



Algoritmos e Programação de Computadores

Revisão: Prova 1

Profa. Sandra Avila

Instituto de Computação (IC/Unicamp)

MC102, 5 Maio, 2020

Agenda

- Muitos exercícios =)

Conteúdo da Prova 1

- Variáveis & Tipos (`int`, `float`, `str`, `bool`)
- Operações & Expressões aritméticas, relacionais, lógicas
- Comandos condicionais (`if-elif-else`)
- Comandos repetitivos (`for`, `while`)
- Listas
- Strings

6. Depois das atribuições a seguir, quais são os valores de x e y ?

```
x = 15  
y = x  
x = 22
```

- a. $x = 15$ e $y = 15$
- b. $x = 22$ e $y = 22$
- c. $x = 15$ e $y = 22$
- d. $x = 22$ e $y = 15$

6. Depois das atribuições a seguir, quais são os valores de x e y?

```
x = 15  
y = x  
x = 22
```

- a. $x = 15$ e $y = 15$
- b. $x = 22$ e $y = 22$
- c. $x = 15$ e $y = 22$
- d. $x = 22$ e $y = 15$**

7. Depois das atribuições a seguir, qual é o valor de a?

```
d = 3
c = 2
b = 4
d = c + b
a = a + 1
print(a)
```

- a. 1
- b. 7
- c. 5
- d. Nada é impresso, ocorre um erro.

7. Depois das atribuições a seguir, qual é o valor de a?

```
d = 3
c = 2
b = 4
d = c + b
a = a + 1
print(a)
```

a. 1

b. 7

c. 5

d. Nada é impresso, ocorre um erro.

8. O que o seguinte código imprime?

```
if (4 + 5 == 10):  
    print("Verdadeiro")  
else:  
    print("Falso")
```

- a. Verdadeiro
- b. Falso
- c. Verdadeiro
Falso
- d. Nada.

8. O que o seguinte código imprime?

```
if (4 + 5 == 10):  
    print("Verdadeiro")  
else:  
    print("Falso")
```

- a. Verdadeiro
- b. Falso**
- c. Verdadeiro
Falso
- d. Nada.

9. O que o seguinte código imprime?

```
if (4 + 5 == 10):  
    print("Verdadeiro")  
else:  
    print("Falso")  
print("Verdadeiro")
```

- a. Verdadeiro
- b. Verdadeiro
Falso
- c. Falso
- d. Falso
Verdadeiro

9. O que o seguinte código imprime?

```
if (4 + 5 == 10):  
    print("Verdadeiro")  
else:  
    print("Falso")  
print("Verdadeiro")
```

- a. Verdadeiro
- b. Verdadeiro
Falso
- c. Falso
- d. Falso**
Verdadeiro

10. O que o seguinte código imprimirá se $x = 3$, $y = 5$ e $z = 2$?

```
if x < y and x < z:  
    print("a")  
elif y < x and y < z:  
    print("b")  
else:  
    print("c")
```

- a. a
- b. b
- c. c
- d. Nada.

10. O que o seguinte código imprimirá se $x = 3$, $y = 5$ e $z = 2$?

```
if x < y and x < z:  
    print("a")  
elif y < x and y < z:  
    print("b")  
else:  
    print("c")
```

- a. a
- b. b
- c. c**
- d. Nada.

11. O que é impresso pelos seguintes comandos?

```
a = "I"  
b = "love"  
c = "Python"  
print(a+b+c)
```

- a. I love Python
- b. Python
- c. IlovePython
- d. Erro, não é possível concatenar três strings.

11. O que é impresso pelos seguintes comandos?

```
a = "I"  
b = "love"  
c = "Python"  
print(a+b+c)
```

- a. I love Python
- b. Python
- c. IlovePython**
- d. Erro, não é possível concatenar três strings.

12. O que o seguinte código imprime?

```
a = "python"  
b = "!"  
print(a+b*3)
```

- a. python!python!python!
- b. pythonpythonpython!
- c. python!!!
- d. Erro, não é possível concatenar e multiplicar strings ao mesmo tempo.

12. O que o seguinte código imprime?

```
a = "python"  
b = "!"  
print(a+b*3)
```

- a. python!python!python!
- b. pythonpythonpython!
- c. **python!!!**
- d. Erro, não é possível concatenar e multiplicar strings ao mesmo tempo.

