

Compressed omikuji

Writer : platypus

概要

- ▶ 英小文字とアスタリスク(*)で構成された文字列が与えられる。
- ▶ すべてのアスタリスクを長さNのある文字列で置き換えることにより、全体を回文にしたい！
- ▶ できるか判定せよ！
- ▶ 文字列問題、Yes,No判定問題

(*)これは注ではありません

愚直

- ▶ とりあえず愚直に回文判定するには？
- ▶ 文字列のアルファベットを一つの数字に、アスタリスクをN項の数列に置き換え、全体を数列に変換する。
- ▶ 後はfor文で前からi番目、後ろからi番目を見ていき、矛盾が生じたらNo、最後まで矛盾なくアルファベットを割り当てられたらYes
- ▶ ただ、この方法だとアスタリスクが大量にあるテストケースでTLEしてしまう。
- ▶ 計算量 $O(N * |S|)$

*はそのまま扱いたい

- ▶ できればアスタリスクは $O(1)$ で丸ごと扱いたい
- ▶ 与えられた文字列の先頭及び末尾から同時に尺取り風に回文条件を当てはめると…？
- ▶ 文字列に関する条件は以下の3つに分解できる！
 1. *の先頭a文字と反転した*の末尾a文字が共通である
⇔*の先頭a文字が回文
 2. *の末尾a文字と反転した*の先頭a文字が共通である
⇔*の末尾a文字が回文
 3. *のa文字目は文字xである
- ▶ 以上の条件を判定するには？

Union find tree

- ▶ 変数集合があり、「変数xと変数yは等しい」「変数xは値Pである」のようなクエリが矛盾を生じないかを調べるにはUnion find treeが有効
- ▶ 値P自体を変数として持ち、異なる値同士が同じと判定されていなければ矛盾なし、そうでなければ矛盾あり
- ▶ *の各文字 $s[0], s[1], \dots, s[N-1]$ およびアルファベット a, b, c, \dots, x, y, z の合計 $N+26$ 変数をUnion find treeに入れ、条件を確認する。
- ▶ ただ、一つの回文条件に $O(N)$ かかるため、オーダーは $O(N * |S| * \alpha(N))$
- ▶ 時間計算量は全く改善されていない！！

回文条件のマージ

- ▶ ある文字列の先頭a文字、および先頭b文字が回文の場合を考えてみる

回文条件のマージ

- ▶ ある文字列の先頭a文字、および先頭b文字が回文の場合を考えてみる

回文(長さa)

回文(長さb)

回文条件のマージ

- ▶ ある文字列の先頭a文字、および先頭b文字が回文の場合を考えてみる

回文(長さa)

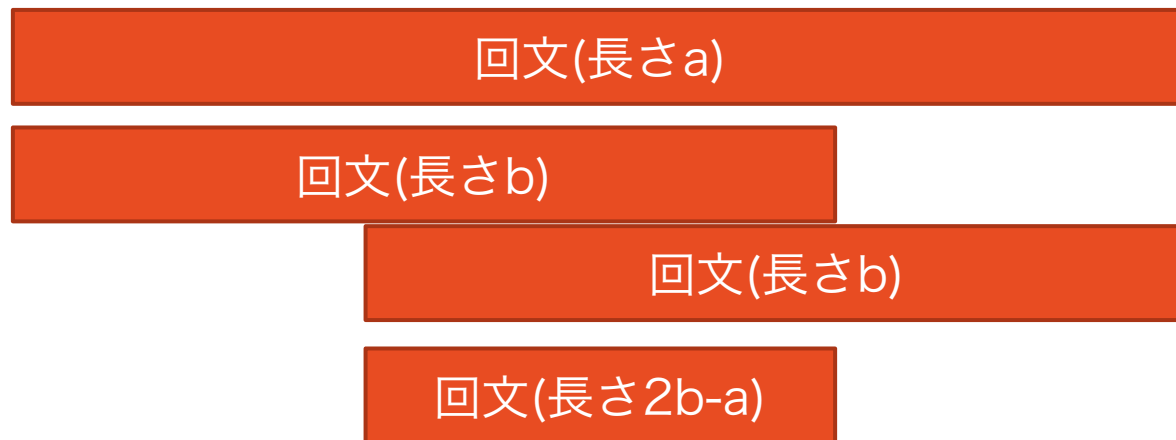
回文(長さb)

