**向量**

#include<bits/stdc++.h>

using namespace std;

typedef long long ll;

struct Vector{

ll x,y,z;

bool two\_dimension;

bool three\_dimension;

Vector(){

x = y = z = 0;

two\_dimension = false;

three\_dimension = false;

}

Vector(ll rhs\_x,ll rhs\_y)

{

two\_dimension = true;

three\_dimension = false;

x = rhs\_x;

y = rhs\_y;

}

Vector(ll rhs\_x,ll rhs\_y,ll rhs\_z)

{

three\_dimension = true;

two\_dimension = false;

x = rhs\_x;

y = rhs\_y;

z = rhs\_z;

}

ll dot\_product(const Vector&rhs)

{

//this 点乘 rhs

ll ans = -1;

if(two\_dimension){

ans = x\*rhs.x + y\*rhs.y;

}

if(three\_dimension){

ans = x\*rhs.x + y\*rhs.y + z\*rhs.z;

}

return ans;

}

ll cross\_product(const Vector&rhs)

{

//this 叉乘 rhs

ll ans = -1;

if(two\_dimension){

ans = x\*rhs.y - rhs.x\*y;

}

if(three\_dimension){

ans = (y\*rhs.z - z\*rhs.y) - (x\*rhs.z - z\*rhs.x) + (x\*rhs.y - y\*rhs.x);

}

return ans;

}

};