
ESTRUTURAS DE CONTROLE

▼ **if, elif e else** - Estruturas condicionais simples e Compostas

▼ Comandos

Estruturas condicionais simples

if teste:

comando se verdadeiro

else:

comando se falso

```
Comando
n1 = 10
n2 = 2
if n1 > n2:
    print(f'{n1} é maior que {n2}')
else:
    print(f'{n1} não é maior que {n2}')
```

Saída

10 é maior que 2

Os comandos **if** e **else** só se repetem uma única vez.

Os comandos **elif** e **else** são opcionais.

Estruturas condicionais compostas

if teste:

comando se verdadeiro

elif teste:

comando se verdadeiro

elif teste:

comando se verdadeiro

else:

comando se falso

```
Comando
```

```
n1 = 7
```

```
n2 = 10
```

```
if n1 > n2:
```

```
    print(f'{n1} é maior que {n2}')
```

```
elif n1 == n2:
```

```
    print(f'{n1} é igual a {n2}')
```

```
else:
```

```
    print(f'{n1} é menor que {n2}')
```

Saída

7 é menor que 10

Os comandos **if** e **else** só se repetem uma única vez.

Os comandos **elif** e **else** são opcionais.

O comando **elif** pode se repetir mais de uma vez, caso necessário.

▼ Exercícios

Exercício 36 - aprovando empréstimo

Escreva um programa para aprovar o empréstimo bancário para a compra de uma casa. Pergunte o valor da casa, o salário do comprador e em quantos anos ele vai pagar. A prestação mensal não pode exceder 30% do salário ou então o empréstimo será negado.

```
casa = float(input('Valor da casa: R$'))
salario = float(input('Salário do comprador: R$'))
anos = int(input('Quantos anos de financiamento? '))
prestacao = casa / (anos * 12)
minimo = salario * 30 / 100
print(f'Para pagar uma casa de R${casa:.2f} em {anos} anos', end=' ')
print(f'a prestação será de R${prestacao:.2f}')
if prestacao <= minimo:
    print('Empréstimo pode ser CONCEDIDO!')
else:
    print('Empréstimo NEGADO!')
```

Exercício 37 - conversor de bases numéricas

Escreva um programa em python que leia um número inteiro qualquer e peça para o usuário escolher qual será a base de conversão: 1 para binário, 2 para octal e 3 para hexadecimal.

```
num = int(input('Digite um número inteiro: '))
print('Escolha uma das bases da conversão:
[ 1 ] converter para BINÁRIO
[ 2 ] converter para OCTAL
[ 3 ] converter para HEXADECIMAL')
opcao = int(input('Sua opção: '))
if opcao == 1:
    print(f'{num} convertido para BINÁRIO é igual a {bin(num)[2:]}')
elif opcao == 2:
    print(f'{num} convertido para OCTAL é igual a {oct(num)[2:]}')
elif opcao == 3:
    print(f'{num} convertido para HEXADECIMAL é igual a {hex(num)[2:]}')
else:
    print('Opção inválida. Tente novamente.')
```

Exercício 38 - comparando números

Escreva um programa que leia dois números inteiros e compare-os. mostrando na tela uma mensagem:

- o primeiro valor é maior
- o segundo valor é maior
- não existe valor maior, os dois são iguais

```
n1 = int(input('Primeiro número: '))
n2 = int(input('Segundo número: '))
if n1 > n2:
    print('O PRIMEIRO valor é maior')
elif n2 > n1:
    print('O SEGUNDO valor é maior')
```

```
else:  
    print('Os dois valores são IGUAIS')
```

Exercício 39 - alistamento militar

Faça um programa que leia o ano de nascimento de um jovem e informe, de acordo com a sua idade, se ele ainda vai se alistar ao serviço militar, se é a hora exata de se alistar ou se já passou do tempo do alistamento. Seu programa também deverá mostrar o tempo que falta ou que passou do prazo.

```
from datetime import date  
atual = date.today().year  
nasc = int(input('Ano de nascimento: '))  
idade = atual - nasc  
print(f'Quem nasceu em {nasc} tem {idade} anos em {atual}')if idade == 18:  
    print('Você tem que se alistar IMEDIATAMENTE!')  
elif idade < 18:  
    saldo = 18 - idade  
    print(f'Ainda faltam {saldo} anos para o alistamento')  
    ano = atual + saldo  
    print(f'Seu alistamento será em {ano}')elif idade > 18:  
    saldo = idade - 18  
    print(f'Você já deveria ter se alistado há {saldo} anos.')  
    ano = atual - saldo  
    print(f'Seu alistamento foi em {ano}.')
```

