



11.1-11.30 注册AWS账号
立享12个月免费套餐+500美元服务抵扣券

AOQNRMGYXLMV

New life, new game.

博客园

首页

联系

管理

随笔 - 672 文章 - 3 评论 - 57

HDU 3594 Cactus 有向仙人掌图判定

题意

给出一个有向图，并给出仙人掌图的定义

1. 图本身是强连通的
2. 每条边属于且只属于一个环

判断输入的图是否是强连通的。

分析

杭电OJ上的数据比较弱，网上一些有明显错误的代码也能AC。

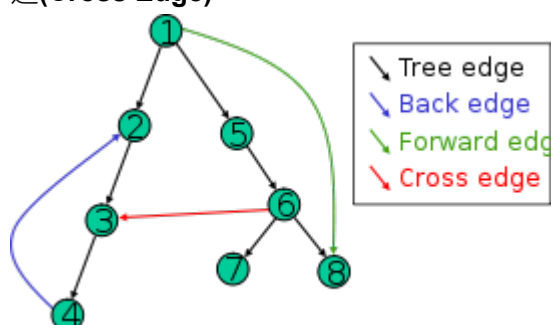
本着求真务实的精神，取网上查阅了相关资料，整理出来一个对自己来说还比较明确的算法。

从DFS森林说起

从有向图的某一点开始进行深度优先遍历，按照遍历的先后顺序会形成一棵树，像这种边被称作树边(Tree Edge)

当然有向图中还可能会存在一些其他的边：

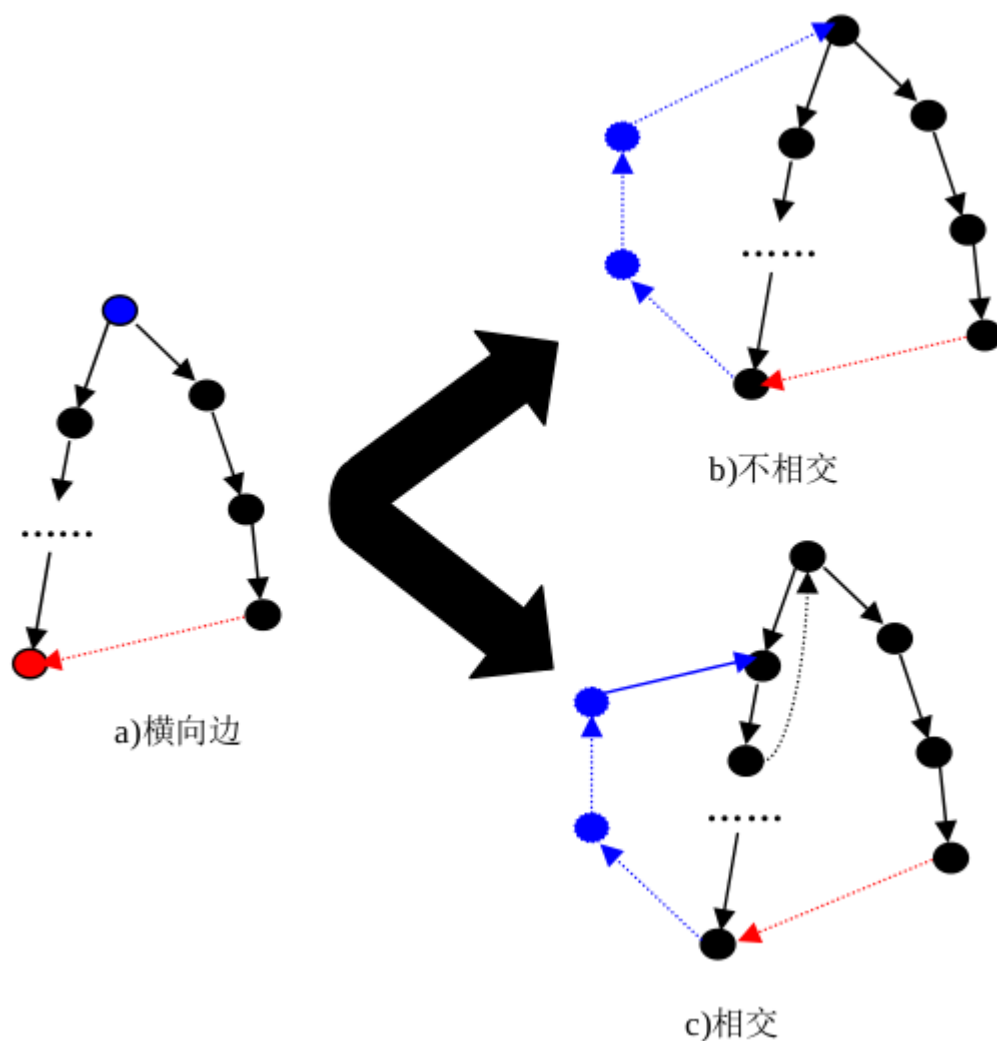
- 从当前节点连向其祖先节点的边叫做反向边(Back Edge)
- 从当前节点连向其后代节点的边叫做前向边(Forward Edge)
- 从当前节点连向其他节点，可能是某个祖先其他分支的节点或者另一颗DFS树的节点，这种边叫做交叉边(Cross Edge)



