



ALGORITMOS COMPUTACIONAIS – Revisão

1. Busque e corrija o erro de sintaxe do código abaixo para exibir cada elemento da lista v em uma linha.

```
V = [7, 1, 5, 4]
for a in v
    print(a)
```

2. Busque e corrija o erro de sintaxe no código

```
while c <= 3:
    print(c)
    c = c + 1
```

para obter a seguinte saída:

```
1
2
3
```

3. Conserte o código abaixo:

```
v = []
for b in v:
    print(b)
```

A saída esperada é:

```
1
2
3
```

4. Escreva um trecho de código usando a iteração com for junto com a função range() para gerar a seguinte saída:

```
14
15
```

16

17

5. Faça um programa que leia uma *string* *s* e teste se ela é um palíndromo (ou seja, se é igual de trás para frente), imprimindo sim ou não. Dica: compare com uma *string* reversa *r*, criada usando `for` e o operador `+`. Teste com a entrada rodador

6. Conserte o erro de lógica no código

```
c = 0
while c >= 2:
    print(c)
    c = c + 1
```

para obter a seguinte saída:

```
0
1
2
```

7. Conserte o erro de lógica no código

```
c = 5
while c <= 3:
    print(c)
    c = c - 1
```

para obter a seguinte saída:

```
5
4
3
```

8. Escreva um trecho de código que define a variável *c* como contador. Use ela para gerar a seguinte saída:

```
c agora é 2
c agora é 4
c agora é 6
c agora é 8
```

9. Escreva um trecho de código que define a variável **c** como contador. Gere a seguinte saída, usando dois laços em sequência:

```
1
2
3
4
3
2
1
```

10. Escreva um trecho de código que define a variável **a** como acumulador e **c** como contador. Use ambas para calcular a soma dos inteiros de 0 a 100, que deve ser impressa apenas no final.

11. Escreva um trecho de código que define a variável **soma** como acumulador e **c** como contador. Use ambas para calcular e imprimir ao final a soma dos múltiplos de 3 entre 3 e 99. Use um **if** e o operador **%** para testar se **c** é múltiplo de 3.

12. Escreva um trecho de código que define a variável **soma** como acumulador e **c** como contador. Use ambas para calcular e imprimir ao final a soma da sequência 3, 6, 9, 12, ... 99. Não use um **if**, mas sim um passo maior para **c**.

13. Escreva um trecho de código que conte na variável quantos quantos são os números múltiplos de 7 que também são pares no intervalo de 1 a 1000. Imprima esta quantidade ao final.

14. Escreva um trecho de código que solicite um número inteiro **n**. Apenas se **n** for maior do que **zero**, imprima os números de **1** a **n** (inclusive). Teste com a entrada 4.

15. Faça um programa que repita até encontrar o valor zero, sempre lendo um valor inteiro para a variável **v** dentro do laço de repetição. Se o valor lido for par, imprima também ele na saída. Teste com a sequência 8, 3, 2, 0.

16. Faça um programa que repita lendo valores inteiros, colocados na variável **v**. Ele deve parar ao encontrar um valor zero. Use um acumulador produto para calcular o produto de todos os valores digitados (menos o último valor zero). Apresente o produto ao final. Teste com a sequência 8, -4, 2, 0.