Exercício 40 - aquele clássico da Média

Crie um programa que leia duas notas de um aluno e calcule sua média, mostrando uma mensagem no final, de acordo com a média atingida:

- média abaixo de 5.0: Reprovado
- média entre 5.0 e 6.9: Recuperação

– média 7.0 ou superior: Aprovado

```
nota1 = float(input('Primeira nota: '))
nota2 = float(input('Segunda nota: '))
media = (nota1 + nota2) / 2
print(f'Tirando {nota1:.1f} e {nota2:.1f}, a média do aluno é {media:.1f}')
if 7 > media >= 5:
    print('O aluno está em RECUPERAÇÃO.')
elif media < 5:
    print('O aluno está REPROVADO.')
elif media >= 7:
    print('O aluno está APROVADO')
```

Exercício 41 - classificando atletas

A confederação nacional de natação precisa de um programa que leia o ano de nascimento de um atleta e mostre sua categoria, de acordo com a idade:

- até 9 anos: mirim
- até 14 anos: infantil
- até 19 anos: júnior
- até 25 anos: sênior
- acima de 25 anos: master

```
from datetime import date
atual = date.today().year
nascimento = int(input('Ano de Nascimento: '))
idade = atual - nascimento
print(f'O atleta tem {idade} anos.')
if idade <= 9:
    print('Classificação: MIRIM')
```

```
elif idade <= 14:
    print('Classificação: INFANTIL')
elif idade <= 19:
    print('Classificação: JUNIOR')
elif idade <= 25:
    print('Classificação: SÊNIOR')
else:
    print('Classificação: MASTER')
```

Exercício 42 - analisando triângulos v2.0

Refaça o desafio 35 dos triângulos, acrescentando o recurso de mostrar que tipo de triângulo será formado:

- equilátero: todos os lados iguais
- isósceles: dois lados iguais
- escaleno: todos os lados diferentes.

```
r1 = float(input('Primeiro segmento: '))
r2 = float(input('Segundo segmento: '))
r3 = float(input('Terceiro segmento: '))
if r1 < r2 + r3 and r2 < r1 + r3 and r3 < r1 + r2:
    print('Os segmentos acima PODEM FORMAR um triângulo ', end='')
    if r1 == r2 == r3:
        print('EQUILATERO!')
    elif r1 != r2 != r3 != r1:
        print('ESCALENO!')
    else:
        print('ISÓSCELES')
else:
    print('Os segmentos acima NÃO PODEM FORMAR triângulo.')
```

Exercícios 43 - índice de massa corporal

Desenvolva uma lógica que leia o peso e a altura de uma pessoa, calcule seu Índice de Massa Corporal (IMC) e mostre seu status, de acordo com a tabela abaixo:

- IMC abaixo de 18,5: abaixo do peso
- entre 18,5 e 25: peso ideal
- 25 até 30: sobrepeso
- 30 até 40: obesidade
- acima de 40: obesidade mórbida

```
peso = float(input('Qual é o seu peso? (kg) '))
altura = float(input('Qual é sua altura? (m) '))
imc = peso / (altura ** 2)
print(f'O IMC dessa pessoa é de {imc:.1f}')
if imc < 18.5:
    print('Você está ABAIXO do Peso normal')
elif 18.5 <= imc < 25:
    print('PARABÉNS, você está na faixa de PESO NORMAL')
elif 25 <= imc < 30:
    print('Você está em SOBREPESO')
elif 30 <= imc < 40:
    print('Você está em OBESIDADE!')
elif imc >= 40:
    print('Você está em OBESIDADE MÓRBIDA, cuidado!')
```

Exercício 44 - gerenciador de pagamentos

Elabore um programa que calcule o valor a ser pago por um produto, considerando o seu preço normal e condição de pagamento:

- à vista dinheiro/cheque: 10% de desconto
- à vista no cartão: 5% de desconto
- em até 2x no cartão: preço formal

