

Aula 1:

Exercícios – Resolução

Disciplina: Fundamentos de Programação

Prof. Luiz Olmes

olmes@unifei.edu.br



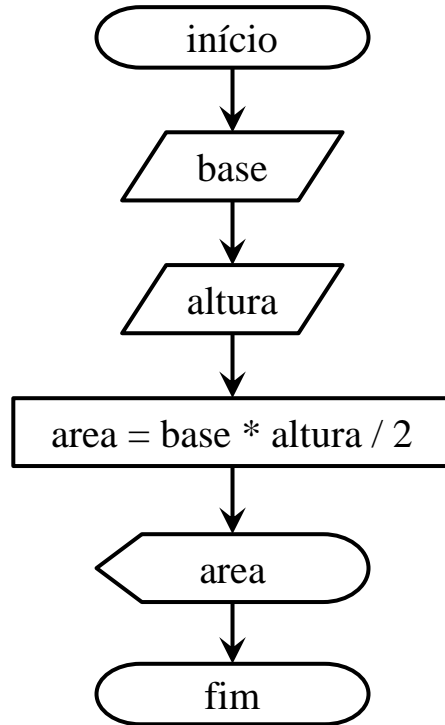
Exercícios – Resolução

- ▶ Nos slides seguintes, você encontra a resolução de alguns exercícios propostos na **Aula 01**.
- ▶ Primeiramente, é apresentada uma explicação sobre o exercício, e só após o seu fluxograma e pseudocódigo.
 - ▶ O professor não fez a descrição narrativa. Isso você consegue fazer sozinho(a)!
 - ▶ Quando o exercício é muito semelhante ao anterior, apenas a explicação será fornecida. Fluxograma e pseudocódigo serão quase idênticos aos indicados.
- ▶ As soluções apresentadas são **possíveis soluções**. Pode ser que você pensou de maneira distinta e sua resposta não ficou exatamente igual.
 - ▶ Procure o professor para discutir sua solução em caso de dúvidas!

1) Fazer um algoritmo para calcular a área de um triângulo. Sabe-se que a área do triângulo é dada pelo produto da base pela altura, dividido por 2.

- ▶ Neste exercício, a fórmula de cálculo é: $area = base * altura / 2$.
- ▶ No caso, como dados de entrada, temos as variáveis que representam a **base** e a **altura**. O “2” não é dado de entrada, pois é uma constante presente na fórmula.
- ▶ O valor de saída é dado pela variável **area**.
- ▶ Como **base**, **altura** e **area** podem assumir valores fracionários (com casas decimais), estes dados são números reais.

1) Fazer um algoritmo para calcular a área de um triângulo. Sabe-se que a área do triângulo é dada pelo produto da base pela altura, dividido por 2.



ALGORITMO “ex1”

VAR

base, altura, area : REAL

INÍCIO

LER base

LER altura

area = base * altura / 2

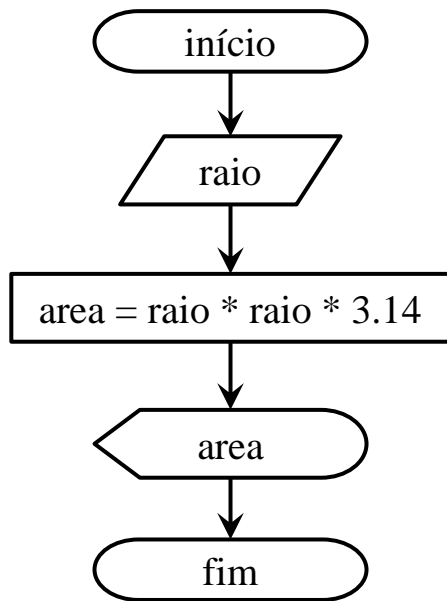
MOSTRAR area

FIM

2) Fazer um algoritmo para calcular a área de um círculo. Sabe-se que a área do círculo é dada pelo produto do raio ao quadrado pela constante $\pi = 3.14$. Nos fluxograma e pseudocódigo, represente π como 'pi'.

- ▶ Neste exercício, a fórmula de cálculo é: $\text{area} = \text{pi} * \text{raio}^2$.
- ▶ No caso, como dados de entrada, temos apenas o valor do **raio**. O “**pi**” não é dado de entrada, pois é uma constante presente na fórmula. O valor de saída é dado pela variável **area**.
- ▶ Representar raio ao quadrado como **raio^2** é coisa de Excel. Na linguagem C, potenciação é feita de outra forma. Porém, o quadrado do raio pode ser obtido por: **raio * raio**
- ▶ Como **area** é fruto de uma multiplicação por um valor fracionário ($\text{pi} = 3.14$), e a medida do **raio** também pode ser assumir valores fracionários (com casas decimais), estes dados são números reais.
- ▶ Nota: dentro das linguagens de programação, o separador de casas decimais é o ponto, não a vírgula.
 - ▶ **pi = 3.14**

2) Fazer um algoritmo para calcular a área de um círculo. Sabe-se que a área do círculo é dada pelo produto do raio ao quadrado pela constante $\pi = 3.14$. Nos fluxograma e pseudocódigo, represente π como 'pi'.



Nesta representação, não declaramos a constante “pi”, mas apenas usamos o seu valor diretamente na fórmula.

ALGORITMO “ex2”

VAR

raio, area : REAL

INÍCIO

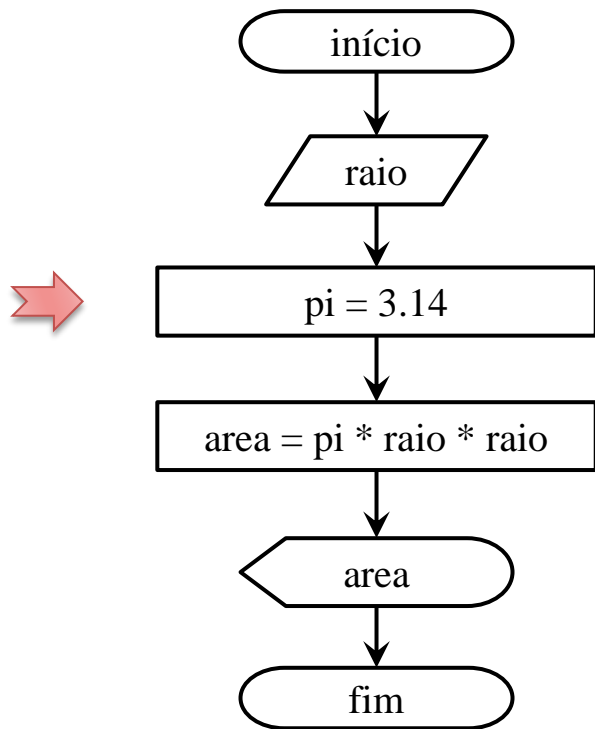
LER raio

area = raio * raio * 3.14

MOSTRAR area

FIM

2) Fazer um algoritmo para calcular a área de um círculo. Sabe-se que a área do círculo é dada pelo produto do raio ao quadrado pela constante $\pi = 3.14$. Nos fluxograma e pseudocódigo, represente π como 'pi'.