13. Quantas vezes a palavra MC102 é impressa pelos seguintes comandos?

```
frase = "Vou arrasar na prova!"  
for f in frase[3:8]:  
    print("MC102")
```

- a. 4
- b. 5
- c. 6
- d. 7

13. Quantas vezes a palavra MC102 é impressa pelos seguintes comandos?

```
frase = "Vou arrasar na prova!"  
for f in frase[3:8]:  
    print("MC102")
```

- a. 4
- b. 5**
- c. 6
- d. 7

14. Quantas vezes a letra a é impressa pelos seguintes comandos?

```
frase = "Vou arrasar na prova!"  
for i in range(len(frase)):  
    if i % 2 == 0:  
        print(frase[i])
```

- a. 0
- b. 1
- c. 2
- d. 3

14. Quantas vezes a letra a é impressa pelos seguintes comandos?

```
frase = "Vou arrasar na prova!"  
for i in range(len(frase)):  
    if i % 2 == 0:  
        print(frase[i])
```

- a. 0
- b. 1**
- c. 2
- d. 3

15. O que o seguinte código imprime?

```
frase = "Vou arrasar na prova!"  
print(frase.count("r") + frase.count("v"))
```

- a. 4
- b. 5
- c. 6
- d. 7

15. O que o seguinte código imprime?

```
frase = "Vou arrasar na prova!"  
print(frase.count("r") + frase.count("v"))
```

- a. 4
- b. 5**
- c. 6
- d. 7

16. O que é impresso pelo trecho de código a seguir?

```
uma_lista = [3, 67, "gato", [56, 57, "gato"], [ ], 3.14, False]
print(len(uma_lista))
```

- a. 6
- b. 7
- c. 8
- d. 9

16. O que é impresso pelo trecho de código a seguir?

```
uma_lista = [3, 67, "gato", [56, 57, "gato"], [ ], 3.14, False]
print(len(uma_lista))
```

- a. 6
- b. 7**
- c. 8
- d. 9

Questão 2

Prova 2018/1

Exercício: Questão 2 (Prova 2018/1)

Para cada um dos seguintes trechos de código, indique o que será exibido pelo programa:

```
n = 10 - 5 * 3
c = 9 > n
print(n)
```

-5

```
n = 10 - 5
c = 9 > n
print(c)
```

True

```
n = 10 // 5
c = 9 > n
print(n, c)
```

2 True

```
n = 4**2
c = 9 > n
print("not c =", not c)
```

not c = True

```
n = 5 - 10
c1 = -10 > n
c2 = -15 > n
r1 = c1 or c2
r2 = c1 and c2
print(c1, c2)
print(r1, r2)
```

**False False
False False**

```
n = 10 % 2
c1 = 9 > n
c2 = 0 > n
r1 = not c1 and c2
r2 = c1 or not c2
print(c1, c2)
print(r1, r2)
```

**True False
False True**

```
n = 2 + 3 * 5
r1 = not not True
r2 = not (10 < n or n == 17)
c1 = 20 < n or 20 > n and n > 10
c2 = n > 0 or 12 <= n and monty
print(n, r1, r2)
print(c1, c2)
```

**17 True False
True True**

Questão 4

Prova 2018/1

Exercício: Questão 4 (Prova 2018/1)

Mário quer escrever um programa que imprime todos os números pares entre 1 e 11, um por linha. A saída esperada por Mário está indicada na tabela ao lado. Ele se lembra que em Python o uso do operador % retorna o resto da divisão de um número por outro. Por exemplo, $5 \% 2$ resulta em 1. Mário escreveu o programa abaixo. O que será impresso quando o código for executado?

```
n = 1
while (n < 11):
    if (n % 2) == 1:
        print(n)
```

| |
|----|
| 2 |
| 4 |
| 6 |
| 8 |
| 10 |

1
1
1
1
...

O programa entrou em loop infinito.

