

# {introcomp}

## Working 04 : Listas e loops

### Objetivos:

- Dominar a construção de estruturas de repetição em Python;
- Aperfeiçoar-se na resolução de problemas;

**Prazo de Envio:** sábado, 17/08, 22:00.

# 1 INTRODUÇÃO

Dando continuidade aos conceitos básicos apresentados nas aulas anteriores, neste *Working* iremos praticar o conteúdo relacionado a *listas* e *loops*, a fim de dominar todos os conceitos básicos que permeiam o alicerce da programação de computadores. A atenção e o empenho são indispensáveis para um bom entendimento do conteúdo!

## 2 Aquecimento

Vamos refletir rapidamente sobre algumas implementações de loops e listas. Caso algum dos conceitos aqui não esteja na memória busque o material de estudos novamente. Não é necessário submeter as respostas em lugar algum, apenas utilize como exemplo.

1. Considere o código abaixo:

```

1 cont = 1
3 for i in range(0, 10 ,1):
    if i == 10:
5         break
7     for j in range(0, 20, 1):
        if i >= 9:
9             break
11 cont = cont + 2;
```

- Para que linha do código acima, o comando break da linha 9 desviará o fluxo de execução do programa?
- O comando break da linha 5 é inútil. Explique por quê.
- Qual o valor final dentro da variável cont?

2. Considere os códigos a seguir:

(a)

```

1     for i in range(20, 0 , -1):
2         //Codigos...
```

(b)

```

2     i=0;
3     while i < 100:
4         //Codigos..
5         i = i + 2
```

Transcreva os códigos de cada item por alguma outra estrutura de repetição diferente.

### 3 PRATICANDO

Agora vamos praticar! Para todos os praticando em que é pedido que se escreva um programa, escreva o código do seu programa e nos envie o .py (código fonte) correspondente. Atenção: Não utilize **NENHUM ACENTO** no seu código fonte, nem mesmo em comentários. As ferramentas de correção automática agradecem.

1. Faça um programa em Python que calcula o produto dos números inteiros digitados pelo usuário. O programa deve permitir que o usuário digite uma quantidade não determinada de números que encerre quando o usuário digita 0. Use as estruturas que achar apropriado.

Exemplo de Entrada	Exemplo de Saida
2 5 0	10
2 3 6 3 0	108

2. Faça um programa que lê um número inteiro 'A' a ser adivinhado e depois leia os números inteiros que serão os chutes, preenchendo uma lista. Quando chute for mais alto que A imprima "Chute alto" na tela e quando for mais baixo imprima "Chute baixo". Quando o número chutado for igual imprima "Acertou!" e imprima a lista de números chutados.

Exemplo de Entrada	Exemplo de Saida
42	Chute baixo
12	Chute baixo
30	Chute alto
45	Acertou!
42	[12, 30, 45, 42]

3. Dado um número N inteiro como entrada, exiba a tabuada de multiplicação de N.

Exemplo de Entrada	Exemplo de Saida
1	1 * 1 = 1 1 * 2 = 2 1 * 3 = 3 1 * 4 = 4 1 * 5 = 5 1 * 6 = 6 1 * 7 = 7 1 * 8 = 8 1 * 9 = 9 1 * 10 = 10

4. A conjectura de Lothar Collatz estabelece uma sequência de números naturais. A partir de um número natural inicial são obedecidos os seguintes critérios: se o número for par, seu sucessor na sequência será sua metade e se o número for ímpar, seu sucessor será uma unidade superior ao seu triplo. Tais operações **sempre** irão levar a sequência ao número 1. A partir de tal base teórica, crie um programa em Python que receba um número inteiro C como entrada e, seguindo tal conjectura, imprima **toda a sequência até o número 1** a partir de tal número C.

Exemplo de Entrada	Exemplo de Saida
5	5 16 8 4 2 1
12	12 6 3 10 5 16 8 4 2 1

## 4 DESAFIO

1. Na matemática, a Sequência de Fibonacci é uma sequência de números inteiros, começando por 0 e 1, na qual, cada termo subsequente corresponde à soma dos dois anteriores. Faça um programa em Python que, dado um número inteiro lido pelo teclado, salve uma lista com essa quantidade de números na sequência e os imprima.

Exemplo de Entrada	Exemplo de Saida
5	[0, 1, 1, 2, 3]
11	[0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55]
18	[0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597]

### NO PRÓXIMO ENCONTRO...

No próximo encontro veremos: matrizes, dicionários e tuplas.