Alternativamente, caso você queira declarar a constante “pi” no código e atribuir valor a ela (o que é mais interessante de fazer), fica assim:

ALGORITMO “ex2”

VAR

raio, área, pi : REAL

INÍCIO

LER raio

➔ pi = 3.14

area = raio * raio * pi

MOSTRAR area

FIM

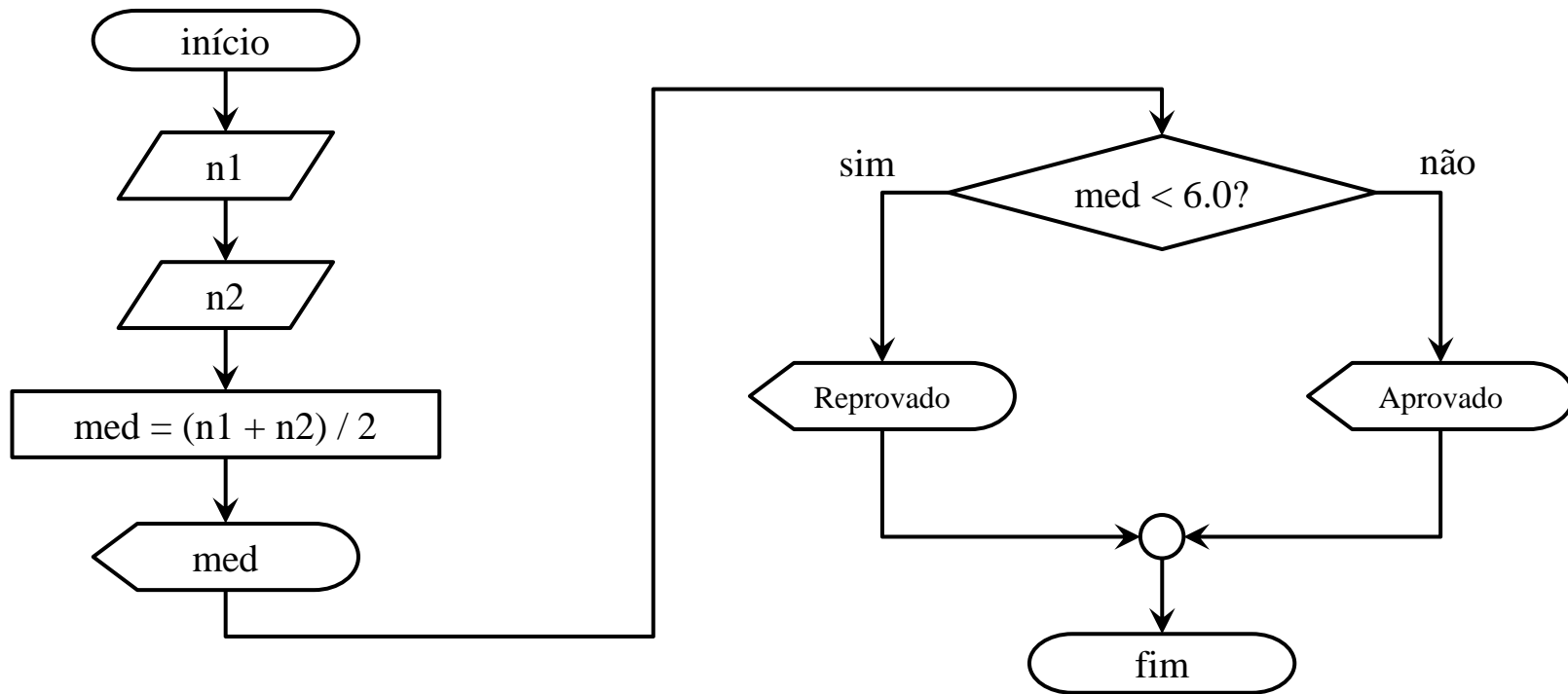
Exercícios 3 e 4

- ▶ São semelhantes aos **Exercícios 1 e 2**. Basta ler as variáveis de entrada, realizar um cálculo matemático e apresentar o resultado.
- ▶ No Exercício 3:
 - ▶ Entrada: o valor da temperatura em Celsius (real)
 - ▶ Saída: o valor da temperatura em Fahrenheit (real)
- ▶ No Exercício 4:
 - ▶ Entrada: o valor do ângulo em graus (real ou inteiro, tanto faz)
 - ▶ Saída: o valor do ângulo em radianos (obrigatoriamente real)

5) Fazer um algoritmo para calcular a média aritmética entre duas notas de um aluno e apresentar a sua situação (aprovado / reprovado), considerando que o mínimo para a aprovação seja 6.0 pontos.

- ▶ Como dados de entrada, temos as duas notas do aluno: $n1$ e $n2$.
- ▶ De posse do valor das notas, deve-se calcular a média (dado de saída): $med = (n1 + n2) / 2$.
 - ▶ Cuidado para não esquecer os parênteses na fórmula. Senão apenas $n2$ é dividido por 2, o que é incorreto.
- ▶ **A média não é dado de entrada.** De posse dos valores das duas notas, o algoritmo tem condições de realizar este cálculo. Então, este cálculo corresponde a uma etapa de processamento.
 - ▶ Em outras palavras, tudo o que o algoritmo pode computar sozinho não deve ser considerado dado de entrada.
- ▶ Após calcular a média, deve-se tomar uma decisão e mostrar a mensagem:
 - ▶ Se a média for menor que 6.0, o aluno está reprovado.
 - ▶ Caso contrário, ele está aprovado.

