

Day2-题解

December 26, 2019

遗迹探索

题意：给 N 个串，问最多能拼成多少个”SAD” 子串；
 $N \leq 200000$

测试点 1-4

测试点 1-2: 暴力;

测试点 1-4

测试点 1-2: 暴力;

测试点 3-4: 我也不大清楚有什么做法, 感觉能做就放在了这里;

测试点 5-6

拼接形成的”SAD” 只有可能是”S + AD” 或”SA + D”;

测试点 5-6

拼接形成的”SAD” 只有可能是”S + AD” 或”SA + D”；

把串分成 9 类：

后缀是”SA”，”S”，或与”SAD” 无关；

前缀是”AD”，”D”，或与”SAD” 无关；

测试点 5-6

拼接形成的"SAD" 只有可能是" $S + AD$ " 或" $SA + D$ ";

把串分成 9 类:

后缀是"SA", "S", 或与"SAD" 无关;

前缀是"AD", "D", 或与"SAD" 无关;

那么" $S + AD$ " 的数量, 就是 $\min(\text{后缀是"S" 的串的数量, 前缀是"AD" 的串的数量})$;

那么" $SA + D$ " 的数量, 就是 $\min(\text{后缀是"SA" 的串的数量, 前缀是"D" 的串的数量})$;

测试点 5-6

拼接形成的"SAD" 只有可能是" $S + AD$ " 或" $SA + D$ ";

把串分成 9 类:

后缀是"SA", "S", 或与"SAD" 无关;

前缀是"AD", "D", 或与"SAD" 无关;

那么" $S + AD$ " 的数量, 就是 $\min(\text{后缀是"S" 的串的数量, 前缀是"AD" 的串的数量})$;

那么" $SA + D$ " 的数量, 就是 $\min(\text{后缀是"SA" 的串的数量, 前缀是"D" 的串的数量})$;

发现有点问题?

测试点 5-6

有些情况下串不能”物尽其用”；

测试点 5-6

有些情况下串不能”物尽其用”；

1: ”S” 后缀与”AD” 前缀数量相等，且属于完全相同的一些串；

测试点 5-6

有些情况下串不能“物尽其用”；

1: “S” 后缀与“AD” 前缀数量相等，且属于完全相同的一些串；
这样“S + AD” 就要少 1；

对“SA + D” 同理；

测试点 5-6

有些情况下串不能”物尽其用”；

1: ”S” 后缀与”AD” 前缀数量相等，且属于完全相同的一些串；
这样”S + AD” 就要少 1；

对”SA + D” 同理；

2: 用到的每个串的前缀和后缀都要使用；

测试点 5-6

有些情况下串不能“物尽其用”；

1: “S” 后缀与“AD” 前缀数量相等，且属于完全相同的一些串；
这样“S + AD” 就要少 1；

对“SA + D” 同理；

2: 用到的每个串的前缀和后缀都要使用；
这样一定会浪费开头的串和结尾的串，答案会减少 1；

测试点 5-6

有些情况下串不能“物尽其用”；

1: “S” 后缀与“AD” 前缀数量相等，且属于完全相同的一些串；
这样“S + AD” 就要少 1；

对“SA + D” 同理；

2: 用到的每个串的前缀和后缀都要使用；
这样一定会浪费开头的串和结尾的串，答案会减少 1；
处理了这两种情况后就可以通过这一部分；

测试点 7-8

一个来自一位同学的神仙做法：

测试点 7-8

一个来自一位同学的神仙做法：
开三个点，记末尾状态分别为”，”S”，”SA”；
把每个串看做一条边，能形成”SAD” 的路径边权设为 1；

测试点 7-8

一个来自一位同学的神仙做法：
开三个点，记末尾状态分别为”， ”S”， ”SA”；
把每个串看做一条边，能形成”SAD” 的路径边权设为 1；
跑有正环的最大费用可行流；

测试点 7-8

一个来自一位同学的神仙做法：
开三个点，记末尾状态分别为”，”S”，”SA”；
把每个串看做一条边，能形成”SAD” 的路径边权设为 1；
跑有正环的最大费用可行流；
但这样似乎无法处理自环的情况；

测试点 9-10

回归测试点 5-6 的想法；
只要处理”A” 的情况；

测试点 9-10

回归测试点 5-6 的想法；
只要处理”A” 的情况；
只有链接在”S” 之后的”A” 是有用的；

测试点 9-10

回归测试点 5-6 的想法；
只要处理”A” 的情况；
只有链接在”S” 之后的”A” 是有用的；
直接枚举这样的”A” 有多少个；
然后套用之前的做法就行了；

测试点 9-10

回归测试点 5-6 的想法；
只要处理”A” 的情况；
只有链接在”S” 之后的”A” 是有用的；
直接枚举这样的”A” 有多少个；
然后套用之前的做法就行了；
注意此时一定可以避免特殊情况 1。

