**Aula 04 - Estruturas de Repetição**

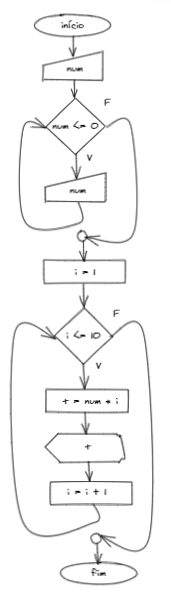
**1. Lógica**

Em determinados programas, precisamos de entrada de dados, processamento e saída de dados. Entretanto, em alguns casos, precisamos também ter decisões e até mesmo repetições de determinados trechos de código. Neste caso, While ou For, como estrutura de repetição no Python.

**1.1. While**

Exemplo:

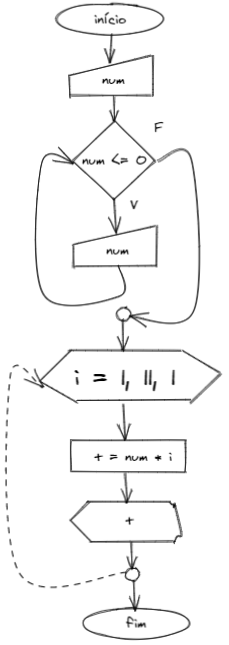
Fazer um programa para receber um número e validar se esse número é positivo. Após isso, exibir a tabuada de 1 a 10 desse número:



**1.2. For**

Exemplo:

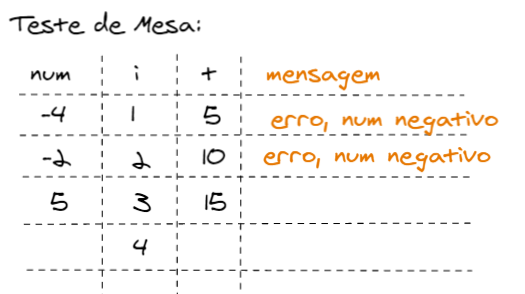
Fazer um programa para receber um número e validar se esse número é positivo. Após isso, exibir a tabuada de 1 a 10 desse número:



**1.3. Teste de Mesa**

Na Lógica de Programação, podemos utilizar um recurso extremamente útil chamado Teste de Mesa. Ele consiste em anotar todos os valores das variáveis conforme o programa vai avançando passo a passo.

Utilizando o exemplo acima, podemos ter o seguinte teste de mesa:



Neste caso, podemos ver que a cada looping do programa os valores estão sendo alterados corretamente, tendo o programa exibindo o resultado da tabuada do número 5 corretamente.

**2. Programação Python**

**2.1. While**

Exemplo:

Fazer um programa para receber um número e validar se esse número é positivo. Após isso, exibir a tabuada de 1 a 10 desse número.

print('Tabuada com WHILE')

num = int(input("Digite um numero para obter a tabuada: "))

while(num <= 0):

print("Não pode numero negativo!")

num = int(input("Digite um outro numero para obter a tabuada: "))

i = 1

while(i < 11):

r = num \* i

print(f'{num} X {i} = {r}')

i = i + 1

**2.2. For**

Exemplo:

Fazer um programa para receber um número e validar se esse número é positivo. Após isso, exibir a tabuada de 1 a 10 desse número.

print('Tabuada com FOR')

num = int(input("Digite um numero para obter a tabuada: "))

while(num <= 0):

print("Não pode numero negativo!")

num = int(input("Digite um outro numero para obter a tabuada: "))

for i in range(1, 11, 1):

r = num \* i

print(f'{num} X {i} = {r}')

**2.3. Manipulando Loopings (While e For)**

**2.3.1. Break**

Caso você queira interromper um looping, você pode utilizar o comando break para isso, veja:

for i in range(1, 4, 1):

numero = int(input("Descubra o número de 1 a 10: "))

if (numero == 8):

print("Parabéns você acertou!")

break

Neste caso, quando o usuário digitar o número 8, o programa imediatamente sairá do looping independentemente se faltar muito ou pouco para terminar o looping.

**2.3.2. Continue**

Caso você queira pular um looping, você pode utilizar o comando “continue” para isso, veja:

for i in range(1, 6, 1):

if (i == 3):

continue

print(i)

Neste caso, quando chegar no número 3, o programa irá pular esse looping e continuará com o próximo looping até o final do ciclo de loopings.