5) Fazer um algoritmo para calcular a média aritmética entre duas notas de um aluno e apresentar a sua situação (aprovado / reprovado), considerando que o mínimo para a aprovação seja 6.0 pontos.



5) Fazer um algoritmo para calcular a média aritmética entre duas notas de um aluno e apresentar a sua situação (aprovado / reprovado), considerando que o mínimo para a aprovação seja 6.0 pontos.

ALGORITMO “ex5”

VAR

n1, n2, med : REAL

INÍCIO

LER n1

LER n2

med = (n1 + n2) / 2

MOSTRAR med

SE med < 6.0 ENTÃO

 MOSTRAR “Reprovado”

SENÃO

 MOSTRAR “Aprovado”

FIM SE

FIM



5) Fazer um algoritmo para calcular a média aritmética entre duas notas de um aluno e apresentar a sua situação (aprovado / reprovado), considerando que o mínimo para a aprovação seja 6.0 pontos.

ALGORITMO “ex5”

VAR

n1, n2, med : REAL

INÍCIO

LER n1

LER n2

med = (n1 + n2) / 2

MOSTRAR med

SE med < 6.0 ENTÃO

 MOSTRAR “Reprovado”

SENÃO

 MOSTRAR “Aprovado”

FIM SE

FIM

O teste do valor da média é feito uma única vez. Se for menor que 6, significa reprovado.

Não é necessário testar novamente para verificar se a média é maior ou igual a 6, pois se a média não for menor que 6, a única possibilidade é ela ser maior ou igual.

5) Fazer um algoritmo para calcular a média aritmética entre duas notas de um aluno e apresentar a sua situação (aprovado / reprovado), considerando que o mínimo para a aprovação seja 6.0 pontos.

ALGORITMO “ex5”

VAR

n1, n2, med : REAL

INÍCIO

LER n1

LER n2

med = (n1 + n2) / 2

MOSTRAR med

SE med < 6.0 ENTÃO

MOSTRAR “Reprovado”

FIM SE

SE med >= 6.0 ENTÃO

MOSTRAR “Aprovado”

FIM SE

FIM

Se você testou duas vezes ($med < 6$ e $med \geq 6$), não significa que “está errado”. O algoritmo continua correto, porém você está executando um teste a mais, e isso começa a ficar estranho, redundante, no algoritmo (e em código). Lembre-se que a eficiência de um algoritmo é medida pela quantidade de operações que ele realiza. Quanto menos operações, mais rápido ele é.

5) Fazer um algoritmo para calcular a média aritmética entre duas notas de um aluno e apresentar a sua situação (aprovado / reprovado), considerando que o mínimo para a aprovação seja 6.0 pontos.

ALGORITMO “ex5”

VAR

n1, n2, med : REAL

INÍCIO

LER n1

LER n2

med = (n1 + n2) / 2

MOSTRAR med

SE med < 6.0 ENTÃO

MOSTRAR “Reprovado”

SENÃO

SE med >= 6.0 ENTÃO

MOSTRAR “Aprovado”

FIM SE

FIM SE

FIM

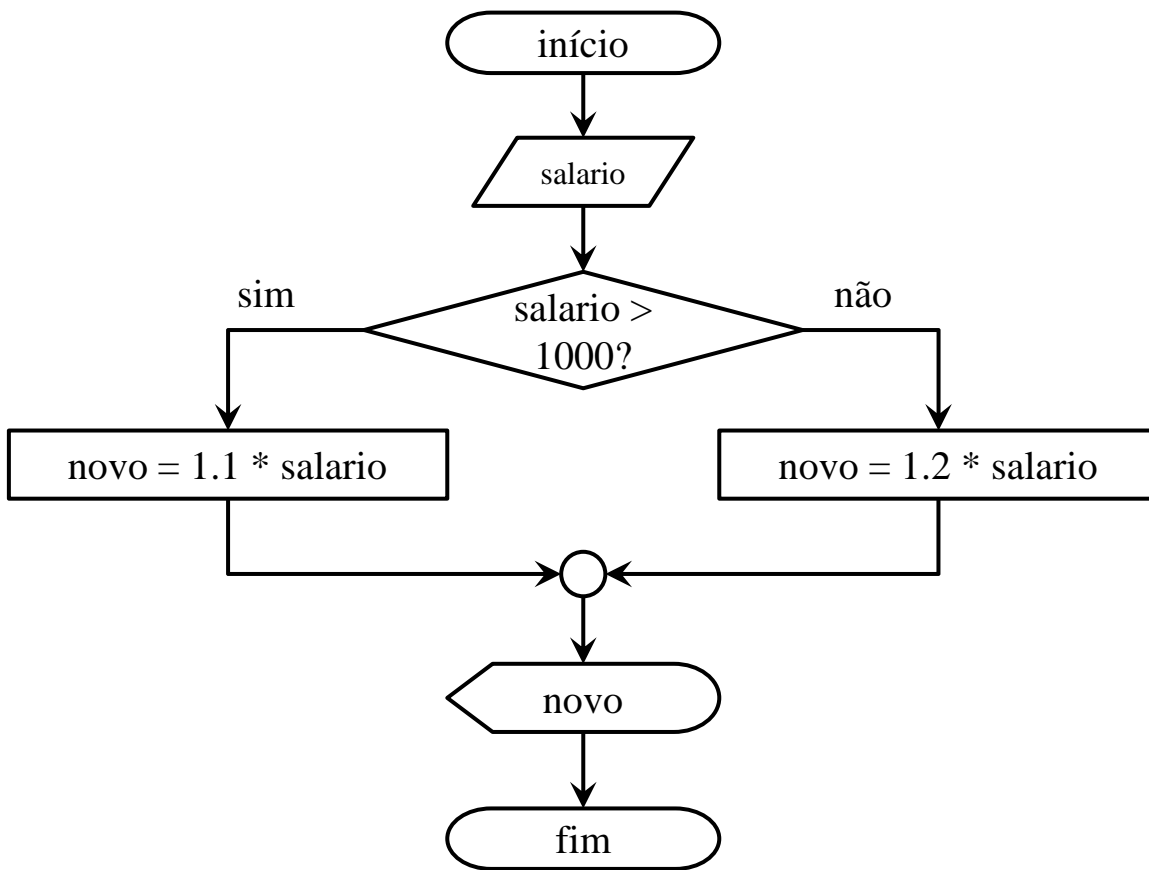
Colocar o teste de “Aprovado” dentro do tratamento da parte negativa, representada pelo SENÃO, também vai produzir o mesmo resultado. Porém, aqui já é algo muito estranho (bizarro!) de ser feito, pela razão descrita no Slide 12.