基因进化

一个数列，有一些前缀可以翻转，按顺序翻转一些前缀使字典序最小；

$$N \leq 300000$$

测试点 1-2

暴力枚举每个位置翻或不翻;

测试点 1-2

暴力枚举每个位置翻或不翻;
 $O(2^N)$

测试点 3-4

我们现在想求出，对于前 i 个数，所能产生的最小的字典序是多少；

测试点 3-4

我们现在想求出，对于前 i 个数，所能产生的最小的字典序是多少；
我们发现，无论后面的怎么翻，之前的一定是越小越好；

测试点 3-4

我们现在想求出，对于前 i 个数，所能产生的最小的字典序是多少；
我们发现，无论后面的怎么翻，之前的一定是越小越好；
对于相邻两个能翻的位置 i, j , ($i < j$)。

测试点 3-4

我们现在想求出，对于前 i 个数，所能产生的最小的字典序是多少；
我们发现，无论后面的怎么翻，之前的一定是越小越好；
对于相邻两个能翻的位置 i, j , ($i < j$)。
前 j 个的最小值要么是前 i 个的最小值接上 i 到 j 这一段；
要么是 i 到 j 的翻转接上前 i 个的最小值（同时在 i 和 j 翻转）；

测试点 3-4

我们现在想求出，对于前 i 个数，所能产生的最小的字典序是多少；
我们发现，无论后面的怎么翻，之前的一定是越小越好；
对于相邻两个能翻的位置 i, j , ($i < j$)。
前 j 个的最小值要么是前 i 个的最小值接上 i 到 j 这一段；
要么是 i 到 j 的翻转接上前 i 个的最小值（同时在 i 和 j 翻转）；
每次翻转前判断哪种方式更优， $O(N)$ 暴力比较；

测试点 5-6

考虑用哈希 + 二分的方式判断优劣程度；
我们现在要维护这些操作：

测试点 5-6

考虑用哈希 + 二分的方式判断优劣程度；
我们现在要维护这些操作：
尾部插入，头部插入，维护一段的哈希值；

测试点 5-6

考虑用哈希 + 二分的方式判断优劣程度；
我们现在要维护这些操作：
尾部插入，头部插入，维护一段的哈希值；
一个朴素的实现就是线段树；

测试点 5-6

考虑用哈希 + 二分的方式判断优劣程度；
我们现在要维护这些操作：
尾部插入，头部插入，维护一段的哈希值；
一个朴素的实现就是线段树；
 $O(N \log^2 N)$

测试点 7-10

其实根本不需要用数据结构维护；

测试点 7-10

其实根本不需要用数据结构维护；
直接用两个队列，记录头插入和尾插入的数；

测试点 7-10

其实根本不需要用数据结构维护；
直接用两个队列，记录头插入和尾插入的数；
维护队列的前缀哈希值和后缀哈希值；
用这些一定可以拼出一段的哈希值；

测试点 7-10

其实根本不需要用数据结构维护；
直接用两个队列，记录头插入和尾插入的数；
维护队列的前缀哈希值和后缀哈希值；
用这些一定可以拼出一段的哈希值；
 $O(N \log N)$

城市破坏

给定一张 N 个点, M 条边的无向图, K 个询问, 求删掉边 C 条边后图是否连通;

$$N, M, K \leq 10^5$$

测试点 1-5

测试点 1-4: $N, M, K \leq 2000$ 直接暴力判断删边后图的连通性;

测试点 1-5

测试点 1-4: $N, M, K \leq 2000$ 直接暴力判断删边后图的连通性;
测试点 5: 删边之后图一定不连通;

