Geometry

Code：

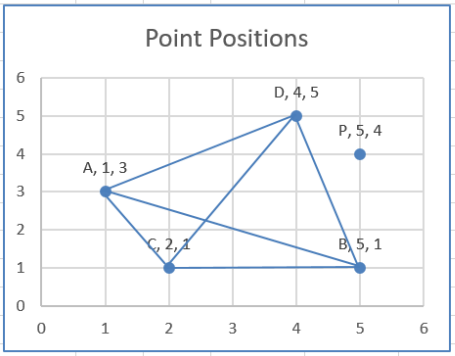
```c
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <stdbool.h>

double H[9];

double Leng(double Xa,double Ya,double Xb,double Yb){
    return sqrt((Xa-Xb)*(Xa-Xb)+(Ya-Yb)*(Ya-Yb));
}

double Perim(double Xa,double Ya,double Xb,double Yb,double Xc,double Yc){
    return Leng(Xa,Ya,Xb,Yb)+Leng(Xa,Ya,Xc,Yc)+Leng(Xb,Yb,Xc,Yc);
}

double Area(double Xa,double Ya,double Xb,double Yb,double Xc,double Yc){
    double p=Perim(Xa,Ya,Xb,Yb,Xc,Yc)/2;
    double AB=Leng(Xa,Ya,Xb,Yb);
    double AC=Leng(Xa,Ya,Xc,Yc);
    double BC=Leng(Xb,Yb,Xc,Yc);
    return sqrt(p*(p-AB)*(p-AC)*(p-BC));
}

double Dist(double Xp,double Yp,double Xa,double Ya,double Xb,double Yb){
    double Spab=Area(Xp,Yp,Xa,Ya,Xb,Yb);
    double AB=Leng(Xa,Ya,Xb,Yb);
    return 2*Spab/AB;
}

void Altitudes(double Xa,double Ya,double Xb,double Yb,double Xc,double Yc,int ha,int hb,int hc){
    H[ha]=Dist(Xa,Ya,Xb,Yb,Xc,Yc);
    H[hb]=Dist(Xb,Yb,Xa,Ya,Xc,Yc);
    H[hc]=Dist(Xc,Yc,Xb,Yb,Xa,Ya);
}

double min(double a,double b){
    if(a<=b) return a;
    else if(a>b) return b;
}

double max(double a,double b){
    if(a>=b) return a;
    else if(a<b) return b;
}

void JudgeCross(float Xa,float Ya,float Xb,float Yb,float Xc,float Yc,float Xd,float Yd){
    // 1st step: Rapid Rejection
    //1: The left x of CD < The right x of AB
    if(  ( min(Xc,Xd)<=max(Xa,Xb) )
    //2: The left x of AB < The right x of CD
       &&( min(Xa,Xb)<=max(Xc,Xd) )
    //3: The down y of CD < The up y of AB
       &&( min(Yc,Yd)<=max(Ya,Yb) )
    //4: The down x of AB <The up x of CD
       &&( min(Ya,Yb)<=max(Yc,Yd) )

    //2nd step: Cross Stand Test
    //1: Point C and D stand on the different side of AB
       &&( ((Xc-Xa)*(Yb-Ya)-(Xb-Xa)*(Yc-Ya)) * ((Xd-Xa)*(Yb-Ya)-(Xb-Xa)*(Yd-Ya)) <=0 )
    //2: Point A and B stand on the different side of CD
       &&( ((Xa-Xd)*(Yc-Yd)-(Xc-Xd)*(Ya-Yd)) * ((Xb-Xd)*(Yc-Yd)-(Xc-Xd)*(Yb-Yd)) <=0 )
       )
         printf("Cross\n");
    else printf("No Cross\n");
}
```

```c
int main(){
    double Xa=1,Ya=3;
    double Xb=5,Yb=1;
    double Xc=2,Yc=1;
    double Xd=4,Yd=5;
    double Xp=5,Yp=4;
    int ha1=0,hb1=1,hc1=2,ha2=3,hb2=4,hc2=5,ha3=6,hb3=7,hc3=8;
    printf("AB = %lf\n",Leng(Xa,Ya,Xb,Yb));
    printf("AC = %lf\n",Leng(Xa,Ya,Xc,Yc));
    printf("AD = %lf\n",Leng(Xa,Ya,Xd,Yd));
    printf("Pabc = %lf\n",Perim(Xa,Ya,Xb,Yb,Xc,Yc));
    printf("Pabd = %lf\n",Perim(Xa,Ya,Xb,Yb,Xd,Yd));
    printf("Pacd = %lf\n",Perim(Xa,Ya,Xc,Yc,Xd,Yd));
    printf("Sabc = %lf\n",Area(Xa,Ya,Xb,Yb,Xc,Yc));
    printf("Sabd = %lf\n",Area(Xa,Ya,Xb,Yb,Xd,Yd));
    printf("Sacd = %lf\n",Area(Xa,Ya,Xc,Yc,Xd,Yd));
    printf("DPab = %lf\n",Dist(Xp,Yp,Xa,Ya,Xb,Yb));
    printf("DPac = %lf\n",Dist(Xp,Yp,Xa,Ya,Xc,Yc));
    printf("DPbc = %lf\n",Dist(Xp,Yp,Xb,Yb,Xc,Yc));
    Altitudes(Xa,Ya,Xb,Yb,Xc,Yc,ha1,hb1,hc1);
    printf("ha of ABC = %lf\n",H[ha1]);
    printf("hb of ABC = %lf\n",H[hb1]);
    printf("hc of ABC = %lf\n",H[hc1]);
    Altitudes(Xa,Ya,Xb,Yb,Xc,Yc,ha2,hb2,hc2);
    printf("ha of ABD = %lf\n",H[ha2]);
    printf("hb of ABD = %lf\n",H[hb2]);
    printf("hc of ABD = %lf\n",H[hc2]);
    Altitudes(Xa,Ya,Xc,Yc,Xd,Yd,ha3,hb3,hc3);
    printf("ha of ACD = %lf\n",H[ha3]);
    printf("hb of ACD = %lf\n",H[hb3]);
    printf("hc of ACD = %lf\n",H[hc3]);

    printf("AB and CD :");
    JudgeCross(Xa,Ya,Xb,Yb,Xc,Yc,Xd,Yd);
    printf("AC and BD :");
    JudgeCross(Xa,Ya,Xc,Yc,Xb,Yb,Xd,Yd);

    return 0;
}
```

Result：

Point	X	Y
A	1	3
B	5	1
C	2	1
D	4	5
P	5	4

AB and CD： Cross
AC and BD: No Cross



Line	AB	AC	AD
	4.472136	2.236068	3.605551
DistanceFromP	2.683282	4.024922	3.000000

Triangle	ABC	ABD	ACD
Perimeter	9.708204	12.200793	4.000000
Area	3.000000	7.000000	4.000000

Heights	ha	hb	hc	ha	hb	hc	ha	hb	hc
	2.000000	2.683282	1.341641	2.000000	2.683282	1.341641	1.788854	2.218801	3.577709