

```
1 #include "../bits/stdc++.h"
2
3 using ld = long double;
4 using Point = std::complex<ld>;
5
6 const ld eps = 1e-9, pi = acos(-1.0);
7
8 namespace std
9 {
10 bool operator<(const Point &lhs, const Point &rhs)
11 {
12     if (lhs.real() < rhs.real() - eps)
13         return true;
14     if (lhs.real() > rhs.real() + eps)
15         return false;
16     return lhs.imag() < rhs.imag();
17 }
18 } // namespace std
19
20 Point input_point()
21 {
22     ld x, y;
23     std::cin >> x >> y;
24     return Point(x, y);
25 }
26
27 bool eq(ld a, ld b)
28 {
29     return (abs(a - b) < eps);
30 }
31
32 ld dot(Point a, Point b)
33 {
34     return real(conj(a) * b);
35 }
36
37 ld cross(Point a, Point b)
38 {
39     return imag(conj(a) * b);
40 }
41
42 // CCW::counter clockwise
43 int ccw(Point a, Point b, Point c)
44 {
45     b -= a;
46     c -= a;
47     if (cross(b, c) > eps)
48         return 1; // a,b,c : counter-clockwise
49     if (cross(b, c) < -eps)
50         return -1; // a,b,c : clockwise
51     if (dot(b, c) < 0)
52         return 2; // c,a,b : on a line
53     if (norm(b) < norm(c))
54         return -2; // a,b,c : on a line
55     return 0; // a,c,b : on a line
56 }
57
```