

Buscando notas menores

Transcrição










Há diversas perguntas que podemos fazer para um *array*. Podemos perguntar:

Qual é o menor elemento?

Qual é o maior elemento?

Qual está no meio?










Para isto, vimos diversas técnicas para arranjar isto, seja para ordenar e assim responder diversas perguntas, ou simplesmente, para responder qual é o maior e menor. Mas existem outras perguntas importantes que podemos querer fazer para um *array*... Por exemplo, se eu fiz uma prova e não fui tão bem. Tirei uma nota 7. Essas são as notas de todos os alunos:

 André 4	 Carlos 8.5	 Ana 10	 Jonas 3	 Juliana 6.7	 Gui 7	 Paulo 9	 Mariana 5	 Lúcia 9.3
--	---	---	--	--	--	--	--	--

A nota 7 é boa ou ruim? Preciso observar as notas dos outros alunos e saber quais foram os resultados. Se todos tiraram uma nota abaixo de 7, a minha nota será um bom resultado. Porém, se todos tiverem tirado notas acima de 7, não será um bom resultado. É importante comparar a minha nota com a pontuação dos demais. O objetivo não é descobrir se "eu sou o melhor" ou se "eu sou o pior"... Mas saber se estou indo bem e se posso me focar em outros temas. Como posso descobrir isto? Se tenho nove alunos na minha sala, em qual colocação fiquei? Ao analisar as demais notas, vou descobrir que quatro alunos tiveram uma nota menor do que a minha. Isto significa que eu fiquei no meio. Fiquei em quinto lugar, tanto entre os melhores e os piores. Mas como eu fiz esta conta? Foi apenas observação, ou contagem com os dedos? Como fiquei em quinto, era possível fazer o cálculo mentalmente, mas o que exatamente fiz para identificar em qual posição eu fiquei?

Simulando quantos são menores

Qual algoritmo nós usamos para descobrir em que posição da sala eu fiquei? Vamos tentar calcular da mesma maneira que fiz anteriormente, mas usando o computador. Uma setinha irá indicar a minha posição:

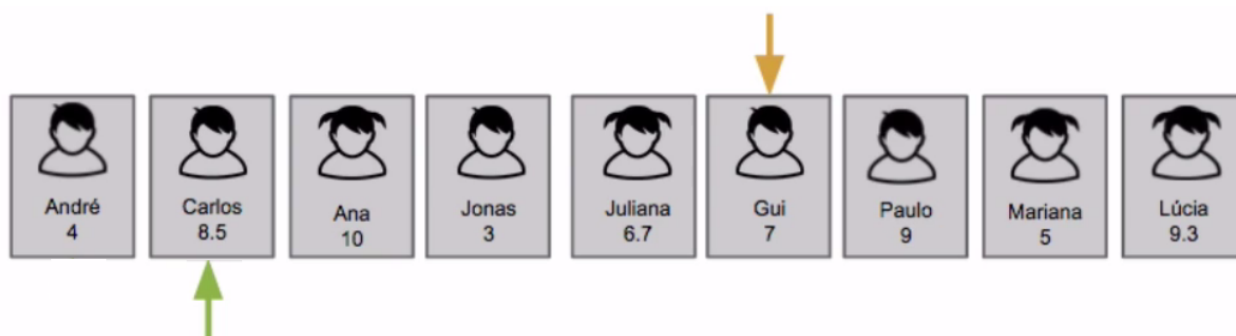
 André 4	 Carlos 8.5	 Ana 10	 Jonas 3	 Juliana 6.7	 Gui 7	 Paulo 9	 Mariana 5	 Lúcia 9.3
---	--	--	---	---	---	---	---	---

Agora iremos observar as notas dos demais alunos e comparar com a minha. Temos o André:

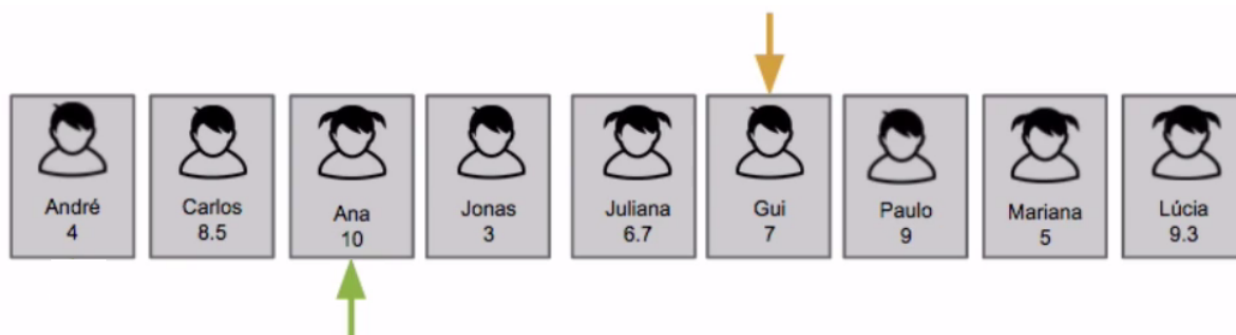


Ele tirou nota 4 e teve um resultado pior do que o meu. Uma pessoa tirou uma nota menor.

