图论中的连通性

复旦大学 李争彦

2021/10/1

最小差值生成树(POJ3522)

- MST（最小生成树）是图论中的一类非常重要的问题
- 给出一个带权图G(V,E)，最小生成树T是图G的一棵子树
- 输出图的最小生成树中最长边与最短边的差

最小差值生成树

- 题意：求最小生成树中最长边和最短边的差值
- $2 \leq N \leq 200, M \leq 5000$

- 使用 Kruskal 或 Prim 算法均可

- 正确性：图的多个最小生成树，最长边和最短边的差值唯一
- 结论：一个图的两棵最小生成树，边权序列排序后结果相同

证明（算法导论23.1-8）

23.1-8 设 T 为图 G 的一棵最小生成树，设 L 为树 T 中一个边权重的有序列表。证明：对于图 G 的任何其他最小生成树 T'，列表 L 也是 T' 中一个边权重的有序列表。

- 假设L'为另一最小生成树T'的有序列表。假设$L' \neq L$，必然T中有边(u,v)，且比其在T'对应边的权值小。
  - 将边(u,v)加入T'，则形成一个环，若环上的值有大于边(u,v)权值，则可用边(u,v)替代L'中的对应边，矛盾。故该环上的值不大于边(u,v)权值
  - 假设该环上的边权值均小于边(u,v)，则可将(u,v)从L中删除，并使用更小权值的边连接两部分，矛盾。故该环上的权值必然有部分和边(u,v)权值相同
  - 使用该边换出L中的边(u,v)，由于交换边权值相同，则L和L'不变。然而T和T'中权值相同的边数增加1。继续该操作，则将导致两棵最小生成树的对应权值均相同。

最小差值生成树(POJ3522)

- 进阶：求一生成树，使得最长边和最短边的差值最小
- $2 \le N \le 200, M \le 5000$

- 思路：按边权排序，枚举最短边权值，每轮进行 Kruskal
- 正确性可同前推导

- 时间复杂度：$O(M^2)$

传话(Codevs1506)

- 兴趣小组的同学来自各个学校，为了增加友谊，晚会上又进行了一个传话游戏，如果 a 认识 b，那么 a 收到某个消息，就会把这个消息传给 b，以及所有 a 认识的人。
- 如果 a 认识 b，b 不一定认识 a。
- 所有人从 1 到 n 编号，给出所有“认识”关系，问如果 i 发布一条新消息，那么会不会经过若干次传话后，这个消息传回给了 i，$1<=i<=n$。

传话(Codevs1506)

- 题意：给定有向图，判断每个点是否存在回路
- $N \le 1000, M \le 10000$

- Floyd 求**传递闭包**，时间复杂度$O(N^3)$
- 以每个点作为起点DFS，BFS遍历，时间复杂度$O(N^2)$

- 假设：数据范围$N \le 10^6, M \le 10^6$，有方法解决吗？

传话

- 原题等价于：给定有向图，判断每个点是否在环中
- $N \leq 10^6, M \leq 10^6$

- 有向图判环 X
  - 判断是否有环存在复杂度为$O(N)$
  - 输出所有环的复杂度为指数级别（例：完全图）

- 有向图的强连通分量 √
  - 对于该子图上每一个点对(V_a, V_b)，存在路径$V_a \rightarrow V_b$以及$V_b \rightarrow V_a$
  - 判断点所在强连通分量大小是否大于1，时间复杂度$O(N + M)$

联合权值([NOIP2014 提高组])

- 题意：给定**无根树**，求相隔的两节点的权值之积的和与最大值
- $N \leq 200{,}000$

- 直接暴力复杂度为$O(N^2)$（例：菊花图）
- 树上DFS / 树形DP
  - 考虑在DFS时权值会产生的位置：
    - 不同的孩子之间
    - 孙子节点和当前节点之间
  - 分别维护即可

```cpp
pii dfs(int u, int f) {
    int psum = 0, pmx1 = 0, pmx2 = 0;
    for(int i = head[u]; i != -1; i = nx[i]) {
        int v = to[i]; if(v == f) continue;
        pii pv = dfs(v, u);
        sum = (sum + 2LL * w[u] * pv.first % MOD) % MOD;
        mx = max(mx, w[u] * pv.second);
        sum = (sum + 2LL * w[v] * psum % MOD) % MOD;
        psum = (psum + w[v]) % MOD;
        if(w[v] > pmx1) {pmx2 = pmx1; pmx1 = w[v];}
        else if(w[v] > pmx2) {pmx2 = w[v];}
    }
    mx = max(mx, pmx1 * pmx2);
    return pii(psum, pmx1);
}
```

