

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:

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

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

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

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

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

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

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

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

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:

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

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

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

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

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

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

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

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

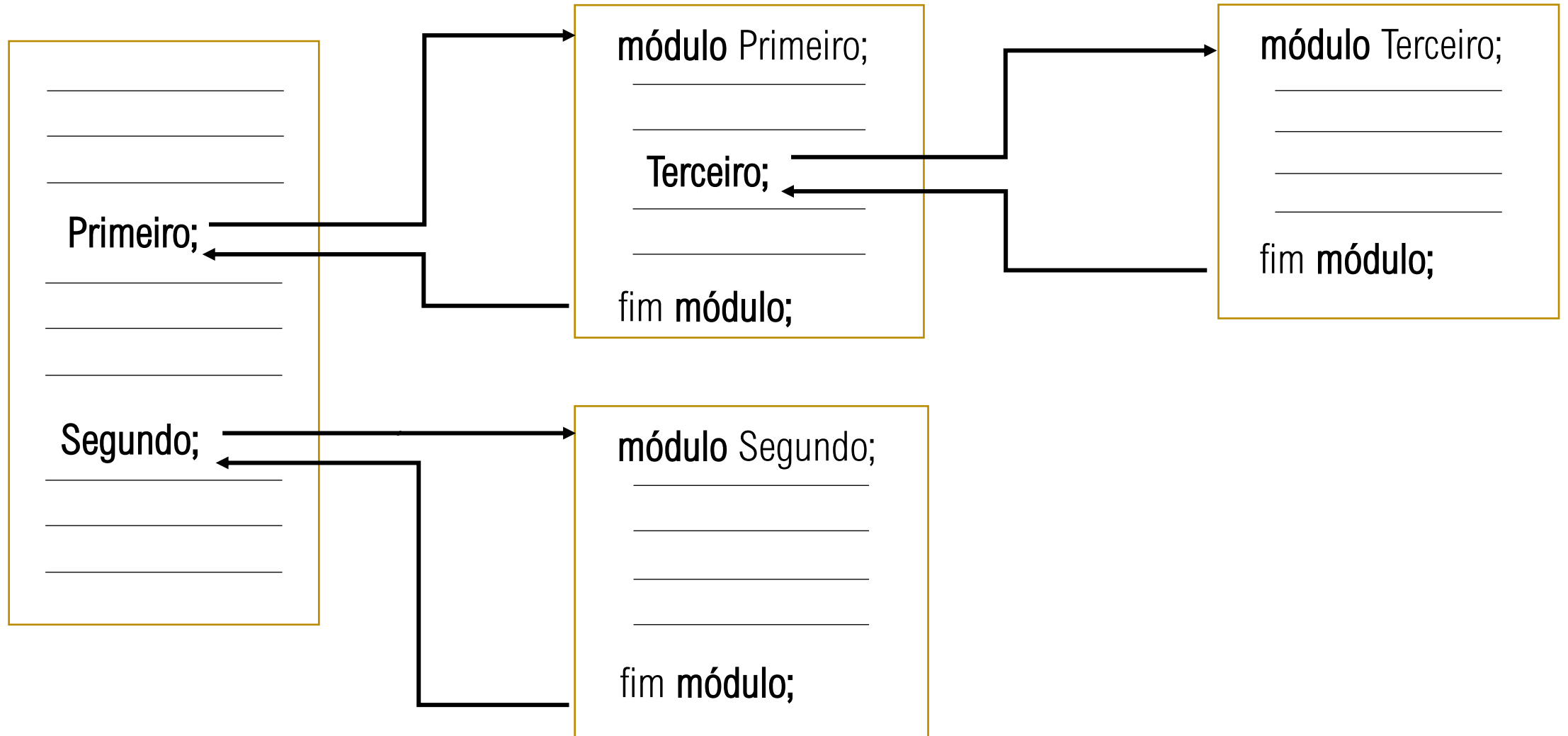
Métodos

- ✔ 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

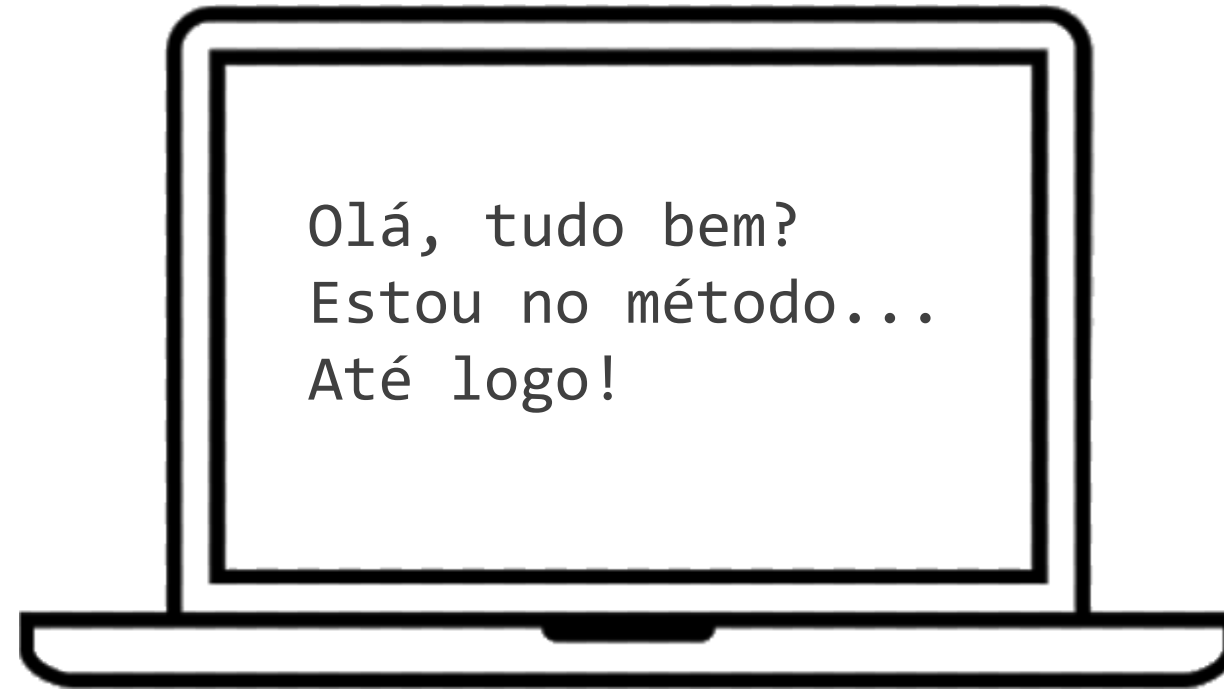


Exemplo – utilizando um método

Na tela:

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

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



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:

