

9



2



<

## 原 支配树 与 tarjan算法

2015年11月19日 15:21:50   a710128   阅读数 8312   文章标签: [算法](#) [支配树](#) [更多](#)

版权声明：本文为博主原创文章，遵循 [CC 4.0 BY-SA](#) 版权协议，转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/a710128/article/details/49913553>

## 简介

## 什么是支配树？支配树是什么？ XD

对于一张有向图（可以有环）我们规定一个起点 $r$ （为什么是 $r$ 呢？因为网上都是这么规定的），从 $r$ 点到图上另一个点 $w$ 可能存在很多条路径（下面将 $r > w$ ）。

如果对于 $r \rightarrow w$ 的任意一条路径中都存在一个点 $p$ ，那么我们称点 $p$ 为 $w$ 的支配点（当然这也是 $r \rightarrow w$ 的必经点），注意 $r$ 点不讨论支配点。下面用 $idom[u]$ 近的支配点。

对于原图上除 $r$ 外每一个点 $u$ ，从 $\text{idom}[u]$ 向 $u$ 建一条边，最后我们可以得到一个以 $r$ 为根的树。这个树我们就叫它“支配树”。

## 相似

这个东西看上去有点眼熟？

支配点和割点（删掉后图联通块数增加）有什么区别？

我们考虑问题给定一个起点 $r$ 和一个终点 $t$ ，询问删掉哪个点能够使 $r$ 无法到达 $t$ 。

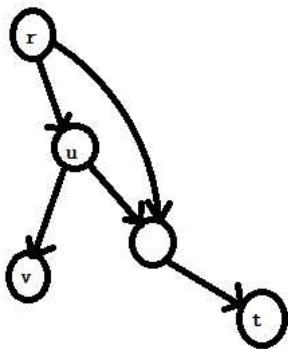
很显然，我们删掉任意一个 $r \rightarrow t$ 的必经点就能使 $r$ 无法到达 $t$ ，删掉任意一个非必经点， $r$ 仍可到达 $t$ 。

从支配树的角度来说，我们只需删掉支配树上 $r$ 到 $t$ 路径上的任意一点即可

从割点的角度来说，我们是不是只需要考虑所有割点，判断哪些割点在 $r \rightarrow t$ 的路径上即可？是否将某个割点删掉即可让 $r$ 无法到达 $t$ ？

这当然是不正确的，我们可以从两个方面来说明它的错误：

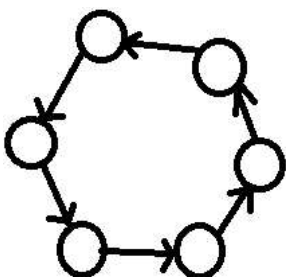
1. 删掉割点不一定使r无法到达t



这个图中点u是关键点（删掉后图联通块个数增加）

并且u在r->t的路径上, 然而删掉点u后r仍然可以到达t

2. 图中不一定存在割点



在这个图中不存在任何割点



所以我们没有办法使用割点来解决这个问题。

## 简化问题

### • 树

对于一棵树，我们用 $r$ 表示根节点， $u$ 表示树上的某个非根节点。很容易发现从 $r \rightarrow u$ 路径上的所有点都是 $u$ 的祖先点，而 $\text{idom}[u]$ 就是 $u$ 的父节点。这个可以在 $O(n)$ 的时间内实现。

### • DAG（有向无环图）

因为是有向无环图，所以我们可以按照拓扑序构建支配树。  
假设当前我们构造到拓扑序中第 $x$ 个节点编号为 $u$ ，那么拓扑序中第 $1 \sim x-1$ 个节点已经处理好了，考虑 $u$ 的祖先点，能够直接到达点 $u$ 的节点，对于这些节点在支配树上的最近公共祖先 $v$ ，这个点 $v$ 就是点 $u$ 在支配树上的父亲。  
如果使用倍增求LCA，这个问题可以在 $O((n+m)\log_2 n)$ 的时间内实现。

