知识精炼

主讲人：邓哲也

曼哈顿距离最大生成树

在二维平面上，有 n 个点，第 i 个点的坐标为 $(x[i], y[i])$.

每两个点之间都有一条边，边权定义为两个点之间的曼哈顿距离。

即对于第 i 个点到第 j 个点的边，

边权为 $|x[i] - x[j]| + |y[i] - y[j]|$

现在请你求出这张图的最大生成树的权值。

$n \le 100000$, $|x[i]|, |y[i]| \le 10^9$

曼哈顿距离最大生成树

如果想要建出完整的图——显然是不可能的。

边的数量是 $O(n^2)$ 数量级的。

传统的 Prim 和 Kruskal 算法不大好用。

曼哈顿距离最大生成树

想想 Boruvka 算法。

利用曼哈顿距离的性质。

曼哈顿距离最大生成树

给定两个点集合 A 和 B，怎么查询 A 中的点 到 B 中的点的最大曼哈顿距离？

曼哈顿距离最大生成树

给定两个点集合 A 和 B，怎么查询 A 中的点 到 B 中的点的最大曼哈顿距离？

暴力 O(n^2) 枚举不可取。

曼哈顿距离最大生成树

给定两个点集合 A 和 B，怎么查询 A 中的点 到 B 中的点的最大曼哈顿距离？

暴力 $O(n^2)$ 枚举不可取。

但是可以考虑一下枚举一个 A 集合里的点 p，然后在小于 $O(n)$ 的时间里在 B 中找到离 p 曼哈顿距离最远的点。

曼哈顿距离最大生成树

这时就要利用到曼哈顿距离的性质了。

$|x1 - x2| + |y1 - y2|$

曼哈顿距离最大生成树

这时就要利用到曼哈顿距离的性质了。

$|x1 - x2| + |y1 - y2|$

绝对值符号是不是有点碍眼？

分四种情况 拆掉绝对值！

曼哈顿距离最大生成树

这时就要利用到曼哈顿距离的性质了。

$|x1 - x2| + |y1 - y2|$

绝对值符号是不是有点碍眼？

分四种情况 拆掉绝对值！

$$x1 - x2 + y1 - y2 = (x1 + y1) - (x2 + y2)$$

$$x1 - x2 + y2 - y1 = (x1 - y1) - (x2 - y2)$$

$$x2 - x1 + y1 - y2 = (y1 - x1) - (y2 - x2)$$

$$x2 - x1 + y2 - y1 = (- x1 - y1) - (- x2 - y2)$$

曼哈顿距离就是上面这四个值中的最大值

曼哈顿距离最大生成树

$$x1 - x2 + y1 - y2 = (x1 + y1) - (x2 + y2)$$
$$x1 - x2 + y2 - y1 = (x1 - y1) - (x2 - y2)$$
$$x2 - x1 + y1 - y2 = (y1 - x1) - (y2 - x2)$$
$$x2 - x1 + y2 - y1 = (- x1 - y1) - (- x2 - y2)$$

对于两个集合 A，B，我们都用 4 个 set 分别来存 (x+y),(x-y),(-x+y),(-x-y)

这样枚举 A 中的一个点 p 时，先考虑第一个式子，在 B 集合的存(x+y)的 set 中找到最小值，用 (x[p] + y[p] - 最小值) 去更新最大值。

同理，考虑第二个式子，在 B 集合的存(x-y)的 set 中找到最小值，用 (x[p] - y[p] - 最小值) 去更新最大值。

以此类推，最后得到的最大值就是 p 到 B 集合中的最远距离。

曼哈顿距离最大生成树

用 set 来支持查询最小值的操作，使得每一步的复杂度都是 O(log n) 的。

这样我们刚刚提出的问题就在 O(n log n) 的时间复杂度下解决了。

曼哈顿距离最大生成树

回到这道题目本身。

Boruvka 算法在计算连通块到别的连通块的最远距离时，也就是我们刚刚提出的子问题。

假设当前连通块是 C，我们要找出 C 到 V - C 的最大距离。（其中 V 是所有点组成的集合）

我们可以对每个连通块建立 4 个 set 存坐标信息，并且对整张图建 4 个 set。查询的时候，先从全局的 set 里删去 C 中的信息，查询完成后，再恢复全局的 set。

曼哈顿距离最大生成树

合并的时候，我们也要合并两个连通块的 set。

具体的做法就是启发式合并，把小的连通块的 set 依次插入到大连通块的 set 中去。

这样可以保证合并的复杂度在 $O(n \log n)$.

曼哈顿距离最大生成树

综上所述，每一轮中更新最大距离需要 O(n log n) 的时间。

合并连通块的 set 需要 O(n log n) 的时间。

总共要循环 O(log n) 次。

故总的时间复杂度为 O(n log^2 n)，完美解决此题。

下节课再见