– 3x ou mais no cartão: 20% de juros

```
# print(f'{"LOJAS GUANABARA":^40}')
print(f'{" LOJAS GUANABARA ":=^40}')
```

```
preco = float(input('Preço das compras: R$'))
print('FORMAS DE PAGAMENTO')
[ 1 ] à vista dinheiro/cheque
[ 2 ] à vista cartão
[ 3 ] 2x no cartão
[ 4 ] 3x ou mais no cartão'''
opcao = int(input('Qual é a opção? '))
if opcao == 1:
    total = preco - (preco * 10 / 100)
elif opcao == 2:
    total = preco - (preco * 5 / 100)
elif opcao == 3:
    total = preco
    parcela = total / 2
    print(f'Sua compra será parcelada em 2x de R${parcela:.2f} SEM JUROS')
elif opcao == 4:
    total = preco + (preco * 20 / 100)
    totalparc = int(input('Quantas parcelas? '))
    parcela = total / totalparc
    print(f'Sua conta será parcelada em {totalparc}x de R${parcela:.2f} COM JUROS')
else:
    total = preco
    print('OPÇÃO INVÁLIDA de pagamento. Tente novamente!')
print(f'Sua compra de R${preco:.2f} vai custar R${total:.2f} no final.')
```

Exercício 45 - game: pedra papel e tesoura

Crie um programa que faça o computador jogar Jokenpô com você.

```
from random import randint
itens = ('Pedra', 'Papel', 'Tesoura')
computador = randint(0, 2)
print('Suas opções:
[ 0 ] Pedra
[ 1 ] Papel
[ 2 ] Tesoura')
jogador = int(input('Qual é a sua jogada? '))
print('-=' * 11)
print(f'Computador jogou {itens[computador]}')
print(f'Jogador jogou {itens[jogador]}')
print('-=' * 11)
if computador == 0:
    if jogador == 0:
        print('EMPATE')
    elif jogador == 1:
        print('JOGADOR VENCE')
    elif jogador == 2:
        print('COMPUTADOR VENCE')
    else:
        print('JOGADA INVÁLIDA!')
elif computador == 1:
    if jogador == 0:
        print('COMPUTADOR VENCE')
    elif jogador == 1:
        print('EMPATE')
    elif jogador == 2:
        print('JOGADOR VENCE')
    else:
        print('JOGADA INVÁLIDA!')
elif computador == 2:
    if jogador == 0:
        print('JOGADOR VENCE')
    elif jogador == 1:
        print('COMPUTADOR VENCE')
    elif jogador == 2:
        print('EMPATE')
```

```
else:  
    print('JOGADA INVÁLIDA!')
```

▼ **for** - estruturas de repetição com variável de controle

▼ Comandos

sintaxe básica for: for < nome variável > in < iterável >:

< nome variável > é o nome da variável que vai receber os elemento de < iterável >.

< iterável > é o container de dados sobre o qual vamos iterar, podendo ser: uma lista, uma tupla, uma string, um dicionário, entre outros.

exemplos com lista

Comando

```
computador = ['Processador', 'Teclado', 'Mouse']  
for item in computador:  
    print(item)
```

Saída

```
Processador  
Teclado  
Mouse
```

Comando

```
lista = [1, 2, 3, 4, 5]
```

```
for item in lista:  
    print(item)
```

Saída

```
1  
2  
3  
4  
5
```

exemplo com string

```
Comando  
for caractere in 'Python':  
    print(caractere)
```

Saída

```
P  
y  
t  
h  
o  
n
```

range() função

ela nos entrega uma sequência de números para utilizarmos!

sintaxes:

< nome variável > é o nome da variável que vai receber os elemento de <iterável>.

< inicio sequência> o inicio de uma sequência.

< fim sequência > o valor final da sequência.

< passo > o passo (ou pulo).

for < nome variável > in range(< inicio_sequência >, < fim_sequência>, < passo >)

da saltos em < passo > do < inicio_sequência > até < fim_sequência>.

exemplo

Comando

```
for num in range(2, 7, 2):  
    print(num)
```

Saída

```
2  
4  
6
```

for < nome variável > in range(< inicio_sequência >, < fim_sequência >)

da saltos em 1 do < inicio_sequência > até < fim_sequência>.

exemplo

Comando

```
for num in range(2, 6):  
    print(num)
```

Saída

2

3

4

5

for < nome variável > in range(< fim_sequência>)

da saltos em 1 do 0 até < fim_sequência>.

exemplo

Comando

```
for num in range(4):  
    print(num)
```

Saída

0

1

2

3

auxiliadores

break

é usado para finalizar um loop, isto é, é usado para parar sua execução.

geralmente vem acompanhado de alguma condição para isso, com um if.

exemplo com auxiliar break

Comando

```
for num in range(10):  
    if num == 5:  
        break  
    else:  
        print(num)
```