对于这两个问题我们能够很简便的求出支配树。

## 有向图

对于一个有向图，我们应该怎么办呢？

### 简单方法

我们可以考虑每次删掉一个点，判断哪些点无法从 $r$ 到达。  
假设删掉点 $u$ 后点 $v$ 无法到达，那么点 $u$ 就是 $r \rightarrow v$ 的必经点（点 $u$ 就是 $v$ 的支配点）。  
这个方法我们可以非常简单的在 $O(nm)$ 的时间内实现。  
其中 $n$ 是点数， $m$ 是边数。

### 更快的方法

这里，我将介绍Lengauer-Tarjan算法。  
这个算法能在很快的时间内求出支配树。  
要介绍这个算法我们还需引入两个定理和一些概念

### 大概步骤

首先来介绍一些这个算法的大概步骤

1. 对图进行DFS（深度优先遍历）并求出搜索树和DFS序。这里我们用 $dfn[x]$ 表示点 $x$ 在dfs序中的位置。
2. 根据半必经点定理计算出所有的半必经点作为计算必经点的根据
3. 根据必经点定理修正我们的半必经点，求出支配点

### 半必经点

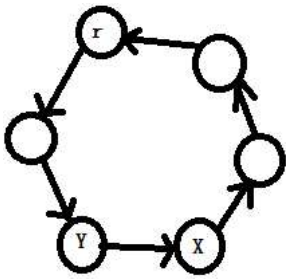
我们用 $\text{idom}[x]$ 表示点 $x$ 的最近支配点，用 $\text{semi}[x]$ 表示点 $x$ 的半必经点。  
那什么是半必经点呢？

对于一个节点 $Y$ ，存在某个点 $X$ 能够通过一系列点 $p_i$ （不包含 $X$ 和 $Y$ ）到达点 $Y$ 且 $\forall i, dfn[p_i] > dfn[Y]$ ，我们就称 $X$ 是 $Y$ 的半必经点，记做 $\text{semi}[Y] = X$

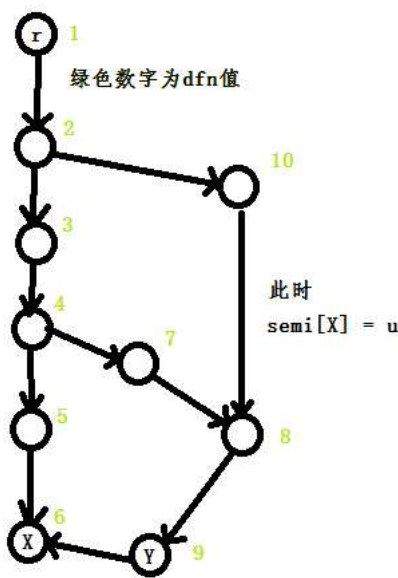
当然一个点 $X$ 的“半必经点”会有多个，而且这些半必经点一定是搜索树中点 $X$ 的祖先（具体原因这里不做详细解释，请自行思考）。  
对于每个点，我们只需要保存其半必经点中 $dfn$ 最小的一个，下文中用 $\text{semi}[x]$ 表示点 $x$ 的半必经点中 $dfn$ 最小的点的编号。  
我们可以更书面一点的描述这个定理：

- 对于一个节点 $Y$ 考虑所有能够到达它的节点，设其中一个为 $X$

- 若 $dfn[X] < dfn[Y]$ ，则 $X$ 是 $Y$ 的一个必经点



- 若 $dfn[X] > dfn[Y]$ ，那么对于 $X$ 在搜索树中的祖先 $Z$ (包括 $X$ )，如果满足 $dfn[Z] > dfn[Y]$ 那



在这些必经点中，我们仅需要 $dfn$ 值最小的  
这个必经点有什么意义呢？

我们求出深搜树后，考虑原图中所有非树边（即不在树上的边），我们将这些边删掉，加入一些新的边  $\{(semi[w], w) | w \in V \text{ and } w \neq r\}$ ，我们会发现构建出的点的支配点是不变的，通过这样的改造我们使得原图变成了DAG

