

two strings 试题讨论

杭州学军中学 罗干

1 问题描述

给定两个字符串A和B，有五种操作：

操作1 在A串开头添加一个字符；

操作2 在A串结尾添加一个字符；

操作3 在B串开头添加一个字符；

操作4 在B串结尾添加一个字符；

操作5 询问当前的B串在当前A串中出现的次数。

保证字符均为小写字母，且A、B串初始非空。

1.1 输入输出

1.1.1 输入格式

第一行第二行分别为初始的A串和B串，第三行一个整数m，表示操作的次数，接下来m行，每行表示一个操作，每行第一个数为一个在1-5之间的数字，若其值不为5，则在数字后会有一小写字母。

1.1.2 输出格式

对于每个询问，每行输出一个整数，表示B串在A串中出现的次数。

1.1.3 数据规模

10%的数据中，最终A串和B串长度之和小于等于200，操作数小于等于10。

30%的数据中，最终A串和B串长度之和小于等于2000，操作数小于等于1000。

100%的数据中，最终A串和B串长度之和小于等于200000，操作数小于等于200000。

1.2 样例

1.2.1 输入

```
ababc
a
7
5
4 b
5
3 a
1 a
5
5
```

1.2.2 输出

```
2
2
1
1
```

1.2.3 样例解释

第几次询问	A串	B串	B串在A串中出现的次数
第1次	ababc	a	2
第2次	ababc	ab	2
第3次	aababc	aab	1
第4次	aababc	aab	1

1.3 朴素算法

1.3.1 模拟法

模拟每个操作，对于每次询问，直接枚举B串在A串中的位置，然后直接比较。

1.3.2 KMP优化模拟法

在模拟法的基础上，对于询问我们用KMP算法去匹配。

1.4 离线优化模拟法

把所有有关A串的操作全部读入组成一个”大A串”。

那么每次询问时当前的A串就相当于询问”大A串”的某一段中几个有个B串。

每次有三四两种操作就重新做一遍KMP，记下哪几个位置与B串匹配。

那么询问就变成了这个几个位置有几个在L R-Len(B)+1之间。

因为L递减、R递增，所以询问就可以O(1)解决。

1.4.1 各操作时间复杂度

操作1 $O(n) \rightarrow O(1)$

操作2 $O(n) \rightarrow O(1)$

操作3 $O(n)$

操作4 $O(n)$

操作5 $O(n) \rightarrow O(1)$

1.5 后缀数组离线处理算法

1.5.1 在线→离线

利用刚刚的离线的思想，将A、B串最终的结果串并在一起，并在中间添加一个分隔符(如‘+’)，用后缀数组来维护后缀之间的关系。

举个例子：

大A串:ABABC

大B串:ABA

最终串: ABABC+ABA

后缀数组排序后:

A

ABA

ABC+ABA

ABABC+ABA

BA

BABC+ABA

BC+ABA

C+ABA

+ABA

1.5.2 求出可行的后缀区间

当前B串在A串第 i 到第 $i+\text{Len}(B)-1$ 处出现, 等同于当前B串与A串第 i 个后缀的最长公共前缀长度(即 Lcp) $\geq \text{Len}(B)$ 。

我们可以用倍增预处理 height 数组的方法, 在 $\log(n)$ 的时间内找到这个区间。

1.5.3 统计合法的A串起始位置

我们定义合法位置为:对于大A串的位置 i , $[i, i+\text{Len}(B)-1]$ 这一段都在当前的A串中。

1.5.4 树状数组维护

我们可以发现:任意一个操作只会改变至多两个位置的合法性, 所以我们可以用树状数组来维护后缀数组中某一段有多少个合法位置。

1.5.5 各操作时间复杂度

操作1 $O(\log(n))$

操作2 $O(\log(n))$

操作3 $O(\log(n))$

操作4 $O(\log(n))$

操作5 $O(\log(n))$

1.6 考查要点

对题目本质的深入思考

对各类数据结构的熟练掌握

对各类模拟算法的优化能力

对离线算法的应用能力