

Segundo Exercício-Programa

Norton Trevisan Roman
Luciano Antonio Digiampietri

Prazo máximo para a entrega: 06/05/2020

1 Cálculo da Raiz Quadrada

Nesse exercício prático você não poderá usar o método `Math.sqrt()`. Em vez disso, você criará um método que calcula \sqrt{x} usando o método de Newton. Segundo esse método, uma aproximação da raiz quadrada de um número $a > 0$ pode ser calculada conforme descrito pelas seguintes equações:

$$x_0 \leftarrow \frac{a}{2}$$
$$x_{i+1} \leftarrow \frac{1}{2} \left(x_i + \frac{a}{x_i} \right)$$

para $i = 0, 1, 2, \dots$

Ou seja, obtemos x_1 fazendo $\frac{1}{2} \left(x_0 + \frac{a}{x_0} \right)$, x_2 fazendo $\frac{1}{2} \left(x_1 + \frac{a}{x_1} \right)$, e assim por diante. Note que quanto mais alto o valor de i , melhor a aproximação de \sqrt{a} dada por x_{i+1} .

Esse processo deve ser repetido enquanto $|x_{i+1} - x_i| \geq \varepsilon$, onde ε é um número positivo que representa a precisão do cálculo da raiz. Assim, a aproximação de \sqrt{x} será o primeiro valor x_{i+1} para o qual $|x_{i+1} - x_i| < \varepsilon$.

Por exemplo, $\sqrt{63}$, com $\varepsilon = 0.001$ fica:

i	x_i	x_{i+1}	$ x_{i+1} - x_i $
0	31.5	16.75	14.75
1	16.75	10.255597014925373	6.494402985074627
2	10.255597014925373	8.19929204939465	2.056304965530723
3	8.19929204939465	7.94144112240038	0.25785092699427015
4	7.94144112240038	7.937255037058497	0.004186085341882695
5	7.937255037058497	7.9372539331938485	1.1038646485772574E-6

(quando repetir esse teste, não se preocupe com diferenças na última casa decimal. Estas podem ocorrer devido à ordem em que as operações foram feitas no código).

Escreva um método que receba como parâmetro um número a , e o valor de ε , retornando o valor de \sqrt{a} com precisão ε . Seu método deve necessariamente ter a seguinte assinatura:

```
static double raizQuadrada(double a, double epsilon)
```

Para o cálculo do valor absoluto $|x_{i+1} - x_i|$, você não poderá usar o método `Math.abs()`, devendo portanto solucionar o problema de outra forma (via um método auxiliar, ou inserindo código em `raizQuadrada()`, por exemplo).

1.1 Entrada

A entrada é composta pelos parâmetros $a \geq 0$, indicando o número cuja raiz se quer calcular, e $0 < \varepsilon < 1$, indicando a precisão do cálculo.

1.2 Saída

Como saída, o método retorna o valor de \sqrt{a} com precisão ε ; ou então -1 , caso $a < 0$, $\varepsilon \leq 0$ ou $\varepsilon \geq 1$.

Para que você teste o funcionamento do método `raizQuadrada()`, pode fazer o método `main()` escrever na tela o valor da raiz de algum número desejado. Isso é apenas para que você teste o método, e deve ser feito somente no método `main`. Não use `println()` diretamente em `raizQuadrada()`

É muito importante que você teste seu método com diferentes valores, buscando erros, e vendo se ele se comporta como requisitado.

1.3 Material a Ser Entregue

Um arquivo, denominado Raiz.java, contendo o método `raizQuadrada()` e qualquer outro método auxiliar que ache necessário. Para sua conveniência, `Raiz.java` será fornecido, cabendo a você então completá-lo.

Atenção!

1. Não modifique a assinatura de `raizQuadrada()`!
2. Para avaliação, apenas o método `raizQuadrada()` será invocado diretamente. Em especial, qualquer código dentro do `main()` será ignorado. Então certifique-se de que o problema seja resolvido chamando-se diretamente somente esse método.

2 Entrega

A entrega será feita única e exclusivamente via edisciplinas, até a data final marcada. Deverá ser postado no edisciplinas um arquivo zip, tendo como nome seu número USP:

número_usp.zip

Dentro do zip deve constar tão somente o arquivo `Raiz.java` com seu código nele. Não esqueça de preencher o cabeçalho constante do arquivo, com seu nome, número USP etc.

A responsabilidade de postagem é exclusivamente sua. Por isso, submeta e certifique-se de que o arquivo submetido é o correto (fazendo seu download, por exemplo). Problemas referentes ao uso do sistema devem ser resolvidos com antecedência.

3 Avaliação

Para avaliação, serão observados os seguintes quesitos:

1. Documentação: se há comentários explicando o que se faz nos passos mais importantes e para que serve o programa (Tanto o método quanto o programa em que está inserido);
2. Apresentação visual: se o código está legível, indentado etc;
3. Corretude: se o programa funciona.

Além disso, algumas observações pertinentes ao trabalho, que influem em sua nota, são:

- Este exercício-programa deve ser elaborado individualmente;
- Não será tolerado plágio, em hipótese alguma;
- Exercícios com erro de sintaxe (ou seja, erros de compilação), receberão nota ZERO.