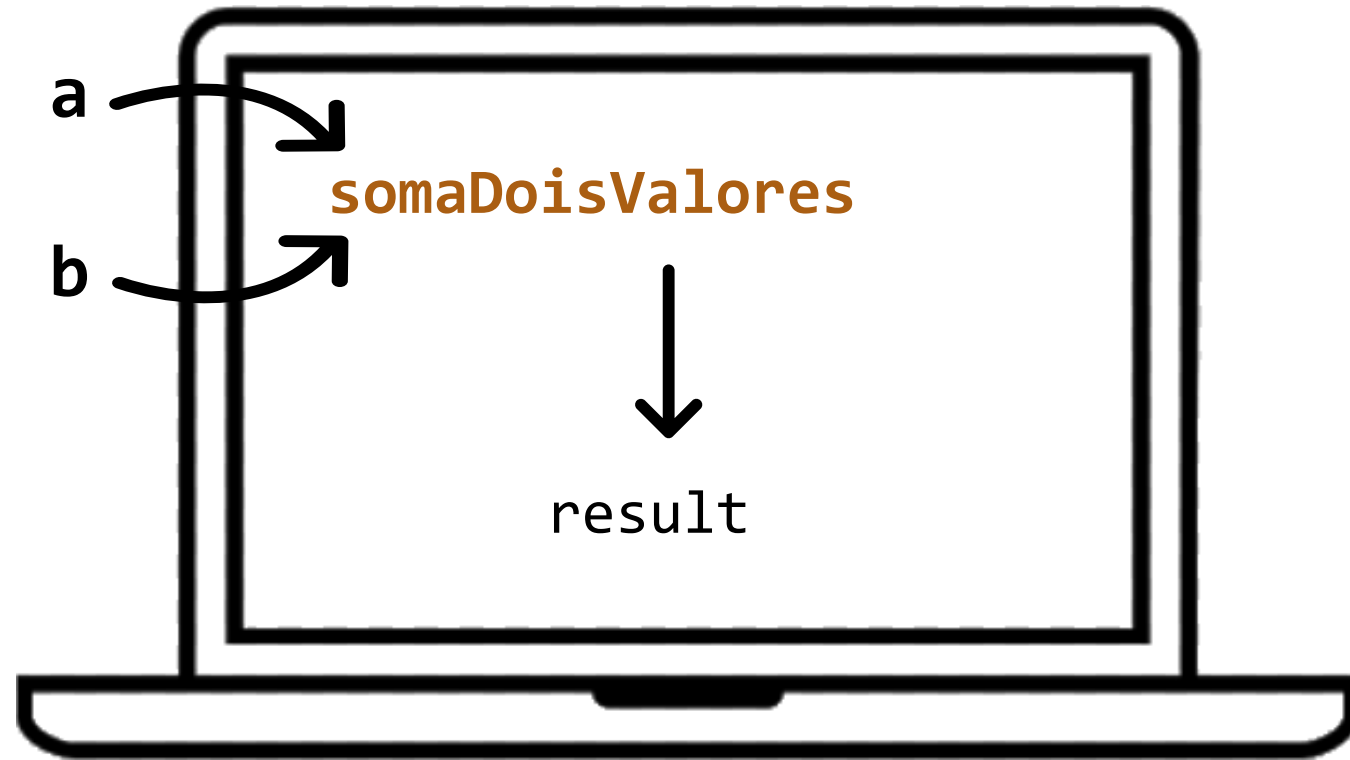
- ✔ o **tipo_retornado** pode ser inteiro, real, caractere, lógico ou até **void** (que significa que não retorna resultado nenhum)
- ✔ a **lista_de_parâmetros** especifica os dados que serão enviados para este método (a lista pode estar vazia)

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

Exemplo – soma de inteiros

```
algoritmo exemplo1
inteiro a, b, s
início
    escreva ("Entre com um número")
    leia (a)
    escreva ("Entre com um número")
    leia (b)
    s ← soma (a, b)
    escreva ("A soma é " + s)
fim

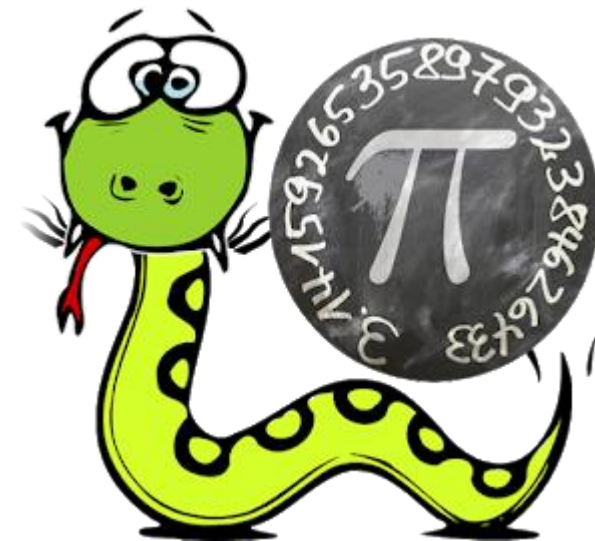
inteiro soma(inteiro n1, inteiro n2)
    inteiro res
    res ← n1+n2
    retorne (res)
fim
```