Funciona? Sim... Porém, escrever algoritmos e códigos dessa forma aparenta ser algo grosseiro, rude, amador...

6) Fazer um algoritmo para calcular o novo salário de um funcionário. Sabe-se que os funcionários que ganham até R\$ 1000,00 terão aumento de 20%, e os demais terão aumento de 10%.

- ▶ Neste exercício, o único dado de entrada é o salário do funcionário. A variável **salario** é, obrigatoriamente, um número real, pois pode conter parte fracionária (centavos).
 - ▶ O novo salário é calculado com base em uma porcentagem. Para aqueles que ganham até R\$ 1000.00, o cálculo fica:
 - ▶ $\text{novo} = 1.2 * \text{salario}$
 - ▶ Ou:
 - ▶ $\text{novo} = \text{salario} + 0.2 * \text{salario}$
 - ▶ Para os demais funcionários, a operação matemática é semelhante, porém considerando 10%.
 - ▶ $\text{novo} = 1.1 * \text{salario}$ OU $\text{novo} = \text{salario} + 0.1 * \text{salario}$
 - ▶ Pode-se usar a mesma variável **novo** em ambas as divisões da tomada de decisão, pois apenas o cálculo que se altera. **Cuidado:** a saída é a variável **novo**, e não a variável **salario**.
-

6) Fazer um algoritmo para calcular o novo salário de um funcionário. Sabe-se que os funcionários que ganham até R\$ 1000,00 terão aumento de 20%, e os demais terão aumento de 10%.



ALGORITMO “ex6”

VAR

salario, novo: REAL

INÍCIO

LER salario

SE salario > 1000.0 ENTÃO

novo = 1.1 * salario

SENÃO

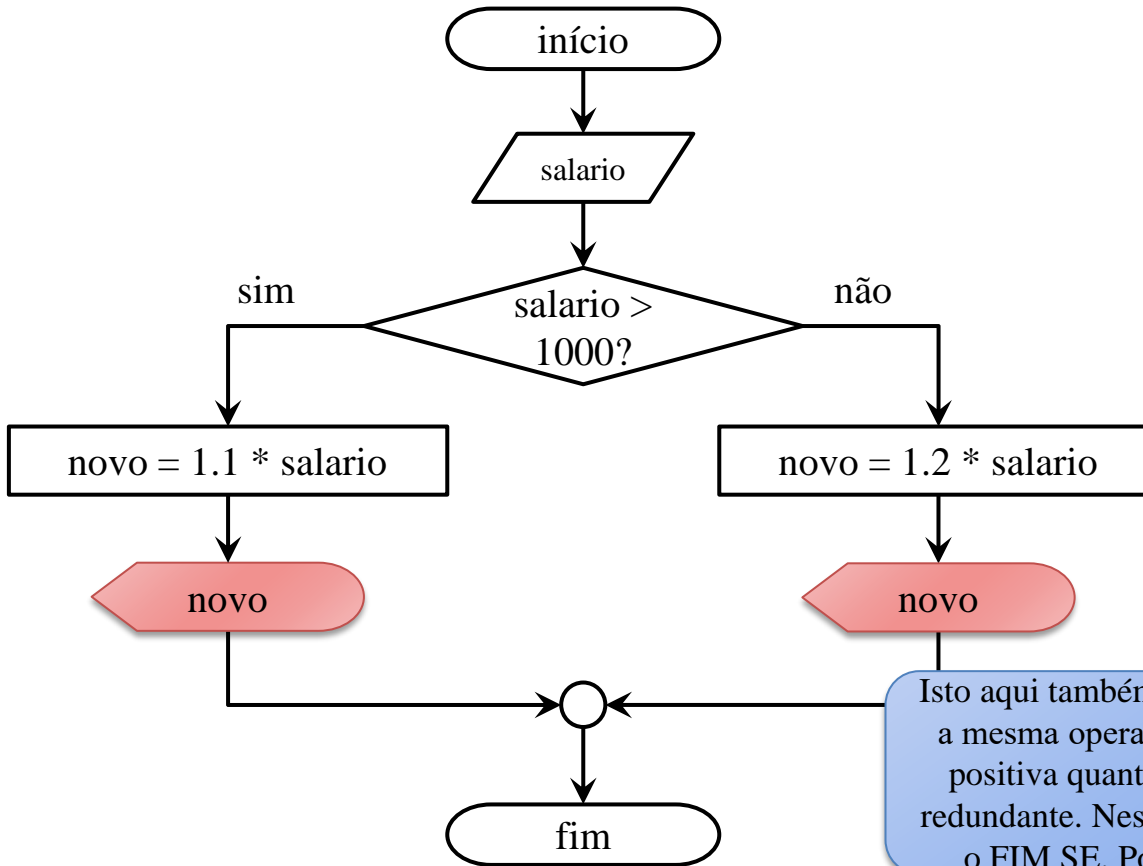
novo = 1.2 * salario

FIM SE

MOSTRAR novo

FIM

6) Fazer um algoritmo para calcular o novo salário de um funcionário. Sabe-se que os funcionários que ganham até R\$ 1000,00 terão aumento de 20%, e os demais terão aumento de 10%.



ALGORITMO “ex6”

VAR

salario, novo: REAL

INÍCIO

LER salario

SE salario > 1000.0 ENTÃO

novo = 1.1 * salario

MOstrar novo

SENÃO

novo = 1.2 * salario

MOstrar novo

FIM SE

FIM

Isto aqui também produz o mesmo resultado final. Porém, note que a mesma operação “**MOstrar novo**” é realizada tanto na parte positiva quanto na parte negativa (ENTÃO e SENÃO), o que é redundante. Nesse caso, ela pode ser realizada uma única vez, após o FIM SE. Portanto, a versão do slide anterior é bem melhor.