是否接下来使用DAG的做法来处理就可以做到 $nlog_2n$ 呢？我没试过，不过我有更好的方法。

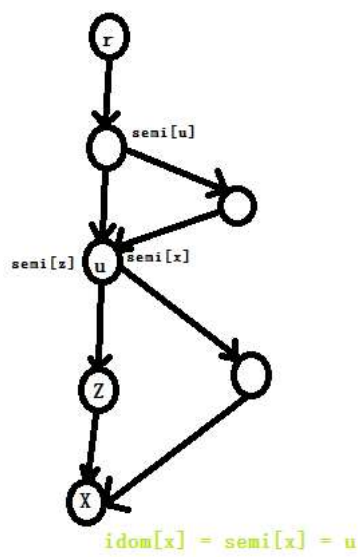
必经点

一个点的必经点有可能是一个点的支配点，也有可能不是。我们需要使用必经点定理对这个必经点进行修正，最后得到必经点

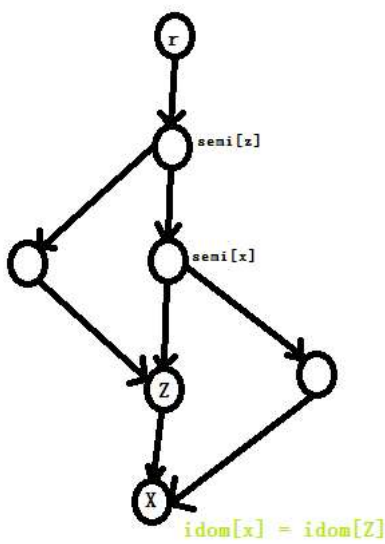
对于一个点 $X$ ，我们考虑搜索树上 $semi[X]$ 到 $X$ 路径上的所有点 $p_0, p_1, p_2, p_3 \dots p_k$ 。对于所有 $p_i (1 \leq i < k)$ ，我们找出 $dfn[semi[p_i]]$ 最小的一个 $p_i$ 记为 $Z$

- 考虑搜索树上 $X$ 与 $semi[X]$ 之间的其他节点（即不包含 $X$ 和 $semi[X]$ ），其中必经点 $dfn$ 值最小的记为 $Z$

- 如果 $semi[Z] = semi[X]$ , 则 $idom[X] = semi[X]$



- 如果 $semi[Z] \neq semi[X]$ , 则 $idom[X] = idom[Z]$



具体实现

对于求半必经点与必经点我们都需要处理一个问题，就是对于一个节点 $X$ 的前驱 $Y$ ，我们需要计算 $Y$ 在搜索树上所有 $dfn$ 值大于 $dfn[X]$ 的祖先中 $semi$ 个，我们可以按 $dfn$ 从大到小的顺序处理，使用并查集维护，这样处理到节点 $X$ 值时所有 $dfn$ 值比 $X$ 大的点都被维护起来了。对于 $Y$ 的所有满足条件的祖先，就是在并查集中 $Y$ 的祖先，可以通过带权并查集的方法，维护祖先中的最小值，并记下 $semi$ 最小的具体是哪个节点。这样我们就能够在 $O((n + m) \times \alpha(n))$ 时间内解决这个问题。

