海亮高级中学
HAILIANG SENIOR HIGH SCHOOL

# Kruskal重构树

任飞宇

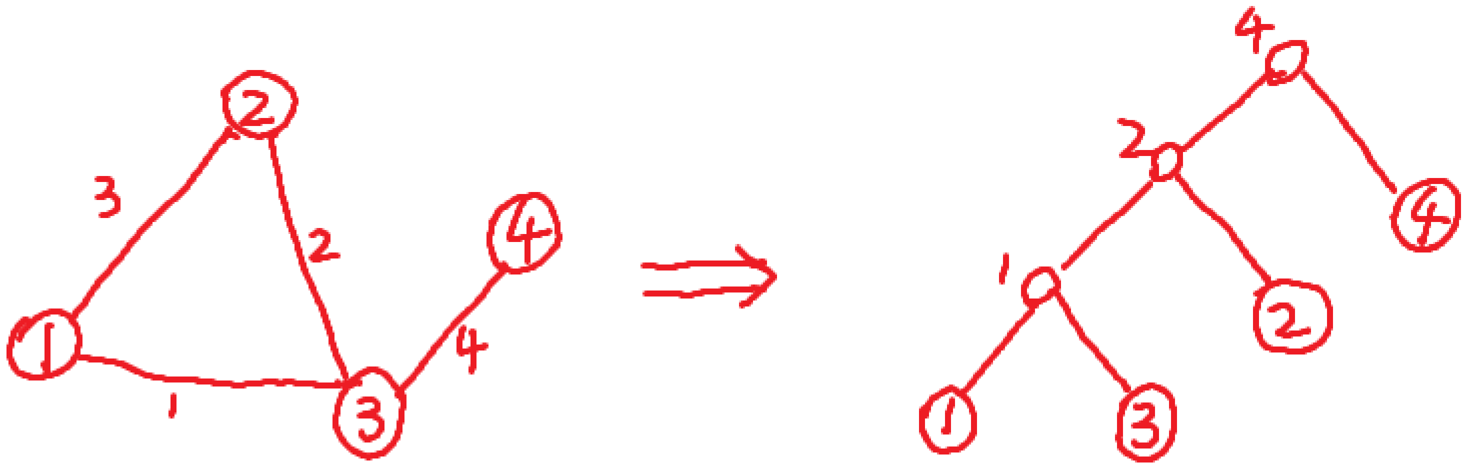
海亮教育
HAILIANG EDUCATION

在跑 Kruskal 的过程中我们会从小到大加入若干条边。现在我们仍然按照这个顺序。

首先新建 $n$ 个集合，每个集合恰有一个节点，点权为 $0$。

每一次加边会合并两个集合，我们可以新建一个点，点权为加入边的边权，同时将两个集合的根节点分别设为新建点的左儿子和右儿子。然后我们将两个集合和新建点合并成一个集合。将新建点设为根。

不难发现，在进行 $n-1$ 轮之后我们得到了一棵恰有 $n$ 个叶子的二叉树，同时每个非叶子节点恰好有两个儿子。这棵树就叫 Kruskal 重构树。



# 性质

1. 是二叉树；非叶节点的权值从下往上是单调的；

2. 原图中两个点之间的所有简单路径上最大边权的最小值 = 最小生成树上两个点之间的简单路径上的最大值 = Kruskal 重构树上两点之间的 LCA 的权值。

3. 到点 $x$ 的简单路径上最大边权的最小值 $\le val$ 的所有点 $y$ 均在 Kruskal 重构树上的某一棵子树内，且恰好为该子树的所有叶子节点。

利用重构树可以把一个图或树上的连通块表示为重构树上的子树或者dfs序上的区间。

# 1. 水灾

牛牛所在的国家闹水灾了！两个城市之间可能有双向道路，每条道路都有个海拔，牛牛想要知道每次对于某些城市，当水的海拔最低达到多高的时候这些城市两两互不能到达。（如果某条道路的海拔小于等于水的海拔那么这条道路就不能通行）牛牛要问你 q 次，请你帮他解答。

