

A : koukyoukokukikou

# 問題概要

- アルファベット小文字からなる文字列が与えられる
  - タイプするときを使う手の左右が入れ替わる回数は何回か？
  - 例 acmicpc
    - (ac) \* (mi) \* (c) \* (p) \* (c)
    - (LL) \* (RR) \* (L) \* (R) \* (L)
- > 4

# 解法

- 文字列内で条件を満たす箇所を数え上げる
  - $(a * c) * (m * i) * (c) * (p) * (c)$
- 使う手の左右が入れ替わっているか、プログラムにどうやって書くのがよいだろうか？

# すごくよくない書き方

- if, switch 文を 2 段にして判定する

```
if (in[i] == 'a')  
    if (in[i + 1] == 'y')  
        ans++;  
    else if (in[i + 1] == 'h')  
        ans++;  
    ...
```

- 書くのに非常に手間がかかる
- バグ混入率が上がり、発見も難しくなる

# やはりよくない書き方

- 左右の判定部を関数にまとめる
- しかし if, switch 文で列挙する

```
bool isLeft(char c1) {  
    if (in[i] == 'a')  
        return true;  
    ...  
}
```

- 多少かもしれませんが欠点は同じ

# いくらかスマートな書き方

- 左手あるいは右手で打つキーのリストを文字列リテラルで与える
- それらの文字列に含まれるかでどちらの手か判定できる

# コメント

- ACM-ICPC はプログラムを書く時間もバグフィックスの時間も非常に限られています
- 速く・正確に・間違った時に修正しやすいプログラムを書くように心がけましょう
- A 問題に時間がかかっているようでは突破は難しいですよ!