Saída

```
0  
1  
2  
3  
4
```

continue

funciona de maneira similar ao break, contudo ao invés de encerrar o loop ele pula todo código que estiver abaixo dele (dentro do loop) partindo para a próxima iteração.

exemplo com auxiliar continue

Comando

```
for num in range(5):  
    if num == 3:  
        print("Encontrei o 3")  
        continue  
    else:  
        print(num)  
print("comando após o IF")
```

Saída

0

comando após o IF

1

comando após o IF

2

comando após o IF

Encontrei o 3

4

comando após o IF

pass

o pass nada mais é que uma forma de fazer um código que não realiza operação nenhuma. o comando pass evita que o código resulte em erro.

exemplo com auxiliador pass

Comando

```
for number in range(6):  
    if number == 3:  
        pass  
    else:  
        print(number)
```

Saída

0

1

2

4

5

▼ Exercícios

Exercício 46 - contagem regressiva

Faça um programa que mostre na tela uma contagem regressiva para o estouro de fogos de artifício, indo de 10 até 0, com uma pausa de 1 segundo entre eles.

```
from time import sleep
for c in range(10, -1, -1):
    print(c)
    sleep(1)
print('BUM! BUM! POOWW!')
```

Exercício 47 - contagem de pares

Crie um programa que mostre na tela todos os números pares que estão no intervalo entre 1 e 50.

```
for c in range(2, 51, 2):
    print(c, end=' ')
```

Exercício 48 - soma ímpares múltiplos de três

Faça um programa que calcule a soma entre todos os números ímpares que são múltiplos de três e que se encontram no intervalo de 1 até 500.

```
soma = 0
for c in range(1, 501, 2):
    if c % 3 == 0:
        soma += c
print(f'Soma: {soma}')
```

Soma: 20667

Exercício 49 - tabuada v.2.0

Refaça o exercício 9, mostrando a tabuada de um número que o usuário escolher, só que agora utilizando um laço for.

```
num = int(input('Digite um número para ver a sua tabuada: '))
for c in range(0, 11):
    print(f'{num} x {c} = {num * c:.2f}')
```

Exercício 50 - soma dos pares

Desenvolva um programa que leia seis números inteiros e mostre a soma apenas daqueles que forem pares. Se o valor digitado for ímpar, desconsidere-o.

```
soma = 0
for c in range(1, 7):
    num = int(input(f'Digite o {c}º número: '))
    if num % 2 == 0:
        soma += num
print(f'A soma dos números pares é {soma}')
```

Exercício 51 - progressão aritmética

Desenvolva um programa que leia o primeiro termo e a razão de uma Progressão Aritmética. No final, mostre os 10 primeiros termos dessa progressão.

```
primeiro = int(input('Primeiro termo da PA: '))
razao = int(input('Razão da PA: '))
decimo = primeiro + (10 - 1) * razao
for c in range(primeiro, decimo + razao, razao):
    print(f'{c}', end=' → ')
print('ACABOU')
```

Exercício 52 - números primos

Faça um programa que leia um número inteiro e diga se ele é ou não um número primo.

```
tot = 0
num = int(input('Digite um número: '))
for c in range(1, num+1):
    if num % c == 0:
        print('\033[33m', end='')
        tot += 1
    else:
        print('\033[31m', end='')
    print(f'{c} ', end='')
print(f'\n\033[30m0 número {num} foi divisível {tot} vezes')
if tot == 2:
    print('E por isso ele É PRIMO!')
else:
    print('E por isso ele NÃO É PRIMO!')
```

Exercício 53 - detector de palíndromo

Crie um programa que leia uma frase qualquer e diga se ela é um palíndromo (é a mesma coisa lendo de frente para trás ou de trás para frente), desconsiderando os espaços.

exemplos de palíndromos: apos a sopa - a sacada da casa - a torre da derrota - o lobo ama o bolo - anotaram a data da maratona

```
frase = str(input('Digite uma frase: ')).strip().upper()
palavras = frase.split()
junto = ''.join(palavras)
inverso = ''
for letra in range(len(junto) - 1, -1, -1):
    inverso += junto[letra]
# ou inverso = junto[::-1]
print(f'O inverso de {junto} é {inverso}')
if inverso == junto:
    print('Temos um palíndromo!')
else:
    print('A frase digitada não é um palíndromo!')
```

