的个数都是奇数个则该测试点在多边形内，否则在多边形外。在这个例子中测试点的左边有5个交点，右边有三个交点，它们都是奇数，所以点在多边形内。