回文(長さb)

回文(長さa)よりこっちにも同じ回文

回文条件のマージ

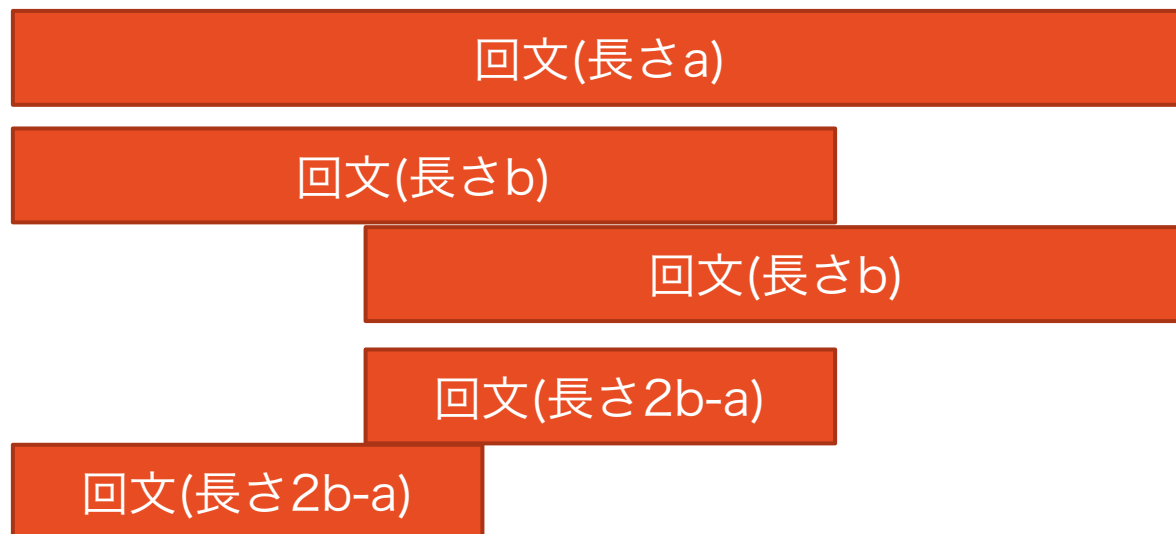
- ▶ ある文字列の先頭 a 文字、および先頭 b 文字が回文の場合を考えてみる



同じ回文同士が重なり合うとそこも回文
(実は文字列と反転文字列が重なり合っても回文)

回文条件のマージ

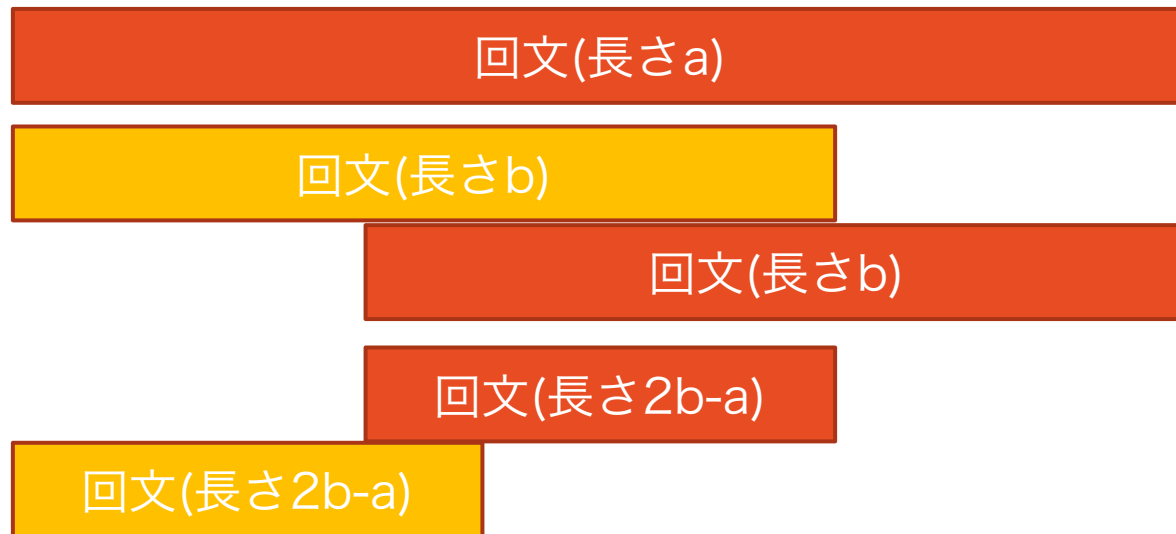
- ▶ ある文字列の先頭 a 文字、および先頭 b 文字が回文の場合を考えてみる



