

Processando a Informação: um livro prático de programação independente de linguagem

Rogério Perino de Oliveira Neves

Francisco de Assis Zampirolli

EDUFABC

editora.ufabc.edu.br

Notas de Aulas inspiradas no livro

Utilizando a(s) Linguagem(ns) de Programação:

C

Exemplos adaptados para Correção Automática no Moodle+VPL

Francisco de Assis Zampirolli

3 de setembro de 2022

Sumário

0.1 Processando a Informação: Cap. 4: Estruturas de Repetição (Laços) - Prática 2	2
0.1.1 Exercícios	2
0.1.2 Formatações avançadas com <code>format</code>	5

0.1 Processando a Informação: Cap. 4: Estruturas de Repetição (Laços) - Prática 2



Este caderno (Notebook) é parte complementar *online* do livro **Processando a Informação: um livro prático de programação independente de linguagem**, que deve ser consultado no caso de dúvidas sobre os temas apresentados.

Este conteúdo pode ser copiado e alterado livremente e foi inspirado nesse livro.

0.1.1 Exercícios

Fontes: [ref1](#); [ref2](#); [ref3](#); [ref4](#)

-
- Desenvolver um algoritmo que leia a altura de 15 pessoas. Este programa deverá calcular e mostrar :

- A menor altura do grupo;
- A maior altura do grupo;

[]: `# escreva o seu código`

-
- Escrever um algoritmo que leia uma quantidade desconhecida de números e conte quantos deles estão em cada um dos seguintes intervalos: [0-25], [26-50], [51-75] e [76-100]. A entrada de dados deve terminar quando for lido um número negativo.

[]: *# escreva o seu código*

3. Escrever um algoritmo que gera e escreve os números ímpares entre 100 e 200. Utilizar o método definido da lista anterior (Prática 1).

[]: *# escreva o seu código*

4. Escreva um algoritmo que leia um valor inicial a_1 e uma razão r e imprima uma sequência em P.A. contendo 10 valores. Relembrando:

$$a_n = a_1 + (n - 1)r$$

[]: *# escreva o seu código*

5. Escreva um algoritmo que leia um valor inicial a_1 e uma razão q e imprima uma sequência em P.G. contendo 10 valores. Relembrando:

$$a_n = a_1 q^{n-1}$$

[]: *# escreva o seu código*

6. Escreva um algoritmo que leia um valor inicial A e imprima a sequência de valores do cálculo de A! e o seu resultado. Formatar a saída exatamente como no exemplo:

$$5! = 5X4X3X2X1 = 120$$

[]: *# escreva o seu código*

7. Foi feita uma pesquisa entre os habitantes de uma região e coletados os dados de altura e sexo (0=masc, 1=fem) das pessoas. Faça um programa que leia 50 dados diferentes e informe:

- a maior e a menor altura encontradas
- a média de altura das mulheres
- a média de altura da população
- o percentual de homens na população

[]: *# escreva o seu código*

8. Rafa tem 1,50 metro e cresce 2 centímetros por ano, enquanto Zé tem 1,10 metro e cresce 3 centímetros por ano. Construa um algoritmo que calcule e imprima quantos anos serão necessários para que Zé seja maior que Rafa.

[]: `# escreva o seu código`

9. Em uma eleição presidencial existem quatro candidatos. Os votos são informados através de códigos. Os dados utilizados para a contagem dos votos obedecem à seguinte codificação:

- 1,2,3,4 = voto para os respectivos candidatos;
- 5 = voto nulo;
- 6 = voto em branco;

Elabore um algoritmo que leia o código do candidato em um voto. Calcule e escreva:

- total de votos para cada candidato;
- total de votos nulos;
- total de votos em branco;

Como finalizador do conjunto de votos, tem-se o valor 0.

[]: `# escreva o seu código`

9. Em uma eleição presidencial existem quatro candidatos. Os votos são informados através de códigos. Os dados utilizados para a contagem dos votos obedecem à seguinte codificação:

- 1,2,3,4 = voto para os respectivos candidatos;
- 5 = voto nulo;
- 6 = voto em branco;

Elabore um algoritmo que leia o código do candidato em um voto. Calcule e escreva:

- total de votos para cada candidato;
- total de votos nulos;
- total de votos em branco;

Como finalizador do conjunto de votos, tem-se o valor 0.

