

题目：F - 连接 (第2题) 官方题解 by en_translator

原题提案者：physics0523

在此问题中， $\$st\$$ 表示将字符串 $\$s\$$ 和 $\$t\$$ 连接后得到的字符串。

如果问题是以下形式，其解法是著名的：

给定 N 个字符串 S_i 。

找出以任意顺序连接 S_i 后得到的字典序最小的字符串。

ECR9-C 正是这个问题（[题解（英文）](#) [参考（日文）]

[\] ; ABC225-F 是这个问题的另一道练习题（题解）。](https://drken1215.hatenablog.com/entry/2022/12/16/194300)

首先，我们将重述这个问题的解法。

上述问题可以通过以下方式解决：

最优字符串是通过按照以下定义的顺序对字符串进行排序，然后按照得到的顺序连接它们而获得的：

- 顺序 $X < Y$ 当且仅当 $XY < YX$ 。

关于证明，请参考上面的题解。

然而，在此问题的约束下 ($N \leq 3 \times 10^5, \sum |S_i| \leq 10^6$)，存在一个陷阱。（这是由 Nyaan 在题目设定期间指出的。）

当比较字符串时，如果朴素地构造 XY 和 YX ，比较的成本是 $O(|X|+|Y|)$ 。

如果在排序过程中某个特定的字符串 S_i 被引用 $O(N)$ 次，计算复杂度将达到 $\Omega(N |S_i|)$ ，当 $|S_i|$ 很大时会导致 TLE。

（也存在规避此问题的解决方案。）出于同样的原因，我们不能在比较上花费 $O(\max(|X|, |Y|))$ 的时间。

但是，如果比较可以在 $O(\min(|X|, |Y|))$ 时间内完成，那么排序总共可以在 $O(\sum |S_i| \log N)$ 时间内完成。

我们如何在 $O(\min(|X|, |Y|))$ 时间内判断 $XY < YX$ ？

除了字符串 X 之外，我们还记忆 Z 数组（应用 Z 算法的结果） Z_X 。

以下是当 $|X| \geq |Y|$ 时的算法。（当 $|X| < |Y|$ 时同理。）

- 将 XY 视为 $X[1, |Y|] X[|Y|+1, |X|], Y$ ，将 YX 视为 $Y X[1, |X|-|Y|], X[|X|-|Y|+1, |X|]$ 。
- 首先，通过从头开始扫描来比较 $X[1, |Y|]$ 和 Y 。
- 如果至此结果仍未确定，我们需要比较 $X[|Y|+1, |X|]$ 和 $X[1, |X|-|Y|]$ 。
 - 我们使用 Z_X 在 $O(1)$ 时间内执行此比较。
 - 如果 $Z_X[|Y|+1] \geq |X|-|Y|$ ，我们认为这些字符串相等，因此我们继续比较下一个字符串。
 - 否则，这些字符串的第 $(Z_X[|Y|+1] + 1)$ 个字符不同，因此检查这些字符以确定顺序。
- 如果至此过程结果仍未确定，则通过从头开始扫描来比较 Y 和 $X[|X|-|Y|+1, |X|]$ 。

令 S'_1, S'_2, \dots, S'_N 是此排序的结果。我们现在将解释如何基于此找到答案。

- 如果 $N=2$, 那么答案是 $S'_2 S'_1$ 。
- 如果存在整数 $1 \leq i < N$ 使得 $S'_i S'_{i+1} = S'_{i+1} S'_i$, 那么交换 S'_i 和 S'_{i+1} 会产生相同的连接字符串, 因此答案是按 S' 中的顺序连接字符串得到的字符串。
- 如果至此答案仍未确定, 则满足:
 - 对于任何整数 $1 \leq i < N$, $S'_i S'_{i+1}$ 都不等于 $S'_{i+1} S'_i$ 。
- 实际上, 答案是以下两个字符串中字典序较小的那个。
 - $S'_1 S'_2 \dots S'_N S'_{N-1}$
 - $S'_1 S'_2 \dots S'_{N-1} S'_{N-2} S'_N$

我们将证明最后一个论断。

首先, 任何逆序数为 2 或更大的排列 (与 S' 相比) 不适用于答案。这是因为减少逆序数总是会产生字典序更小的字符串 (换句话说, 如果逆序数为 2 或更大, 则至少存在两个字典序更小的字符串。)

接下来, 考虑两个字符串 $S'_1 S'_2 \dots S'_{N-1} S'_N$ 和 $S'_1 S'_2 \dots S'_{N-1} S'_{N-2}$ 。

由于我们可以得到它们, 任何前缀不是 $S'_1 S'_2 \dots S'_{N-2}$ 的字符串都不可能是答案。

由于对于任何整数 $1 \leq i < N$, $S'_i S'_{i+1}$ 和 $S'_{i+1} S'_i$ 都不同, 在逆序数为 1 的排列中, 那些交换 S'_{N-2} 之前字符串的排列不可能是答案。

因此, 剩下以下两个候选:

- $S'_1 S'_2 \dots S'_N S'_{N-1}$
- $S'_1 S'_2 \dots S'_{N-1} S'_{N-2} S'_N$

例如, 对于 $S' = (\text{a, b, c})$, 前者是答案; 对于 $S' = (\text{a, aab, b})$, 后者是答案。

剩下的就是实现这个算法。

可以在 $O(\sum |S_i| \log N)$ 时间内对字符串进行排序。

即使使用朴素实现, 也可以判断是否存在 i 使得 $S'_i S'_{i+1} = S'_{i+1} S'_i$ 。

最后一步中两个候选字符串的比较也可以在 $O(\sum |S_i|)$ 时间内完成。

因此, 该问题已在总共 $O(\sum |S_i| \log N)$ 时间内解决。

示例代码 (C++):

发布于: 约1小时前

最后更新: 约1小时前