

Algoritmos e Programação I

Módulo 4: estruturas de programação

Prof. Dr. Said Sadique Adi



Estruturas de programação

- A maioria das linguagens de programação oferecem um conjunto de estruturas padrões que facilitam a implementação de um programa:
 - **Estruturas sequenciais:** representam os blocos de instruções que sempre são executados e sequencialmente, na ordem em que foram codificados;
 - **Estruturas de decisão ou seleção:** permitem a escolha de um conjunto de instruções a serem executadas, escolha que depende de uma condição, representada por meio de expressões aritméticas, lógicas ou racionais;
 - **Estruturas de repetição:** permitem que um conjunto de instruções sejam executadas repetidas vezes.

Estruturas de decisão ou seleção

- São providas pela linguagem Python por meio do comando **if** (que significa **se** em português), cuja sintaxe é:

if <condição_do_if>:

**bloco indentado com as instruções a serem executadas
caso a condição do if seja verdadeira**

instruções após o if

Seleção de uma via

Estruturas de decisão ou seleção

- Quando da necessidade de executar blocos de forma alternativa, utilizamos o comando **if-else** (se-senão em português), cuja sintaxe é:

if <condição>:

bloco indentado com as instruções a serem executadas caso a condição seja verdadeira

else:

bloco indentado com as instruções a serem executadas caso a condição seja falsa

instruções após o if-else

Seleção de duas vias

Estruturas de decisão ou seleção

- Quando temos mais do que duas alternativas, podemos aninhar comandos do tipo **if-else**, seguindo a sintaxe abaixo:

if <condição 1>:

bloco com as instruções a serem executadas caso a condição 1 seja verdadeira

elif <condição 2>:

bloco com as instruções a serem executados caso a condição 2 seja verdadeira

elif <condição 3>:

bloco com as instruções a serem executados caso a condição 3 seja verdadeira

else :

bloco com as instruções a serem executados caso as condições 1, 2 e 3 sejam falsas

instruções após o if-elif-else

Seleção de três ou mais vias

Estruturas de repetição

- São providas pelo Python por meio dos comandos **for** e **while**
- Sintaxe do comando for:

for <variável> in <sequência>:

**bloco indentado com as instruções a serem executadas para
cada valor assumido pela variável**

instruções após o for

- A variável sequência deve ser de um tipo que possa ser iterado (string, lista, tupla, etc).
- No lugar da sequência pode-se usar a função range(start, stop, step).

Estruturas de repetição

- Sintaxe do comando while:

while <condição>:

**bloco indentado com as instruções a serem executadas
enquanto a condição for verdadeira.**

instruções após o while

- É útil quando não sabemos a priori quantas vezes o bloco indentado será executado.

Comandos relacionados

- **break:** quando inserido dentro de um laço força o programa a sair dele.
- **continue:** quando inserido dentro de um laço força o programa a iniciar uma nova iteração do laço a partir da linha em que ele foi inserido.
- **pass:** instrução válida que não faz absolutamente nada.

Exemplo de uso das estruturas de programação

```
numero = int(input("Digite um número positivo: "))
while(numero != 0):
    if numero % 2 == 0:
        print("O número", numero, "não é primo.")
    else:
        contador = 0;
        for i in range(1, numero + 1, 1):
            if numero % i == 0:
                contador = contador + 1
        if contador == 2:
            print("O número", numero, "é primo.")
        else:
            print("O número", numero, "não é primo.")
numero = int(input("Digite um número positivo: "))
```