```
vector<int> adj[2100];
vector<int> radj[2100];
inline int TRUE(int x) {return x + x;}
inline int FALSE(int x) {return x + x + 1;}
inline void link(int x,int y) {adj[x].push_back(y);radj[y].push_back(x);}
int n,m;
vector<int> st;
bool visit[2100];
int fa[2100];

void dfs_1(int now) {
    if (visit[now]) return;
    visit[now] = true;
    for (int i = 0; i < adj[now].size(); i++) {
        int next = adj[now][i];
        dfs_1(next);
    }
    st.push_back(now);
}
void dfs_2(int now,int blk) {
    if (visit[now]) return;
    visit[now] = true;
    fa[now] = blk;
    for (int i = 0; i < radj[now].size(); i++) {
        int next = radj[now][i];
        dfs_2(next,blk);
    }
}

bool check() {
    memset(visit,0,sizeof(visit));
    for (int i = 0; i < (n << 1); i++) {
        if (!visit[i]) {
            dfs_1(i);
        }
    }
    memset(visit,0,sizeof(visit));
    memset(fa,0xff,sizeof(fa));
    int nn = 0;
    while (st.size()) {
        int now = *st.rbegin(); st.pop_back();
        if (!visit[now]) {
            dfs_2(now, ++nn);
        }
    }
    for (int i = 0; i < n; i++) {
        if (fa[TRUE(i)] == fa[FALSE(i)]) {
            return false;
        }
    }
    return true;
}
```