

Problem A

王様の視察 / King's Inspection

原案: 繁富

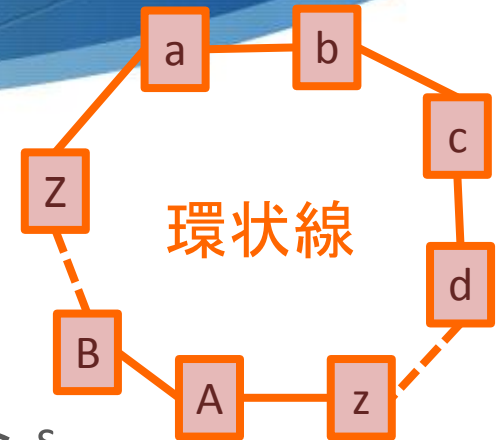
問題: 伊藤・西田

解説: 伊藤

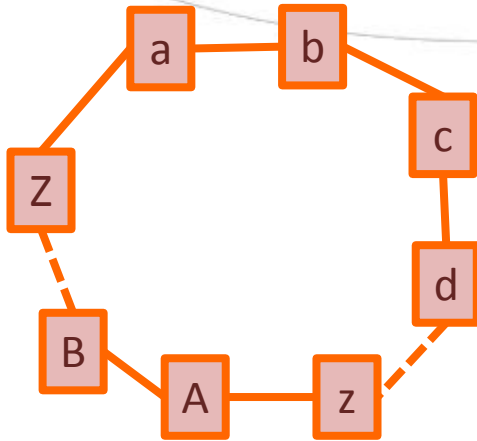


問題概要

- ◆ 環状線に52個の駅 (a-zA-Z) が存在する
- ◆ 入力
 - ◆ 王様が視察する駅の順番が暗号化されたリスト s
 - ◆ 暗号を復号化するための鍵のリスト $k_1 \sim k_n$
- ◆ 暗号を復号化して、王様が視察する駅の順番を出力せよ
- ◆ 鍵について
 - ◆ n 個目の鍵の次は、1個目の鍵を使用する
 - ◆ 暗号化された駅 x を鍵 k_i で復号化すると、 x の k_i 個手前の駅になる



問題例



例) $s = \text{"acc"}$, $k1 = 1$, $k2 = 2$

暗号文	a	c	c
鍵	1	2	1
復号文	Z	a	b

- 1文字目: 'a' の1つ手前($k1$ を使用)の駅になるので、'Z'
- 2文字目: 'c' の2つ手前($k2$ を使用)の駅になるので、'a'
- 3文字目: 鍵は、1つ目に戻り $k1$ を使用して、'b'

解法

- ◆ 問題に指定されている通りに、素直に解きましょう
- ◆ どんな解き方でも大丈夫です

解法の一例

- ◆ string st = “abcd...zAB...Z” という駅リストを作成
- ◆ 復号化したい駅のインデックスを参照 => st[idx] とする
- ◆ 鍵 ki で st[idx] を復号化する場合、以下の式で求まる
 - ◆ $st[(idx - ki + 52) \% 52]$
 - ◆ st[idx] のki 個手前の駅を求める処理
 - ◆ (idx - ki)に、52を足して、剰余をとることで循環構造を実現
- ◆ 例) 鍵 1 で ‘a’ を復号化する場合
 - ◆ ‘a’ に対応するインデックスは、0
 - ◆ $st[(0 - 1 + 52) \% 52] = st[51] = ‘Z’$

A問題について

- ◆ 一番簡単な問題
 - ◆ 基本的なプログラミング能力があれば解ける問題
 - ◆ これは、本番でも変わらないでしょう
 - ◆ **正確に高速に**解くことが、B問題以降を多く解くためのステップアップ
- ◆ ミスはしないように
 - ◆ 速く解くことも大事だけど、**ミスがないように、落ち着いて**コーディング
 - ◆ **1回のミスだけで、1200ペナルティ**加算です
 - ◆ 簡単な問題で、ペナルティをもらうのだけは、避けたいところです

結果

- 💧 First AC
 - 💧 Operasan (280秒)
 - 💧 ↑zero
- 💧 Total AC
 - 💧 122