# 凸包

#include<cstdio>  
#include<cstring>  
#include<algorithm>  
#include<iostream>  
#include<cmath>  
using namespace std;  
  
struct Point {  
    double x,y;  
    Point (double x=0,double y=0) : x(x), y(y){}  
};  
const int M = 10101;  
const double pi = acos(-1.0);  
bool operator <  (Point A,Point B){  
    return A.x==B.x?A.y<B.y:A.x<B.x;  
}  
int r[M];  
Point p[M];  
double cross(Point A ,Point B){  
    return (double)((A.x\*B.y)-(A.y\*B.x));  
}  
double dist(Point A,Point B){  
    return sqrt((A.x-B.x)\*(A.x-B.x)+(A.y-B.y)\*(A.y-B.y));  
}  
Point operator - (Point A ,Point B){  
    return Point(A.x-B.x,A.y-B.y);  
}  
int n;  
double L;  
int main(){  
    freopen("data.in","r",stdin);  
    while(cin>>){  
    for(int i=1;i<=n;i++){  
        cin>>p[i].x>>p[i].y;  
    }  
    sort(1+p,1+p+n);  
    int m=0;  
    for(int i=1;i<=n;i++){  
        while(m>1&&cross(p[i]-p[r[m-1]],p[r[m]]-p[r[m-1]])>0) m--;  
        r[++m]=i;  
    }  
    int k=m;  
    for(int i=n-1;i>=1;i--){  
        while(m>k&&cross(p[i]-p[r[m-1]],p[r[m]]-p[r[m-1]])>0) m--;  
        r[++m]=i;  
    }  
    if(n>1) m--;  
    double ans=0;  
    for(int i=1;i<m;i++){  
        //    printf("%d\n",r[i]);  
        ans+=dist(p[r[i]],p[r[i+1]]);  
    }  
    ans+=dist(p[r[1]],p[r[m]]);  
    cout<<ans<<endl;  
    }  
    return 0;  
}