Pausa para programação

Exemplo: programa Python que utiliza uma função.
A função somaDoisValores deve ser declarada antes
de ser chamada.

```
def somaDoisValores(a, b):  
    result = a + b  
    return(result)
```

```
x = float(input("Entre com o 1º valor: "))  
y = float(input("Entre com o 2º valor: "))  
print("A soma dos valores é ", somaDoisValores(x, y) )
```



Pausa para programação

#Exemplo: programa Java que utiliza uma função.

```
import java.util.Scanner;
class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Entre com o 1º valor");
        float x = sc.nextFloat();
        System.out.println("Entre com o 2º valor");
        float y = sc.nextFloat();
        System.out.println("A soma é " + somaDoisValores(x,y));
    }
    public static float somaDoisValores (float a, float b){
        return a + b;
    }
}
```



Pausa para programação

#Exemplo: programa C# que utiliza uma função.

```
using System;
public class Program{
    public static void Main(){
        float x, y;
        Console.Write("Entre com o 1º valor");
        x = float.Parse(Console.ReadLine());
        Console.Write("Entre com o 2º valor");
        y = float.Parse(Console.ReadLine());
        Console.Write("A soma é " + somaDoisValores(x,y));
    }

    public static float somaDoisValores(float a, float b){
        return a+b;
    }
}
```



Exemplo – método *void*

algoritmo exemplo2

caractere msg

inteiro n

início

escreva ("Digite um número: ")

leia (n)

escreva ("Digite um texto: ")

leia (msg)

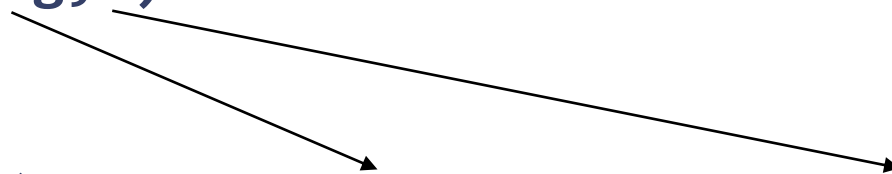
exibirMensagem(msg,n)

fim

void **exibirMensagem**(**caractere** **texto**, **inteiro** nro)

escreva ("Mensagem: " + **texto** + " número"+ **nro**)

fim



Comentários sobre a utilização de métodos

- ✓ Podemos utilizar tantos métodos como necessário.
- ✓ Um método poderá ser chamado qualquer quantidade de vezes.
- ✓ Um método poderá chamar a outro método.
- ✓ Um método poderá se autoexecutar (recursividade).



Vantagens da utilização de métodos

- ✓ Permitem dividir a lógica de um algoritmo em partes específicas.
- ✓ Facilitam a reutilização de código existente.
- ✓ A programação fica mais clara e organizada.
- ✓ A manutenção de programas fica mais fácil.



Métodos com retorno – Exemplo

Escreva um método com retorno que receba como parâmetros a base e a altura de um triângulo, calcule e retorne o valor de sua área.

$$\text{area} = \text{base} * \text{altura} / 2$$

Faça um algoritmo que solicite a base e altura de um triângulo ao usuário, e utilizando a função definida acima, calcule e mostre o valor da área.

Após o cálculo anterior, defina outros valores no código de base e altura e utilizando o mesmo método acima, mostre o valor da área.

Métodos com retorno – Exemplo

algoritmo Principal

início

real base, altura

escreva("Qual a base?")

leia(base)

escreva("Qual a altura?")

leia(altura)

escreva("Área " +
 calculaAreaTriangulo(base, altura))

escreva("Área " +
 calculaAreaTriangulo(27.5, 44.5))

fim

```
real calculaAreaTriangulo(real  
    base, real altura) {  
    real area  
    area = base * altura / 2  
    retorne (area)  
}
```

Método



Exercícios de aplicação

1- Faça um algoritmo que leia dois valores inteiros representando, respectivamente, um valor de hora e um de minutos e informe quantos minutos se passaram desde o início do dia. **Exemplo:**

valores lidos: 13 e 15

impressão: 795 minutos

2- Elabore um algoritmo que leia as dimensões de uma caixa (largura, altura e profundidade, os três valores fornecidos em centímetros) e calcule o volume dessa caixa em m^3 .