按边的分类考虑仙人掌图

接下来默认图是强连通的，后面不再强调。

1. 如果 $u \rightarrow v$ 是一条前向边，必然有一条从 v 到 u 的路径 $Path$ 。这样 $Path$ 就和前向边 $u \rightarrow v$ 构成了一个环，同时也和树边上的 u 到 v 的路径构成了一个环，而且这两个环有公共路径 $Path$ 。因此得到结论：仙人掌图中不含前向边。
2. 如果 $u \rightarrow v$ 是一条交叉边，它们的最近公共祖先为 anc 。同样也有一条从 v 到 anc 的路径 $Path_{v \rightarrow anc}$ ，这条路径和 v 到 anc 的路径或相交或不相交。同样也构成了两个有公共边的环，因此得到结论：仙人掌图中不含交叉边。



因此，除了树边只剩下反向边，而且可以看出每有一条反向边 $u \rightarrow v$ ，它和树边上的路径 $v \rightarrow u$ 构成了一个环。

下面想办法保证每条树边至多被一个环所包含：

- 一个点最多有一条反向边
- 在当前节点记录一个可以返回的最小的DFS序，保证反向边指向的节点的DFS序不能小于该值，否则会出现有公共边的两个环。

这是通过一遍DFS实现的。

```
#include <stdio>
#include <string>
#include <algorithm>
```

```

#include <stack>
#include <vector>
using namespace std;

const int maxn = 10000 + 10;

int n, m;
vector<int> G[maxn];

stack<int> S;
int dfs_clock, pre[maxn], low[maxn];
int scc_cnt, sccno[maxn];

void dfs(int u) {
    pre[u] = low[u] = ++dfs_clock;
    S.push(u);
    for(int v : G[u]) {
        if(!pre[v]) {
            dfs(v);
            low[u] = min(low[u], low[v]);
        } else if(!sccno[v]) low[u] = min(low[u], pre[v]);
    }
    if(low[u] == pre[u]) {
        scc_cnt++;
        for(;;) {
            int x = S.top(); S.pop();
            sccno[x] = scc_cnt;
            if(x == u) break;
        }
    }
}

//Tarjan算法求强连通分量
void find_scc() {
    dfs_clock = scc_cnt = 0;
    memset(pre, 0, sizeof(pre));
    memset(sccno, 0, sizeof(sccno));
    for(int i = 0; i < n; i++) if(!pre[i])
        dfs(i);
}

//第二遍DFS保证是仙人掌图

//color[u]为0表示还没有访问, 为1表示正在访问, 为2表示已经访问完毕
int color[maxn];

bool dfs2(int u, int minBack) {    //minBack表示反向边能指向的最小的DFS序
    color[u] = 1;
    int backs = 0; //反向边的个数, 至多只能有一个
    for(int v : G[u]) if(color[v] == 1) {    //找到一条反向边
        backs++;
        if(backs > 1) return false;
        if(pre[v] < minBack) return false;    //反向边指向的节点的DFS序小于最小值
    }
    if(backs) minBack = pre[u];
    for(int v : G[u]) {
        if(color[v] == 2) return false;    //前向边或交叉边
        if(color[v] == 0)    //树边

