

SCC0124 – Introdução à Programação para Engenharias

1. Faça um programa que receba uma frase e retorne quais são os dois últimos caracteres.
2. Faça um programa que receba uma letra e uma frase. Após isso, o programa retornará quantas vezes a letra informada aparece na frase.

3. Faça uma função que retorne quantas vogais tem uma string. Exemplo de uso da função:

```
palavra = input('Digite uma palavra: ')  
n = quantasVogais(palavra)  
print ( 'Tem', n, vogais.)
```

4. Faça uma função que retorne quantas consoantes tem uma string. Assuma que a palavra contém apenas letras minúsculas. Exemplo de uso da função:

```
frase = input('Digite uma frase: ')  
n = quantasConsoantes(frase)  
print ( 'Tem', n, 'consoantes.')
```

5. Repita o exercício 4, mas agora considere que a palavra pode conter numerais também. Exemplo de uso da função:

```
frase = input('Digite uma frase: ')  
n = quantasConsoantes(frase)  
print ( 'Tem', n, 'consoantes.')
```

6. Uma técnica usada para comparar strings de forma a ignorar diferenças de maiúscula ou minúscula é a seguinte (teste o código no PyCharm):

```
texto1 = 'AbCdE'
texto2 = 'aBcDe'
if (texto1==texto2):
    print ('Textos sao iguais! ')
else:
    print ('Textos sao diferentes! ')

if (texto1.upper()==texto2.upper()):
    print ('Ignorando maiusculas/minusculas, sao iguais! ')
else:
    print ('Textos sao diferentes! ')
```

A função do **upper()** retorna a string com todos os caracteres em maiúsculo. Utilizando essa técnica, repita o exercício 4, mas agora considerando que podem haver letras maiúsculas e minúsculas na string.

7. Corrija os programas abaixo para obter a saída desejada:

a)

```
palavra = input('Digite uma palavra: ')
x = int(palavra)

print ('A palavra tem', x, 'caracteres. ')
```

Saída obtida:

Digite uma palavra:teste

Traceback (most recent call last):

ValueError: invalid literal for int() with base 10: 'teste'

Process finished with exit code 1

Saída desejada:

Digite uma palavra:teste

A palavra tem 5 caracteres.

b)

```
palavra = input('Digite uma palavra: ')  
  
for i in range(0, len(palavra)):  
    if (palavra[i] == '0'):  
        palavra = palavra[0:i] + 'Z' + palavra[i:len(palavra)]  
  
print (palavra)
```

Saída obtida:

Digite uma palavra:abc0def0ghi

abcZZZZZZZZ0def0ghi

Saída desejada (troca todo '0' por 'Z'):

Digite uma palavra:abc0def0ghi

abcZdefZghi

c)

```
palavra = input('Digite uma palavra: ')  
  
for i in range(0, len(palavra)):  
    if (palavra[i] == '*'):  
        palavra = palavra[0:i] + '***' + palavra[i+1:len(palavra)]  
  
print (palavra)
```

Saída obtida:

Digite uma palavra:qwe*qwe

qwe*****qwe

Saída desejada (troca todo '*' por '**'):

Digite uma palavra:qwe*qwe

qwe**qwe

d)

```
palavra = input('Digite uma palavra: ')
temp = ''
for i in range(0, len(palavra)):
    if (palavra[i] == '*'):
        temp = temp + '**'
    else:
        temp = temp + palavra[i]

print (temp)
```

Saída obtida:

```
Digite uma palavra:qwe*qwe*
qqq**qqq**
```

Saída desejada (troca todo '*' por ''):**

```
Digite uma palavra:qwe*qwe*
qwe**qwe**
```

8. Faça uma função que receba uma string e retire todos os caracteres que não são vogais. Por exemplo, a frase 'a prova eh facil!' ficaria 'aoaeai'. Exemplo de uso da função:

```
frase = input('Digite uma frase:')
fraseNova = removerNaoVogais(frase)
print ( fraseNova)
```

9. Faça uma função que retorne True se a string for um palíndromo e False caso contrário. Palíndromos são palavras que não se alteram quando lidas de traz para frente. Exemplos: ovo, mirim, arara...