Exercício: Questão 4 (Prova 2018/1)

Sem usar laços (`for` ou `while`), escreva um programa que produza a saída esperada por Mário utilizando um único comando `print`.

```
print("2\n4\n6\n8\n10")
```

```
2
4
6
8
10
```

Exercício: Questão 4 (Prova 2018/1)

Escreva um programa que produza a saída esperada por Mário imprimindo um número a cada iteração em um laço `for`.

```
for i in range(2,11,2):  
    print(i)
```

Considere que Mário quer imprimir os números múltiplos de 3 de 1 a 11, em ordem decrescente, um por linha. Escreva um programa que produza esta saída imprimindo um número a cada iteração em um laço `while`.

```
n = 9  
while (n > 1):  
    print(n)  
    n = n - 3
```

| |
|----|
| 2 |
| 4 |
| 6 |
| 8 |
| 10 |

Questão 5

Prova 2018/1

Exercício: Questão 5 (Prova 2018/1)

João codificou o programa abaixo. Mário, que é amigo de João, disse que o programa não iria compilar, já que a lista declarada na linha 1 é composta por inteiros e strings. Apresente uma justificativa se você concorda com Mário ou indique qual seria a saída do programa.

```
1. lista = [0, -1, "Pedro", "Maria"]
2. tamanho = len(lista)
3. for i in range(tamanho):
4.     print(lista[i])
```

```
0
-1
Pedro
Maria
```

Exercício: Questão 5 (Prova 2018/1)

Mário escreveu uma variação do programa codificado por João. Indique o que será exibido na tela quando o programa for executado.

```
1. lista = ["Mário", "Mároi", "Mário", "Maria"]
2. for a in lista:
3.     if (a == "Mário"):
4.         for n in range(5,7):
5.             print(n)
6. print("Fim")
```

```
5
6
5
6
Fim
```

Exercício: Questão 5 (Prova 2018/1)

Reescreva a linha 4, de forma que o programa exiba em ordem crescente todos os números inteiros de 89 (inclusive) a 100 (inclusive).

```
1. lista = ["Mário", "Mároi", "Mário", "Maria"]
2. for a in lista:
3.     if (a == "Mário"):
4.         for n in range(5,7):
5.             print(n)
6. print("Fim")
```

| | |
|----------|--------------------------------|
| Linha 4: | for n in range(89,101): |
|----------|--------------------------------|

Exercício: Questão 5 (Prova 2018/1)

Reescreva a linha 4, de forma que o programa exiba em ordem crescente todos os números ímpares de 89 (inclusive) a 99 (inclusive).

```
1. lista = ["Mário", "Mároi", "Mário", "Maria"]
2. for a in lista:
3.     if (a == "Mário"):
4.         for n in range(5,7):
5.             print(n)
6. print("Fim")
```

| | |
|----------|----------------------------------|
| Linha 4: | for n in range(89,100,2): |
|----------|----------------------------------|

