

Tec. de Desenvolvimento de Algoritmos

MÉTODOS

Na aula passada...

- ✓ Algoritmos?
- ✓ Interpretador/Compilador
- ✓ Tipos de Dados
- ✓ Variáveis
- ✓ Entrada e saída
- ✓ Funções matemáticas



Exemplos de laboratório

Dadas as variáveis abaixo:

$$L = V$$

$$M = F$$

$$B = 2$$

$$C = 3$$

$$X = 2.0$$

$$Y = 10.0$$

Determine o resultado da avaliação das expressões a seguir:

$$\text{a) } X * (X + Y) =$$

$$\text{b) } X * Y + B * C =$$

$$\text{c) } X * (Y + B) * C =$$

$$\text{d) } L \text{ OU } M =$$

$$\text{e) } (L \text{ E } (\text{NÃO } M)) =$$

$$\text{f) } (L \text{ E } (\text{NÃO } M)) \text{ OU } (M \text{ E } (\text{NÃO } L)) =$$

$$\text{g) } X > Y \text{ E } C \leq B =$$

$$\text{h) } (B \geq 5) \text{ OU } ((C > X) \text{ E } (X - Y + B > 3 * Y)) =$$

Exemplos de laboratório

Dadas as variáveis abaixo:

$$L = V$$

$$M = F$$

$$B = 2$$

$$C = 3$$

$$X = 2.0$$

$$Y = 10.0$$

Determine o resultado da avaliação das expressões a seguir:

$$a) X * (X + Y) = 24.0$$

$$b) X * Y + B * C = 26.0$$

$$c) X * (Y + B) * C = 72.0$$

$$d) L \text{ OU } M = V$$

$$e) (L \text{ E } (\text{NÃO } M)) = V$$

$$f) (L \text{ E } (\text{NÃO } M)) \text{ OU } (M \text{ E } (\text{NÃO } L)) = V$$

