



### AO VIVO

O que os outros  
estão resolvendo.

### LISTAR

Liste todas as  
suas submissões.

### TENTADO

Problemas ainda  
não resolvidos.

### FAQS

Precisa de ajuda?

### RESPOSTAS

O que isso  
significa?

### FÓRUM

Busque por ajuda  
no Fórum.

## CÓDIGO FONTE

[EDITAR & ENVIAR](#)

VISUALIZE O CÓDIGO FONTE DE SUAS SUBMISSÕES, JUNTO COM ALGUNS DETALHES EXTRAS.

### SUBMISSÃO # 42895688

PROBLEMA: 1024 - Criptografia  
RESPOSTA: **Accepted**  
LINGUAGEM: Python 3.9 (Python 3.9.4) [+1s]  
TEMPO: 1.207s  
TAMANHO: 2,5 KB  
MEMÓRIA: -  
SUBMISSÃO: 18/12/2024 08:26:32

### CÓDIGO FONTE

```
1 # -*- coding: utf-8 -*-
2
3 from sys import stdin
4
5
6 def apply_first_step(digit_str):
7     if not digit_str.isalpha():
8         return digit_str
9     new_digit = ord(digit_str) + 3
10    return chr(new_digit)
11
12
13 def apply_third_step(digit_str):
14     new_digit = ord(digit_str) - 1
15     return chr(new_digit)
16
17
18 def run_challenge_simple():
19     amount_lines = input()
20     if amount_lines == '':
21         return
22
23     amount_lines = int(amount_lines)
24     if amount_lines <= 0:
25         return
26
27     for index_line in range(amount_lines):
28         line_input_str = input()
29         if line_input_str == '':
30             print('')
31             continue
32
33         line_input_lst = list(line_input_str)
34         for index_char in range(len(line_input_str)):
35             line_input_lst[index_char] = apply_first_step(line_input_lst[index_char])
36
37         line_input_lst.reverse()
38
39         for index_char in range(len(line_input_str) // 2, len(line_input_str)):
40             line_input_lst[index_char] = chr(ord(line_input_lst[index_char]) - 1)
41
42         print(''.join(line_input_lst))
43
44
45 def run_challenge():
46     amount_lines = input()
47     if amount_lines == '':
48         return
49
50     amount_lines = int(amount_lines)
51     if amount_lines <= 0:
52         return
53
54     for index_line in range(amount_lines):
55         line_input_str = input()
56         if line_input_str == '':
57             print('')
58             continue
59         line_input_lst = list(line_input_str)
60         for original_index in range(0, len(line_input_lst) // 2):
61             inverted_index = len(line_input_lst) - original_index - 1
62
63             line_input_lst[inverted_index] = apply_first_step(line_input_lst[inverted_index])
64             line_input_lst[original_index] = apply_first_step(line_input_lst[original_index])
65
66             line_input_lst[original_index], line_input_lst[inverted_index] = line_input_lst[inverted_index], line_input_lst[original_index]
```

```
69         line_input_lst[inverted_index] = apply_third_step(line_input_lst[inverted_index])
70
71     if (len(line_input_lst)) % 2 != 0:
72         original_index = (len(line_input_lst)) // 2 + (len(line_input_lst)) % 2
73         original_index -= 1
74         line_input_lst[original_index] = apply_first_step(line_input_lst[original_index])
75         line_input_lst[original_index] = apply_third_step(line_input_lst[original_index])
76
77     print(''.join(line_input_lst))
78
79
80 if __name__ == '__main__':
81     run_challenge()
82
```