[]: `# escreva o seu código`

10. Faça um programa que desenhe na tela losangos ou triângulos utilizando somente o caractere “%” (veja exemplos abaixo). O usuário é quem escolhe o que deve ser

impresso. O usuário também deve ter a opção de escolher o tamanho (em linhas) da figura a ser desenhada.

```

      %
    %%
  %%%
 %%%%
 %%%%%
 %%%%%%
 %%%%%%
  %%%%
    %%
      %

```

```

      %
    %%
  %%%
 %%%%
 %%%%%
 %%%%%%
 %%%%%%
  %%%%
    %%
      %

```

[]: *# escreva o seu código*

-
11. Construa um algoritmo para o jogo da velha. Esse jogo consiste em um tabuleiro de dimensão 3x3 de valores O ou X. Os usuários devem informar a linha e a coluna que desejam preencher. A partir da terceira jogada de cada jogador é necessário verificar se houve algum ganhador. Também é possível que o resultado do jogo seja empate (nenhum jogador preencheu uma coluna, uma linha ou uma diagonal).

[]: *# escreva o seu código*

-
12. Faça um programa que calcule e mostre o maior divisor comum de dois números a e b, usando o algoritmo básico de Euclides (temp é uma variável inteira, temporária):

```

enquanto b for diferente de 0:
    temp = a
    a = b
    b = temp % b

```

Esse algoritmo deixará o resultado (MDC) em a, no final.

[]: *# escreva o seu código*

0.1.2 Formatações avançadas com format

<https://docs.python.org/3/library/string.html#format-specification-mini-language>

```
[ ]: #format_spec ::=  [[fill]align][sign][#][0][width][,][.precision][type]_
    ↪
```

```
[ ]: print('{0}, {1}, {2}'.format('a', 'b', 'c'))
    print('{} , {} , {}'.format('a', 'b', 'c')) # 3.1+ only
    print('{2}, {1}, {0}'.format('a', 'b', 'c'))
    print('{2}, {1}, {0}'.format(*'abc'))        # unpacking argument_
    ↪sequence
    print('{0}{1}{0}'.format('abra', 'cad'))    # arguments' indices can be_
    ↪repeated
```

```
[ ]: 'Coordinates: {latitude}, {longitude}'.format(latitude='37.24N',_
    ↪longitude='-115.81W')
```

```
[ ]: coord = {'latitude': '37.24N', 'longitude': '-115.81W'}
    'Coordinates: {latitude}, {longitude}'.format(**coord)
```

```
[ ]: c = 3-5j
    ('The complex number {0} is formed from the real part {0.real} and the_
    ↪imaginary part {0.imag}.').format(c)
```

```
[ ]: class Point:
    def __init__(self, x, y):
        self.x, self.y = x, y
    def __str__(self):
        return 'Point({self.x}, {self.y})'.format(self=self)

    str(Point(4, 2))
```

```
[ ]: coord = (3, 5)
    'X: {0[0]}; Y: {0[1]}'.format(coord)
```

```
[ ]: "repr() shows quotes: {!r}; str() doesn't: {!s}".format('test1',_
    ↪'test2')
```

```
[ ]: '{:<30}'.format('left aligned')
```

```
[ ]: '{:>30}'.format('right aligned')
```

```
[ ]: '{:^30}'.format('centered')
```

```
[ ]: '{:*^30}'.format('centered') # use '*' as a fill char
```

```
[ ]: '{:+f}; {:+f}'.format(3.14, -3.14) # show it always
```

```
[ ]: '{: f}; {: f}'.format(3.14, -3.14)  # show a space for positive numbers

[ ]: '{:-f}; {:-f}'.format(3.14, -3.14)  # show only the minus -- same as '{:
    ↪f}; {:f}'
    '3.140000; -3.140000'

[ ]: "int: {0:d};  hex: {0:x};  oct: {0:o};  bin: {0:b}".format(42)

[ ]: "int: {0:d};  hex: {0:#x};  oct: {0:#o};  bin: {0:#b}".format(42)

[ ]: '{:,}'.format(1234567890)

[ ]: points = 19
    total = 22
    'Correct answers: {:.2%}'.format(points/total)

[ ]: import datetime
    d = datetime.datetime(2010, 7, 4, 12, 15, 58)
    '{:%Y-%m-%d %H:%M:%S}'.format(d)
```