

# 计算几何

## 精度问题

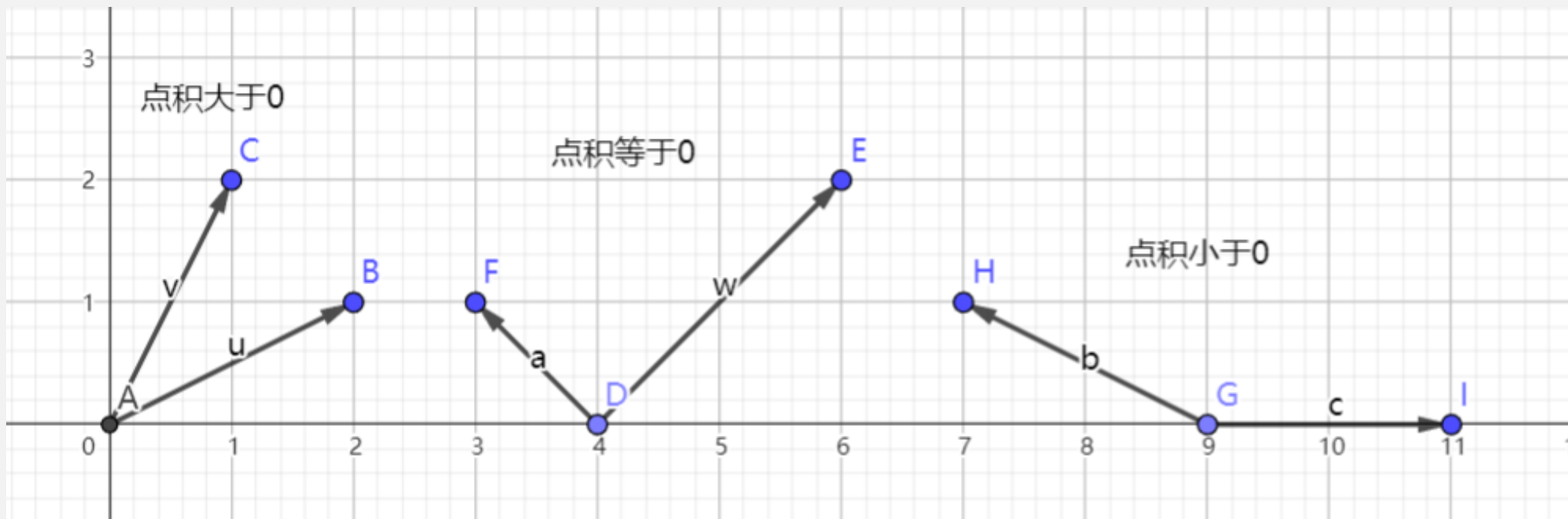
- long double的区分精度大约为 $2^{-64}$ ,  $1e-15 \sim 1e-18$
- double的区分精度大约为 $2^{-53}$ ,  $1e-12 \sim 1e-15$
- 精度问题, 求两个 $1e9$ 内的点的斜率, 误差为 $1e-18$
- 两个db相较:
- `const db EPS = 1e-9;`

```
inline int sign(db a) { return a < -EPS ? -1 : a > EPS; }
```

```
inline int cmp(db a, db b) { return sign(a - b); }
```

