



67%

ATIVIDADES  
3 de 8

DISCORD  
ALURA

FÓRUM DO  
CURSO

VOLTAR  
PARA  
DASHBOARD

MODO  
NOTURNO



17.1k xp

a

03

## Implementando a troca de letras

### Transcrição

Para resolvermos o problema de digitação anterior, criaremos a função `troca_letra()`, que substituirá uma letra equivocada pela correta.

No caso do termo "lógica" escrito erroneamente, nosso algoritmo precisará pegar a letra “i” e substituir por “o”, corrigindo para "lógica".

A nova função receberá as `fatias` e, como já comentamos, será a junção das funções `insere_letras()` e `deletando_caracteres()` criadas anteriormente.

Então poderemos utilizar as funções que já implementamos para escrevermos a `troca_letra()`; copiaremos o corpo de `insere_letras()` e colaremos dentro da nova função para fazermos as adaptações cabíveis.

Primeiro, precisaremos deletar a primeira letra de index `0` da segunda parte da palavra `D`, então pegaremos do index `1` em diante como fizemos em `deletando_caracteres()`.

```
def troca_letra(fatias):  
    novas_palavras = []  
    letras = 'abcdefghijklmnopqrstuvwxyzáâãäåèéêëìíîóôõùúûç'
```



67%

ATIVIDADES  
3 de 8

DISCORD  
ALURA

FÓRUM DO  
CURSO

VOLTAR  
PARA  
DASHBOARD

MODOS  
NOTURNOS



17.1k xp

a

```
for E, D in fatias:
    for letra in letras:
        novas_palavras.append(E + letra + D[1:])
return novas_palavras
```

COPIAR CÓDIGO

Com isso, teremos a operação de `troca_letra()` implementada.

Antes de prosseguirmos, melhoraremos nosso código, pois estas operações, o gerador de palavras e o avaliador estão muito distantes entre si. Então os colocaremos em uma única célula para facilitar as modificações.

Em seguida, precisaremos chamar a nova função `troca_letra()` recebendo as `fatias` dentro de `gerador_palavras()`.

Ao final, teremos a `palavra_exemplo` sendo igual a "lógica" e imprimiremos as `palavras_geradas`.

```
def insere_letras(fatias):
    novas_palavras = []
    letras = 'abcdefghijklmnopqrstuvwxyzàáâãèéêîíîðóôõùúç'
    for E, D in fatias:
        for letra in letras:
            novas_palavras.append(E + letra + D)
    return novas_palavras
```

```
def deletando_caracteres(fatias):
    novas_palavras = []
```



67%

ATIVIDADES  
3 de 8

DISCORD  
ALURA

FÓRUM DO  
CURSO

VOLTAR  
PARA  
DASHBOARD

MODO  
NOTURNO



17.1k xp



```
for E, D in fatias:
    novas_palavras.append(E + D[1:])
return novas_palavras
```

```
def troca_letra(fatias):
    novas_palavras = []
    letras = 'abcdefghijklmnopqrstuvwxyzáâãäåêëèìíóôõöùûüç'
    for E, D in fatias:
        for letra in letras:
            novas_palavras.append(E + letra + D[1:])
    return novas_palavras
```

```
def gerador_palavras(palavra):
    fatias = []
    for i in range(len(palavra)+1):
        fatias.append((palavra[:i],palavra[i:]))
    palavras_geradas = insere_letras(fatias)
    palavras_geradas += deletando_caracteres(fatias)
    palavras_geradas += troca_letra(fatias)
    return palavras_geradas
```

```
palavra_exemplo = "lógica"
palavras_geradas = gerador_palavras(palavra_exemplo)
print(palavras_geradas)
```

COPIAR CÓDIGO



67%

ATIVIDADES  
3 de 8

DISCORD  
ALURA

FÓRUM DO  
CURSO

VOLTAR  
PARA  
DASHBOARD

MODOS  
NOTURNO



17.1k xp

a

Rodando esta célula, teremos todas as as palavras geradas pelos algoritmos, inclusive pela troca de letras, e dentre elas a correta "lógica".

Portanto, a função `troca_letra()` está funcionando conforme o esperado.

Mas ainda existe mais um tipo de erro para corrigirmos, e o veremos a seguir para podermos testar o algoritmo final do corretor.