```

```

        if(!dfs2(v, minBack)) return false;;
    }
    color[u] = 2;
    return true;
}

int main()
{
    int T; scanf("%d", &T);
    while(T--) {
        scanf("%d%d", &n, &m);
        for(int i = 0; i < n; i++) G[i].clear();
        while(m--) {
            int u, v; scanf("%d%d", &u, &v);
            G[u].push_back(v);
        }

        find_scc();
        if(scc_cnt > 1) { puts("NO"); continue; }

        memset(color, 0, sizeof(color));
        if(!dfs2(0, 0)) puts("NO");
        else puts("YES");
    }

    return 0;
}

```

参考资料

1.仙人掌图分析

2.[my solution](#)注：这份代码没有考虑只能有一条反向边的限制，但也能在**UVa**上**AC**

标签: [图论](#) , [强连通分量](#) , [仙人掌图](#)

谢谢推荐! 关注成功

收藏该文



AOQNRMGYXLMV

关注 - 27

粉丝 - 58

关注成功

1

推荐

0

反对

支持成功

« 上一篇: [HDU 3861 The King's Problem](#) 强连通分量 最小路径覆盖

» 下一篇: [HDU 2460 Network](#) 边双连通分量 缩点

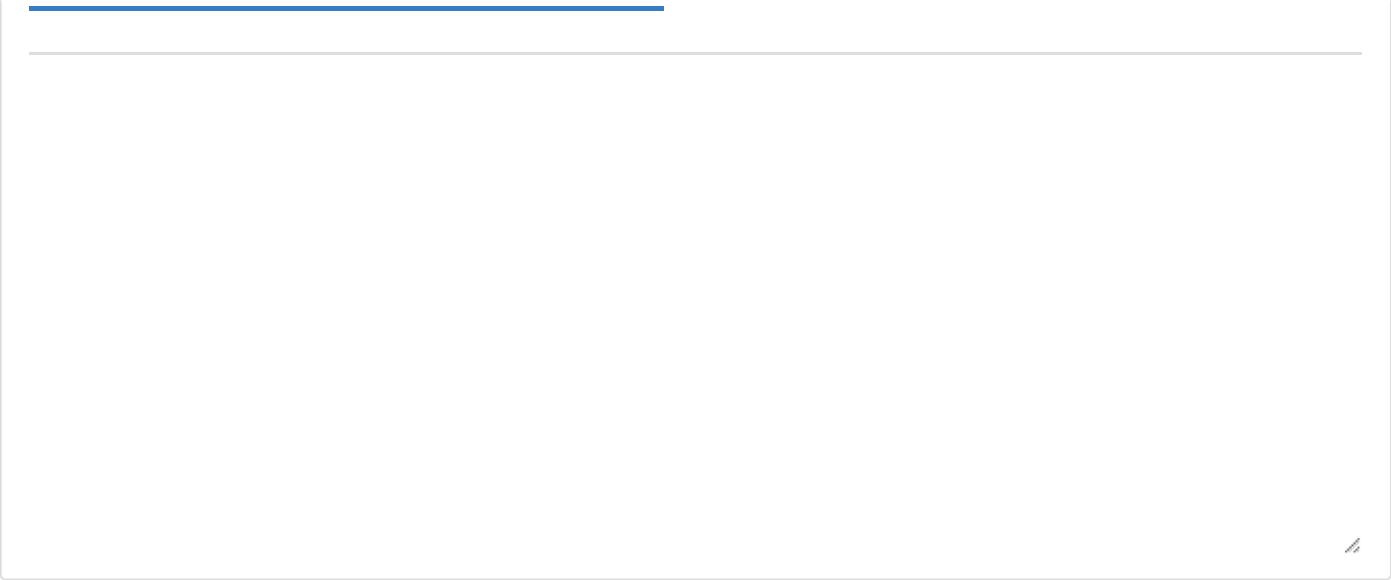
posted @ 2016-04-05 01:18 AOQNRMGYXLMV 阅读(806) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