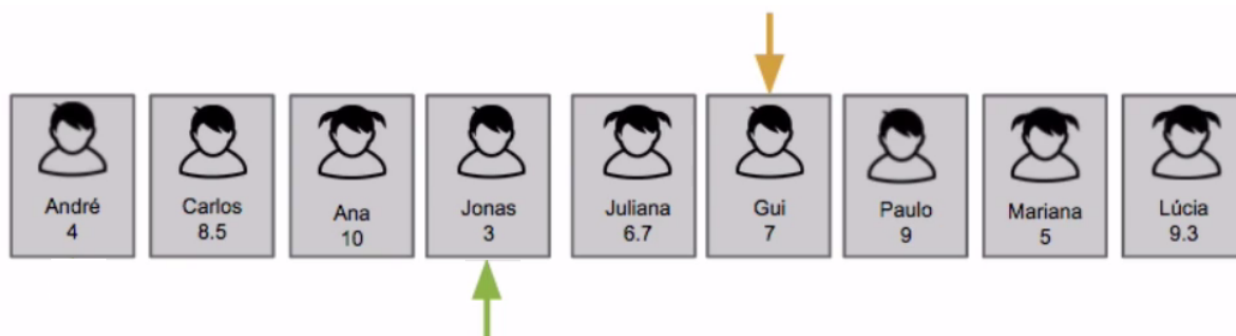
O Carlos tirou 8.5 e foi se saiu melhor do que eu.



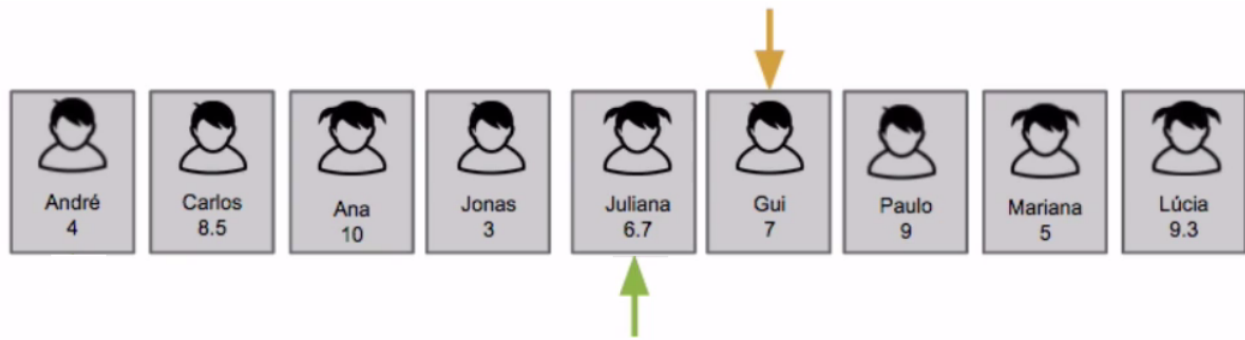
A Ana tirou 10 e foi melhor do que eu.



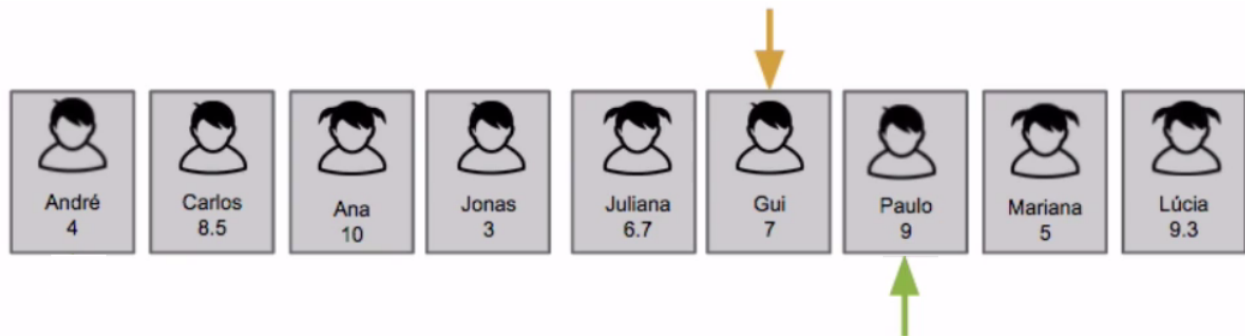
O Jonas tirou 3 e foi pior do que eu. Duas pessoas tiraram notas menores do que a minha.