回文(長さ b)よりこっちにも回文(長さ $2b-a$)

回文条件のマージ

- ▶ ある文字列の先頭 a 文字、および先頭 b 文字が回文の場合を考えてみる



回文条件のマージ

- ▶ ある文字列の先頭 a 文字、および先頭 b 文字が回文の場合を考えてみる

回文(長さ b)

回文(長さ $2b-a$)

回文条件のマージ

- ▶ ある文字列の先頭 a 文字、および先頭 b 文字が回文の場合を考えてみる

回文(長さ b)

回文(長さ $2b-a$)

- ▶ 再帰的なことが起きている…？

回文条件のマジ

- ▶ ある文字列の先頭 a 文字と先頭 b 文字が回文の場合、先頭 $2b-a$ 文字も回文！
- ▶ ただし $2b-a > 0$ の時
- ▶ これは再帰的な条件なので、 $2b-a > 0$ ($\Leftrightarrow b > a/2$) である限りずっと続く
- ▶ 再帰的に適用すると、先頭 a 文字は周期 $b-a$ であり、かつ部分文字列 $[0, a\%(b-a))$ と $[a\%(b-a), b-a)$ は回文である ($x\%y$ は x を y で割った余り) という条件にたどり着くことがわかる。
- ▶ この「周期 $b-a$ 」をうまく使い計算量を落とせないか??

ダブリング

- ▶ 「先頭???文字が回文」という情報を1,2,4,8,16,...ごとに区切り、グループに分けていく
- ▶ 2^i 以上 $2^{(i+1)}$ 未満のグループの要素を仮に $p[1], p[2], \dots, p[M]$ と置く。
($p[1] < p[2] < \dots < p[M]$)
- ▶ $p[1] * 2 - p[M] > 0$ のため、前述の考察は成立し、先頭 $p[M]$ 文字は周期 $p[M] - p[1]$ である。
- ▶ 同様に周期 $p[M] - p[2], p[M] - p[3], \dots, p[M] - p[M-1]$ でもあるため、全体をまとめると、先頭 $p[M]$ 文字は周期 $\text{GCD}(p[M] - p[1], p[M] - p[2], \dots, p[M] - p[M-1])$ である。

(\therefore 一般的に、長さ $a+b$ 以上の文字列が周期 a かつ周期 b の場合、周期 $\text{GCD}(a, b)$)
(GCDは最大公約数)

計算量

- ▶ 各グループについての計算量を考える。
- ▶ 先頭 p 文字の周期が q だとわかった場合、周期の要素ごとにつなげていけばよいので、合計で $O(p)$ 回繋ぐ
- ▶ また、一つの回文情報に付き $O(q)$ で処理できる。
(すべてを見なくても、周期の性質上先頭 q 文字さえ見ればよいので)
- ▶ 回文情報の個数は高々 $O(p/q)$ である(理由： q は最大公約数のため、 p 以下の整数で q を約数に持つような整数は高々 p/q 個のため)
- ▶ よって、全体の回文情報を反映させるためにも $O(p/q * q) = O(p)$ 回繋ぐ
- ▶ 以上から、各グループごとに $O(p * \alpha(p))$ の計算量がかかる。

計算量

- ▶ グループのサイズは $1, 2, 4, \dots$, と2の累乗のため、全体の計算量は $1 * \alpha(1) + 2 * \alpha(2) + 4 * \alpha(4) + \dots = O(N * \alpha(N))$ である。
- ▶ 条件列挙に必要な尺取り法は $O(|S|)$
- ▶ よって全体で $O(|S| + N * \alpha(N))$ でこの問題を解くことができた。

まとめ

- ▶ 回文に関する問題は中～高難易度の問題が多い
- ▶ 典型知識も多いが、案外まとめてあるサイトは少ないので自分で積極的に調べてみよう
- ▶ Yes,No問題なので、非常にバグ検出が難しく、かつ嘘解法にも気づきにくい
- ▶ ゆっくりと証明を進めていき、丁寧にコードに書き写そう