```c
#include <stdio.h>
#include <math.h>
#define N 10

struct punto{
        double x;
        double y;
};

struct rettangolo{
        struct punto A;
        struct punto B;
};

struct punto leggi_valori(){
        struct punto p;
        scanf("%lf %lf", &p.x, &p.y);
        return p;
}

void stampa_punto(struct punto p){
        printf("(%.3lf, %.3lf)\n", p.x, p.y);
        return;
}

double distanza(struct punto p1, struct punto p2){
        return sqrt((p1.x-p2.x)*(p1.x-p2.x)+(p1.y-p2.y)*(p1.y-
p2.y));
}

int punti_interni(struct punto p, struct rettangolo r){
        if((p.x<=r.B.x && p.x>=r.A.x) && (p.y<=r.A.y && p.y>=r.B.y))
                return 1;
                else return 0;
}

double area(struct rettangolo r){
        return (r.B.x-r.A.x)*(r.A.y-r.B.y);
}

void indici_distanza(struct punto *vet, int n, int *i_max, int
*j_max){
        int i, j;
        double dist, max;
        max=distanza(vet[*i_max], vet[*j_max]);
        for(i=0; i<n-1; i++){
                for(j=i+1; j<n; j++){
                        dist=distanza(vet[i], vet[j]);
                        if(dist>max){
                                max=dist;
                                *i_max=i;
                                *j_max=j;
                }
```

```
        }

  }
return;
}


int main(){
        unsigned int n;
        struct punto vet[N];
        struct punto origine={0, 0};
        struct rettangolo r;
        int i;
        int i_max=0, j_max=1;
        scanf("%u", &n);
        printf("Inserire il numero dei punti: ");
        if(n>N)
                n=N;


        for(i=0; i<n; i++)
                vet[i]=leggi_valori();

        r.A=leggi_valori();
        r.B=leggi_valori();

        printf("[PUNTI]\n");
        for(i=0; i<n; i++)
                stampa_punto(vet[i]);

        printf("[DISTANZE]\n");
        for(i=0;i<n;i++)
                printf("%.3lf\n", distanza(origine, vet[i]));

        printf("[INTERNI]\n");
        for(i=0;i<n;i++){
                if(punti_interni(vet[i], r))
                        stampa_punto(vet[i]);
        }
        printf("[AREA]\n");
        printf("%.3lf\n", area(r));

        printf("[COPPIA]\n");
        indici_distanza(vet, n, &i_max, &j_max);
        stampa_punto(vet[i_max]);
        stampa_punto(vet[j_max]);

return 0;
}
```