**3. Exercícios**

23. Criar uma rotina de entrada que aceite somente um valor positivo.

num = int(input('Digite um número: '))

while(num<=0):

print('Erro! Digite apenas números positivos!')

num = int(input('Digite um número: '))

print('Digitou número positivo!')

24. Entrar com dois valores via teclado, onde o segundo deverá ser maior que o primeiro. Caso contrário solicitar novamente apenas o segundo valor.

a = int(input('Digite o primeiro valor: '))

b = int(input('Digite o segundo valor: '))

while(b<=a):

print('O segundo valor deve ser maior do que o primeiro valor!')

b = int(input('Digite o segundo valor: '))

print('Obrigado!')

25. Entrar via teclado com o sexo de determinado usuário, aceitar somente “F” ou “M” como respostas válidas.

s = input('Digite o sexo (F/M): ')

while ( (s.upper() != 'M') and (s.upper() != 'F') ):

print('Erro! Digite apenas F ou M!')

s = input('Digite o sexo (F/M): ')

print('Obrigado!')

26. Exibir a tabuada do número cinco no intervalo de um a dez.

**26W**

num=5

i=1

while(i<=10):

r = num \* i

print(f'{num} X {i} = {r}')

i=i+1

**26FOR**

num = 5

for i in range(1, 11, 1):

r = num \* i

print(f'{num} X {i} = {r}')

27. Entrar via teclado com um valor qualquer. Travar a digitação, no sentido de aceitar somente valores positivos. Após a digitação, exibir a tabuada do valor solicitado, no intervalo de um a dez.

**27W**

num=int(input('Digite um número para obter sua tabuada: '))

i=1

while(num<=0):

print('Erro! Digite apenas números positivos!')

num=int(input('Digite um número para obter sua tabuada: '))

while(i<=10):

r = num \* i

print(f'{num} X {i} = {r}')

i=i+1

**27FOR**

num=int(input('Digite um número para obter sua tabuada: '))

while(num<=0):

print('Erro! Digite apenas números positivos!')

num=int(input('Digite um número para obter sua tabuada: '))

for i in range(1, 11, 1):

r = num \* i

print(f'{num} X {i} = {r}')

28. Entrar via teclado com um valor (X) qualquer. Travar a digitação, no sentido de aceitar somente valores positivos. Solicitar o intervalo que o programa que deverá calcular a tabuada do valor digitado, sendo que o segundo valor (B), deverá ser maior que o primeiro (A), caso contrário, digitar novamente somente o segundo. Após a validação dos dados, exibir a tabuada do valor digitado, no intervalo decrescente, ou seja, a tabuada de X no intervalo de B para A.

num = int(input('Digite um número para obter a sua tabuada: '))

while(num <= 0):

print('Erro! Digite apenas números positivos!')

num = int(input('Digite um número para obter a sua tabuada: '))

a = int(input('Digite o intervalo inicial para obter a tabuada: '))

while (a <= 0):

print('Erro! Digite apenas números positivos!')

a = int(input('Digite o intervalo inicial para obter a tabuada: '))

b = int(input('Digite o intervalo final para obter a tabuada: '))

while (b <= a):

print('Erro! O intervalo final deve ser maior do que o inicial!')

b = int(input('Digite o intervalo final para obter a tabuada: '))

for i in range(b, a-1, -1):

r = num \* i

print(f'{num} X {i} = {r}')

29. Exibir a tabuada dos valores de um a vinte, no intervalo de um a dez. Entre as tabuadas, solicitar que o usuário pressione uma tecla.

for num in range(1, 21, 1):

for i in range(1, 11, 1):

r = num \* i

print(f'{num} X {i} = {r}')

input('Pressione uma tecla para continuar...')