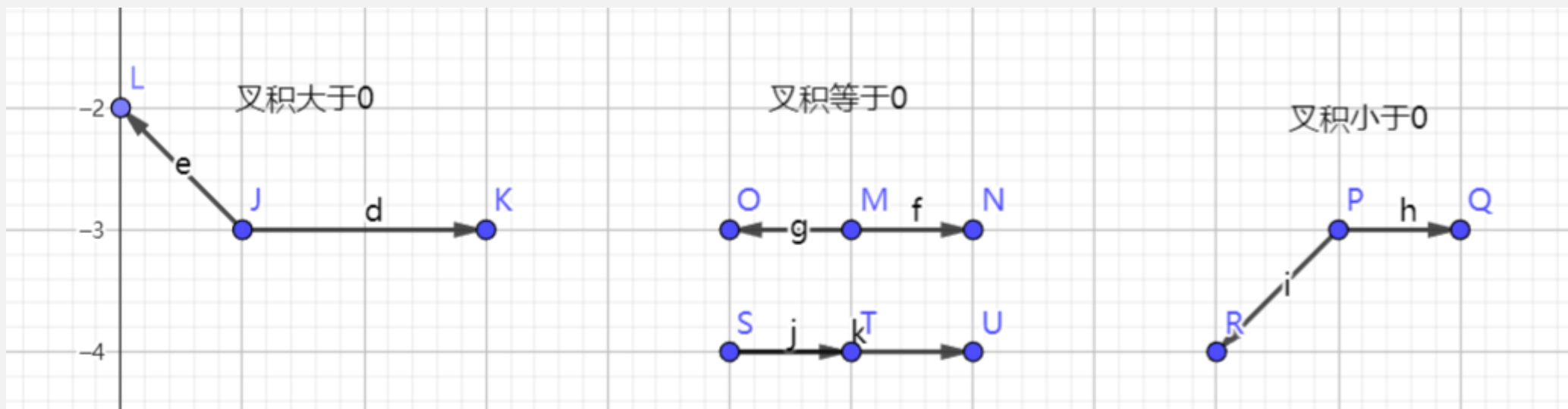
## 点积、叉积

- `db dot(P p) { return x * p.x + y * p.y; }`
- 点积反应了两个向量之间的夹角大小



## 点积、叉积

- `db det(P p) { return x * p.y - y * p.x; }`
- 反应了两个夹角之间的相对位置关系



# 常用函数

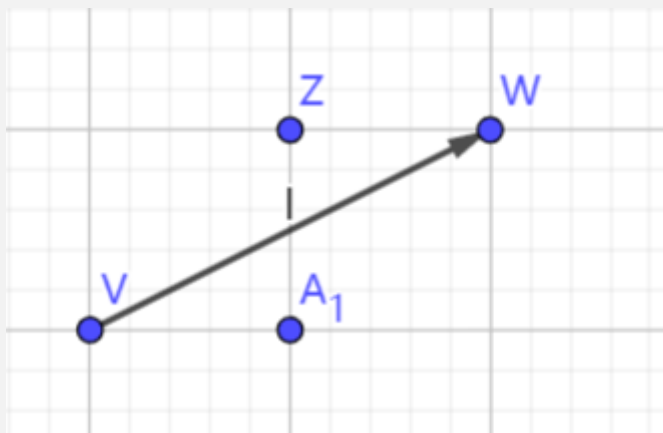
- ```
db disTo(P p) { return (*this - p).abs(); } //两点距离
db disTo2(P p) { return (*this - p).abs2(); } //两点距离的平方
db alpha() { return atan2(y, x); } //求极角
void readint() {
    int x_, y_;
    cin >> x_ >> y_;
    x = x_, y = y_;
} //输入整数
void readdb() { cin >> x >> y; }

void write() { cout << "(" << x << ", " << y << ")" << endl; } //输出
db abs() { return sqrt(abs2()); } //原点距离
db abs2() { return x * x + y * y; } //原点距离的平方
P rot90() { return P(-y, x); } //原点旋转90
int quad() const { return sign(y) == 1 || (sign(y) == 0 && sign(x) >= 0); } //判断点在上半边还是下半边
P unit() { return *this / abs(); } //单位向量

P rot(db an) {
    return {x * cos(an) - y * sin(an), x * sin(an) + y * cos(an)};
} // 绕原点旋转an度表示
```

## 点关系

- 约定,逆时针为正,顺时针为负
- ```
#define cross(p1, p2, p3)((p2.x-p1.x)*(p3.y-p1.y)-(p3.x-p1.x)*(p2.y-p1.y))  
#define crossOp(p1, p2, p3) sign(cross(p1,p2,p3))
```
- 表示第三个参数在p1p2向量的方向