$$g) X > Y \text{ E } C \leq B = F$$

$$h) (B \geq 5) \text{ OU } ((C > X) \text{ E } (X - Y + B > 3 * Y)) = F$$

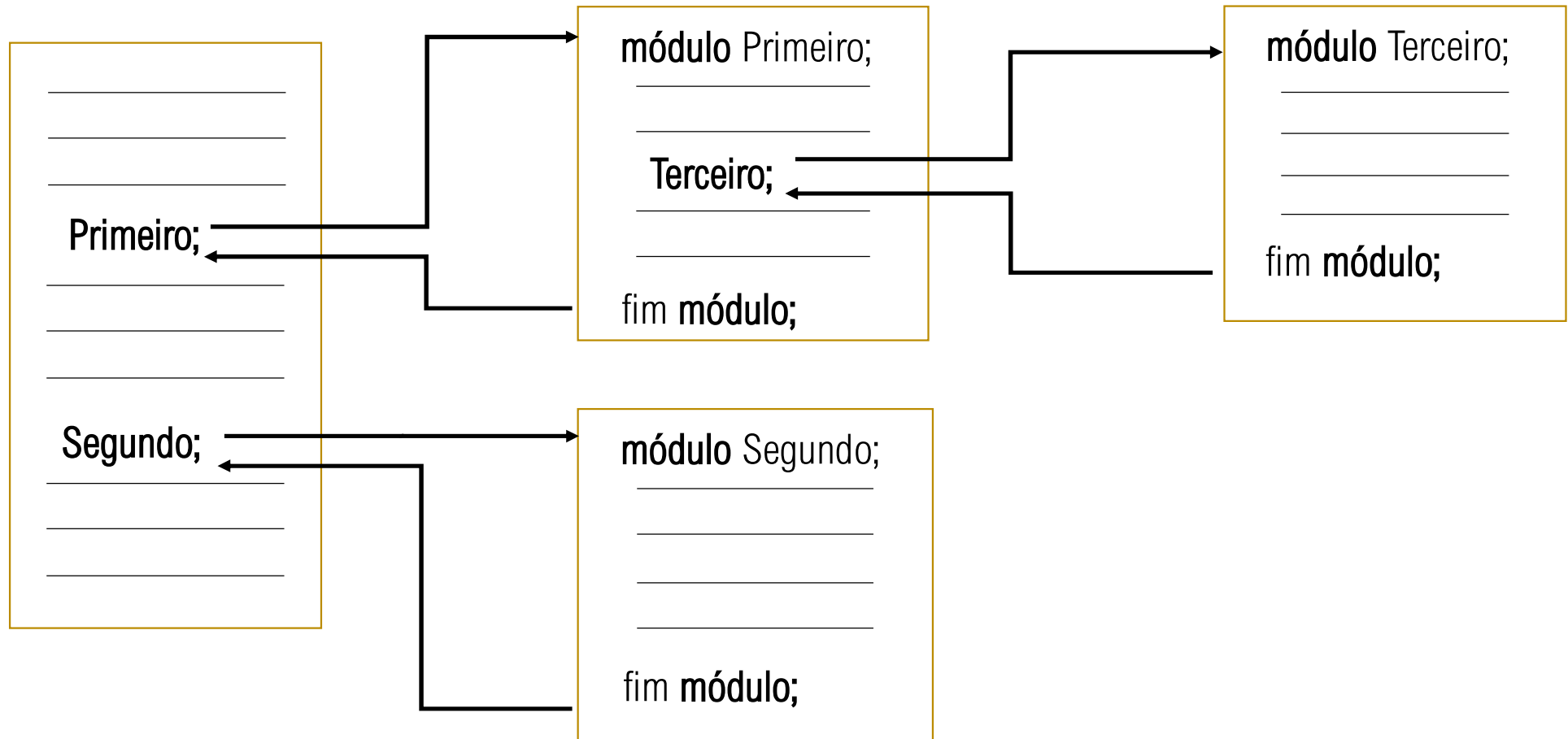
Métodos

- ✔ Um algoritmo pode ser simplificado quando dividido em várias sub-rotinas (métodos). Os métodos podem ser classificados em: **procedimentos** (sem retorno de valor) e **funções** (com retorno de valor).
- ✔ Quando um método é chamado por um algoritmo, ele é executado e ao seu término o controle de processamento retorna automaticamente para a primeira linha de instrução após a linha que efetuou a chamada do método.

Obs: **método** é um conceito muito utilizado na Programação Orientada a Objetos, para denominar, em forma geral, funções e procedimentos



Métodos

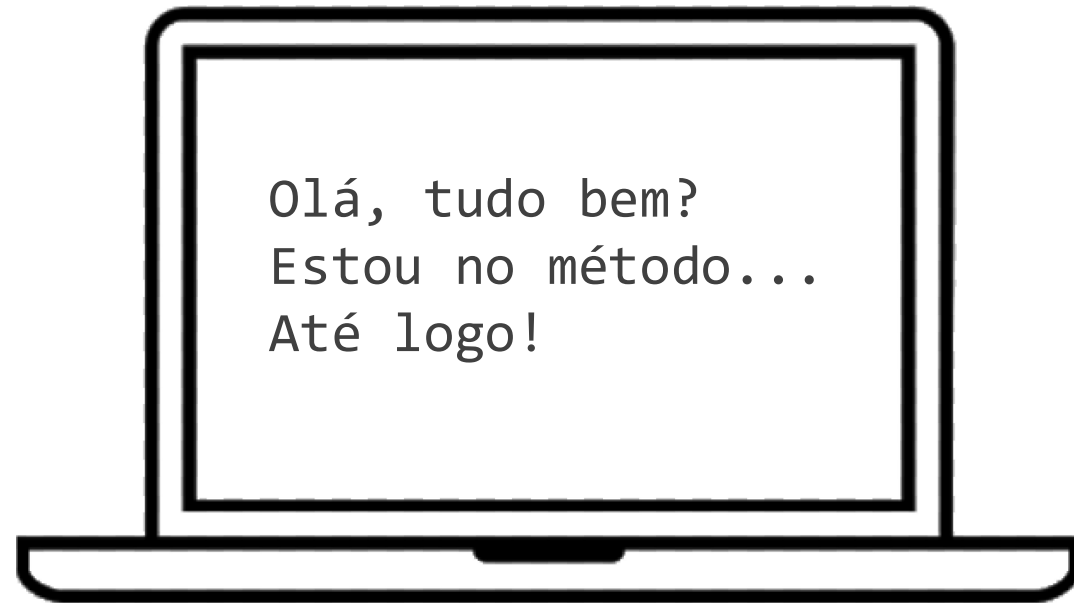


Exemplo – utilizando um método

```
algoritmo exemplo1
início
    exibirMensagens()
    escreva ("Até logo!")
fim

void exibirMensagens ()
início
    escreva ("Olá, tudo bem?")
    escreva ("Estou no método...")
fim
```

Na tela:



Este método é um procedimento, porque não retorna um valor.

