

次小生成树



主讲人：邓哲也



次小生成树

上一讲中，我们讲到了可以枚举删除最小生成树上的每一条边，再求一次最小生成树，比较它们是否一样。

但这样实在是太耗时了。

次小生成树

考虑我们先求出了一个最小生成树 T 。

对于没有选进 T 的边 (u, v) ，这条边与 T 中 u 到 v 中的路径形成了一个环。

我们只要用 (u, v) 替换掉 T 中 u 到 v 的路径中边权最小的那条边，就可以得到一个与 T 不同的，边权和尽可能小的生成树。

只要枚举完所有的 (u, v) ，再取最小值，就是次小生成树。

次小生成树

现在问题变成了，给定一颗树 T 。

每次查询 从 u 到 v 的路径上边权的最小值。

用倍增即可，时间复杂度 $O(n \log n)$

算上求 T 的时间复杂度， $O(m \log n)$

总的时间复杂度为 $O((n + m) \log n)$

试一试! POJ 1679

给一个顶点数 ≤ 100 , 问它的最小生成树是否唯一。

样例:

3 3

1 2 1

2 3 2

3 1 3

Unique

4 4

1 2 2

2 3 2

3 4 2

4 1 2

Not Unique

下节课再见