

---

# JAG模擬国内予選2022

## C: i18n

原案: darsein  
問題文: darsein  
データセット: hasi  
解答: darsein, rian, tsutaj, 他多数

---

# 問題概要

英単語の省略として、接頭辞と接尾辞を1文字ずつ以上残し、残りの部分を文字数で置き換えることを考える (例: internaionalization → int13tion)

辞書に含まれるN個の単語を省略し、置き換えに用いた数の合計を最大化せよ。ただし、辞書中の複数の単語の省略形に対応する形には省略できない。

制約:

$$1 \leq N \leq 1000, 3 \leq |S_i| \leq 30$$

# 想定解法

- $L = \max_i(|S_i|)$  とする
- すべての考えうる変換を試す
  - 例: "love" -> "l2e", "l1ve", "lo1e", "love"
  - 各言語において, 数値を文字列に変換する方法を覚えておくといよい
  - 接頭辞の長さ  $O(L)$  通りと接尾辞の長さ  $O(L)$  通りなので,  $O(L^2)$  個の変換がありえる
  - 変換の計算にそれぞれ  $O(L)$
- すべての単語のすべての変換の個数を数えてメモする
  - マップを用いて, 変換した文字列をキー, 個数を値にするとよい
  - 例: ["love", "live"] -> {"love": 1, "live": 1, "l2e": 2, "l1ve": 2, "lo1e": 1, "li1e": 1}
  - ハッシュマップで  $O(L)$  (キーの照会に文字列比較で  $O(L)$  かかるため)
- 変換の個数が複数あるもの (マップの値が2以上のもの) は使えない. 各単語ごとに, 変換の値が1のもので最大の値を見つける.

計算量:  $O(NL^3)$ 、今回の制約 & TLEの緩い国内予選形式ならこの計算量で通る

# おまけ: 計算量改善

- 計算量のボトルネックは、変換の $O(L)$ と文字列比較の $O(L)$ 
  - ここが落とせれば  $O(NL^2)$  になる
  - 文字列比較をやめるには？ → 文字列のハッシュ値を計算できればよい
- 考察: 長さが異なる単語同士の省略形は必ず異なる
  - actually と accessibility はどちらも  $aKc$ 、 $aLc$  などの形に省略できるが、 $K$ や $L$ が必ず異なる
  - よって、同じ長さのものをまとめて、その中だけで考えてもよい
- 長さが同じ → 省略形の比較に文字数はいらぬ
  - actually も assembly も  $a6y$  と略せるが、同じ長さなので  $a\#y$  と略しても問題ない
  - さらに、接尾辞と接頭辞を入れ替えても問題ない
- $\{\text{suffix}\}\#\{\text{prefix}\}$  のハッシュ値をローリングハッシュで求めれば、ハッシュ値の計算も変換1つあたり $O(1)$ にできる

# ジャッジ解

beet: 67行 (C++)

darsein: 95行 (C++)

hasi: 65行 (C++)

HIR: 228行 (C++)

hos: 108行 (C++)

rian: 44行 (C++)

tomerun: 60行 (C++)

# 統計情報

- AC / Trying teams
  - 80/97 (82.47%)
- First Acceptance
  - nowcow (00:06:46)