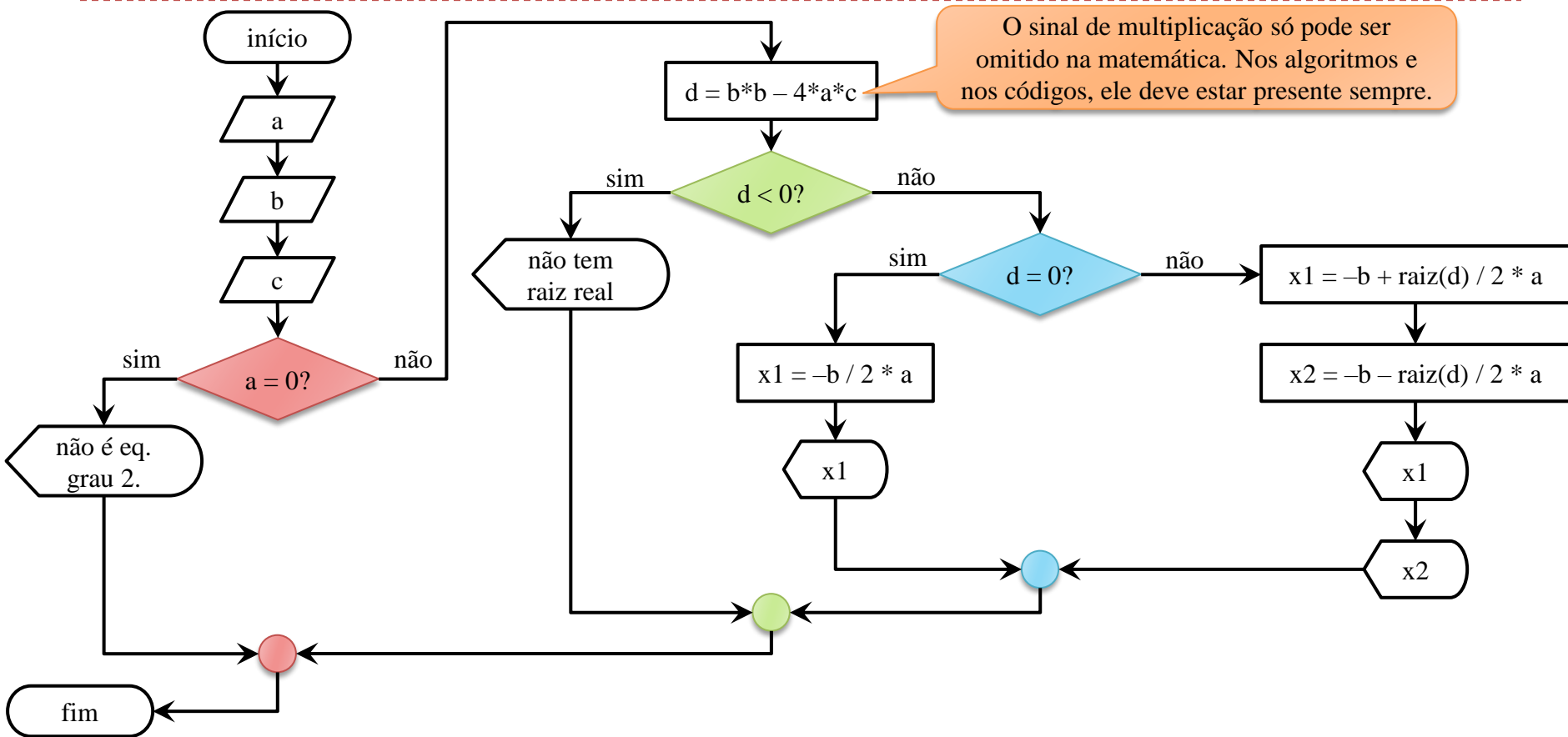
7) Fazer um algoritmo para resolver uma equação do primeiro grau.

- ▶ É semelhante aos exercícios 1, 2, 3 e 4.
- ▶ Equações de 1º grau têm a forma: $ax + b = 0$
- ▶ Basta ler os valores de a e b e calcular o valor de x através da fórmula: $x = -b / a$
 - ▶ Entradas: a e b (reais)
 - ▶ Saída: x (real)
- ▶ O problema deste exercício é que o valor de a tem que ser diferente de zero, pois será usado no denominador. Procede-se como no **Exemplo 5** da **Aula 01**.

8) Fazer um algoritmo para resolver uma equação do segundo grau. Seu algoritmo deve verificar se a equação é realmente de segundo grau e tratar o valor do discriminante delta adequadamente.

- ▶ Equações de 2º grau têm a forma: $ax^2 + bx + c = 0$
- ▶ Basta ler os valores de a , b e c e calcular o valor de x através da fórmula de Bhaskara, onde:
 - ▶ Entradas: a , b e c (reais)
 - ▶ Saída: $x1$ e $x2$ (reais)
- ▶ O valor de d (delta) não é entrada. Como no **Exercício 5**, ele pode ser calculado pelo algoritmo. Portanto, trata-se de uma etapa de processamento.
- ▶ O problema deste exercício é que o valor de a tem que ser diferente de zero, senão não é equação de grau 2. Procede-se como no **Exemplo 5** da **Aula 01**.

8) Fazer um algoritmo para resolver uma equação do segundo grau. Seu algoritmo deve verificar se a equação é realmente de segundo grau e tratar o valor do discriminante delta adequadamente.



8) Fazer um algoritmo para resolver uma equação do segundo grau. Seu algoritmo deve verificar se a equação é realmente de segundo grau e tratar o valor do discriminante delta adequadamente.

```
1. ALGORITMO "ex8"
2. VAR
3.     a, b, c, d, x1, x2: REAL
4. INÍCIO
5.     LER a
6.     LER b
7.     LER c
8.
9.     SE a < 0 ENTÃO
10.        MOSTRAR "não é eq. grau 2"
11.    SENÃO
12.        d = b * b - 4 * a * c
13.        SE d < 0 ENTÃO
14.            MOSTRAR "não tem raiz real"
15.        SENÃO
16.            SE d = 0 ENTÃO
17.                x1 = -b / 2 * a
18.                MOSTRAR x1
19.            SENÃO
20.                x1 = -b + raiz(d) / 2 * a
21.                x2 = -b - raiz(d) / 2 * a
22.                MOSTRAR x1
23.                MOSTRAR x2
24.            FIM SE
25.        FIM SE
26.    FIM SE
27. FIM
```

