


模拟退火相关题目

1.计算函数最值（非单峰函数）

image-20201212232425335

1.Strange fuction

题目传送门[HDU2899](https://acm.hdu.edu.cn/showproblem.php?pid=2899)

思路：本题的状态函数就是题目中的数学函数，因为是求最小值，所以每次取最小即可，其他细节见代码。

```
XXXXXXXXXX
#include<bits/stdc++.h>
using namespace std;
const double eps=1e-8;//终止温度
double y;
double fun(double x){ //计算函数结果
    return 6*pow(x,7.0)+8*pow(x,6.0)+7*pow(x,3.0)+5*pow(x,2.0)-y*x;
}
double solve(){
    double T=100;//初始温度
    double delta=0.997;//降温系数
    double x=50.0;//初始x
    double now=fun(x),ans=now;//当前计算的函数值，ans最后的答案
    while(T>eps){ //如果没有降低终止温度
        double nx=x+(rand()*2-RAND_MAX)*T;//随T的降低，对x的改变减少 T可能增加，可能减少
        if(nx>=0&&nx<=100){ //nx处于范围内
            double nt=fun(nx);//计算nx的值
            ans=min(ans,nt);//取最小
            if(now-nt>eps){ //如果新状态更小
                x=nx,now=nt;//x取新状态，now=nt
            }
        }
        T*=delta;//降温
    }
    return ans;
}
```

```

int main() {
    int t;
    cin>>t;
    while(t--){
        cin>>y;
        printf("%.4lf\n", solve());
    }
    return 0;
}

```

求平衡点。

题目传送门: [P1337](#)

本题的状态函数是: n 个点距离答案坐标的距离*各自重量的求和 也是取最小。因为距离乘上重量 越小, 代表重物的重力势能越低, 也就越平稳。下面上代码。

```

#include<bits/stdc++.h>
using namespace std;
struct p{
    double x,y,w;
}a[1005];
const double eps=1e-18,delta=0.9986;// delta 玄学delta 一般越接近1得到的可行解精度越高
int n;
double x,y,now;
double fun(double x,double y){ //状态函数
    double sum=0,dx,dy;
    for(int i=1;i<=n;i++){
        dx=x-a[i].x,dy=y-a[i].y;
        sum+=sqrt(dx*dx+dy*dy)*a[i].w;
    }
    return sum;
}
void sa(){
    double t=3000;
    while(t>eps){
        double nx=x+(rand()*2-RAND_MAX)*t,ny=y+(rand()*2-RAND_MAX)*t;
        double nt=fun(nx,ny),de=nt-now;
    }
}

```

```

        if(de<0)
        {
            x=nx, y=ny, now=nt;
        }

        else if(exp(-de/t)*RAND_MAX>rand())
            x=nx, y=ny;

        t*=delta;
    }
}

int main() {
    cin>>n;

    for(int i=1;i<=n;i++) cin>>a[i].x>>a[i].y>>a[i].w, x+=a[i].x, y+=a[i].y;

    x/=n, y/=n;

    now=fun(x, y);

    sa();

    printf("%.3lf %.3lf\n", x, y);

    return 0;
}

```