代码

这里提供 Codechef May15 中 GRAPHCNT 题目的代码

```
1  #include <iostream>
2  #include <cstdio>
3  using namespace std;
4  typedef long long lld;
5  const int MaxN = 100000 + 10, MaxE = (5 * 100000) * 2 + MaxN;
6  int head[MaxN], pre[MaxN], dom[MaxN], to[MaxE], nxt[MaxE], top;
7  void addedge(int *h,int fr,int tt)
8  {
9      top ++;
10     nxt[top] = h[fr];
11     to[top] = tt;
12     h[fr] = top;
```

```

13 }
14 int n, m;
15 void init()
16 {
17     scanf("%d%d", &n, &m);
18     int a, b;
19     for(int i = 1; i <= m; ++i)
20     {
21         scanf("%d%d", &a, &b);
22         addedge(head, a, b);
23         addedge(pre, b, a);
24     }
25 }
26 int bcj[MaxN], semi[MaxN], idom[MaxN], best[MaxN], dfn[MaxN], id[MaxN], fa[MaxN], dfs_clock;
27 int push(int v)
28 {
29     if(v == bcj[v]) return v;
30     int y = push(bcj[v]);
31     if(dfn[semi[best[bcj[v]]]] < dfn[semi[best[v]]]) best[v] = best[bcj[v]];
32     return bcj[v] = y;
33 } //带权并查集路径压缩
34 void dfs(int rt)
35 {
36     dfn[rt] = ++dfs_clock;
37     id[dfs_clock] = rt;
38     for(int i = head[rt]; i; i = nxt[i])
39         if(!dfn[to[i]])
40         {
41             dfs(to[i]);
42             fa[to[i]] = rt;
43         }
44 }
45 } //求出dfs序
46 void tarjan()
47 {
48     for(int i = dfs_clock, u; i >= 2; --i)
49     { //按dfs序从大到小计算
50         u = id[i];
51         for(int j = pre[u]; j; j = nxt[j]) //semi
52         {
53             if(!dfn[to[j]]) continue;
54             push(to[j]);
55             if(dfn[semi[best[to[j]]]] < dfn[semi[u]]) semi[u] = semi[best[to[j]]];
56         }
57         addedge(dom, semi[u], u);
58         bcj[u] = fa[u]; u = id[i - 1];
59         for(int j = dom[u]; j; j = nxt[j]) //idom
60         {
61             push(to[j]);
62             if(semi[best[to[j]]] == u) idom[to[j]] = u;
63             else idom[to[j]] = best[to[j]];
64         }
65     }
66     for(int i = 2, u; i <= dfs_clock; ++i)
67     {
68         u = id[i];
69         if(idom[u] != semi[u]) idom[u] = idom[idom[u]];
70     }
71 }
72 int sons[MaxN];
73 lld ans;
74 void calc_son()
75 {
76     for(int i = dfs_clock, u; i >= 2; --i)
77     {
78         u = id[i];
79         ++ sons[u];
80         if(idom[u] != 1) sons[idom[u]] += sons[u];
81         else ans -= ((lld)sons[u] * (lld)(sons[u] - 1)) / 2ll;
82     }
83 }

```



9



2



```
84     }
85 }
86 void solve()
87 {
88     for(int i = 1; i <= n; ++i) bcj[i] = semi[i] = best[i] = i;
89     dfs_clock = 0;
90     dfs(1);
91     tarjan();
92     ans = ((1ll)dfs_clock * (1ll)(dfs_clock - 1)) / 2ll;
93     calc_son();
94     cout << ans << endl;
95 }
96 int main()
97 {
98     init();
99     solve();
100    return 0;
}
```

9



2









**人脸识别主要算法原理**  
人脸识别算法  
4675阅读



想对作者说点什么



似火^: 对于一个节点Y, 存在某个点X能够通过一系列点pi (不包含X和Y) 到达点Y且 $\forall i \text{ dfn}[i] > \text{dfn}[Y]$ , 我们就称X是Y的半必经点, 记做 $\text{semi}[Y]=X$  是说  $\text{dfn} [ i ]$  还是  $\text{dfn} [ \text{pi} ]$ ? 为什么是 i 呢? (1个月前 #2楼)



YouSiki: 并非所有半必经点都一定是X的搜索树祖先, 只有最终的那个DFS序最小的半必经点一定是其搜索树祖先。 (2年前 #1楼)

**全网最!详细!tarjan算法讲解**  
阅读数 3万+

转自: 点击打开全网最详细tarjan算法讲解, 我不敢说别的。反正其他tarjan算法讲解, 我看了半天才看懂。我写的... [博文](#) 来自: [K键盘里的青春K](#)

**感性理解支配树**  
阅读数 1410

这篇文章给想要性感感性地学习支配树的同学。 [博文](#) 来自: [litble的成\(tui\)长\(fe...](#)

**支配树与Lengauer-Tarjan算法**  
阅读数 150

伪目录给出支配树的定义给出一些性质介绍快速构造支配树的Lengauer-Tarjan算法及具体实现支配树是啥一个有源... [博文](#) 来自: [Canopus](#)

**Tarjan算法详解**  
阅读数 1万+

在有向图G中, 如果两个顶点间至少存在一条路径, 称两个顶点强连通(stronglyconnected)。如果有向图G的每两... [博文](#) 来自: [刘伟的博客](#)