8) Fazer um algoritmo para resolver uma equação do segundo grau. Seu algoritmo deve verificar se a equação é realmente de segundo grau e tratar o valor do discriminante delta adequadamente.

```
1. ALGORITMO "ex8"
2. VAR
3.     a, b, c, d, x1, x2: REAL
4. INÍCIO
5.     LER a
6.     LER b
7.     LER c
8.
9.     SE a < 0 ENTÃO
10.        MOSTRAR "não é eq. grau 2"
11.     SENÃO
12.        d = b * b - 4 * a * c
13.        SE d < 0 ENTÃO
14.            MOSTRAR "não tem raiz real"
15.        SENÃO
16.            SE d = 0 ENTÃO
17.                x1 = -b / 2 * a
18.                MOSTRAR x1
19.            SENÃO
20.                x1 = -b + raiz(d) / 2 * a
21.                x2 = -b - raiz(d) / 2 * a
22.                MOSTRAR x1
23.                MOSTRAR x2
24.            FIM SE
25.        FIM SE
26.     FIM SE
27. FIM
```

Cada losango do fluxograma representa um SE, e cada conector representa um FIM SE. Eles estão mostrados no pseudocódigo na mesma cor em que estão no fluxograma.

ENTÃO corresponde ao lado “sim” do losango.
SENÃO corresponde ao lado “não” do losango.

Os recuos à direita indicam qual estrutura está dentro de qual estrutura. Por exemplo, no fluxograma, o primeiro teste é: **a < 0?**. O lado “não” desta tomada de decisão contém todo o restante do algoritmo. Por isso, no pseudocódigo, há um recuo à direita para todo o código que está dentro do SENÃO deste teste, mostrado em vermelho. O mesmo acontece para os demais testes.

9) Fazer um algoritmo que receba o ano de nascimento de uma pessoa e o ano atual (dois valores de entrada). A seguir, calcular e mostrar: a) a idade da pessoa em anos. b) a idade da pessoa em meses. c) a idade da pessoa em dias.

- ▶ Aqui tem-se duas variáveis de entrada: **anoNasc** e **anoAtual** (inteiras).
 - ▶ Não pode haver espaço no nome das variáveis.
- ▶ Não podemos considerar o **anoAtual** como 2023, fixo no cálculo que o algoritmo realiza, como se fosse uma constante. Se assim for feito, quando estivermos em 2024, 2025, etc., o algoritmo vai produzir resultados errados. Por isso, o ano atual deve ser uma variável de entrada.
- ▶ Há três variáveis de saída:
 - ▶ **idadeAnos** = $\text{anoAtual} - \text{anoNasc}$
 - ▶ **idadeMeses** = $\text{idadeAnos} * 12$
 - ▶ **idadeDias** = $\text{idadeAnos} * 365$ (ignore anos bissextos, senão esta questão fica muito difícil...)
- ▶ Fluxograma e pseudocódigo são semelhantes aos dos **Exercícios 1 a 4** (ler variável, fazer uma continha e mostrar a resposta).

Exercícios 10 e 11

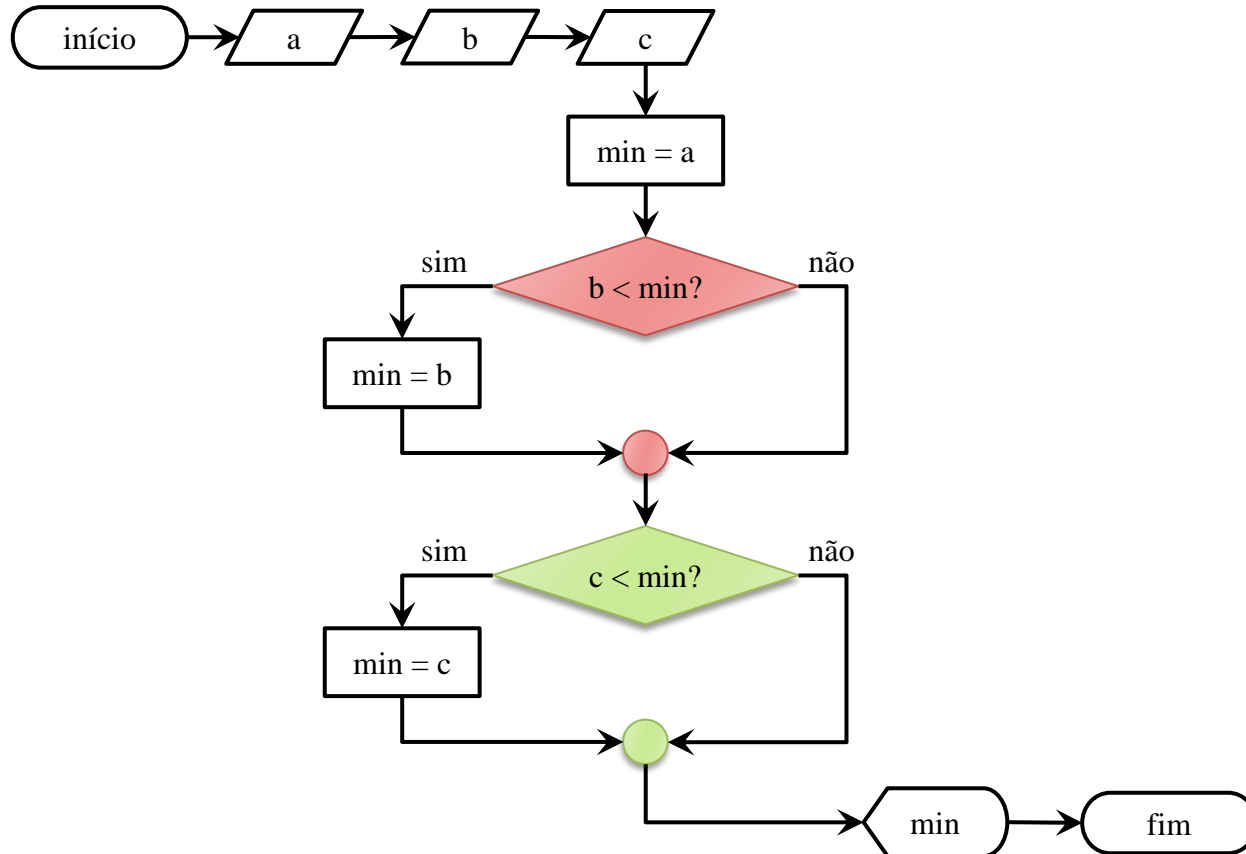
- ▶ **Exercício 10:** uma única tomada de decisão resolve esta questão.
- ▶ **Exercício 11:** o mais interessante desta lista. O **Exercício 8** pode ser grande, mas não tem nada de desafiador, até porque você já sabe resolver equações de grau 2 desde o Ens. Fundamental. Basta apenas não se perder no fluxograma do 8.
- ▶ No caso do **Exercício 11**, uma possível solução, que não é a melhor, é testar os 3 valores A, B, e C da seguinte forma:
 - ▶ $A > B$ e $A > C...$ $B > A$ e $B > C...$ Sempre testando todas as possibilidades.