测试点 6-8

只删一条边；

测试点 6-8

只删一条边；
只有删到桥时图不连通；

测试点 9-11

$$M \leq N + 20$$

测试点 9-11

$$M \leq N + 20$$

先随便取出一棵原图的生成树；
如果删的边都不在树上一定连通；

测试点 9-11

$$M \leq N + 20$$

先随便取出一棵原图的生成树；

如果删的边都不在树上一定连通；

否则最多把生成树分成 $C + 1$ 个部分；

用剩下的 $M - (N - 1)$ 判断这些部分是否能连通；

测试点 9-11

$$M \leq N + 20$$

先随便取出一棵原图的生成树；

如果删的边都不在树上一定连通；

否则最多把生成树分成 $C + 1$ 个部分；

用剩下的 $M - (N - 1)$ 判断这些部分是否能连通；

如何判断一个点属于哪一部分？

测试点 9-11

$$M \leq N + 20$$

先随便取出一棵原图的生成树；

如果删的边都不在树上一定连通；

否则最多把生成树分成 $C + 1$ 个部分；

用剩下的 $M - (N - 1)$ 判断这些部分是否能连通；

如何判断一个点属于哪一部分？

对于每一条在树上的被割掉的边，我们标记下这条边两侧所对应的点；

那么一个点一定和被标记且距离它最近的点同属一个部分；

测试点 9-11

$$M \leq N + 20$$

先随便取出一棵原图的生成树；

如果删的边都不在树上一定连通；

否则最多把生成树分成 $C + 1$ 个部分；

用剩下的 $M - (N - 1)$ 判断这些部分是否能连通；

如何判断一个点属于哪一部分？

对于每一条在树上的被割掉的边，我们标记下这条边两侧所对应的点；

那么一个点一定和被标记且距离它最近的点同属一个部分；

而两点间的距离可以 $O(1)$ 求；

所以对于每个询问，复杂度是 $O(C * (M - (N - 1)))$ 的；

测试点 12-20

删边判断连通性的问题有一个比较巧妙的思路：

测试点 12-20

删边判断连通性的问题有一个比较巧妙的思路：

就你给每条边定义一个权值，如果删的边恰好使图不连通了，他们的异或是 0；

这里的恰好指的是一旦删的边少了任意一条都不能使图不连通；

测试点 12-20

删边判断连通性的问题有一个比较巧妙的思路：

就你给每条边定义一个权值，如果删的边恰好使图不连通了，他们的异或是 0；

这里的恰好指的是一旦删的边少了任意一条都不能使图不连通；

首先如果割的边恰好使一个点不连通了，异或应该是 0，所以一个点的所有出边的异或和应该是 0；

测试点 12-20

删边判断连通性的问题有一个比较巧妙的思路：

就你给每条边定义一个权值，如果删的边恰好使图不连通了，他们的异或是 0；

这里的恰好指的是一旦删的边少了任意一条都不能使图不连通；

首先如果割的边恰好使一个点不连通了，异或应该是 0，所以一个点的所有出边的异或和应该是 0；

然后我们发现，对于任意一个子集，如果恰好与外部的点不连通了，且子集中每个点出边异或和都是 0，那么那些被割掉的边异或和也为 0；

测试点 12-20

删边判断连通性的问题有一个比较巧妙的思路：

就你给每条边定义一个权值，如果删的边恰好使图不连通了，他们的异或是 0；

这里的恰好指的是一旦删的边少了任意一条都不能使图不连通；

首先如果割的边恰好使一个点不连通了，异或应该是 0，所以一个点的所有出边的异或和应该是 0；

然后我们发现，对于任意一个子集，如果恰好与外部的点不连通了，且子集中每个点出边异或和都是 0，那么那些被割掉的边异或和也为 0；

证明：考虑把子集内每个点的所有出边都异或到一起，那么它们的异或值为 0；

然后对于子集内部的边，它们会被异或两次，所以它们的异或值为 0；

所以所有从子集中连出去的边的异或值为 0；

测试点 12-20

现在问题是如何构造边的权值是它满足一个点的所有出边的异或和为 0;

测试点 12-20

现在问题是如何构造边的权值是它满足一个点的所有出边的异或和为 0;

其实可以存在一个点它的异或值不为 0, 因为不连通后总有一个部分所有点的异或值为 0;

测试点 12-20

现在问题是如何构造边的权值是它满足一个点的所有出边的异或和为 0;

其实可以存在一个点它的异或值不为 0, 因为不连通后总有一个部分所有点的异或值为 0;

还是考虑随机选取一棵生成树 (1 号点为根);

对于非树边, 我们随机一个 1 到 $2^{63} - 1$ 的权值;

对于树边, 我们可以给定一个权值, 使它向下连向的节点的异或和为 0;

测试点 12-20

现在问题是如何构造边的权值是它满足一个点的所有出边的异或和为 0;

其实可以存在一个点它的异或值不为 0, 因为不连通后总有一个部分所有点的异或值为 0;

还是考虑随机选取一棵生成树 (1 号点为根);

对于非树边, 我们随机一个 1 到 $2^{63} - 1$ 的权值;

对于树边, 我们可以给定一个权值, 使它向下连向的节点的异或和为 0;

这样的话, 除了 1 号点, 每个点的所有出边异或和都为 0;

测试点 12-20

现在问题是如何构造边的权值是它满足一个点的所有出边的异或和为 0;

其实可以存在一个点它的异或值不为 0, 因为不连通后总有一个部分所有点的异或值为 0;

还是考虑随机选取一棵生成树 (1 号点为根);

对于非树边, 我们随机一个 1 到 $2^{63} - 1$ 的权值;

对于树边, 我们可以给定一个权值, 使它向下连向的节点的异或和为 0;

这样的话, 除了 1 号点, 每个点的所有出边异或和都为 0;

如果当前删的边不能恰好使图不连通, 那么异或值一定是一些随机边权异或起来;

随机边权异或起来得到 0 的概率是 $\frac{1}{2^{63}}$;

所以出现偏差的概率几乎是 0;