

# Programação I

## Strings (ficha 9) *v2.0*

1. Avalie as expressões seguintes mentalmente e confirme no interpretador de Python.

- (a) `s="Jogos Olimpicos"; s[0]`
- (b) `s="teste"; len(s)`
- (c) `s='universidade'; s[3:]`
- (d) `s='universidade'; s[3:-1]`
- (e) `s='universidade'; s[1::2]`
- (f) `s='universidade'; len( s[:4] )`

2. Confirme que não é possível modificar um carácter de uma string.

```
s='universidade'; s[0]='U'  
# espera-se um erro!
```

3. Escreva a função `letras(str)` que recebe uma string como argumento e escreve uma letra por linha. Implemente uma versão com ciclo `while` e outra com ciclo `for`.

```
print(letras('Aula'))  
A  
u  
l  
a
```

4. Escreva a função `letras_inv(str)` que recebe uma string como argumento e escreve as letras por ordem inversa (do fim para o início).

```
print(letras_inv('Aula'))  
a  
l  
u  
A
```

5. Escreva a função `escala_in(str)` que imprime uma string numa escala invertida.

```
print(escala_in('Aula'))  
Aula  
Aul  
Au  
A
```

6. Escreva uma função `espelho(str)` que recebe uma palavra e que imprime a palavra e o reverso da mesma juntas.

```
print(espelho('ab'))
abba
print(espelho('123'))
12321
```

7. Um palíndromo é uma palavra que é escrita da mesma forma da esquerda para a direita ou da direita para a esquerda; por exemplo "aia" e "noon". Escreva a função `palindromo(str)` que recebe como argumento uma string e devolve `True` se for um palíndromo e `False` de outro modo.

```
print(palindromo('reviver'))
True
```

8. Implemente as seguintes funções sobre strings:

- (a) `vogal(str)`: verifica se o argumento corresponde a uma letra que seja vogal. Devolve `True` ou `False`.
- (b) `conta_vogais(str)`: devolve o número de vogais existentes na string recebida como argumento. Utilize a função definida no ponto anterior.

9. Implemente a função `ocorrencias(letra, string)` que conta o nº de vezes que `letra` aparece em `string`.

```
print(ocorrencias('a', 'aula'))
2
```

10. Implemente a função `ocorrencias(letra, string, indice)` que conta o nº de vezes que `letra` aparece na `string`. `indice` indica o índice a partir do qual a contagem deve ser feita.

```
print(ocorrencias('u', 'aula', 2))
0
```

11. Implemente a função `ocorrencias_case(letra, string, indice)` que conta o nº de vezes que `letra` aparece na `string` a partir de `indice`. A contagem não deve fazer distinção entre maiúsculas e minúsculas

```
print(ocorrencias_case('a', 'Aula', 0))
2
```

12. Crie uma função que tem como parâmetro uma frase (*string*) e conta o número de palavras dessa frase. Considere que as palavras estão separadas por espaços ou vírgulas.

```
print(conta_palavras('computador, caderno e caneta'))
4
print(conta_palavras('  ,  '))
0
```

13. Implemente a função `rot13(str)` que recebe uma string e aplica a codificação ROT13 (ver slides da aula teórica). Reaplicando a função à string codificada obterá a string original. Utilize as funções `chr()` e `ord()`.