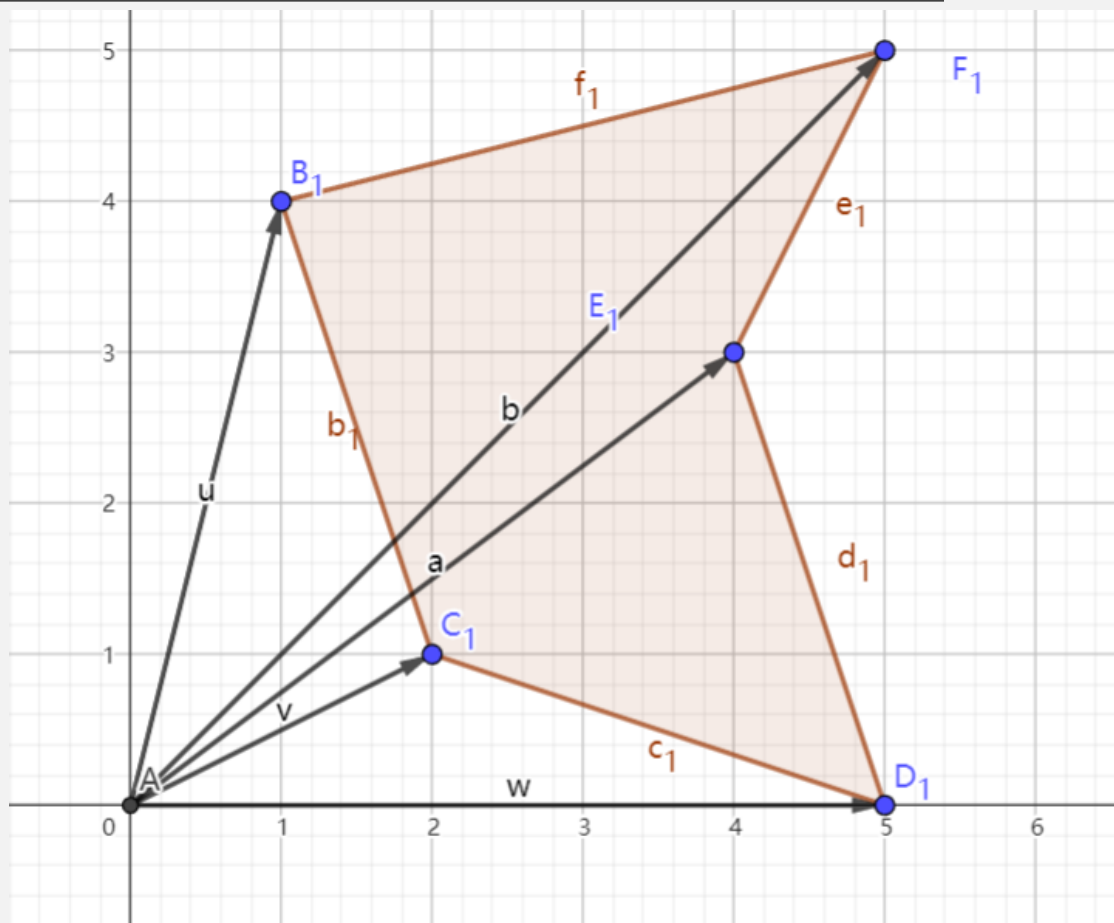


## 例题 I

- <https://codeforces.com/gym/101242>
- <https://codeforces.com/gym/103861/problem/D>