Obs: Efetue os cálculos solicitados nos exercícios anteriores dentro de **métodos** e **retorne** o valor calculado.

Resposta: Exercício 1 – versão A

algoritmo calculaMinutos

inteiro a, b, resmin

início

escreva ("Digite a quantidade de horas:")

leia (a)

escreva ("Digite a quantidade de minutos:")

leia (b)

 resmin ← **minutos**(a, b)

escreva ("A quantidade de minutos que passaram é " + resmin)

fim

inteiro minutos (**inteiro** horas, **inteiro** minut)

inteiro res

 res ← horas*60 + minut

retorne (res)

fim

Resposta: Exercício 1 – versão B

algoritmo calculaMinutos

inteiro a, b

início

escreva ("Digite a quantidade de horas:")

leia (a)

escreva ("Digite a quantidade de minutos:")

leia (b)

escreva ("A quantidade de minutos que passaram é “ + **minutos(a, b)**)

fim

inteiro **minutos** (**inteiro** horas, **inteiro** minut)

inteiro res

 res \leftarrow horas*60 + minut

retorne (res)

fim

Resposta: Exercício 2 – versão A

```
algoritmo volumeCaixa
real larg, alt, prof
início
    escreva ("Digite a largura da caixa em cm:")
    leia (larg)
    escreva ("Digite a altura da caixa em cm:")
    leia (alt)
    escreva ("Digite a profundidade da caixa em cm:")
    leia (prof)
    escreva ("O volume da caixa é " + volume(larg,alt,prof) + " m³" )
fim
real volume (real l, real a, real p)
    real res
    res = (l/100 * a/100 * p/100)
    retorne (res)
fim
```


Resposta: Exercício 2 – versão B

```
algoritmo volumeCaixa
real larg, alt, prof, vol
início
    escreva ("Digite a largura da caixa em cm:")
    leia (larg)
    escreva ("Digite a altura da caixa em cm:")
    leia (alt)
    escreva ("Digite a profundidade da caixa em cm:")
    leia (prof)
    vol = volume (larg,alt,prof)
    escreva ("O volume da caixa é " + vol + " m³")
fim
real volume (real l, real a, real p)
    real res
    res = (l/100 * a/100 * p/100)
    retorne (res)
fim
```

Exercícios de aplicação

3- Faça um algoritmo que leia dois valores inteiros representando, respectivamente, um valor de hora e um de minutos e informe quantos minutos se passaram desde o início do dia. **Exemplo:**

valores lidos: 13 e 15

impressão: 795 minutos

4- Elabore um algoritmo que leia as dimensões de uma caixa (largura, altura e profundidade, os três valores fornecidos em centímetros) e calcule o volume dessa caixa em m^3 .

Obs.: **Calcule e visualize** os resultados solicitados dentro de métodos. Nesta ocasião os métodos não retornam valor, eles calculam e visualizam os resultados.

Resposta: Exercício 3

```
algoritmo calculaMinutos
```

```
inteiro a, b
```

```
início
```

```
    escreva ("Digite a quantidade de horas:")
```

```
    leia (a)
```

```
    escreva ("Digite a quantidade de minutos:")
```

```
    leia (b)
```

```
    minutos(a, b)
```

```
fim
```

```
void minutos (inteiro horas, inteiro minut)
```

```
    inteiro res
```

```
    res = horas*60 + minut
```

```
    escreva ("A quantidade de minutos que passaram é " + res)
```

```
fim
```

Resposta: Exercício 4

```
algoritmo volumeCaixa
real larg, alt, prof
início
    escreva ("Digite a largura da caixa em cm:")
    leia (larg)
    escreva ("Digite a altura da caixa em cm:")
    leia (alt)
    escreva ("Digite a profundidade da caixa em cm:")
    leia (prof)
    volume(larg,alt,prof)
fim

void volume (real l, real a, real p)
    real res
    res = (l/100 * a/100 * p/100)
    escreva "O volume da caixa é " + res + " m³"
fim
```

Exercícios de fixação

1- Faça um método que receba como parâmetros o Km inicial, Km final, quantidade de litros gastos e preço do litro. **Calcule e mostre:**

- Distância percorrida;
- Consumo médio;
- Valor gasto;

Faça um algoritmo principal que solicite para o usuário o valor da quilometragem inicial, final, a quantidade de litros gastos e o preço do litro e mostre a distância percorrida, o consumo médio e o valor gasto, para isso utilize o método definido acima.

