

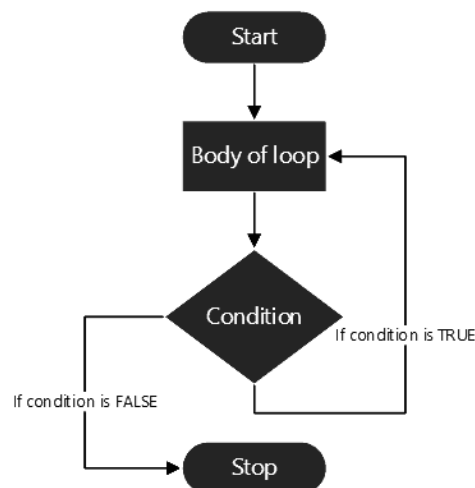
Aula 03 – Estruturas de Repetição

While

Dinâmica: Faça um programa que solicite ao usuário três números e imprima os números na tela. Agora amplie isso para 5 números, para 8....

Você percebe que a estrutura do código cresce de maneira proporcional à quantidade de variáveis, deixando-o muito extenso e deselegante?

Já que estamos executando os mesmos comandos várias vezes, podemos usar uma estrutura de repetição. A estrutura de repetição segue o fluxograma a seguir:



O **while** é uma estrutura de repetição ou loop que executa indefinidamente até a condição inicial ser satisfeita. Vejamos a solução do desafio para 10 casos com o uso do while:

```
n = 0

while n < 10:
    num = int(input('Digite um número inteiro:'))
    print('O número digitado foi', num)
    n += 1
```

Perceba que se em algum momento não satisfizermos a condição inicial, os comandos executarão indefinidamente. Por isso, precisamos atualizar o valor de n, caso contrário teremos um loop infinito.

For

Utilizamos o for em Python para iterar **em cima de uma sequência definida** (uma lista, uma tupla, uma string, um dicionário... na maioria dos casos é uma lista mesmo). Percorremos todos os elementos da lista de tamanho n, repetindo n vezes o conjunto de operações que está no corpo da lista. No caso do for, não precisamos de contador, pois o número de repetições é o próprio tamanho da lista.

Exemplo 1: iterando em uma lista

```
frutas = ['maçã', 'banana', 'abacaxi']
for fruta in frutas:
    print(fruta)
```

Exemplo 2: iterando em uma string

```
for x in 'banana':
    print(x)
```

Exemplo 3: interrompendo a iteração

```
frutas = ['maçã', 'banana', 'abacaxi']
for fruta in frutas:
    if fruta == 'banana':
        break
    print(fruta)
```

Elementos Importantes da Estrutura de Repetição

- **Contador:** nos ajuda a contar as iterações (repetições do loop). O valor adicionado em um contador é sempre constante.
- **Acumulador:** nos ajuda a armazenar os resultados de forma cumulativa (soma, produto, etc.). O valor adicionado em um acumulador pode variar.
- **Incrementar/decrementar:** somar ou diminuir o contador por um valor constante.
- **Break:** comando que interrompe a execução de um loop.
- **Continue:** comando que continua o loop a partir do próximo elemento.

Exercícios

Exercício 1 - Tabuada Simples

Faça um programa que exibe a tabuada do 1 ao 10 de um número digitado pelo usuário. A saída do programa deverá ser no formato "2 x 2 = 4".

Exercício 2 - Investimento Financeiro

Faça um programa que pergunte o depósito inicial e a taxa de juros de um investimento financeiros. O programa deverá exibir o valor do saldo mês a mês para os 24 primeiros meses e o total ganho com juros no período. Considere juros compostos.

Exercício 3 - Caixa registradora

Faça um programa que leia um valor monetário e imprima a quantidade de cédulas necessárias para o pagamento desse valor ao cliente.

Exercício 4 - Crescimento Populacional

Supondo que a população de um país A seja da ordem de 80000 habitantes com uma taxa anual de crescimento de 3% e que a população de B seja 200000 habitantes com uma taxa de crescimento de 1.5%. Faça um programa que calcule e escreva o número de anos necessários para que a população do país A ultrapasse ou iguale a população do país B, mantidas as taxas de crescimento.

Exercício 5 - Programa de Lanchonete (**DIFÍCIL**)

O cardápio de uma lanchonete é o seguinte:

Especificação	Código	Preço
Cachorro Quente	100	R\$ 1,20
Bauri Simples	101	R\$ 1,30
Bauri com ovo	102	R\$ 1,50
Hambúrguer	103	R\$ 1,20
Cheeseburger	104	R\$ 1,30
Refrigerante	105	R\$ 1,00

Faça um programa que leia o código dos itens pedidos e as quantidades desejadas. Calcule e mostre o valor a ser pago por item (preço * quantidade) e o total geral do pedido. Considere que o cliente deve informar quando o pedido deve ser encerrado.