# 面积

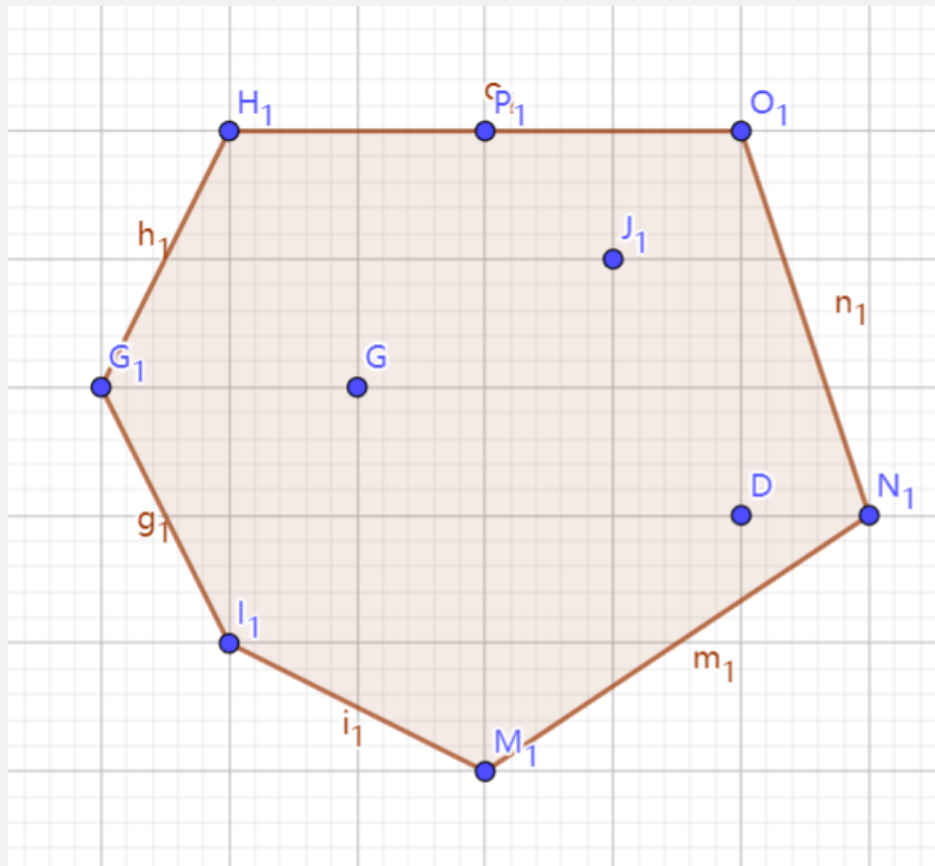
- 以边分段
- 有正负面积
- 绕边叉积一圈即可





# 凸包

- 用橡皮筋围起来的边界
- 如何求凸包:排列+单调队列



## 旋转卡壳

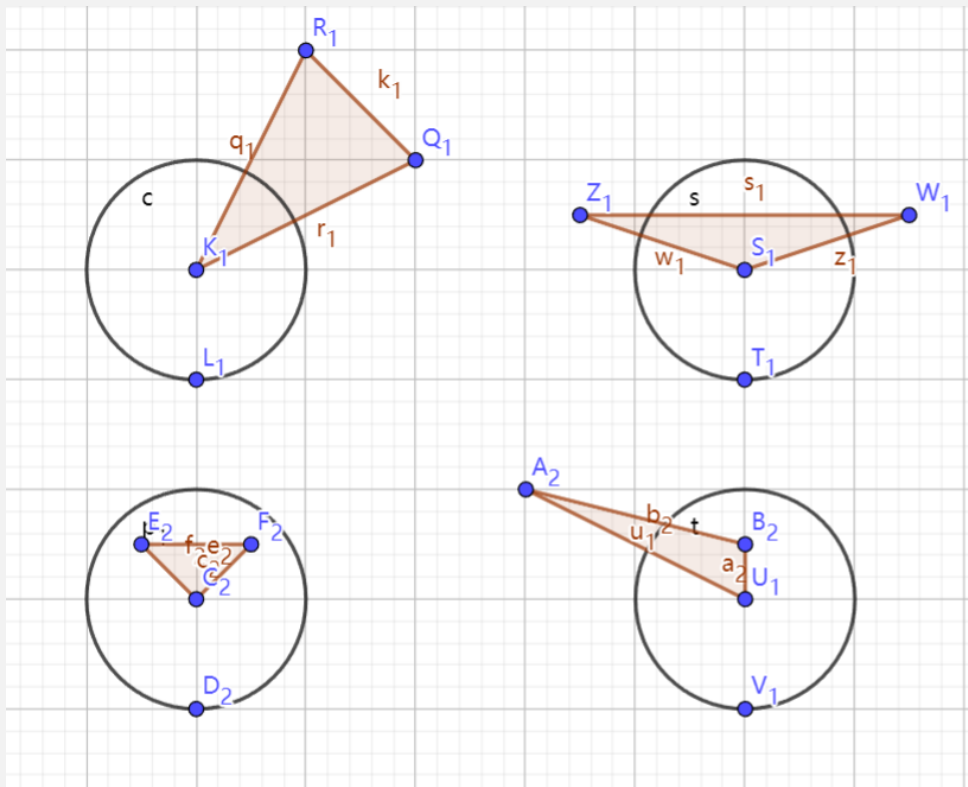
- 多个点在凸包上旋转
- 例:求覆盖凸包的矩形最小面积



- <https://www.luogu.com.cn/problem/P3187>
- <https://codeforces.com/gym/104417/problem/M>

