

Around the world 解题报告

长沙市一中 张天扬

1 试题来源

POI2014

<http://www.lydsy.com/JudgeOnline/problem.php?id=3827>

2 试题大意

一个环上有 n 个点，第 i 个点和第 $i+1$ 个点间的距离为 L_i 。

现在有 Q 次询问，每次限定一个 k ，每次前进的距离不能超过 k 。我们可以任意选择一个点出发，要求顺时针绕环一圈后回到起始点，中间只能停留在给定的 n 个点上。求最少需要停留多少次能够绕环一圈。

$n \leq 10^6, Q \leq 100$ 。

$\sum L_i \leq 10^9, k \leq 10^9$ 。

空间限制：24MB

3 算法介绍

首先，因为空间限制只有24MB，因此我们至多只能开6个大小为 n 的数组。

3.1 一个很慢的算法

我们考虑，如果已经知道起点，怎么计算答案？显然，我们每次一定会走到能走的最远的一个点上去。那么我们可以先计算出每个点下一步会走到哪个点上。

那么我们发现：有些时候，从相邻的两个点出发，第一步能到的点是相同的，那么后面的很多步都会是相同的，只有最后一步有可能不同。这种时候就产生了重复计算。我们要想办法避免重复计算。

考虑拆环为链。我们把环从 n 和 1 中间的地方拆开并复制一份。那么我们的问题就变成了：从编号为 i 的点开始，多少次可以走到编号为 $i + n$ 的点上？

我们不妨将每个点下一步会走到的点设为它的父亲，那么整个图就会变成一个树形的结构。考虑用并查集维护，我们按照 $1 - n$ 的顺序考虑每个点，假设当前考虑的第 i 号点，如果第 i 号点所在并查集的父节点的编号小于 $i + n$ ，就把它和它的父亲并起来。重复这个过程直到父节点的编号大于等于 $i + n$ ，这时父节点和 i 号点在树中的高度差就是从 i 号点出发的答案，每个点的高度容易预处理出来。

复杂度 $O(Qn\alpha(n))$ ，一共需要开大小为 $5n$ 的数组。

3.2 一个很快的算法

感谢彭雨翔同学教我这一做法。

依旧考虑拆环为链，但是现在我们不倍增长度。

考虑从一个点 i 开始，走到的最后一个点 j （再向前走就会回到链的开头）。如果从 j 不能走回 i ，那么我们不用考虑从 i 开始的情况！这是因为，比如说我们从 j 开始能走到的最远的点是 x ，那么一定有 $x < i$ ，那么从 x 开始就不会比从 i 开始劣。

那么我们可以通过一个简单的递推求出每个点会走到的最后一个点以及中间需要停留多少次。那么就可以很轻松的更新答案了。

复杂度 $O(Qn)$ ，一共需要开大小为 $4n$ 的数组。