

报文回路

题目描述：

IGMP 协议中响应报文和查询报文，是维系组播通路的两个重要报文，在一条已经建立的组播通路中两个相邻的 HOST 和 ROUTER，ROUTER 会给 HOST 发送查询报文，HOST 收到查询报文后给 ROUTER 回复一个响应报文，以维持相互之间互通的关系，一旦这个关系断裂，那么这条组播通路就“异常”了。现通过某种手段，抓取到了 HOST 和 ROUTER 两者通讯的所有响应报文和查询报文，请分析该组播通路是否“正常”

输入描述：

第一行抓到的报文数量 C（C<=100），后续 C 行依次输入设备节点 D1 和 D2，表示从 D1 到 D2 发送了单向的报文，D1 和 D2 用空格隔开。

输出描述：

组播通路是否“正常”，正常输出 True，异常输出 False。

补充说明：

示例 1

输入：

```
5
1 2
2 3
3 2
1 2
2 1
```

输出：

```
True
```

说明：

示例 2

输入：

```
3
1 3
```

3 2

2 3

输出:

False

说明:

```
import java.util.Scanner;
import java.util.Map;
import java.util.HashMap;
import java.util.Set;
import java.util.HashSet;
public class Main {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        int m = Integer.parseInt(in.nextLine());
        Map<Integer,Set<Integer>> map = new HashMap<>();
        //初始化
        int[] allLeft = new int[m];
        for(int i=0;i<m;i++){
            String[] numArray=in.nextLine().split(" ");
            int x = Integer.parseInt(numArray[0]);
            int y = Integer.parseInt(numArray[1]);
            allLeft[i]=x;
            if(map.containsKey(x)){
                Set<Integer> list = map.get(x);
                list.add(y);
                map.put(x,list);
            }else{
                Set<Integer> list = new HashSet<>();
                list.add(y);
                map.put(x,list);
            }
        }
        String result = "True";
        for(int i=0;i<m;i++){
            Integer z=allLeft[i];
            Set<Integer> set=map.get(z);
            for(Integer x:set){
                if(!map.containsKey(x)){
                    result = "False";
                    break;
                }
            }
            Set<Integer> validSet=map.get(x);
            boolean has = false;
```

```
        if(!validSet.contains(z)){
            result = "False";
            break;
        }
    }
}
System.out.println(result);
}
```