一句话题意：给一个 n 个节点 m 条带权边的无向连通图，有 q 次询问，每次询问图中 $k_i$ 个互不相同的点，你可以选择一个数 x，然后将图中所有边权小于等于 x 的边删除。求当删除这些边后 $k_i$ 个点互不连通时，x 的最小值。

**强制在线**

$1 \le n, m, q \le 5 \times 10^5$

$1 \le \sum_{i=1}^{q} k_i \le 10^6, 1 \le k_i \le n$

$1 \le u, v \le n, 1 \le w_i \le 10^9$

# 1. 水灾

考虑使海拔下降（不断地出现一些道路）直到有两个关键点连通。

于是从大到小加边，建重构树。

两个关键点要连通，就要不断加边，一直加到它们的LCA。

问题转为从这些关键点的LCA中找一个权值最大的。

把关键点按dfs序排序，我们只需要找相邻关键点的LCA。

# 2. Qpwoeirut and Vertices

有一个$n$个点$m$条边的图和$q$个询问，第$i$个询问形如：
图中只保留编号为1~$k$的边，使得编号为$[l_i,r_i]$区间内的所有点连通，这个$k$最小是多少？

$2 \le n \le 10^5, 1 \le m, q \le 2 \cdot 10^5$

# 2. Qpwoeirut and Vertices

有一个$n$个点$m$条边的图和$q$个询问，第$i$个询问形如：
图中只保留编号为1~$k$的边，使得编号为$[l_i, r_i]$区间内的所有点连通，这个$k$最小是多少？
$2 \le n \le 10^5, 1 \le m, q \le 2 \cdot 10^5$

按编号从小到大加边建重构树，每次询问等价于求编号在$[l_i, r_i]$内所有点的LCA。
这些点在DFS序上不一定是连续的，如何快速求所有点的LCA呢？
多个点的LCA，就是DFS序最小的$x$，和DFS序最大的$y$，这两点的LCA。
证明：这个LCA在DFS序上的区间包含$x$和$y$，那么也一定包含DFS序上位于$x$和$y$之间的点。

# 3. Graph and Queries

给定一个 $n$ 个点 $m$ 条边的无向图，第 $i$ 个点的点权初始值为 $p_i$,所有 $p_i$ 互不相同。

接下来进行 $q$ 次操作，分为两类：

- 1 v 查询与 $v$ 连通的点中，$p_u$ 最大的点 $u$ 并输出 $p_u$，然后让 $p_u = 0$。
- 2 i 将第 $i$ 条边删掉。

$1 \le n \le 2 \cdot 10^5; 1 \le m \le 3 \cdot 10^5; 1 \le q \le 5 \cdot 10^5$

# 3. Graph and Queries

给定一个 $n$ 个点 $m$ 条边的无向图，第 $i$ 个点的点权初始值为 $p_i$,所有 $p_i$ 互不相同。

接下来进行 $q$ 次操作，分为两类：

- 1 v 查询与 $v$ 连通的点中， $p_u$ 最大的点 $u$ 并输出 $p_u$，然后让 $p_u = 0$。
- 2 i 将第 $i$ 条边删掉。

$1 \le n \le 2 \cdot 10^5; 1 \le m \le 3 \cdot 10^5; 1 \le q \le 5 \cdot 10^5$

删边不好做，考虑倒着加边。但是每个询问会受之前询问的影响。

分析一下，一个询问，等价于保留后面的所有边，点v所在连通块的最大点权。

于是我们倒着加边建重构树，然后正向回答所有询问，每个询问等价于找区间内的最大点权并置为0，可以用线段树维护。

# 4. 归程



本题的故事发生在魔力之都，在这里我们将为你介绍一些必要的设定。 魔力之都可以抽象成一个 $n$ 个节点、$m$ 条边的无向连通图（节点的编号从 $1$ 至 $n$）。我们依次用 $l, a$ 描述一条边的**长度、海拔**。

作为季风气候的代表城市，魔力之都时常有雨水相伴，因此道路积水总是不可避免的。由于整个城市的排水系统连通，因此**有积水的边一定是海拔相对最低的一些边**。我们用**水位线**来描述降雨的程度，它的意义是：所有海拔**不超过**水位线的边都是**有积水**的。

