

```

        int ones = 0;
        int tmp = num;
        while (tmp != 0) {
            ones += (tmp & 1) != 0 ? 1 : 0;
            tmp >>= 1;
        }
        return num - ones;
    }
}

```

判断一个点是否在矩形内部

【题目】

在二维坐标系中，所有的值都是 `double` 类型，那么一个矩形可以由 4 个点来代表 $(x1,y1)$ 为最左的点、 $(x2,y2)$ 为最上的点、 $(x3,y3)$ 为最下的点、 $(x4,y4)$ 为最右的点。给定 4 个点代表的矩形，再给定一个点 (x,y) ，判断 (x,y) 是否在矩形中。

【难度】

尉 ★★☆☆

【解答】

本题的解法有很多种，本书提供的方法先解决如果矩形的边不是平行于 x 轴就是平行于 y 轴的情况下，该如何判断点 (x,y) 是否在其中，具体请参看如下代码中的 `isInside` 方法。

```

public boolean isInside(double x1, double y1, double x4, double y4,
                        double x, double y) {
    if (x <= x1) {
        return false;
    }
    if (x >= x4) {
        return false;
    }
    if (y >= y1) {
        return false;
    }
    if (y <= y4) {
        return false;
    }
    return true;
}

```

这种情况是比较简单的，因为矩形的边不是平行于 x 轴就是平行于 y 轴，所以判断该

点是否完全在矩形的左侧、右侧、上侧或下侧，如果都不是，就一定在其中。如果矩形的边不平行于坐标轴呢？也非常简单，就是高中数学的知识，通过坐标变换把矩阵转成平行的情况，在旋转时所有的点跟着转动就可以。旋转完成后，再用上面的方式进行判断。具体请参看如下代码中的 isInside 方法。

```
public boolean isInside(double x1, double y1, double x2, double y2,
    double x3, double y3, double x4, double y4, double x, double y)
{
    if (y1 == y2) {
        return isInside(x1, y1, x4, y4, x, y);
    }
    double l = Math.abs(y4 - y3);
    double k = Math.abs(x4 - x3);
    double s = Math.sqrt(k * k + l * l);
    double sin = l / s;
    double cos = k / s;
    double x1R = cos * x1 + sin * y1;
    double y1R = -x1 * sin + y1 * cos;
    double x4R = cos * x4 + sin * y4;
    double y4R = -x4 * sin + y4 * cos;
    double xR = cos * x + sin * y;
    double yR = -x * sin + y * cos;
    return isInside(x1R, y1R, x4R, y4R, xR, yR);
}
```

判断一个点是否在三角形内部

【题目】

在二维坐标系中，所有的值都是 double 类型，那么一个三角形可以由 3 个点来代表，给定 3 个点代表的三角形，再给定一个点(x,y)，判断(x,y)是否在三角形中。

【难度】

尉 ★★☆☆

【解答】

本书提供两种解法，第一种解法是从面积的角度来解决这道题，第二种解法是从向量的角度来解决。解法一在逻辑上没有问题，但是没有解法二好，下面会给出详细的解释。

先来介绍解法一，如果点 O 在三角形 ABC 内部，如图 9-1 所示，那么，有面积 $ABC =$