Exercício: Crime

- Utilizando listas, faça um programa que faça 5 perguntas para uma pessoa sobre um crime. As perguntas são:
 - “Telefonou para a vítima?”
 - “Esteve no local do crime?”
 - “Mora perto da vítima?”
 - “Devia para a vítima?”
 - “Já trabalhou com a vítima?”
- Se a pessoa responder positivamente a 2 questões ela deve ser classificada como “Suspeita”, entre 3 e 4 como “Cúmplice” e 5 como “Assassino”. Caso contrário, ele será classificado como “Inocente”.

```
res = []
res.append(input("Telefonou para a vítima? 1/Sim ou 0/Não: "))
res.append(input("Esteve no local do crime? 1/Sim ou 0/Não: "))
res.append(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
res.append(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
res.append(input("Já trabalhou com a vítima? 1/Sim ou 0/Não: "))
```

```
res = []
res.append(input("Telefonou para a vítima? 1/Sim ou 0/Não: "))
res.append(input("Esteve no local do crime? 1/Sim ou 0/Não: "))
res.append(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
res.append(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
res.append(input("Já trabalhou com a vítima? 1/Sim ou 0/Não: "))

soma_respostas = 0
for i in res: # soma o número de respostas
    soma_respostas = soma_respostas + int(i)
```

```
res = []
res.append(input("Telefonou para a vítima? 1/Sim ou 0/Não: "))
res.append(input("Esteve no local do crime? 1/Sim ou 0/Não: "))
res.append(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
res.append(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
res.append(input("Já trabalhou com a vítima? 1/Sim ou 0/Não: "))

soma_respostas = 0
for i in res: # soma o número de respostas
    soma_respostas = soma_respostas + int(i)

if (soma_respostas < 2):
    print("\nInocente")
elif (soma_respostas == 2):
    print("\nSuspeita")
elif (3 <= soma_respostas <= 4):
    print("\nCúmplice")
elif (soma_respostas == 5):
    print("\nAssassino")
```

```
lista_perguntas = ["Telefonou para a vítima? 1/Sim ou 0/Não: ",  
                  "Esteve no local do crime? 1/Sim ou 0/Não: ",  
                  "Mora perto da vítima? 1/Sim ou 0/Não: ",  
                  "Devia para a vítima? 1/Sim ou 0/Não: ",  
                  "Já trabalhou com a vítima? 1/Sim ou 0/Não: "]
```

```
lista_perguntas = ["Telefonou para a vítima? 1/Sim ou 0/Não: ",
                  "Esteve no local do crime? 1/Sim ou 0/Não: ",
                  "Mora perto da vítima? 1/Sim ou 0/Não: ",
                  "Devia para a vítima? 1/Sim ou 0/Não: ",
                  "Já trabalhou com a vítima? 1/Sim ou 0/Não: "]

res = []
soma_respostas = 0
for i in range(len(lista_perguntas)):
    print(lista_perguntas[i])
    res.append(input()) # adiciona as respostas na lista res
    soma_respostas += int(res[i]) # soma o número de respostas
```

```
lista_perguntas = ["Telefonou para a vítima? 1/Sim ou 0/Não: ",
                  "Esteve no local do crime? 1/Sim ou 0/Não: ",
                  "Mora perto da vítima? 1/Sim ou 0/Não: ",
                  "Devia para a vítima? 1/Sim ou 0/Não: ",
                  "Já trabalhou com a vítima? 1/Sim ou 0/Não: "]

res = []
soma_respostas = 0
for i in range(len(lista_perguntas)):
    print(lista_perguntas[i])
    res.append(input()) # adiciona as respostas na lista res
    soma_respostas += int(res[i]) # soma o número de respostas

status = ["Inocente", "Suspeita", "Cúmplice", "Cúmplice", "Assassino"]
if soma_respostas < 2:
    print(status[0])
else:
    print(status[soma_respostas-1])
```

Essa solução não utiliza listas

```
res1 = int(input("Telefonou para a vítima? 1/Sim ou 0/Não: "))
```

```
res2 = int(input("Esteve no local do crime? 1/Sim ou 0/Não: "))
```

```
res3 = int(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
```

```
res4 = int(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
```

```
res5 = int(input("Já trabalhou com a vítima? 1/Sim ou 0/Não: "))
```

```
# Essa solução não utiliza listas
```

```
res1 = int(input("Telefonou para a vítima? 1/Sim ou 0/Não: "))
```

```
res2 = int(input("Esteve no local do crime? 1/Sim ou 0/Não: "))
```

```
res3 = int(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
```

```
res4 = int(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
```

```
res5 = int(input("Já trabalhou com a vítima? 1/Sim ou 0/Não: "))
```

```
# soma o número de respostas
```

```
soma_respostas = res1 + res2 + res3 + res4 + res5
```

