一、树的删边游戏

给出一棵有根树，两人轮流操作，每人每次可以选择一条边删去，不与根节点相连的部分将被移走。无法操作者输。

结论：叶子节点的 值为 ；中间节点的 值为它的所有子节点的「 值加 」的异或和。这样就可以推出根节点的 值。

这种问题还有变形,比如变成删除某一个真(假)子树(即是否能删除根节点)的,本质上和上面的这种删除一条边是一样的模型,都是中间节点的 值为它的所有子节点的「 值加 」的异或和。这样就可以推出根节点的 值。

void dfs(int x,int fa)

{

for (int i=last[x];i;i=e[i].next)

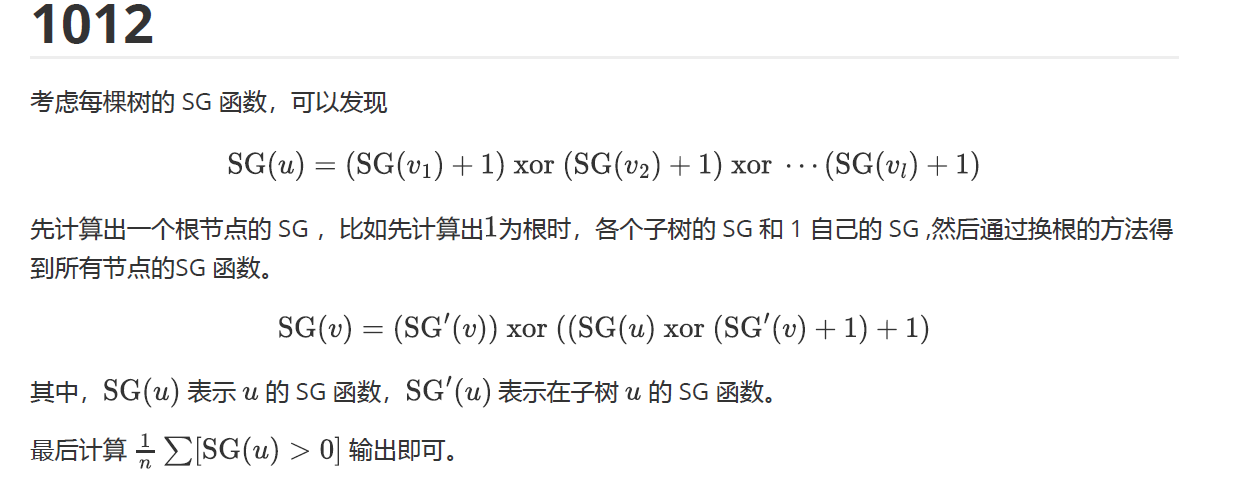
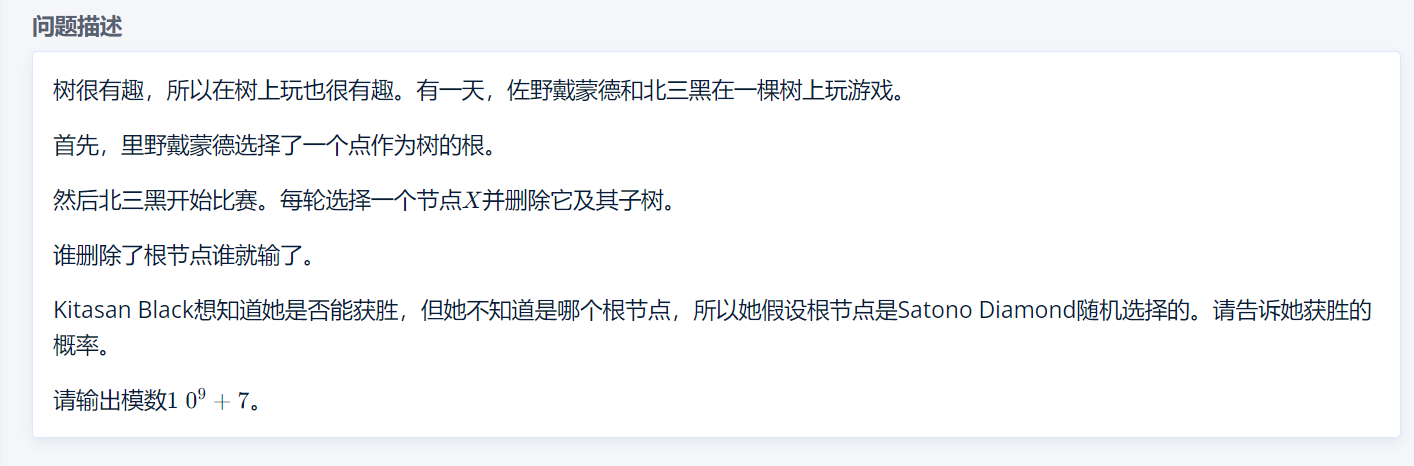
{

if (e[i].to==fa) continue;

dfs(e[i].to,x);

sg[x]^=sg[e[i].to]+1;