**人脸识别主要算法原理**

**支配树 Dominator Tree**  
阅读数 775

写在前面今天亮神花了一早上给讲了支配树, 但由于今天网(没)速(人)不(跟)好(上), 我们残忍地看了他一... [博文](#) 来自: [VioletSu的博客](#)

**支配树**  
阅读数 20

入门版 (DAG图) 洛谷P2597[ZJOI2012]灾难https://www.luogu.org/problem/P2597给的图为DAG图建反向图通... [博文](#) 来自: [七九河开的博客](#)

**学习一个支配树**  
阅读数 389

http://blog.csdn.net/qq\_35649707/article/details/64125918http://blog.csdn.net/GEOTCBRL/article/detai... [博文](#) 来自: [onepointo](#)

**【学习小记】支配树【图论】**  
阅读数 61

Preface给定一个有向图和一个起点ststst, 我们需要知道起点到某个点的关于必经点的信息。若起点到点v的所有路... [博文](#) 来自: [Never give in.](#)

**[洛谷P5180]【模板】支配树**  
阅读数 87

对于DAGDAGDAG, 拓扑排序++ +倍增LCALCALCA就可以做了 (不会的可以先做[ZJOI2012]灾难) 对于一般有向... [博文](#) 来自: [\\*★,°\\*:.☆\(￣▽￣\)/\\$:\\*...](#)

【惊呆】3分钟就萎了? 教你这样做, 一招突破40分钟!

家然 · 猎媒

快速构造支配树的Lengauer-Tarjan算法

本篇口胡写给我自己这样的老是证错东西的口胡选手以及那些想学支配树，又不想啃论文原文的人... 大概会讲的...

【学习笔记】支配树

【前言】本文为博主的转载，由于博主看到的文章同样是转载的，无法注明原文出处。博主在原文的基础上修改了格...



键盘里的青春

644篇文章

排名:2000+

关注



litble

276篇文章

排名:千里之外

关注



wang33123621

305篇文章

排名:千里之外

关注



玩人

137篇文章

排名:7000+

关注

ubuntu18.04 挂载ntfs硬盘无法写入解决办法

win10和ubuntu18.04双系统，在ubuntu下通过/etc/fstab挂载ntfs硬盘无写入权限，尝试通过chmod修改写入权限...

Dominator Tree--支配树

在学习支配树之前，请保证已经会写lca（tarian求法）简介支配树是什么？支配树能干什么？对于一个DAGDAGDA...

Ubuntu 挂在 NTFS 格式的硬盘

发现Ubuntu的文件管理器中有显示的对应的NTFS格式的硬盘，但打不开，出现错误如下用ntfsfix命令修复虽然可以...

内幕:200斤瘦成90斤，女人减肥就用它，效果比抽脂还狠！

争霸减肥 · 猎媒

Lengauer-Tarjan算法--支配树构造（bzoj 2815: [ZJOI2012]灾难）

模型：一个有向图G，设定一个点r，要求点r能到达G中所有的点，如果这样的点不存在，新建并向所有点连边支配点...

支配树（以HDU4694 Important Sisters为例）

定义在一个流程图（单源有向图）中，如果从源点到某个点p必须要经过某个点q，则q是p的支配点，我们记除了该...

BJ模拟 医院【支配树】

题目描述：过年辣！放假辣！可是有个地方必须有人守着——医院。护士们也想放假，但是她们必须轮流放。现在医...

2019.01.19【BZOJ3281】小P的烦恼（支配树）

传送门解析：首先本题数据不是DAG。。。对的就是数据有误不然有更加优秀的做法。。。思路：必经边是吧，那就...

支配树dominator tree学习笔记

说实话冬令营之前我都没见过支配树的题，也不知道这是个什么东西（但是学完之后感觉这个东西已经烂大街了qaq...

女孩子千万不要让男票发现这传奇！开局一条龙吸引力太大了！

贪玩游戏 · 顶新

支配树(Dominator Tree)学习笔记

因为是学习笔记所以是边学边写，以防自己到时候忘掉了=w=首先，支配树是一棵树（废话），用来求解必经点问题...

支配树(Dominator tree)学习笔记 及HDU4694Important Sisters

前置技能：LCA(Tarjan)今年冬令营上讲的东西现在才学惭愧惭愧。。首先做出图G的dfs树T其中S为起点定义T中节...

