

## A、用户管理系统

---

**解题思路：**通读题意，就是要看给定的两个节点，**x和y是不是以x为LCA。**

所以采取使用链式前向星存图，然后使用dfs序+rmq的做法，最后再使用ST表对查询进行优化。

**注意事项：**本题需要自己找出根节点在哪里，所以在读入边信息时，确定每一个节点的父节点是谁。最后遍历一遍fa数组就可以知道根节点了。

## G、模拟宇宙

---

**解题思路：**有向图，经过一个节点起，最多可以经过多少节点。可以尝试将所有边反向。再使用拓扑排序的kahn算法。每一次将入度为0的节点一次性全部加入队列，并赋给他们相同的值cnt。入队操作结束后cnt++。然后从队列中取元素（直到取完），将其关联的边删除（如果期间有新的入度为0的节点出现，则入队）。循环下去直至队空。

## H、救援行动

---

**解题思路：**最小生成树问题，考虑到边比较多，是个稠密图，所以使用prim算法。边多所以用邻接矩阵存图，在初始化一个vis数组表示是否被访问过，一个dis数组表示新节点到mst距离。

## P、利萨斯陷落

---

**解题思路：**即求欧拉路径的最小字典序。无向图的欧拉路径：所有节点度为偶数，或者仅有两个节点度为奇数，其他节点度都为偶数。所以在main函数先判断各节点的度是不是满足构成欧拉路径的要求，不满足就直接return。选出遍历起点（有奇数度的点，就从小那个奇数度的节点开始，全是偶数度就从1开始dfs）随后进行dfs即可，由于要找最小字典序，所以还可以使用一个优先队列，每次取出邻接点中可用的最小的节点，将储存结果放入栈中，最后输出即可。

**注意事项：**还有一个坑，就是给的节点，可能不是一个连通图，（所以即使度满足条件，也可能无法构成欧拉路径）。所以我们需要在输入的时候就使用并查集来处理，判断一下图是否连通。

## R、你好世界

---

**解题思路：**本题的难点主要在于建图的方式。如果按照题意建图就太复杂了。所以我们考虑进行优化。给定a，b区间，那我们可以对a到b建立一条边而内部用长度为0的边连接起来，从而大大减少了复杂度。

**注意事项：**相邻节点长度为0的边是从后指向前的有向边，而外部的a到b的边是从前指到后的有向边。

## V、客户

---

**解题思路：**典型的匈牙利算法求二分图的最大匹配。

循环处理需求一方的每个点即可

```
for (int i = 1; i <= m; i++) {  
    memset(vis, 0, sizeof(vis));  
    if (dfs(i)) ans++;  
}
```

```
bool dfs(int u) {  
    for (int v : g[u]) {  
        if (vis[v]) continue;  
        vis[v] = 1;  
        if (!match[v] || dfs(match[v])) {  
            match[v] = u;  
            return true;  
        }  
    }  
    return false;  
}
```