

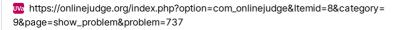
[UVA796] Critical Links

≡ Algorithm	Tarjan
© Created	@Jul 22, 2020 11:36 PM
• Difficulty	省选/NOI-
→ Related to 近期更 新 (Property)	[UVA796]Critical Links
© URL	https://onlinejudge.org/index.php? option=com_onlinejudge&Itemid=8&category=9&page=show_problem&problem=737

题目链接:

Online Judge

Online Judge





Critical Links

problemUrl]: https://uva.onlinejudge.org/index.php?
option=com_onlinejudge&Itemid=8&category=9&page=show_problem&problem=737 [PDF]
(https://uva.onlinejudge.org/external/7/p796.pdf) ![]

nttps://www.luogu.com.cn/problem/UVA796

题解:

[UVA796]Critical Links

这就是经(e)典(xin)的模板题,要注意输入和输出的格式(我就是莫名其妙的 \overline{WA} 了一个小时才发现输出每个testcase之间要有换行(貌似样例给的输出错了)。

其中,加了一些小技巧,如位运算的亦或"成对变换"性质,pair的自动排序等。

有两个注意事项:

- 1. 0不是结束的标志,只是说给你的图中有0个节点。
- 2. 每输出完一个testcase就要换行
- 3. 要排序,每个二元组内部也要有序

下面是AC代码:

```
//
// Created by admin on 2020/7/22.
#include <bits/stdc++.h>
using namespace std;
const int maxn = 1e5+10;
const int max_edge = 1e6+5;
struct Edge
   int to, next;
   bool cut;
}edge[max_edge];
int tot, head[maxn];
int id, dfn[maxn], low[maxn];
int num;
void adde(int u, int v)
   edge[tot].to = v;
   edge[tot].next = head[u];
   edge[tot].cut = false;
   head[u] = tot++;
}
void tarjan(int u, int f)
   dfn[u] = low[u] = ++id;
   for(int i = head[u]; i; i = edge[i].next)
       int v = edge[i].to;
       if(v == f) continue;
       if(!dfn[v])
           tarjan(v, u);
           low[u] = min(low[u], low[v]);
           //桥>,割点>=,可证明,略
           if(low[v]>dfn[u])
               //由于链式前向星是连着存的,所以就用 ^ 得到所用有向边代替的另一个边 (0^1=1, 1^1=0)
               edge[i].cut = edge[i^1].cut = true;
               num++;
           }
       else low[u] = min(low[u], dfn[v]);
   }
int main()
   while(cin>>n)
       id = num = tot = 0;
       memset(head, 0, sizeof(head));
       memset(dfn, 0, sizeof(dfn));
       memset(low, 0, sizeof(low));
       int u, m, v;
       for(int i = 1; i<=n; i++)
```

[UVA796]Critical Links 2

```
scanf("%d (%d)",&u,&m);
              for(int j = 1; j<=m; j++)
                  scanf("%d",&v);
                  adde(u, v);
adde(v, u);
         }
         for(int i = 0; i<n; i++)
             if(!dfn[i])
         tarjan(i, i);
vector<pair<int, int> >a;
         for(int u = 0; u<n; u++)
              for(int i = head[u]; i; i = edge[i].next)
                  if(edge[i].cut && u < edge[i].to)</pre>
                      a.push_back(make_pair(u, edge[i].to));
         sort(a.begin(), a.end());
         printf("%d critical links\n", num);
         for(int i = 0; i<a.size(); i++)
    printf("%d - %d\n", a[i].first, a[i].second);</pre>
        printf("\n");
}
```

[UVA796]Critical Links 3