【DAG】寻找桥边、必经点（支配树）

在许多题目中，常常都需要一下操作：给出S、T，找一个桥边（必经点）Solution1先从S走到T，记录下路径（包括...

HDU 4694: Important Sisters（支配树）

题意：御坂和她的n-1个克隆姐姐（编号1~n-1）构成一张御坂网络，如果第x号"姐姐"和第y号"姐姐"有条x到y的有...

2017.11.2 支配树上LCA 解题报告

题目描述给出一个无向图（n

27岁美女揭开驼奶市场秘密，亲赴大漠，发现惊人真相

好伙伴 · 猎媒

9

2

来自: DZYO的博客

来自: cz\_xuyixuan的博客

来自: 玩人

来自: huying915的专栏

来自: 苟为蒟蒻又何妨

来自: huihuiwith的博客

来自: 加载中...

来自: zyyylx的博客

来自: cdsszjj的博客

来自: zxyoi\_dreamer的...

来自: L\_0\_Forever\_LF的...

来自: 时光真疯狂, 我一路...

来自: Don't fake it till y...

来自: 伊克赛艇旗舰店

来自: 加载中...

来自: onepointo

阅读数 2467

阅读数 421

阅读数 6431

阅读数 226

阅读数 787

阅读数 537

阅读数 304

阅读数 76

阅读数 239

阅读数 928

阅读数 746

阅读数 1959

阅读数 832

阅读数 330

阅读数 151

https://blog.csdn.net/a710128/article/details/49913553

7/11



<div><div>【模板】Lengauer-Tarjan算法求支配树</div><div>参考题目：洛谷P5180解析：填了N久之前留下的大坑，题解什么时候再写吧代码：#include&lt;bits/stdc++&gt;...</div></div>	<div><div><div>❤</div><div>9</div></div><div><div>来自：zyxoi_dreamer的...</div></div></div>	<div>阅读数 114</div>
<div><div>洛谷 P2597 [ZJOI2012]灾难 灭绝树（支配树）</div><div>题面题意给出一张食物网,问当每种生物分别灭绝后会让其他一共几种生物灭绝(没有食物).做法可以发现生物之间的灭...</div></div>	<div><div><div>🔗</div><div>2</div></div><div><div>来自：zyyyylx的博客</div></div></div>	<div>阅读数 255</div>
<div><div>codeforces 757F Team Rocket Rises Again（最短路+支配树）</div><div>今天刚刚看了支配树，然而怨我太弱并没有看明白Lengauer-Tarjan算法，比较懵QAQ.....不过我倒是学会了在DAG...</div></div>	<div><div><div>📖</div><div>1</div></div><div><div>来自：XHRlyb的博客</div></div></div>	<div>阅读数 296</div>
<div><div>bzoj 3281: 小P的烦恼 支配树算法+dp</div><div>题意小P最近遇上了大麻烦，他的高等代数挂科了。于是他只好找高代老师求情。善良的高代老师答应不挂他，但是...</div></div>	<div><div><div>📖</div><div>1</div></div><div><div>来自：beginend</div></div></div>	<div>阅读数 324</div>
<div><div>康复计划#4 快速构造支配树的Lengauer-Tarjan算法</div><div>本篇胡写给我自己这样的老是证错东西的口胡选手以及那些想学支配树，又不想啃论文原文的人... 大概会讲的...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：狼啸极行的博客</div></div></div>	<div>阅读数 172</div>
<div><div><a href="#">推动全社会公益氛围形成，使公益与空气和阳光一样触手可及。</a> 公益缺你不可，众多公益项目等你PICK——百度公益 让公益像「空气和阳光」一样触手可及！ gongyi.baidu.com</div></div>		
<div><div>支配树学习笔记</div><div>今天晚上沉迷毒瘤常数优化spfa过支配树题，然而一直调不过（ET），隔壁队长都把支配树给学了，于是就看了看支...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：DJ的博客</div></div></div>	<div>阅读数 145</div>
<div><div>支配树(dominator tree)学习笔记</div><div>抖m学抖mtree的笔记2333</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：GEOTCBRL</div></div></div>	<div>阅读数 2713</div>