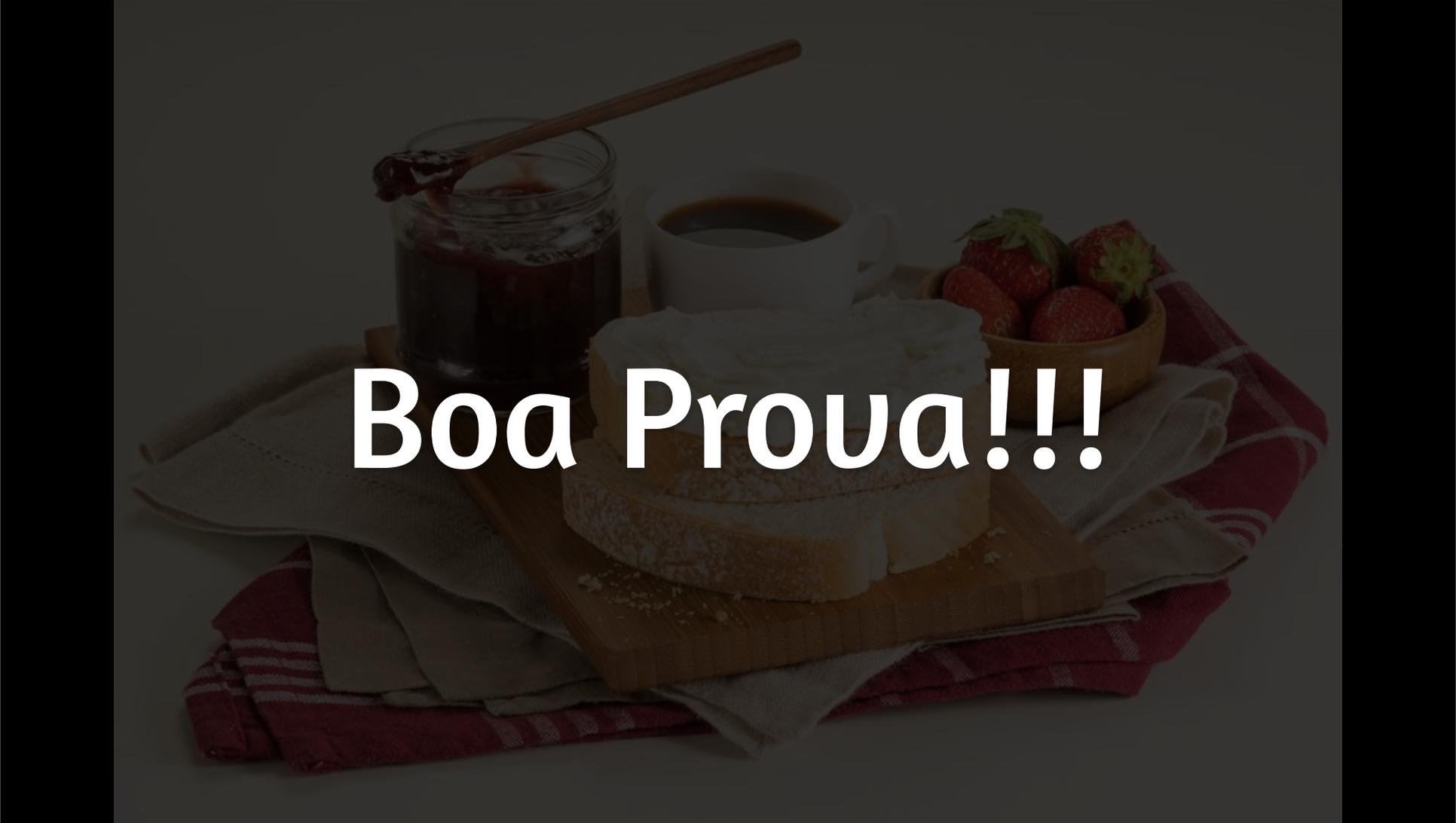
```
# Essa solução não utiliza listas
res1 = int(input("Telefonou para a vítima? 1/Sim ou 0/Não: "))
res2 = int(input("Esteve no local do crime? 1/Sim ou 0/Não: "))
res3 = int(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
res4 = int(input("Mora perto da vítima? 1/Sim ou 0/Não: "))
res5 = int(input("Já trabalhou com a vítima? 1/Sim ou 0/Não: "))

# soma o número de respostas
soma_respostas = res1 + res2 + res3 + res4 + res5

if (soma_respostas < 2):
    print("\nInocente")
elif (soma_respostas == 2):
    print("\nSuspeita")
elif (3 <= soma_respostas <= 4):
    print("\nCúmplice")
elif (soma_respostas == 5):
    print("\nAssassino")
```

Referências & Exercícios

- <https://wiki.python.org.br/EstruturaSequencial>: 18 exercícios
 - <https://wiki.python.org.br/EstruturaDeDecisao>: 28 exercícios
 - <https://wiki.python.org.br/EstruturaDeRepeticao>: 51 exercícios
 - <https://wiki.python.org.br/ExerciciosComStrings>: 14 exercícios
 - <https://wiki.python.org.br/ExerciciosListas>: 24 exercícios
- 135 exercícios**



Boa Prova!!!