

# RUPC2019day2

## M: 1333

原案・解説: beet

# 概要

長さ $N$ の文字列 $S$ が与えられる。以下のクエリを $Q$ 回処理せよ。

クエリ

$S[L:R]$ を $S$ の $L$ 文字目から $R$ 文字目まで（両端を含む）からなる文字列とする。

$S[L:R]$ を適当な文字列 $A, B, C, X$ を用いて $AXBXCX$ と表すことを考える

そのような $X$ の中で最長のものの長さを入力する。 $(1 \leq |A|, |B|, |C|, |X|)$

例

sensanbyakusanjuusan

sen**san**byaku**san**juu**san**

# 方針

$|X| > 1$  のとき、 $X$ の先頭文字を $a$ 、残りを $Y$ とすると、( $X = 'a' + Y$ )

$$A' = Aa$$

$$B' = Ba$$

$$C' = Ca$$

$$X' = Y$$

とおくことで、一文字減らせる→単調性がある→二分探索

# 方針

長さを決め打ちして、 $AXBXCX$ と表せるかを判定する

$X$ は $S[L:R]$ のsuffixなので、長さが定まると一意に定まる

ある区間の中に $X$ が存在するかを求めたい→Suffix Array

# 方針

prefixがXになるようなindexの集合は、Suffix Array上の区間になる

ある区間に対して特定の値より小さい中で最大値を求めたい

→Wavelet Matrix

## 例 : Suffix Array

10: an  
7: ansan  
4: ansansan  
2: isansansan  
0: itisansansan  
11: n  
8: nsan  
5: nsansan  
9: san  
6: sansan  
3: sansansan  
1: tisansansan

S = itisansansan

“an”をprefixに持つのは  
10, 7, 4 : [0, 2] に対応

# 別解

Segment Treeにソート済みの点列を載せてlower\_boundしても間に合います  
logが一個多いんですが定数倍が軽くて落とせなかった

セグ木サイコー！ 一番好きなデータ構造です

なんか並列二分探索してるチームもあったけどよくわからん

# 統計

FA:

オンサイト : rupc\_suimin (3:26:23)

オンライン : IH19980412 (2:26:20)

Success Rate: 44% (4 / 9)

# ジャッジ解

beet                    c++                    314行

beet                    java                    330行

haji                    c++                    251行

tubuann c++                    301行