有向图(POJ 2186)

- 有一些奶牛,他们之间有互相崇拜的关系,并且这个关系是有传递性的,求被所有奶牛崇拜的奶牛数量
- $N \leq 10^5, M \leq 5 \times 10^5$

有向图(POJ 2186)

- 题意：求从任意一点出发均可达的点数量
- $N \le 10^5, M \le 5 \times 10^5$

- 有向图的强连通分量 + 拓扑排序

- 缩点后重新建图，判定出度为0的只有一个分量
- 若有多个分量出度为0，表示不存在任意可达的点

```
void Tarjan(int u) {
    low[u] = dfn[u] = ++idx;
    stk[++tp] = u;
    instack[u] = true;
    for(int i = head[u]; i != -1; i = nx[i]) {
        int v = to[i];
        if(!dfn[v]) {Tarjan(v); low[u] = min(low[u], low[v]);}
        else if(instack[v] && low[u] > dfn[v]) low[u] = dfn[v];
    }
    if(low[u] == dfn[u]) {
        scc++;
        int v;
        do {
            v = stk[tp--];  instack[v] = false;
            num[scc]++;
            belong[v] = scc;
        } while(v != u);
    }
}
```

经过的边

- 一个含有n个顶点m条边的图，如果要遍历这张图上的所有点，有多少条边是一定会经过的？
- $N, M \le 200{,}000$

经过的边

- 题意：求遍历有向图时，必然会经过的边数量
- $N, M \leq 200{,}000$

- 思路：Tarjan 求桥（割边）数量，注意判**重边**
- 桥：对于连通图$G = (V, E)$，若$E' \subseteq E$且$G' = (V, E \setminus E')$不是连通图，则E'是图G的一个**边割集**。大小为一的点割集称作桥 (Bridge)或割边。

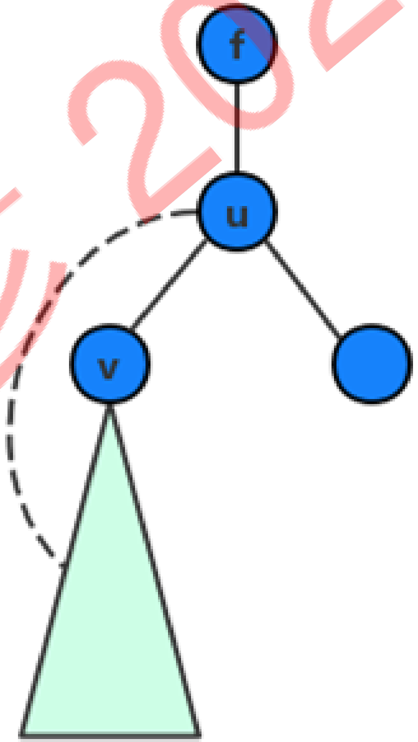
经过的边

- 题意：求遍历有向图时，必然会经过的边数量
- $N, M \leq 200{,}000$

- 思路：Tarjan 求桥（割边）数量，注意判**重边**
- 桥：对于连通图$G = (V, E)$，若$E' \subseteq E$且$G' = (V, E \setminus E')$不是连通图，则E'是图G的一个**边割集**。大小为一的点割集称作桥 (Bridge)或割边。

