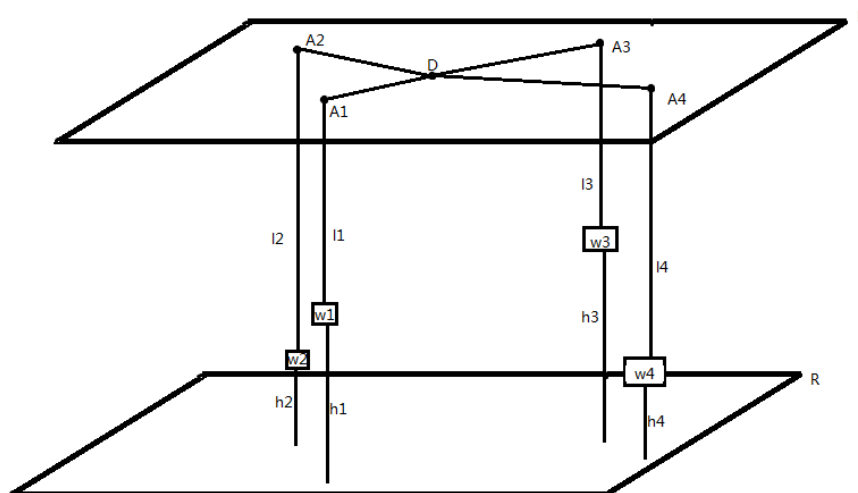


[JSOI2004]吊打XXX

[题目传送门](#)

首先你会打这道题，你需要知道关于广义费马点的概念。



看这个图 ↑

假如有4个重物从4个点上用同长的绳子拴起来（绳子的起点绑在一起），那么最终的状态一定存在一个点D使4个重物达到平衡。点D即为广义费马点。

那么我们对这些点进行分析。

这些点平衡后势能总和达到最低，即 $\sum w_i * h_i$ 最小。

因为L平面到R平面的距离一定，所以此时 $\sum w_i * l_i$ 达到最大。

又由于绳长一定，所以此时 $\sum Len_{(D,A_i)} * w_i$ 达到最小。

对于这道题，可以使用模拟退火的方法求广义费马点。

洗白脸多退火几次。

Code :

```
#include <ctime>
#include <cmath>
#include <cstdio>
#include <cstdlib>
using namespace std;
```

```

int read(){
    char c;while(c=getchar(),c!='-'&&(c<'0' || c>'9'));
    int x=0,y=1;c=='-'?y=-1:x=c-'0';
    while(c=getchar(),c>='0'&&c<='9')x=x*10+c-'0';
    return x*y;
}

const int Maxn=1e3+5;
const double Td=0.983;
int N;
double xAns,yAns,Ans=2e17;
struct Node{int x,y,w;}D[Maxn];

double Gen(double x,double y){
    double Sum=0;for(int i=1;i<=N;i++){
        double Dx=x-D[i].x,Dy=y-D[i].y;
        Sum+=sqrt(Dx*Dx+Dy*Dy)*D[i].w;
    }return Sum;
}

void Mnth(){
    double fx=xAns,fy=yAns;
    double Temp=1926.0817;
    while(Temp>1e-14){
        double xTemp=xAns+(rand()*2-RAND_MAX)*Temp;
        double yTemp=yAns+(rand()*2-RAND_MAX)*Temp;
        double Ne=Gen(xTemp,yTemp);
        if(Ne<Ans)fx=xTemp,fy=yTemp,xAns=fx,yAns=fy,Ans=Ne;
        else if(exp((Ans-Ne)/Temp)*RAND_MAX>rand())fx=xTemp,fy=yTemp;
        Temp*=Td;
    }
}

void Solve(){
    srand(time(0));
    for(int i=1;i<=23;i++)Mnth();
}

int main()
{
    N=read();for(int i=1;i<=N;i++)D[i].x=read(),D[i].y=read(),D[i].w=read();
    Solve();printf("%.31f %.31f",xAns,yAns);
    return 0;
}

```