<div><div>HDU 4694 [支配树]</div><div>题目传送门典型的支配树裸题，就是询问给定点到某点i的全部路径都经过的某点j。具体讲解就不写了，网上很详细...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：lemonoil的博客</div></div></div>	<div>阅读数 398</div>
<div><div>Codeforces 757F: Team Rocket Rises Again（支配树）</div><div>Description 2017新年到！Bash小朋友想给他的好朋友们送礼物。在他的国家里，有 n 座城市和 m 条双向道路（这...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：DZYO的博客</div></div></div>	<div>阅读数 555</div>
<div><div>[HDU]P4694 Important Sisters 支配树</div><div>题目大意：给你n(n&amp;amp;amp;lt;=5*104)n(n&amp;amp;amp;lt;=5*104)n(nm(m&amp;amp;amp;lt;=105)...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：liangzihao1的博客</div></div></div>	<div>阅读数 64</div>
<div><div>千万不要在乱喝驼奶了！我是认真的，再忙也要看！</div><div>世百泽·猎媒</div></div>		
<div><div>HDU4694: Important Sisters（支配树）</div><div>DescriptionThereareNclonesofMisakaMikoto(sisters)formingtheMisakanetwork.Somepairsofsistersareconne...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：DZYO的博客</div></div></div>	<div>阅读数 408</div>
<div><div>bzoj 2815: [ZJOI2012]灾难 支配树</div><div>题意给出一个有向无环图，问把每个点分别删掉后有多少个点不能从入读为0的点到达。n...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：beginend</div></div></div>	<div>阅读数 327</div>
<div><div>hihocoder #1343 : Stable Members(支配树)</div><div>时间限制:10000ms单点时限:1000ms内存限制:256MB描述RecentlyLittleHijoinedanalgorithmlearninggroup.The...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：hexianhao的博客</div></div></div>	<div>阅读数 672</div>
<div><div>hdu4694 Important Sisters 支配树</div><div>有关支配树和Lengauer_Tarjan算法，可以见这个ppt。 注意有不能到达的点，输出0；然后就是裸的跑支配树。A...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：lych的博客</div></div></div>	<div>阅读数 1612</div>
<div><div>HDU-4964 Emmet（模拟）</div><div>一道很简单的字符串模拟题，给的操作也少但是由于第一次使用char数组实现，感觉好多地方写跪了，导致一直WA...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：idealism_xxm的专栏</div></div></div>	<div>阅读数 327</div>
<div><div>陈小龙坦言：这游戏不充钱都能当全服大哥，找到充值入口算我输！</div><div>贪玩游戏·顶新</div></div>	<div><div><div>👑</div><div>1</div></div><div><div>来自：98kaiii的博客</div></div></div>	<div>阅读数 32</div>
<div><div>学习记录-DAG上的支配树</div><div>基本概念作用：在有向图中，当询问从起点到终点的路径必须经过的点，即去掉这个点以及周围的边，就不能从起点...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>来自：98kaiii的博客</div></div></div>	<div>阅读数 32</div>
<div><div>支配数</div><div>支配数：数组中某个元素出现的次数大于数组总数的一半时就成为支配数，其所在位序成为支配点；比如a=[3,3,1,2,...</div></div>	<div><div><div>🔗</div><div>1</div></div><div><div>博文 来自：weixin_42784553...</div></div></div>	<div>阅读数 62</div>



