

dfs1.0

题目描述(p1433)

房间里放着n块奶酪。一只小老鼠要把它们都吃掉，问至少要跑多少距离？老鼠一开始在(0,0)点处。

输入格式 第一行一个数n ($n \leq 15$)

接下来每行2个实数，表示第i块奶酪的坐标。

两点之间的距离公式= $\sqrt{(x1-x2)^2+(y1-y2)^2}$

输出格式

一个数，表示要跑的最少距离，保留2位小数。

输入输出样例

```
输入      输出 7.41
4
1  1
1 -1
-1 1
-1 -1
```

思路

dfs实质是暴力枚举

关注有什么值在动态变化，剪枝很重要，以及是否需要回溯

小tip 方向数组

这道题的思路是先用方向数组dis[x][y]算出x y 之间的距离，然后进行dfs枚举，当发现当前已走路程已经大于ans，便提前结束掉当前dfs，即剪枝。

这道题需要回溯

talk is cheap,show me the code

```

#include <iostream>
#include <cmath>
#include<cstdio>
#include<cstring>
using namespace std;
int n;
double ans=1e9;
double a[20],b[20];
double dis[20][20];
int vis[20]; //记录访问到哪里
void dfs(int i,int x,double len){
    if(len>ans) return; //剪枝
    if(i==n) {
        ans=min(ans,len); //dfs末尾
        return;
    }
    for(int q=1;q<=n;q++){
        if(vis[q]==0){
            vis[q]=1;
            dfs(i+1,q,dis[x][q]+len);
            vis[q]=0; //回溯
        }
    }
}

int main(){
    cin>>n;
    memset(vis,0,sizeof(vis));
    for(int i=1;i<=n;i++){
        cin>>a[i]>>b[i];
    }
    for(int i=0;i<=n;i++){ //建立dis, 使dfs可以直接调用距离, 减少重复运算
        for(int q=0;q<=n;q++){
            dis[i][q]=sqrt((a[i]-a[q])*(a[i]-a[q])+(b[i]-b[q])*(b[i]-b[q]));
        }
    }
    dfs(0,0,0);
    printf("%.2f",ans);
    return 0;
}

```