Entretanto, o **Exercício 11** pode admitir outras variações. Suponha que ao invés de 3 valores, fossem 10 valores... Se formos usar o símbolo de tomada de decisão para testar cada possibilidade, com 10 valores, o fluxograma ficaria excessivamente enorme, quase precisando de uma parede inteira para ser escrito!

11) Fazer um algoritmo que receba três números como entrada e mostre qual o menor deles.

- ▶ A solução mais elaborada do **Exercício 11** é a seguinte:
- ▶ Considere uma variável auxiliar para armazenar o valor do menor número. Vamos chamá-la de **min** (mínimo).
- ▶ Após ler as três entradas (**a**, **b** e **c**), **suponha** que qualquer um dos valores seja o valor mínimo. Você pode considerar qualquer uma. Na solução apresentada, pegamos o primeiro deles: **a**
 - ▶ **Isto é uma suposição!** A variável **a** pode ser a menor, mas pode não ser. As comparações seguintes é que irão determinar se essa suposição é verdadeira ou não.
- ▶ Agora basta verificar se as outras duas (**b** e **c**) são menores que o mínimo (sendo que **min = a** no início, devido à suposição). Se for, realiza-se a troca pelo menor valor até o momento. Ao fim, todas as três variáveis terão sido analisadas e o menor valor encontrado.

11) Fazer um algoritmo que receba três números como entrada e mostre qual o menor deles.



```
ALGORITMO "ex11"
VAR
    a, b, c, min: INTEIRO
INÍCIO
    LER a
    LER b
    LER c

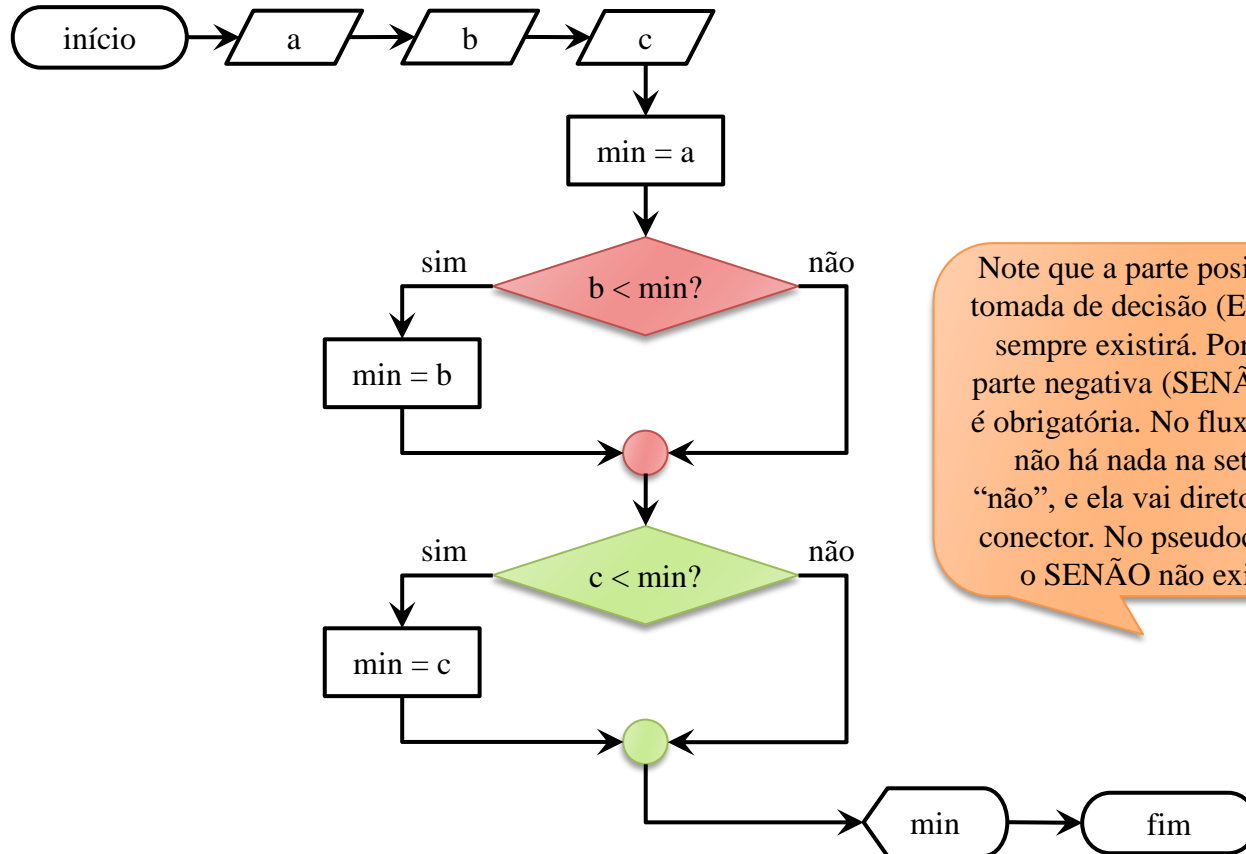
    min = a

    SE b < min ENTÃO
        min = b
    FIM SE

    SE c < min ENTÃO
        min = c
    FIM SE

    MOSTRAR min
FIM
```

11) Fazer um algoritmo que receba três números como entrada e mostre qual o menor deles.