Exercício 54 - grupo da maioridade

Crie um programa que leia o ano de nascimento de sete pessoas. No final, mostre quantas pessoas ainda não atingiram a maioridade e quantas já são maiores. a partir de 21 anos é maior de idade.

```
from datetime import date
menor = maior = 0
for c in range(1, 8):
    nascimento = int(input(f'Digite o ano de nascimento da {c}ª pessoa: '))
    if date.today().year - nascimento < 21:
        menor += 1
    else:
        maior += 1
print(f'{maior} pessoas são maiores de idade')
print(f'{menor} pessoas são menores de idade')
```

Exercício 55 - maior e menor da sequência

Faça um programa que leia o peso de cinco pessoas. No final, mostre qual foi o maior e o menor peso lidos.

```

maior = 0
menor = 0
for c in range(1, 6):
    peso = float(input(f'Peso da {c}ª pessoa (kg): '))
    if c == 1:
        menor = peso
        maior = peso
    else:
        if peso < menor:
            menor = peso
        if peso > maior:
            maior = peso
print(f'O maior peso foi de {maior}kg')
print(f'O menor peso foi de {menor}kg')

```

Exercício 56 - analisador completo

Desenvolva um programa que leia o nome, idade e sexo de 4 pessoas. No final do programa, mostre:

- a média de idade do grupo
- qual é o nome do homem mais velho
- quantas mulheres têm menos de 20 anos.

```

somaidade = 0
mediaidade = 0
maioridadehomem = 0
nomevelho = ''
totmulher20 = 0
for c in range(1, 5):
    print(f'----- {c}ª PESSOA -----')
    nome = str(input('Nome: ')).strip()
    idade = int(input('Idade: '))
    sexo = str(input('Sexo [M/F]: ')).strip()
    somaidade += idade

```

```
if c == 1 and sexo in 'Mm':
    maioridadehomem = idade
    nomevelho = nome
if sexo in 'Mm' and idade > maioridadehomem:
    maioridadehomem = idade
    nomevelho = nome
if sexo in 'Ff' and idade < 20:
    totmulher20 += 1
mediaidade = somaidade / 4
print(f'A média de idade do grupo é de {mediaidade} anos')
print(f'O homem mais velho tem {maioridadehomem} anos e se chama {nomevelho}.')
print(f'Ao todo são {totmulher20} mulheres com menos de 20 anos')
```

▼ **while** - estruturas de repetição com teste lógico

▼ Comandos

sintaxe básica while: while < condição >:

while Significa Enquanto

< condição > é uma expressão que pode ser reduzida à True ou False, podendo ser:

- A verificação do valor de uma variável;
- Determinada estrutura de dados alcançar um tamanho;
- O retorno de uma função se igualar a determinado valor;
- Algum valor externo ser alterado (por exemplo um valor armazenado em Banco de Dados).

exemplo 1

Comando

```
c = 1
while c < 5:
    print(c)
    c = c + 1
print('Fim')
```

Saída

```
1
2
3
4
Fim
```

exemplo 2

Comando

```
r = 'S'
while r == 'S':
    n = int(input('Digite um valor: '))
    r = str(input('Quer continuar? [S/N] ')).upper()
print('Fim')
```

Saída

```
Digite um valor: 3
Quer continuar? [S/N] S
Digite um valor: 6
Quer continuar? [S/N] s
```

```
Digite um valor: 8
Quer continuar? [S/N] n
Fim
```

exemplo 3

```
Comando
n = s = 0
while True:
    n = int(input('Digite um número: '))
    if n == 999:
        break
    s += n
print(f'A soma vale {s}')
```

```
Saída
Digite um número: 2
Digite um número: 3
Digite um número: 999
A soma vale 5
```

auxiliares

break

é usado para finalizar um loop, isto é, é usado para parar sua execução.

geralmente vem acompanhado de alguma condição para isso, com um if.

exemplo com auxiliar break

Comando

```
num = 0
while num < 5:
    num += 1
    if num == 3:
        break
    print(num)
```

Saída

```
1
2
```

Comando

```
n = s = 0
while True:
    n = int(input('Digite um número: '))
    if n == 999:
        break
    s += n
print(f'A soma vale {s}')
```

Saída

```
Digite um número: 2
Digite um número: 3
Digite um número: 999
A soma vale 5
```

continue

funciona de maneira similar ao break, contudo ao invés de encerrar o loop ele pula todo código que estiver abaixo dele (dentro do loop) partindo para a próxima iteração.

exemplo com auxiliar continue

```
Comando
num = 0
while num < 5:
    num += 1
    if num == 3:
        continue
    print(num)
```

```
Saída
1
2
4
5
```

pass

o pass nada mais é que uma forma de fazer um código que não realiza operação nenhuma. o comando pass evita que o código resulte em erro.