}

还有一个hdu的例题: <https://acm.hdu.edu.cn/contest/problem?cid=1094&pid=1012>

二、POJ 3710 Christmas Game

*Description*

有 个局部连通的图，两人轮流操作，每人每次可以选择一条边删去，不与根节点相连的部分将被移走。无法操作者输。（图是通过从基础树上加一些边得到的，环与环之间无公共边，且只与基础树有一个公共点。）

*Solution*

与「树的删边游戏」不同的在于图中 出现了环。所以环可以单独考虑。

分析环的性质：环保证不共用边，且只与基础树有一个公共点。因此所有的环都是从树中某一个点连出又连回同一个点的简单环。

对于长度为奇数的环，去掉其中任意一条边后，剩下的两个链长度 同奇偶，异或之后的 值不可能为奇数，并且可能为 。所以进行 运算后它的 值为 。

对于长度为奇数的环，去掉其中任意一条边后，剩下的两个链长度 不同奇偶，异或之后的 值不可能为 。所以进行 运算后它的 值为 。实际上它并没有贡献。

所以可以去掉所有的偶环，将所有的奇环变为长度为 的链，转化为「树的删边游戏」的模型。

算出每一棵树的 值，再把所有根节点的 值异或起来即可。

具体实现时，不需要做「将所有的奇环变为长度为 的链」的操作，只需考虑奇环的贡献。

#include<cstdio>  
#define int long long  
using namespace std;  
const int N=1e5+5;  
int t,n,m,x,y,cnt,hd[N],to[N<<1],nxt[N<<1],top,s[N],sg[N],vis[N],ans; //vis=0: 未访问过 vis=1: 访问过且不是某个环上的点（不考虑树上的点） vis=-1: 访问过且是某个环上的点   
void add(int x,int y){  
 to[++cnt]=y,nxt[cnt]=hd[x],hd[x]=cnt;  
}  
void dfs(int x,int fa){  
 s[++top]=x,vis[x]=1,sg[x]=0;  
 bool flag=0;  
 for(int i=hd[x];i;i=nxt[i]){  
 int y=to[i];  
 if(y==fa&&!flag){flag=1;continue;} //第一次连向父节点   
 if(vis[y]==1){  
 int cnt=1,now=x;  
 while(now!=y) cnt++,vis[now]=-1,now=s[--top];  
 if(cnt&1) sg[y]^=1; //奇环   
 }  
 else if(!vis[y]){  
 dfs(y,x);  
 if(~vis[y]) sg[x]^=(sg[y]+1); //不在环上的才能更新   
 }  
 }  
 if(~vis[x]) --top; //非环上的，及时出栈   
}  
signed main(){  
 while(~scanf("%lld",&t)){  
 ans=0;  
 while(t--){  
 scanf("%lld%lld",&n,&m),cnt=top=0;  
 for(int i=1;i<=n;i++) hd[i]=vis[i]=0;  
 for(int i=1;i<=m;i++){  
 scanf("%lld%lld",&x,&y);  
 add(x,y),add(y,x);  
 }  
 dfs(1,0),ans^=sg[1]; //根节点的 SG 值异或起来   
 }  
 puts(ans?"Sally":"Harry");  
 }  
 return 0;  
}

三、图的删边游戏

「POJ3710 Christmas Game」去掉环的限制。即：给定一个无向连通图，有一个点作为图的根。两人轮流操作，每人每次可以从图中选择一条边删去，不与根节点相连的部分将被移走。无法操作者输。

对于这个模型，有一个定理——Fusion Principle。

结论：对无向图做如下改动：将图中的任意一个偶环缩成一个新点，任意一个奇环缩成一个新点加一条新边；所有连到原先环上的边全部改为与新点相连。这样的改动不会影响图的 值。

因此可以将任意一个无向图改成树结构。然后就能把「图的删边游戏」转化为「树的删边游戏」了。

四、参考资料[《组合游戏略述——浅谈 SG 游戏的若干拓展及变形》](https://wenku.baidu.com/view/25540742a8956bec0975e3a8)（大部分内容摘自这儿）