2019/9/3	支配树 与 tarjan算法 - 我数学不好 - CSDN博客		
<b>python 支配数：数组中某个元素出现的次数大于数组总数的一半时就成为支配数，其所在位序成为支配点；</b> 支配数：数组中某个元素出现的次数大于数组总数的一半时就成为支配数，其所在位序成为支配点；比如a=[3,3,1,2,...	 9	阅读数 363	来自： <a href="#">Darkman_EX</a> 的博客
<b>在流程图中求支配点的一种快速算法+[CodeChef FEB14]Graph Challenge解题报告（求半支配点）</b> 0.说明本文译自Tarjan的论文：https://www.cs.princeton.edu/courses/archive/fall03/cs528/handouts/a%20fas...	 2	阅读数 1388	来自： <a href="#">Tham</a> 在思索中前...
<b>2019.01.16【CodeChef-GRAPHCNT】 Counting on a directed graph（支配树）</b> 传送门解析：考虑我们要统计的这个东西，只需要“存在”一条满足条件的路径就行了。换句话说，合法点对的必经...	 2	阅读数 82	来自： <a href="#">zxyoi_dreamer</a> 的...
 <b>人脸识别主要算法原理</b> 人脸识别算法 4675阅读			
<b>ipad pro键盘快捷键</b> 主屏幕前往主屏幕⌘shiftH 搜索⌘space 切换应用⌘tab 文稿快捷键剪切⌘X 复制⌘C...		阅读数 1万+	博文 来自： <a href="#">青墨</a>
<b>在带有双硬盘的Windows10系统上安装Ubuntu16.04系统</b> 最近在看深度学习，需要使用TensorFlow跑程序。虽然在Windows系统上也可以使用GPU进行加速，好不容易安装...		阅读数 1281	博文 来自： <a href="#">sym</a> 的博客
<b>Verilog HDL——移位运算符</b> VerilogHDL学习笔记——语法		阅读数 2万+	博文 来自： <a href="#">proton_boke</a> 的博客
<b>高手经验：一个新手的verilog学习经验</b> 我学verilog语言进行FPGA设计也就半年时间，很多的东西就是在EDACN上学到的，现在想说说自己对使用verilog...		阅读数 3992	博文
<b>如何从ipad pro上通过SSH远程Linux</b> 如何在ipadpro上进行SSH一直在寻找方法已优化上学的体验，而抛弃笨重的游戏本便是重中之重。博主拥有一台20...		阅读数 2839	博文 来自： <a href="#">weixin_44640638</a> ...
<b>这经典传奇！变态+99999！爆率+99999！卸载算我输！</b> 贪玩游戏·顶新			
<b>iPad Pro 2018 壁纸全套</b> 苹果 iPad Pro 2018 新壁纸全套打包下载。		12-20	下载



a710128

TA的个人主页 >

关注

原创10

粉丝3

喜欢7

评论4

等级：

博客 已

访问：1万+

积分：255

排名：38万+

Web前端免费公开课

授课模式：在线直播+课后录  
从零基础到中高级Html5开发  
工程师

最新文章

费用流做二分图最大权匹配

【清单】边角知识清单









【经典】进化树问题  
解方程（有点精度问题）  
网络流 -- 流量平衡？

分类专栏		
C	这个不需要分类	6篇
C	算法	5篇
C	经典问题	1篇

归档		
2015年11月		1篇
2015年10月		1篇
2015年2月		2篇
2014年11月		2篇
2014年9月		1篇
2014年8月		1篇
2014年7月		2篇
<a href="#">展开</a>		

热门文章	
支配树 与 tarjan算法	阅读数 8311
【经典】进化树问题	阅读数 1124
费用流做二分图最大权匹配	阅读数 1073
Javascript 解数独	阅读数 557
网络流 -- 流量平衡？	阅读数 466

最新评论	
支配树 与 tarjan算法	qq_43644454: 对于一个节点Y，存在某个点X能够通过一系列点pi（不包含X和Y）到达点Y且v ...
支配树 与 tarjan算法	YOUSIKI: 并非所有半必经点都一定是X的搜索树祖先，只有最终的那个DFS序最小的半必经点- ...
费用流做二分图最大权匹配	a710128: 好像图片里把符号打反了
【经典】进化树问题	u010274222: 然而RQNoj上，直接把1~2， 2~3.....N-1~N， N~1的和加起来除以2就能过...



腾讯云

学生服务器套餐

10元/月

• 1核2G • 1M宽带  
• 50GB存储

立即购买





9





2



















CSDN学院



CSDN企业招聘

 QQ客服

 kefu@csdn.net

 客服论坛

 400-660-0108

工作时间 8:30-22:00

关于我们 招聘 广告服务 网站地图

 百度提供站内搜索 京ICP备19004658号

©1999-2019 北京创新乐知网络技术有限公司

网络110报警服务 经营性网站备案信息

北京互联网违法和不良信息举报中心

中国互联网举报中心 家长监护 版权申诉

 9



 2