Estrutura geral de um método (algoritmo)

tipo_retornado **nome_do_método** (**lista_de_parâmetros**)

inicio

declaração de variáveis (se necessário)

comandos

fim

onde:

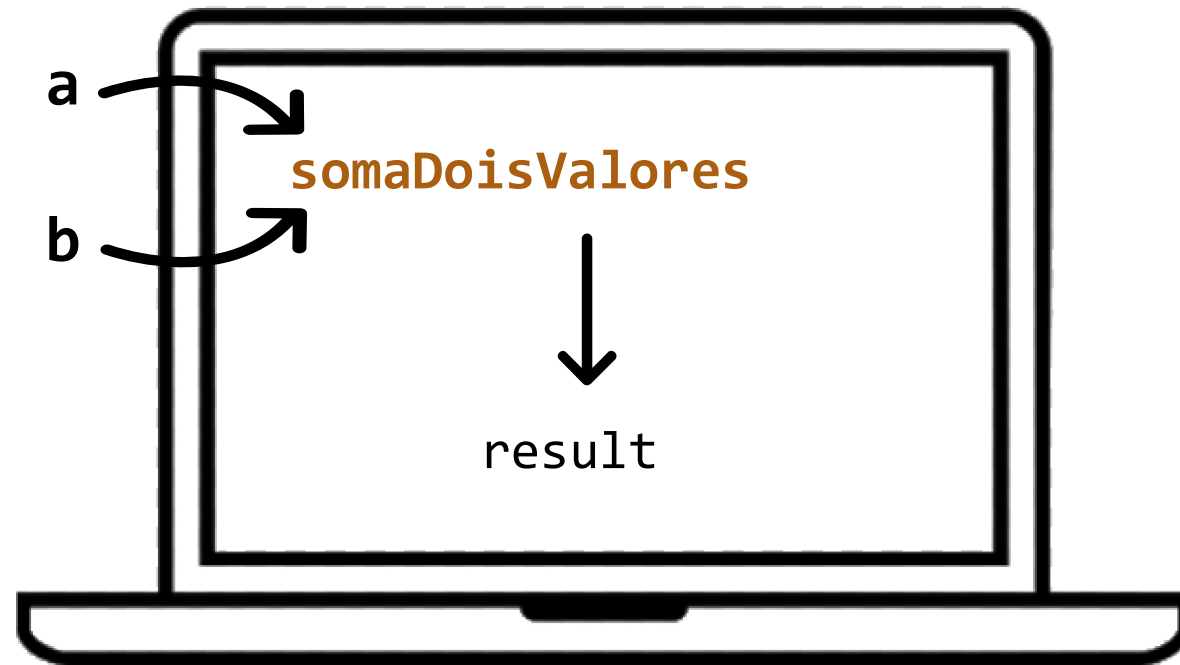
- ✔ o **tipo_retornado** pode ser inteiro, real, caractere, lógico ou até **void** (que significa que não retorna resultado nenhum)
- ✔ a **lista_de_parâmetros** especifica os dados que serão enviados para este método (a lista pode estar vazia)

Parâmetros por valor e por referência

- ✓ Parâmetro por valor: um valor será enviado para o método. O valor poderá ser alterado, mas não afetará a variável utilizada na chamada.
- ✓ Parâmetro por referência: neste caso, será enviada para o método uma referência (o endereço) da variável utilizada na chamada. Se o método alterar o valor deste parâmetro, o valor da variável utilizada na chamada também será modificado.
- ✓ Cada linguagem de programação estabelece como considerará os parâmetros por valor e por referência. Exemplo: Visual Basic utiliza ByVal e ByRef.
- ✓ Em Java, Python e outras linguagens de programação, os vetores e matrizes são considerados como parâmetros por referência.

Exemplo – um método que soma dois valores

```
real somaDoisValores (real a, real b)
  real result
  result ← a + b
  retorne ( result )
fim
```



Este método é uma função, porque retorna um valor.

Exemplo completo - pseudocódigo

algoritmo somarDoisValores

real x,y

início

escreva ("Entre com o 1º valor:")

leia (x)

escreva ("Entre com o 2º valor:")

leia (y)

escreva ("A soma é " + **somaDoisValores** (x, y))

fim

real **somaDoisValores** (**real** a, **real** b)

real result

 result ← a + b

retorne (result)

fim

→ entrada

→ retorno