30. Exibir a soma dos números inteiros positivos do intervalo de um a cem.

t=0

for i in range(1, 101, 1):

#t = t + i

t += i

print(f'A soma dos números de 1 a 100 é: {t}')

Checkpoint 01

exec = "S"

historico = ""

while (exec.upper() == "S"):

nome = input('Digite o nome do usuário: ')

num = int(input('Digite um número para obter a sua tabuada: '))

while(num <= 0):

print('Erro! Digite apenas números positivos!')

num = int(input('Digite um número para obter a sua tabuada: '))

a = int(input('Digite o intervalo inicial para obter a tabuada: '))

while (a <= 0):

print('Erro! Digite apenas números positivos!')

a = int(input('Digite o intervalo inicial para obter a tabuada: '))

b = int(input('Digite o intervalo final para obter a tabuada: '))

while (b <= a):

print('Erro! O intervalo final deve ser maior do que o inicial!')

b = int(input('Digite o intervalo final para obter a tabuada: '))

historico = historico + 'Nome: ' + nome + ' | Tabuada: ' + str(num) + ' | Intervalo: ' + str(a) + ' a ' + str(b) + '\n'

for i in range(b, a-1, -1):

r = num \* i

print(f'{num} X {i} = {r}')

exec = input('Deseja executar novamente o programa (S/N)?')

print(f'Histórico das execuções: \n{historico}')

31. Entrar via teclado com dez valores positivos. Consistir a digitação e enviar mensagem de erro, se necessário. Após a digitação, exibir:

a) O maior valor;

b) A soma dos valores;

c) A média aritmética dos valores;

soma = 0

for i in range(1, 11, 1):

num = int(input(f'Digite o {i}° número: '))

while(num <= 0):

print('Erro! Número negativo!')

num = int(input(f'Digite o {i}° número: '))

if (i == 1):

maior = num

elif (num > maior):

maior = num

soma = soma + num

media = soma / 10

print(f'O maior número é: {maior}')

print(f'A soma dos valores é: {soma}')

print(f'A média é: {media}')

32. Entrar via teclado com “N” valores quaisquer. O valor “N” (que representa a quantidade de números) será digitado, deverá ser positivo, mas menor que vinte. Caso a quantidade não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente. Após a digitação dos “N” valores, exibir:

a) O maior valor;

b) O menor valor;

c) A soma dos valores;

d) A média aritmética dos valores;

e) A porcentagem de valores que são positivos;

f) A porcentagem de valores negativos;

soma = 0

qtd\_pos = 0

qtd\_neg = 0

qtd = int(input('Digite a quantidade de valores que deseja digitar: '))

while((qtd <= 0) or (qtd > 20)):

print('Erro! Valor inválido, a quantidade deve estar entre 1 e 20!')

qtd = int(input('Digite a quantidade de valores que deseja digitar: '))

for i in range(1, qtd+1, 1):

num = int(input(f'Digite o {i}° número: '))

if (i == 1):

maior = num

menor = num

if (num > maior):

maior = num

if (num < menor):

menor = num

if (num >= 0):

qtd\_pos = qtd\_pos + 1

else:

qtd\_neg = qtd\_neg + 1

soma = soma + num

media = soma / qtd

por\_pos = (qtd\_pos \* 100) / qtd

por\_neg = (qtd\_neg \* 100) / qtd

print(f'O maior número é: {maior}')

print(f'O menor número é: {menor}')

print(f'A soma dos valores é: {soma}')

print(f'A média é: {media:.1f}')

print(f'O percentual de positivos é: {por\_pos:.2f}%')

print(f'O percentual de negativos é: {por\_neg:.2f}%')

33. Exibir os trinta primeiros valores da série de Fibonacci. A série: 1, 1, 2, 3, 5, 8, …

a = 1

b = 0

for i in range(1, 31, 1):

f = a + b

print(f)

a = b

b = f

34. Exibir os vinte primeiros valores da série de Bergamaschi. A série: 1, 1, 1, 3, 5, 9, 17, …

a = 1

b = 1

c = -1

for i in range(1, 21, 1):

f = a + b + c

print(f)

a = b

b = c

c = f

35. Calcular e exibir a soma dos “N” primeiros valores da seqüência abaixo. O valor “N” será digitado, deverá ser positivo, mas menor que cem. Caso o valor não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente.

