# 树的点分治

就是每次通过两次dfs记录s[i]表示以i为根的子树有多少结点来找到树的重心（即拿去这个点所有块中结点最多的块结点最小）。这样每次树大小至少减少一半，然后递归处理这log层，对于每层来说，暴力dfs统计答案即可。

poj1741

题意：给每条边的边权，求树上距离小于等于m的点对有多少个。

我们对于每个点统计经过这个点的路径长度小于m的有多少个。

这可以通过暴力dfs这颗子树来实现每个点到当前根的距离d[i]。

排个序从小到大枚举即可。

int t=sz;

sort(1+d,1+d+sz);

ans[num]=0;

for(int i=1;i<=sz;i++){

if(d[i]>m) break;

while(t&&d[t]+d[i]>m){

t--;

if(!t) break;

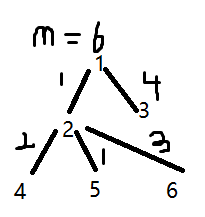
}

ans[num]+=t;

if(d[i]\*2<=m) ans[num]--;//自己和自己不能构成一对

}

但是这种统计方法明显会有漏洞



这样(4,5)，（5，6）在2，1都会被统计，而（4，5）不会统计在1中

于是我们在统计2的时候还需找到上一次被分治的点（1号点），查询是否有d[x]+d[y]<=m-2\*dist[1][2].避免被重复统计。