Note que a parte positiva da tomada de decisão (ENTÃO) sempre existirá. Porém, a parte negativa (SENÃO) não é obrigatória. No fluxograma não há nada na seta de “não”, e ela vai direto para o conector. No pseudocódigo, o SENÃO não existe.

```
ALGORITMO “ex11”
VAR
    a, b, c, min: INTEIRO
INÍCIO
    LER a
    LER b
    LER c

    min = a

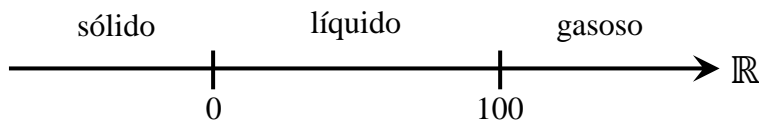
    SE b < min ENTÃO
        min = b
    FIM SE

    SE c < min ENTÃO
        min = c
    FIM SE

    MOSTRAR min
FIM
```

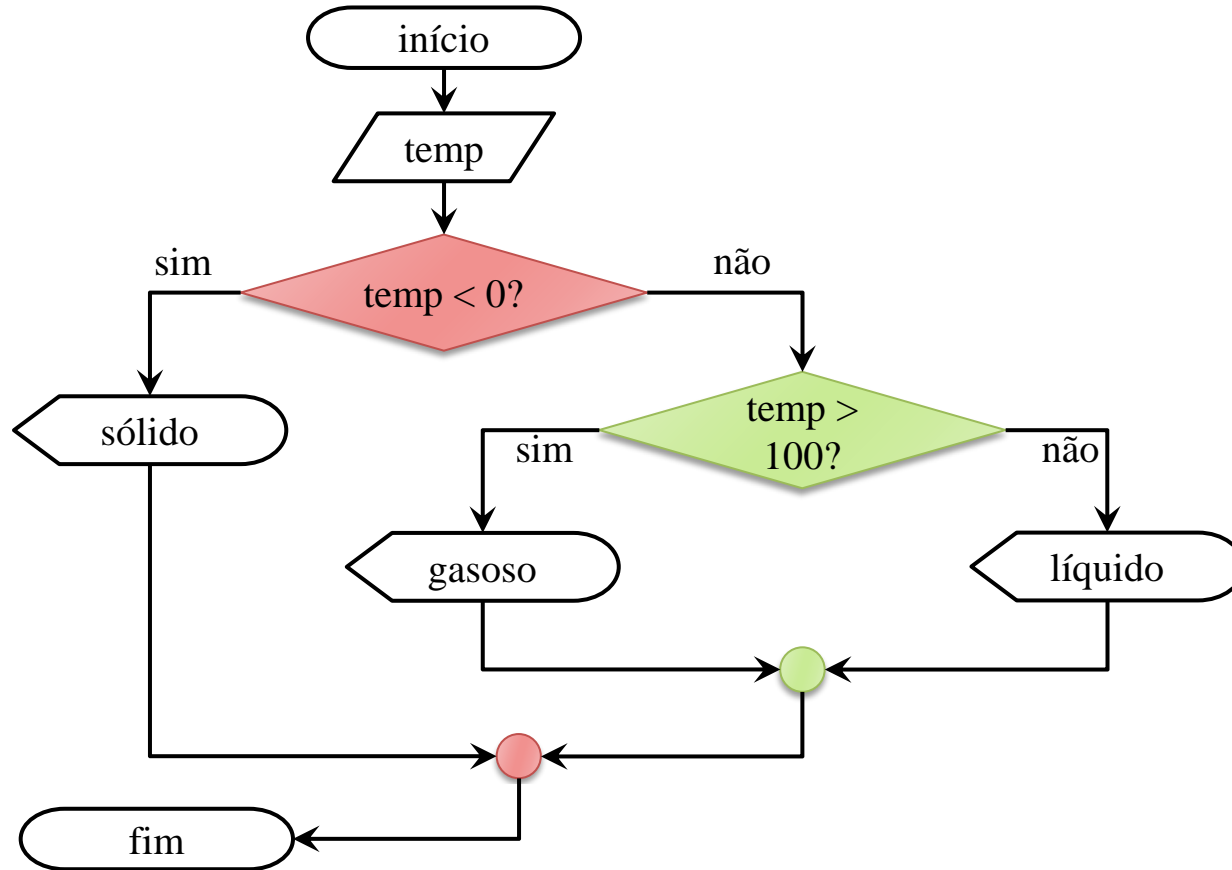
12) Fazer um algoritmo que receba um número representando a temperatura da água e mostre em qual estado físico (sólido, líquido ou gasoso) ela se encontra.

- ▶ Nesta questão, existe apenas um valor de entrada, que é a temperatura da água (**temp**). A temperatura pode admitir valores fracionários, então é um valor real.
- ▶ Este tipo de questão é bem comum. Nela, teremos duas marcações na reta dos números reais (zero e 100 graus), o que a divide em 3 partes (sólido, líquido, gasoso):



- ▶ O truque para fazer comparação de valores mais facilmente é:
 - ▶ Resolver os dois extremos: sólido e gasoso.
 - ▶ O que sobrar, isto é, a parte negativa (último SENÃO), é a parte do meio.
 - ▶ Dessa forma, você evita ter que comparar dois valores para o estado líquido: se é maior que 0 e ao mesmo tempo menor que 100 (até porque não aprendemos a fazer isso ainda...).

12) Fazer um algoritmo que receba um número representando a temperatura da água e mostre em qual estado físico (sólido, líquido ou gasoso) ela se encontra.



ALGORITMO “ex12”

VAR

temp: REAL

INÍCIO

LER temp

SE temp < 0 ENTÃO
MOSTRAR “sólido”

SENÃO

SE temp > 100 ENTÃO
MOSTRAR “gasoso”

SENÃO

MOSTRAR “líquido”

FIM SE

FIM SE

FIM

13) Fazer um algoritmo que receba um número representando a temperatura corporal de um paciente. Caso o valor seja maior que 39, mostre a mensagem “Febre alta”. Caso o valor esteja entre 38 e 39, mostre a mensagem “Febre baixa”. Caso contrário, mostre a mensagem “Temperatura Normal”.

- ▶ Este exercício é idêntico ao anterior.
- ▶ O que se altera são apenas as marcações na reta dos números reais e as mensagens de saída.

Aula 1:

Exercícios – Resolução

Disciplina: Fundamentos de Programação

Prof. Luiz Olmes

olmes@unifei.edu.br