A seqüência: 2, 5, 10, 17, 26, ....

n = int(input("digite o valor de n: "))

while n>100 or n<=0:

print("O valor deve ser positivo e menor que 100")

n = int(input("digite n novamente: "))

curr = 2

incr = 3

for i in range(n):

print(curr)

curr += incr

incr += 2

36. Calcular e exibir a soma dos “N” primeiros valores da seqüência abaixo. O valor “N” será digitado, deverá ser positivo, mas menor que cinqüenta. Caso o valor não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente.



n = int(input("digite o valor de n: "))

while (n <= 0) or (n > 50):

print("O valor deve ser positivo e menor que 50")

n = int(input("digite n novamente: "))

a = 1

b = 2

soma = 0.0

for i in range(1, n+1, 1):

print(f'{a} / {b}')

a += 1

b += 1

soma = soma + a/b

print(f'A soma da sequência é: {soma:.2f}')

37. Calcular e exibir a soma dos “N” primeiros valores da seqüência abaixo. O valor “N” será digitado, deverá ser positivo, mas menor que cinqüenta. Caso o valor não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente.



n = int(input("digite o valor de n: "))

while (n <= 0) or (n > 50):

print("O valor deve ser positivo e menor que 50")

n = int(input("digite n novamente: "))

a = 2

b = 1

c = 1

soma = 0.0

for i in range(1, n+1, 1):

print(f'{a} / {b}')

soma = soma + (a/b)

c = c + 2

a = a + c

b = ((i+1) \* (i+1) \* (i+1))

print(f'A soma da sequência é: {soma:.2f}')

38. Entrar via teclado com “N” valores quaisquer. O valor “N” (que representa a quantidade de números) será digitado, deverá ser positivo, mas menor que vinte. Caso a quantidade não satisfaça a restrição, enviar mensagem de erro e solicitar o valor novamente. Após a digitação dos “N” valores, exibir:

a) O maior valor;

b) O menor valor;

c) A soma dos valores;

d) A média aritmética dos valores;

e) A porcentagem de valores que são positivos;

f) A porcentagem de valores negativos;

Após exibir os dados, perguntar ao usuário se deseja ou não uma nova execução do programa. Consistir a resposta no sentido de aceitar somente “S” ou “N” e encerrar o programa em função dessa resposta.

continuar = 's'

while(continuar == 's'):

qtd = *int*(input('Digite a quantidade de valores à serem digitados: '))

while((qtd <= 0) or (qtd > 20)):

print('Quantidade inválida! Favor escolher entre 1 e 20.')

qtd = *int*(input('Digite a quantidade de valores à serem digitados: '))

maior\_valor = 0

menor\_valor = 0

zeros = 0

soma = 0

num\_pos = 0

num\_neg = 0

for i in range(1, qtd+1, 1):

numero = *int*(input(f'Digite o numero {i}: '))

soma = soma + numero

if (numero > 0):

num\_pos = num\_pos + 1

elif (numero < 0):

num\_neg = num\_neg + 1

else:

zeros = zeros + 1

if (i == 1):

maior\_valor = numero

menor\_valor = numero

else:

if (numero > maior\_valor):

maior\_valor = numero

if (numero < menor\_valor):

menor\_valor = numero

print(f'O maior valor é: {maior\_valor}')

print(f'O menor valor é: {menor\_valor}')

print(f'A soma dos números é: {soma}')

print(f'A quantidade de números zeros: {zeros}')

media = soma / qtd

print(f'A média aritmética é: {media:.2f}')

por\_num\_pos = (num\_pos / qtd) \* 100

print(f'Porcentagem dos valores positivos: {por\_num\_pos}%')

por\_num\_neg = (num\_neg / qtd) \* 100

print(f'Porcetagem dos valores negativos: {por\_num\_neg}%')

continuar = input('Deseja realizar uma nova execução? (s/n): ')

while((continuar != 's') and (continuar != 'n')):

print('Opção inválida! Apenas digite s ou n!')

continuar = input('Deseja realizar uma nova execução? (s/n): ')

39. Calcular o fatorial de um valor que será digitado. Este valor não poderá ser negativo. Enviar mensagem de erro e solicitar o valor novamente, se necessário. Perguntar se o usuário deseja ou não fazer um novo cálculo, consistir a resposta em “S” ou “N”.

N! = N x N-1 x N-2 x N-3 x ....... x (N - (N-1))

Ex: 5! = 5 x 4 x 3 x 2 x 1 = 120

num = int(input('Digite um número para obter o seu fatorial: '))

while (num < 0):

print('Erro! Digite apenas números positivos!')

num = int(input('Digite um número para obter o seu fatorial: '))

if (num == 0) or (num == 1):

fat = 1

else:

fat = num

while (num > 1):

fat = fat \* (num-1)

num = num - 1

print(f'O fatorial é: {fat}')