exemplo com auxiliar pass

```
Comando
while False:
    pass
```

▼ Exercícios

Exercício 57 - validação de dados

Faça um programa que leia o sexo de uma pessoa, mas só aceite os valores 'M' ou 'F'. Caso esteja errado, peça a digitação novamente até ter um valor correto.

```
sexo = str(input('Informe seu sexo: [M/F] ')).strip().upper()[0] #[0] pega só a primeira letra
while sexo not in 'MmFf':
    sexo = str(input('Dados inválidos. Por favor, informe seu sexo: ')).strip().upper()[0]
print(f'Sexo {sexo} registrado com sucesso')
```

Exercício 58 - jogo da adivinhação v2.0

Melhore o jogo do desafio 28 onde o computador vai “pensar” em um número entre 0 e 10. Só que agora o jogador vai tentar adivinhar até acertar, mostrando no final quantos palpites foram necessários para vencer.

```
from random import randint
computador = randint(0, 10)
print('Sou seu computador... Acabei de pensar em um número entre 0 e 10.')
print('Será que você consegue adivinhar qual foi? ')
acertou = False
palpites = 0
while not acertou:
    jogador = int(input('Qual é seu palpite? '))
    palpites += 1
    if jogador == computador:
        acertou = True
```

```
else:
    if jogador < computador:
        print('Mais... Tente mais uma vez.')
    elif jogador > computador:
        print('Menos... Tente mais uma vez.')
print(f'Acertou com {palpites} tentativas. Parabéns!')
```

Exercício 59 - criando um menu de opções

Crie um programa que leia dois valores e mostre um menu na tela:

[1] somar

[2] multiplicar

[3] maior

[4] novos números

[5] sair do programa

Seu programa deverá realizar a operação solicitada em cada caso.

```
from time import sleep
n1 = int(input('Primeiro valor: '))
n2 = int(input('Segundo valor: '))
opcao = 0
while opcao != 5:
    print(''
          [ 1 ] somar
          [ 2 ] multiplicar
          [ 3 ] maior
          [ 4 ] novos números
          [ 5 ] sair do programa'')
    opcao = int(input('>>>> Qual é a sua opção? '))
    if opcao == 1:
        soma = n1 + n2
        print(f'A soma entre {n1} + {n2} é {soma}')
```

```

elif opcao == 2:
    produto = n1 * n2
    print(f'O resultado de {n1} x {n2} é {produto}')
elif opcao == 3:
    if n1 > n2:
        maior = n1
    else:
        maior = n2
    print(f'Entre {n1} e {n2} o maior é {maior}')
elif opcao == 4:
    print('Informe os números novamente:')
    n1 = int(input('Primeiro valor: '))
    n2 = int(input('Segundo valor: '))
elif opcao == 5:
    print('Finalizando...')
else:
    print('Opção inválida. Tente novamente')
print('== ' * 10)
sleep(2)
print('Fim do programa! Volte sempre!')

```

Exercício 60 - cálculo do fatorial

Faça um programa que leia um número qualquer e mostre o seu fatorial.

exemplo: $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$

```

n = int(input('Digite um número para calcular o seu Fatorial: '))
c = n
f = 1
print(f'Calculando {n}! = ', end='')
while c > 0:
    print(f'{c}', end = ' ')
    print(f' x ' if c > 1 else ' = ', end='')
    f *= c
    c -= 1

```

```
c -= 1
print(f'{f}')
```

Exercício 61 - progressão aritmética v2.0

Refaça o desafio 51, lendo o primeiro termo e a razão de uma PA, mostrando os 10 primeiros termos da progressão usando a estrutura while.

```
print('Gerador de PA')
print('-=' * 10)
primeiro = int(input('Primeiro termo: '))
razao = int(input('Razão da PA: '))
termo = primeiro
cont = 1
while cont <= 10:
    print(f'{termo} → ', end='')
    termo += razao
    cont += 1
print('FIM')
```

Exercício 62 - super progressão aritmética v3.0

Melhore o desafio 61, perguntando para o usuário se ele quer mostrar mais alguns termos. O programa encerrará quando ele disser que quer mostrar 0 termos.

```
print('Gerador de PA')
print('-=' * 10)
primeiro = int(input('Primeiro termo: '))
razao = int(input('Razão da PA: '))
termo = primeiro
cont = 1
total = 0
mais = 10
while mais != 0:
```

```

total = total + mais
while cont <= total:
    print(f'{termo} → ', end='')
    termo += razao
    cont += 1
print('PAUSA')
mais = int(input('Quantos termos você quer mostrar a mais? '))
print(f'Progressão finalizada com {total} termos mostrados.')