Yazid 是一名来自魔力之都的 OIer，刚参加完 ION2018 的他将踏上归程，回到他温暖的家。Yazid 的家恰好在魔力之都的 $1$ 号节点。对于接下来 $Q$ 天，每一天 Yazid 都会告诉你他的出发点 $v$，以及当天的水位线 $p$。

每一天，Yazid 在出发点都拥有一辆车。这辆车由于一些故障不能经过有积水的边。Yazid 可以在任意节点下车，这样接下来他就可以步行经过有积水的边。但车会被留在他下车的节点并不会再被使用。 需要特殊说明的是，第二天车会被重置，这意味着：

- 车会在新的出发点被准备好。
- Yazid 不能利用之前在某处停放的车。

Yazid 非常讨厌在雨天步行，因此他希望在完成回家这一目标的同时，最小化他**步行经过的边**的总长度。请你帮助 Yazid 进行计算。

本题的部分测试点将强制在线，具体细节请见【输入格式】和【子任务】。

# 4. 归程

每个询问中，海拔高于水位线的边形成一些连通块，我们要在$v_i$所在连通块中找一个点，到点1的距离最小。

预处理每个点到点1的最短距离，从大到小加边建重构树，从$v_i$向上倍增找到连通块对应的子树，答案就是子树里所有点的最短距离的最小值。

# 5. Life is a Game



n 个点，m 条边的无向连通图。

每个点上有个能力值，经过点时可以获得该能力值 ai，但每个点只能获得一次。

每条边上有个能力值限制 wi，只有当能力值超过 wi 时才能通过这条边。

有 Q 次询问，每次询问给出初始位置 x 和初始能力值 k，询问能获得的最大能力值是多少。

$n, m, q <= 1e5, ai <= 1e4, wi <= 1e9$

# 5. Life is a Game



n 个点，m 条边的无向连通图。
每个点上有个能力值，经过点时可以获得该能力值 ai，但每个点只能获得一次。
每条边上有个能力值限制 wi，只有当能力值超过 wi 时才能通过这条边。
有 Q 次询问，每次询问给出初始位置 x 和初始能力值 k，询问能获得的最大能力值是多少。
$n, m, q <= 1e5, ai <= 1e4, wi <= 1e9$

位于某个点时，可以到达的点是边权不超过当前能力值的所有边形成的一个连通块（假设能力值不变）。想到按边权从小到大加边建重构树。

在重构树上，我们从初始点出发，如果当前能力值比父节点大，就可以走到父节点子树内的所有点，收集能力值。这样不断地向上，扩大子树的范围直到不能继续，就得到答案了。

问题是一步步向上爬太慢了。

设重构树上向上走，当前走到点x，那么此时能力值就是初始值k加上x子树内能力值之和$s[x]$，可以走上父亲y等价于$k + s[x] > w[y]$，移项即$k > w[y] - s[x]$。

最终停下的位置就是第一个不满足条件的位置，可以用倍增快速找。

# 6. 狼人

在日本的茨城县内共有 $N$ 个城市和 $M$ 条道路。这些城市是根据人口数量的升序排列的，依次编号为 0 到 $N-1$。每条道路连接两个不同的城市，并且可以双向通行。由这些道路，你能从任意一个城市到另外任意一个城市。

你计划了 $Q$ 个行程，这些行程分别编号为 0 至 $Q-1$。第 $i(0 \leq i \leq Q-1)$ 个行程是从城市 $S_i$ 到城市 $E_i$。

你是一个狼人。你有两种形态：**人形**和**狼形**。在每个行程开始的时候，你是人形。在每个行程结束的时候，你必须是狼形。在行程中，你必须要变身（从人形变成狼形）恰好一次，而且只能在某个城市内（包括可能是在 $S_i$ 或 $E_i$ 内）变身。

狼人的生活并不容易。当你是人形时，你必须避开人少的城市，而当你是狼形时，你必须避开人多的城市。对于每一次行程 $i(0 \leq i \leq Q-1)$，都有两个阈值 $L_i$ 和 $R_i(0 \leq L_i \leq R_i \leq N-1)$，用以表示哪些城市必须要避开。准确地说，当你是人形时，你必须避开城市 $0, 1, \ldots, L_i - 1$；而当你是狼形时，则必须避开城市 $R_i + 1, R_i + 2, \ldots, N-1$。这就是说，在行程 $i$ 中，你必须在城市 $L_i, L_i + 1, \ldots, R_i$ 中的其中一个城市内变身。