Exercícios de fixação

2- Escreva um método com retorno que receba como parâmetros os lados de um retângulo, calcule e retorne o valor de sua área.

$$\text{area} = \text{lado} * \text{lado}$$

Faça um algoritmo principal que solicite os valores dos lados de um retângulo ao usuário, e utilizando a função definida acima, calcule e mostre o valor de área.

3- Construir um método que receba como parâmetros o valor de uma compra e a quantidade de parcelas e calcule e **retorna** o valor da parcela, sabendo que a loja acrescenta 5% de juros para as compras parceladas.

No algoritmo principal, solicite para o usuário o valor de uma compra e a quantidade de parcelas e utilizando o método descrito acima, mostre o valor da parcela.

Exercícios de fixação

4- Elabore um algoritmo para calcular as raízes de uma equação de segundo grau ax^2+bx+c , conhecidos os coeficientes a , b , c .

Lembre que as raízes x_1 e x_2 são calculadas pela fórmula

$$x_{1,2} = \frac{-b \pm \sqrt{\Delta}}{2a} \quad \Delta = b^2 - 4.a.c$$

Elabore métodos separados, funções, `getDelta(a,b,c)`, `getX1(a,b,c)` e `getX2(a,b,c)`. Também, elabore o método principal para efetuar a leitura de dados e chamar às restantes funções.

Exercícios de fixação

5- Elabore um algoritmo para calcular a velocidade de três objetos diferentes (com velocidade constante).

Conhecemos (são dados digitados pelo usuário), para cada objeto, a distância percorrida e o tempo que necessitou para percorrer essa distância.

Utilize um método geral que **calcule e retorne** a velocidade de um objeto, fornecidos como parâmetros os dados de distância e tempo.

Exercícios de fixação

6- Escreva um programa que obtenha o valor de um produto, aplique neste valor um desconto de 9% e mostre o novo valor.

7- A loja de pregos Pregão comercializa dois tipos de pregos, o telheiro e o quadrado. Cada telheiro custa R\$ 1,05 e o quadrado custa R\$ 0,51. Ao final do dia, o gerente quer saber quanto arrecadou com a venda dos pregos do tipo telheiro e quadrado (juntos), e quanto deve separar de comissão (10% do total arrecadado). Faça um programa para obter as quantidades de pregos do tipo telheiro e quadrado que foram vendidos, calcule e mostre o valor arrecadado com as vendas e o valor da comissão.

Exercícios de fixação

8- Escreva um método chamado `valorCofre` que aceite três parâmetros inteiros chamados `vinteCinco`, `dez` e `cinco`, que conterão a quantidade de moedas de 25, 10 e 5 centavos respectivamente de um cofre. O método deve calcular e retornar o valor em reais (R\$).

Faça um algoritmo principal que solicite ao usuário a quantidade de moedas de 25, 10 e 5 centavos e usando o método definido acima, mostre o valor em reais.

Exercícios de fixação

9- Faça um procedimento que recebe como parâmetros o peso e a altura de uma pessoa, calcula e mostra o valor de seu IMC (índice de massa corporal). Sabendo que:

$$\text{IMC} = \text{peso} / \text{altura}^2.$$

Faça um algoritmo que solicite o peso e altura de uma pessoa, em seguida calcule e mostre o IMC utilizando o procedimento criado.

Exercícios de fixação

10- Escreva um método chamado `converteDolar` que recebe dois parâmetros reais, um é o valor total em reais (R\$) e o segundo a cotação do dólar. O método deve calcular e retornar o valor em Dólar (\$).

Faça um algoritmo principal que solicite ao usuário o valor que o mesmo deseja converter e a cotação do dólar e usando o método definido acima, mostre o valor em dólar.



That's all Folks!