```

Exercício 63 - sequência de fibonacci v1.0

Escreva um programa que leia um número N inteiro qualquer e mostre na tela os N primeiros elementos de uma Sequência de Fibonacci.

Exemplo: 0 → 1 → 1 → 2 → 3 → 5 → 8

```

print('-'*30)
print('Sequência de Fibonacci')
print('-'*30)
n= int(input('Quantos termos você quer mostrar? '))
t1 = 0
t2 = 1
print('~'*30)
print(f'{t1} → {t2}', end='')
cont = 3
while cont <= n:
    t3 = t1 + t2
    print(f' → {t3}', end='')
    t1 = t2
    t2 = t3
    cont += 1
print(' → FIM')
print('~'*30)

```

Exercício 64 - tratando vários valores v1.0

Crie um programa que leia vários números inteiros pelo teclado. O programa só vai parar quando o usuário digitar o valor 999, que é a condição de parada. No final, mostre quantos números foram digitados e qual foi a soma entre eles (desconsiderando o flag).

```
num = cont = soma = 0
num = int(input('Digite um número [999 para parar]: '))
while num != 999:
    soma += num
    cont += 1
    num = int(input('Digite um número [999 para parar]: '))
print(f'Você digitou {cont} números e a soma entre eles foi {soma}.')
```

Exercício 65 - maior e menor valores

Crie um programa que leia vários números inteiros pelo teclado. No final da execução, mostre a média entre todos os valores e qual foi o maior e o menor valores lidos. O programa deve perguntar ao usuário se ele quer ou não continuar a digitar valores.

```
resp = 'S'
soma = quant = media = maior = menor = 0
while resp in 'Ss':
    num = int(input('Digite um número: '))
    soma += num
    quant += 1
    if quant == 1:
        maior = menor = num
    else:
        if num > maior:
            maior = num
        if num < menor:
            menor = num
    resp = str(input('Quer continuar? [S/N] ')).upper().strip()[0]
media = soma / quant
print(f'Você digitou {quant} números e a média foi {media}.')
print(f'O maior valor foi {maior} e o menor foi {menor}.')
```

Exercício 66 - vários números com flag

Crie um programa que leia números inteiros pelo teclado. O programa só vai parar quando o usuário digitar o valor 999, que é a condição de parada. No final, mostre quantos números foram digitados e qual foi a soma entre elas (desconsiderando o flag).

```
soma = cont = 0
while True:
    num = int(input('Digite um valor (999 para parar): '))
    if num == 999:
        break
    cont += 1
    soma += num
print(f'A soma dos {cont} valores foi {soma}!')
```

Exercício 67 - tabuada v3.0

Faça um programa que mostre a tabuada de vários números, um de cada vez, para cada valor digitado pelo usuário. O programa será interrompido quando o número solicitado for negativo.

```
while True:
    n = int(input('Quer ver a tabuada de qual valor? '))
    print('-' * 30)
    if n < 0:
        break
    for c in range(1, 11):
        print(f'{n} x {c} = {n * c}')
    print('-' * 30)
print('PROGRAMA TABUADA ENCERRADO. Volte sempre!')
```

Exercício 68 - jogo do par ou ímpar

Faça um programa que jogue par ou ímpar com o computador. O jogo só será interrompido quando o jogador perder, mostrando o total de vitórias consecutivas que ele conquistou no final do jogo.

```
from random import randint
v = 0
while True:
    jogador = int(input('Diga um valor: '))
    computador = randint(0, 10)
    total = jogador + computador
    tipo = ' '
    while tipo not in 'PI':
        tipo = str(input('Par ou Ímpar? [P/I] ')).strip().upper()[0]
    print(f'Você jogou {jogador} e o computador {computador}. Total de {total} ', end="")
    print('DEU PAR' if total % 2 == 0 else 'DEU ÍMPAR')
    if tipo == 'P':
        if total % 2 == 0:
            print('Você VENCEU!')
            v += 1
        else:
            print('Você PERDEU!')
            break
    elif tipo == 'I':
        if total % 2 == 1:
            print('Você VENCEU!')
            v += 1
        else:
            print('Você PERDEU!')
            break
    print('Vamos jogar novamente...')
print(f'GAME OVER! Você venceu {v} vezes.')
```