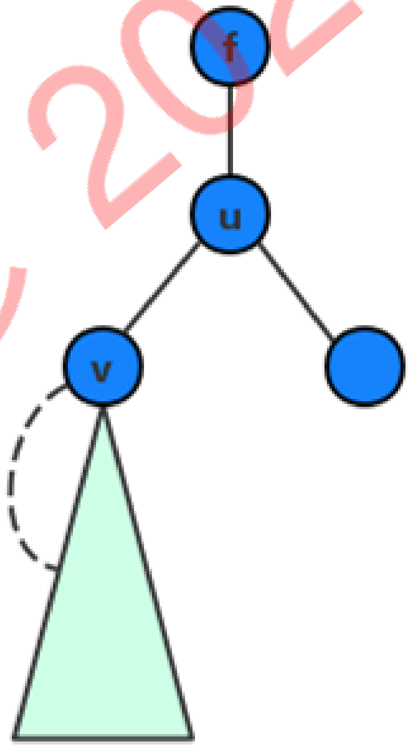
经过的边

- 对于一条边(u, v)，如果v及其后代能够通过反向边访问u及u之前时间的点，那么如果删除边(u, v)，则以v为根的子树仍能和u连通



经过的边

- 相反的，若$\text{low}[v] > \text{dfn}[u]$，说明删除边$(u, v)$后，整个图不连通。



```
void Tarjan(int u, int fa) {
    low[u] = dfn[u] = ++idx;
    for (int i = head[u]; i != -1; i = nx[i]) {
        int v = to[i]; if(v == fa) continue;
        if(!dfn[v]) {
            Tarjan(v, u);
            if(low[v] > dfn[u]) {
                if(mp[pii(min(u, v), max(u, v))] == 1) bridge++;
            }
            low[u] = min(low[u], low[v]);
        } else if(low[u] > dfn[v]) low[u] = dfn[v];
    }
}
```

网络（POJ1144）

- 一个电话信号传输公司正在搭建一个新的电话电缆网络。他们想要连接从 1 到 N 标号的各个地点，标号不重复。电缆是双向作用的，在每个地方电缆都链接一个信号站。每个地点都有一个信号站。信号站与信号站之间可能通过若干个信号站实现连通。有的时候一个地点断电了那么部分信号站便都收不到信号。信号传输公司意识到这是因为一个地方信号站的失效而导致的其他地方信号站的失效，这样的信号站被称为中转站。现在信号传输公司想找出有多少个这样的中转站，请帮助他们。
- $N, M \leq 200{,}000$

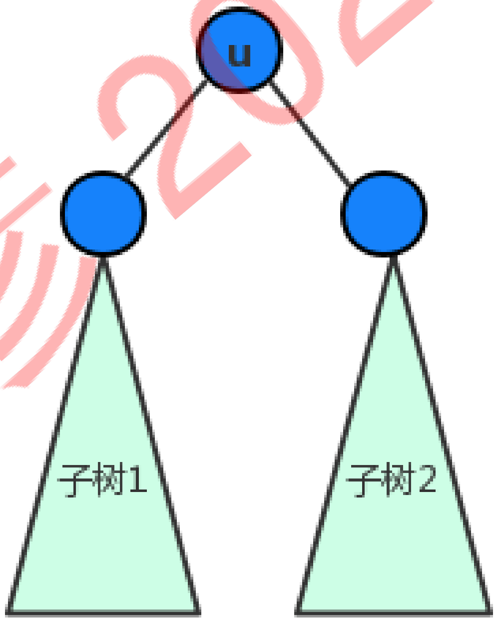
网络

- 题意：输出无向图中的点，使得删除后图不再连通
- $N, M \leq 200{,}000$

- 思路：Tarjan 求割点数量，注意细节（数据较弱）
- 割点：对于连通图$G = (V, E)$，若$V' \subseteq V$且$G[V\backslash V']$不是连通图，则V'是图G的一个**点割集**。大小为一的点割集称作割点(Cut)。