# 原神,启动!

- 圆与多边形面积交
- 圆与三角形的关系





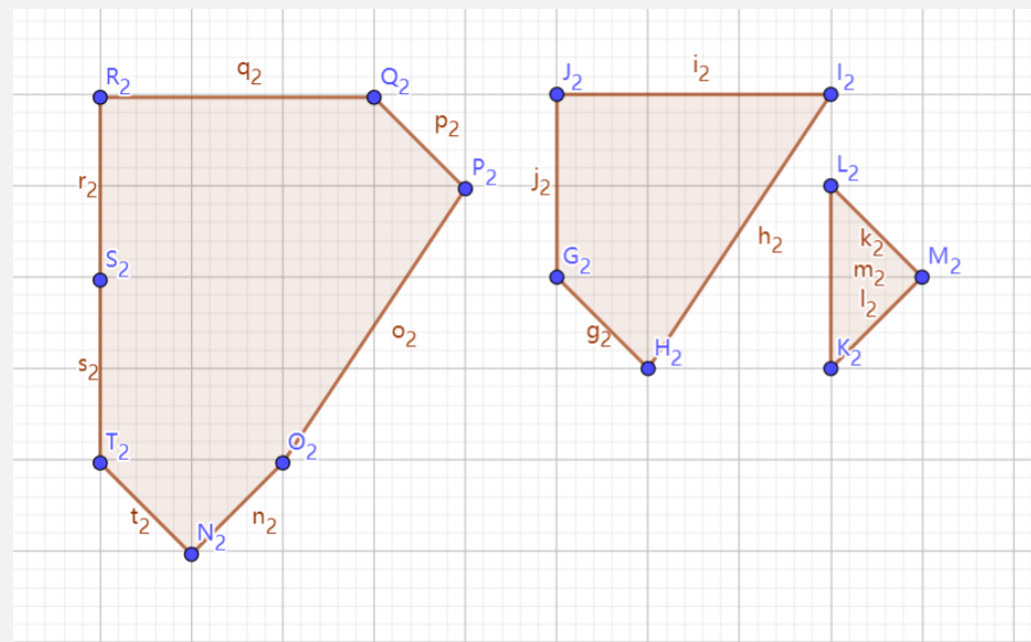
- 面积交
- 周长

## PICK 公式

- $S = a + \frac{b}{2} - 1$   
a表示多边形内部的点数
- b表示多边形边界上的点数
- S表示多边形的面积

# 闵可夫斯基和

- 两个点集所构成的膨胀集
- 翻译:合并俩凸包



## 1004

---

题意：给你一个凸多边形  $A$  和一个凸多边形  $B$  和一个圆  $P$  (以  $X$  为圆心， $R$  为半径)，问你在圆  $P$  内随机选一个点  $S$ ，问当凸多边形沿着向量  $\vec{OS}$  移动，得到凸多边形  $A'$ ，问  $A'$  与  $B$  相交的概率有多大。（相交意味着存在一个点  $w$ ，满足  $w \in A'$  and  $w \in B$ 。



## 1004

题意：给你一个凸多边形  $A$  和一个凸多边形  $B$  和一个圆  $P$  (以  $X$  为圆心， $R$  为半径)，问你在圆  $P$  内随机选一个点  $S$ ，问当凸多边形沿着向量  $\vec{OS}$  移动，得到凸多边形  $A'$ ，问  $A'$  与  $B$  相交的概率有多大。（相交意味着存在一个点  $w$ ，满足  $w \in A'$  and  $w \in B$ 。

首先，我们通过闵可夫斯基和可以构造出顺着怎样的向量  $\vec{OS}$  移动之后的  $A'$  会与  $B$  相撞，不难发现这些  $S$  构成了一个凸包（对闵可夫斯基不熟悉的话可以看一下这道入门题 <https://www.luogu.com.cn/problem/P4557>）。故题目就转化成这个凸包和圆  $P$  相交区域的面积，可以用三角剖分等方法。

最后输出的概率要注意一下不要输出-0.

有的队伍可能精度还有问题（唔验题的时候没有考虑到，可以优化一下自己的圆和多边形求交的板子，可以通过这道题来看看自己的板子精度是否合格<http://acm.hdu.edu.cn/showproblem.php?pid=2892>）

时间复杂度  $O(n \log n)$ .