

D. BOULEVARD

PROBLEM DESCRIPTION

N 個人從不同的時間、起點、終點出發，問每個人各會與幾個人相遇。

SOLUTION TECHINQUES

數學

SOLUTION SKETCHES

枚舉人對，看判斷會不會相遇。

注意同方向也要判斷。

TIME COMPLEXITY

$O(N^2)$ ，N 為人數。

SOLUTION PROGRAM FOR REFERENCE

```
#include<stdio>

struct node
{
    int t, s, f;
    bool v;
};

int ans[1005];
node p[1005];

bool same(int a, int b)
{
    if(p[a].s<p[a].f){
        if(p[a].s<p[b].s && p[a].f<p[b].s) return 0;
        if(p[b].s<p[a].s && p[b].f<p[a].s) return 0;
        if((p[a].s-p[a].t)!=(p[b].s-p[b].t)) return 0;
    }
    else{
        if(p[a].f<p[b].f && p[a].s<p[b].f) return 0;
        if(p[b].f<p[a].f && p[b].s<p[a].f) return 0;
        if((p[a].s+p[a].t)!=(p[b].s+p[b].t)) return 0;
    }
    return 1;
}

bool enc(int a, int b)
{
    double t;
    t=(double)(p[b].s+p[b].t+p[a].t-p[a].s)/2;
    if(t<p[a].t) return 0;
    if(t<p[b].t) return 0;
    if(t>p[a].f+p[a].t-p[a].s) return 0;
    if(t>p[b].s+p[b].t-p[b].f) return 0;
    return 1;
}

int main()
{
    int n, i, j;
    scanf("%d", &n);
    for(i=0;i<n;i++) ans[i]=0;
    for(i=0;i<n;i++){
        scanf("%d %d %d", &p[i].t, &p[i].s, &p[i].f);
        if(p[i].s<p[i].f) p[i].v=1;
        else p[i].v=0;
    }
    for(i=0;i<n;i++){
        for(j=i+1;j<n;j++){
            if(p[i].v==p[j].v){
                if(same(i, j)){
                    ans[i]++;
                    ans[j]++;
                }
            }
        }
    }
}
```

```
        }
        else{
            if((p[i].v&&enc(i, j))||p[j].v&&enc(j, i)){
                ans[i]++;
                ans[j]++;
            }
        }
    }
}
for(i=0;i<n-1;i++) printf("%d ", ans[i]);
printf("%d\n", ans[n-1]);
}
```