

## 2. Estruturas de Controle – Decisão e Repetição

As estruturas de controle são essenciais para que o computador possa **decidir** e **repetir** tarefas, desviando-se da execução sequencial padrão.

### Estruturas Condicionais (Decisão):

Permitem que comandos sejam executados ou ignorados com base no resultado de um teste lógico.

- **if:** Testa uma condição. Se for verdadeira (True), o bloco de código indentado a seguir é executado.
- **else:** Executa um bloco de código se a condição do if for falsa. É a cláusula "senão".
- **elif:** Permite testar múltiplas condições sequenciais, servindo como uma contração de *else if* e sendo preferível a múltiplos if aninhados, pois reduz a indentação e melhora a legibilidade.
- **Blocos de Comando:** São definidos exclusivamente pela **indentação** (reco, geralmente com a tecla TAB), diferente de outras linguagens que usam chaves {}.

### Operadores de Controle:

Para construir as condições e realizar cálculos, são utilizados diversos operadores:

- **Aritméticos:** Usados para operações matemáticas. Incluem + (adição/concatenação de strings), - (subtração), \* (multiplicação/repetição de strings), / (divisão), \*\* (potenciação, também disponível via pow(base, exp)), // (quociente inteiro da divisão) e % (resto da divisão inteira).
- **Comparação (Relacionais):** Produzem um valor lógico (True ou False). Incluem == (igual), != (diferente), > (maior que), < (menor que), >= (maior ou igual a) e <= (menor ou igual a).
- **Lógicos:** Combinam condições e realizam testes compostos. Os principais são and (E), or (OU) e not (NÃO). A **Tabela Verdade** é o recurso usado para definir o valor lógico de proposições complexas que utilizam esses conectivos.

### Estruturas de Repetição (Iteração):

Usadas quando é necessário repetir um conjunto de instruções.

- **while:** O laço de repetição continua a executar as instruções **enquanto** a condição for verdadeira. É mais adequado quando o número de iterações é **indeterminado**. É crucial que o laço inclua uma instrução para modificar a condição, evitando o **"loop infinito"**.
- **for:** Repete as instruções para **cada item** em uma sequência, sendo mais adequado quando o número de iterações é **definido**.
  - **Função range():** É comumente usada com o for para gerar uma sequência de números, definindo uma faixa. Pode receber 1 argumento (quantidade de números, começando em 0), ou 2/3 argumentos (início, fim (aberto) e opcionalmente o incremento ou razão da progressão aritmética).
- **break:** Comando usado para **quebrar** a execução de uma estrutura de repetição e forçar a saída do laço