你的任务是，对每一次行程，判定是否有可能在满足上述限制的前提下，由城市 $S_i$ 走到城市 $E_i$。你的路线可以有任意长度。

- $2 \leq N \leq 200,000$
- $N-1 \leq M \leq 400,000$
- $1 \leq Q \leq 200,000$

# 6. 狼人

题意：给定一个图，多次询问是否存在一条$S_i$到$E_i$的路径，满足路径前面几个点编号$\geq L_i$，后面几个点编号$\leq R_i$，中间的分界点必须同时满足两个条件。$S_i$或$E_i$也可以是分界点。

从中间的分界点入手，基于连通块的思想，
只保留编号$\geq L_i$的点，跟S在同一个连通块的点才能作为分界点；
只保留编号$\leq R_i$的点，跟T在同一个连通块的点才能作为分界点；

如果两个连通块有交，答案就是YES。

考虑建两个重构树。这里不是边权而是点的编号，如何处理？

分析一条边什么时候加入，只有连的两个点都存在，边才算加入了。所以边权是两点编号的min(max)

S的连通块在第一颗重构树里对应一个区间，T的连通块在第二颗重构树里对应一个区间，
问两个区间是否有交？（注意两个DFS序是两个不同的排列）

第一个排列中每个数记一个在第二个排列中的下标，询问相当于第一个排列的某个区间里是否有下标在某个范围内。

一个二维矩阵数点问题。扫描线

# 7. Groceries in Meteor Town

给定 N 个点的树，起初每个节点都是黑色。

三种操作：

1. 把下标为 [l,r] 的点染成白色；
2. 把下标为 [l,r] 的点染成黑色；
3. 询问从节点 x 出发到达任意一个白色节点的简单路径上经过的边，最大可能的权值。不存在则输出-1.

$2 \leq n, q \leq 3 \cdot 10^5$

# 7. Groceries in Meteor Town

给定 N 个点的树，起初每个节点都是黑色。

三种操作：

1. 把下标为 [l,r] 的点染成白色；
2. 把下标为 [l,r] 的点染成黑色；
3. 询问从节点 x 出发到达任意一个白色节点的简单路径上经过的边，最大可能的权值。不存在则输出-1.

$2 \leq n, q \leq 3 \cdot 10^5$

假设没有修改。询问的东西等价于从小到大加边，最后一次x跟白点合并，此次合并的边权。

考虑从小到大建重构树，要求的就是LCA(x,w1)，LCA(x,w2)，LCA(x,w3)…中的最大值。

这个点实际上就是LCA(x,w1,w2,w3…)。

还是找DFS序最大和最小的两个点。

用线段树维护所有白色位置的最值。

# 8. 超级加倍

给定一棵树。

我们认为一条从 $x \to y$ 的简单路径是好的，当且仅当路径上的点中编号最小的是 $x$，最大的是 $y$。

请求出好的简单路径条数。

# 8. 超级加倍

给定一棵树。

我们认为一条从 $x \to y$ 的简单路径是好的，当且仅当路径上的点中编号最小的是 $x$，最大的是 $y$。

请求出好的简单路径条数。

考虑点分治并不好做（一条路径不能分成重心出发的两条路径）。
条件可以转化为x-y路径编号>=x且<=y。考虑连通块的思想

保留编号>=x的点，x在一个连通块C1里；保留编号<=y的点，y在一个连通块C2里。

路径x-y合法，当且仅当y在C1里且x在C2里。

建两颗重构树，C1和C2可以表示为区间。

问题转化为：有两个排列，每个点$i$在排列1里对应区间$[L_i, R_i]$，在排列2里对应区间$[L_i', R_i']$，
问有多少$i$，$j$满足 $L_i \le j \le R_i$且$L_j' \le i \le R_j'$

考虑扫描线枚举i，只保留满足第二个不等式的j，于是问题变成单点加减，区间求和。