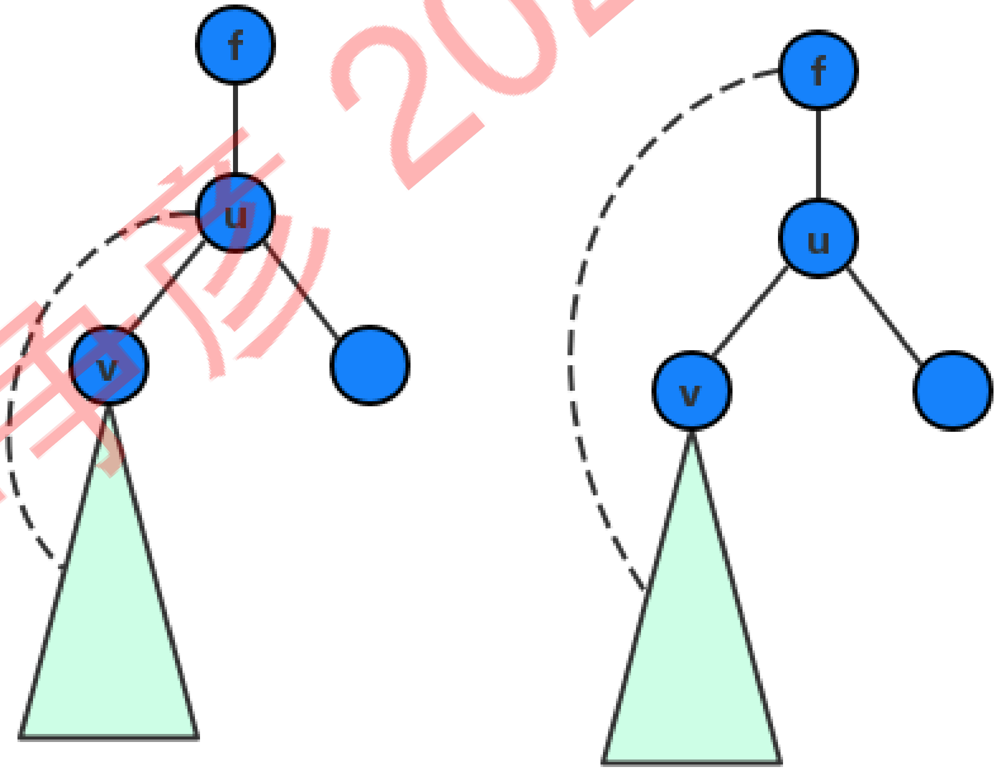
网络

- 当根结点u有两个或更多的子结点时才是割点。去掉u后会形成两个连通块。



网络

- 非根结点u是个割点当且近当u存在一个子结点v，使得v及其后代都没有反向边连回u的祖先
- 对于树边(u,v)当有一个v满足$\mathrm{low}[v] \geq \mathrm{dfn}[u]$时，$u$是割点



```
int low[N], dfn[N], idx, cut[N];
int Tarjan(int u, int fa) {
    low[u] = dfn[u] = ++idx;
    int son = 0;
    for(int i = head[u]; i != -1; i = nx[i]) {
        int v = to[i]; if(v == fa) continue;
        if(!dfn[v]) {
            Tarjan(v, u); son++;
            low[u] = min(low[u], low[v]);
            if(u != fa && low[v] >= dfn[u]) cut[u] = true;
        } else if(low[u] > dfn[v]) low[u] = dfn[v];
    }
    if(u == fa && son > 1) cut[u] = true;
}
```

网络

- 当原图$n = 2$时，不存在符合题意的的点

- 考虑：原图不连通的情况
  - 当原图有三个及更多的连通分量时：原图全部的点符合要求
  - 当原图有两个连通分量时：
    - 如果有一个连通分量大小为1，则另一连通分量的所有点均可行
    - 否则原图全部的点符合要求
  - 当原图连通时：Tarjan求所有的割点

```
for(int i = 1; i <= n; i++) {
    if(!vis[i]) {
      vector<int> v;
      dfs(i, v);
      vv.push_back(v);
    }
  }
  if(vv.size() > 2) {
    printf("%d\n", n);
    for(int i = 1; i <= n; i++) printf("%d%c", i, " \n"[i == n]);
  } else if(vv.size() == 2) {
    bool f = vv[0].size() == 1 || vv[1].size() == 1;
    if(f) {
      int p = vv[0].size() == 1;
      printf("%d\n", n - 1);
      sort(vv[p].begin(), vv[p].end());
      for(int i = 0; i < vv[p].size(); i++) printf("%d%c", vv[p][i], " \n"[i + 1 == vv[p].size()]);
    } else {
      printf("%d\n", n);
      for(int i = 1; i <= n; i++) printf("%d%c", i, " \n"[i == n]);
    }
  } else {
    Tarjan(1, 1);
    int cnt = 0; for(int i = 1; i <= n; i++) if(cut[i]) cnt++;
    printf("%d\n", cnt);
    for(int i = 1; i <= n; i++) if(cut[i]) printf("%d ", i);
  }
```

例题选讲

Mining Your Own Business

- 题意：求在一无重边无向图中，选择尽量少的点，使得删除图中任何一个点后，图的每个连通分量至少包含一个被选的点，输出任一方案
- $N \leq 5 \times 10^4$

- HDU3844 / UvaLive5135

Mining Your Own Business

- 若原图为点双连通图，则任意选择两点

- 考虑无向图按点双缩点，形成一棵树
- 故若原图不为点双，则必然有两个双连通分量只有一个割点
- 对于有多个割点的点双，如选择其中一个割点，则该点双能够通过另一割点与其他点连通
- 最优方案为在只有一个割点的点双中，选择非割点