发表评论 【福利】注册AWS账号，立享12个月免费套餐

编辑

预览



[退出](#) [订阅评论](#) [我的博客](#)

[Ctrl+Enter]快捷键提交

[首页](#) [新闻](#) [博问](#) [专区](#) [闪存](#) [班级](#)

相关博文：

- [【HDU 3594 Cactus】tarjan+仙人掌图](#)
- [hdu 3594 Cactus](#)
- [仙人掌图判定及求直径HDU3594 BZOJ1023](#)
- [BZOJ 1023 \[SHOI2008\]cactus仙人掌图](#)
- [hdu 3594 Cactus](#)
- » [更多推荐...](#)

最新 IT 新闻：

- [从阿里、京东、腾讯手里“吸金”，这家欧洲奢侈品电商凭什么？](#)
- [携程四君子：中国最美创业故事](#)
- [运营了15年的魔兽世界 居然要把等级压回60级？](#)
- [库克大发慈悲？ 30%“苹果税”直砍一半，网友：这是用“富人税”养平民](#)
- [谷歌宣布针对Chrome扩展程序的重大隐私政策更新](#)
- » [更多新闻...](#)

公告

昵称： AOQNRMGYXLMV

园龄： 6年5个月

粉丝： 58

关注： 27

关注成功

<	2020年11月						>
日	一	二	三	四	五	六	
1	2	3	4	5	6	7	
8	9	10	11	12	13	14	
15	16	17	18	19	20	21	
22	23	24	25	26	27	28	
29	30	1	2	3	4	5	
6	7	8	9	10	11	12	

搜索



我的标签

2015暑假集训(58)

DP(57)

数据结构(46)

图论(31)

数学(30)

线段树(19)

树形DP(16)

主席树(11)

数论(10)

树链剖分(9)

更多

随笔分类

BFS(20)

DFS(15)

暴力(3)

背包问题(10)

并查集(16)

动归/递推(75)

二分查找(12)

- 概率/期望(13)
- 高精度(8)
- 构造(3)
- 计算几何(36)
- 记忆化搜索(7)
- 模拟(13)
- 排序检索(2)
- 树状DP(4)
- 数据结构(61)
- 数论(33)
- 数学(121)
- 数学_组合游戏(26)
- 贪心(29)
- 图论(28)
- 图论_网络流(8)
- 线段树(17)
- 杂记(5)
- 杂题(39)
- 状态压缩(2)
- 状压DP(6)
- 字符串(36)
- 最长公共子序列(6)
- 最大子段问题(7)

巨巨们的博客

- cxlove
- Staginner
- kuangbin
- 不慌不忙、不急不躁
- iwtwiiioi
- ACdreamers
- PureFrog
- 英雄哪里出来

最新评论

1. Re:【读书笔记】莫比乌斯函数与莫比乌斯反演
orz

--Cro-Marmot
2. Re:【读书笔记】莫比乌斯函数与莫比乌斯反演
大佬您的LaTeX挂掉了。。。QAQ

--zjp_shadow
3. Re:UVa 1606 (极角排序) Amphiphilic Carbon Molecules
请问统计统计p[L]到p[R]之间的点是什么意思啊 为什么当区间大于180度停止啊

--SWTCZB
4. Re:CodeForces 552D Closest Equals 树状数组
唔, 貌似是522D

--ldvzchuscvol

5. Re:CodeForces 771C Bear and Tree Jumps 树形DP

@ AOQNRMGYXLMV谢谢了, 理解了, Thanks!!!...

--ldvzchuscvol

阅读排行榜

1. 【计算几何模板】求两个圆的交点(5082)
2. 【读书笔记】莫比乌斯函数与莫比乌斯反演(2076)
3. CodeForces 8D Two Friends 判断三个圆相交(1418)
4. SPOJ QTREE4 - Query on a tree IV 树分治(1268)
5. UVa 1572 (拓扑排序) Self-Assembly(980)

Copyright © 2020 AOQNRMGYXLMV
Powered by .NET 5.0.0 on Kubernetes