Exercício 69 - análise de dados do grupo

Crie um programa que leia a idade e o sexo de várias pessoas. A cada pessoa cadastrada, o programa deverá perguntar se o usuário quer ou não continuar. No final, mostre:

- A) quantas pessoas tem mais de 18 anos.
- B) quantos homens foram cadastrados.
- C) quantas mulheres tem menos de 20 anos.

```
tot18 = toth = totm20 = 0
while True:
    idade = int(input('Idade: '))
    sexo = ' '
    while sexo not in 'MF':
        sexo = str(input('Sexo: [M/F] ')).strip().upper()[0]
    if idade >= 18:
        tot18 += 1
    if sexo == 'M':
        toth += 1
    if sexo == 'F' and idade < 20:
        totm20 += 1
    resp = ' '
    while resp not in 'SN':
        resp = str(input('Quer continuar? [S/N] ')).strip().upper()[0]
    if resp == 'N':
        break
print(f'Total de pessoas com mais de 18 anos = {tot18}')
print(f'Ao todo temos {toth} homens cadastrados')
print(f'E temos {totm20} mulheres com menos de 20 anos')
```

Exercício 70 - estatísticas em produtos

Crie um programa que leia o nome e o preço de vários produtos. O programa deverá perguntar se o usuário vai continuar ou não. No final, mostre:

- A) qual é o total gasto na compra.
- B) quantos produtos custam mais de R\$1.000.

C) qual é o nome do produto mais barato.

```
total = totmil = menor = cont = 0
barato = ''
while True:
    produto = str(input('Nome do produto: '))
    preco = float(input('Preço: R$'))
    cont += 1
    total += preco
    if preco > 1000:
        totmil += 1
    if cont == 1:
        menor = preco
        barato = produto
    else:
        if preco < menor:
            menor = preco
            barato = produto
    resp = ' '
    while resp not in 'SN':
        resp = str(input('Quer continuar? [S/N] ')).strip().upper()[0]
    if resp == 'N':
        break
print(f'{" FIM DO PROGRAMA ":-^40}')
print(f'O total da compra foi R${total:.2f}')
print(f'Temos {totmil} produtos custando mais de R$1.000')
print(f'O produto mais barato foi {barato} que custa R${menor:.2f}')
```

Exercício 71 - simulador de caixa eletrônico

Crie um programa que simule o funcionamento de um caixa eletrônico. No início, pergunte ao usuário qual será o valor a ser sacado (número inteiro) e o programa vai informar quantas cédulas de cada valor serão entregues.

obs: considere que o caixa possui cédulas de 50 reais, 20 reais, 10 reais e 1 real.

```

print('=' * 30)
print(f'{"BANCO CEV":^30}')
print('=' * 30)
valor = int(input('Que valor você quer sacar? R$'))
total = valor
ced = 50
totced = 0
while True:
    if total >= ced:
        total -= ced
        totced += 1
    else:
        if totced > 0:
            print(f'Total de {totced} cédulas de R${ced}')
        if ced == 50:
            ced = 20
        elif ced == 20:
            ced = 10
        elif ced == 10:
            ced = 1
        totced = 0
        if total == 0:
            break
print('=' * 30)
print('Volte sempre ao BANCO CEV! Tenha um bom dia!')

```

▼ Outros

time.sleep()

O Python possui suporte integrado para colocar seu programa em hibernação. O timemódulo tem uma função sleep() que você pode usar para suspender a execução do thread de chamada por quantos segundos você especificar.

exemplo

Comando

```
from time import sleep
for c in range(5, -1, -1):
    print(c)
    sleep(1)
```

Saída

```
5
4
3
2
1
0
```

date.today()

Dessa forma você consegue obter a data atual no Python.

Comando

```
from datetime import date
print(date.today())
```

Saída

```
2023-03-01
```

date.today().year

Retorna o ano da data atual.

Comando

```
from datetime import date  
print(date.today().year)
```

Saída

2023

date.today().month

Retorna o mês da data atual.

Comando

```
from datetime import date  
print(date.today().month)
```

Saída

3

date.today().day

Retorna o dia da data atual.

Comando

```
from datetime import date  
print(date.today().day)
```

Saída

1

random.randint

O randint() método retorna um elemento selecionado de número inteiro do intervalo especificado.

Retorna um número inteiro aleatório no intervalo especificado.

random.randint(Inicio, fim)

Comando

```
import random  
print(random.randint(3, 9))
```

Saída

```
4 #qualquer número no intervalo
```

