### Kick Start 2021 - Round H

# Analysis: Transform the String

Let us first define the two operations that we can perform.

- **Clockwise**: Changing a letter to the one following it. For example, changing from `c` to `d`.
- **Counter-clockwise**: Changing a letter to the one preceding it. For example, changing from `a` to `z`.

Let us denote the ASCII value of a character $c_x$ by $ASCII(c_x)$. If we move the padlock from a character $c_a$ to another character $c_b$ such that $ASCII(c_a) < ASCII(c_b)$, the number of operations required in clockwise direction = $ASCII(c_b) - ASCII(c_a)$ and the number of operations required in counter-clockwise direction = $26 - (ASCII(c_b) - ASCII(c_a))$.

For example, if we move the padlock from `c` to `e`:

- Number of operations required in clockwise direction = $ASCII(e) - ASCII(c) = 2$.
- Number of operations required in counter-clockwise direction = $26 - (ASCII(e) - ASCII(c)) = 24$.

Similarly, if we move the padlock from a character $c_a$ to another character $c_b$ such that $ASCII(c_a) > ASCII(c_b)$, the number of operations required in clockwise direction = $26 - (ASCII(c_a) - ASCII(c_b))$ and the number of operations required in counter-clockwise direction = $ASCII(c_a) - ASCII(c_b))$.

For example, if we move the padlock from `g` to `b`:

- Number of operations required in clockwise direction = $26 - (ASCII(g) - ASCII(b)) = 21$.
- Number of operations required in counter-clockwise direction = $ASCII(g) - ASCII(b) = 5$.

Thus minimum number of operations required to change a character in the padlock from $c_a$ to $c_b$ = $min(abs(ASCII(c_a) - ASCII(c_b)), 26 - abs(ASCII(c_a) - ASCII(c_b)))$.

Let us call the above expression $f(c_a, c_b)$.

## Approach 1

### Test Set 1

When length of $\mathbf{F}$ = 1 we need to change every character in $\mathbf{S}$ to that in $\mathbf{F}$. Therefore, the answer is the sum of $f(c_s, c_f)$ for every character $c_s$ in $\mathbf{S}$ and $c_f$ in $\mathbf{F}$.

### Test Set 2

For this case, $\mathbf{F}$ can have multiple characters.

For each character in $\mathbf{S}$ we need to find a character in $\mathbf{F}$ such that $f(c_s, c_f)$ is minimized. Therefore, for every character $c_s$ in $\mathbf{S}$, we iterate over all possible characters $c_f$ in $\mathbf{F}$ and find minimum of $f(c_s, c_f)$ and add the minimum value to the final answer.

## Approach 2

For each character in $\mathbf{S}$, move the padlock in the clockwise direction and count the number of operations until we reach a character that belongs to $\mathbf{F}$. Similarly, for the same character in $\mathbf{S}$, move the padlock in the counter-clockwise direction and count the number of operations until we reach a character that belongs to $\mathbf{F}$. Compare the number of operations in both directions and add minimum of the two to the final answer.