A Juliana tirou 6.7 e foi pior do que eu. Três pessoas tiraram notas menores.



Quando chegamos na minha posição, ignoramos. O Paulo tirou 9 e se saiu melhor do que eu.



A Mariana tirou 5 e se saiu pior do que eu. Temos quatro pessoas com notas menores do que a minha. A Lúcia tirou 9.3 e teve um melhor resultado.



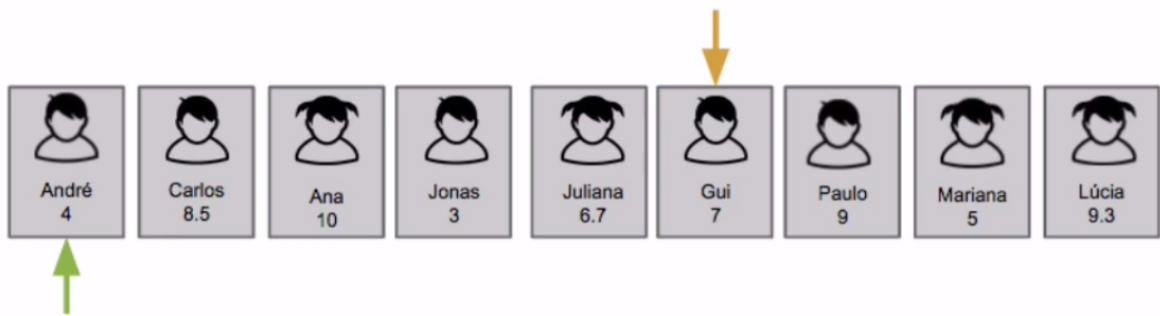
Se formos contado o número de pessoas com notas menores do que a minha, ficaremos sabendo o quão baixa minha nota está. Se contarmos quantas pessoas forma melhores do que eu, saberemos o quão alto minha nota está. Basta contar quantas pessoas foram melhores e piores.

Iremos agora anotar quantas notas são menores do que a minha usando a variável `menores`.

Vamos simular o algoritmo?

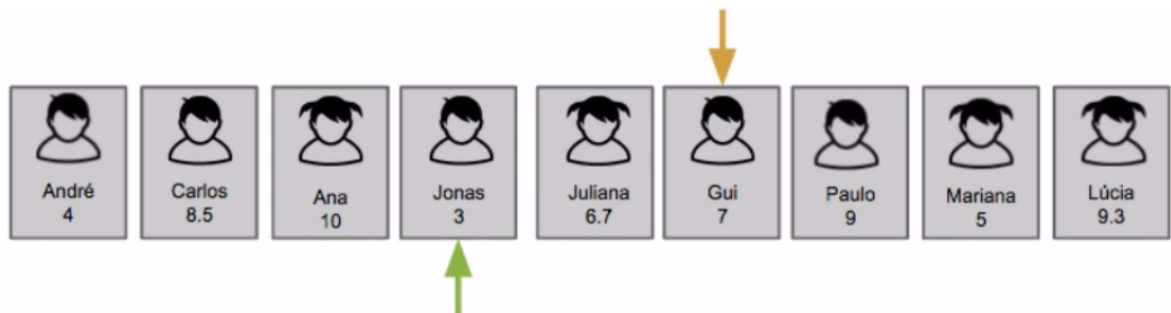


Entre Gui e o André, qual elemento é menor? O André. Seguiremos para a direita e somaremos +1 na variável `menores`.



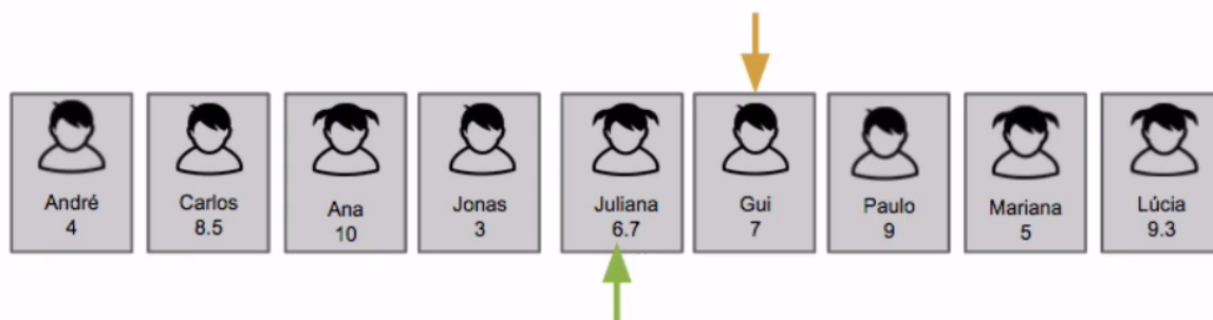
1
menores

Iremos comparar com a nota do Carlos. Ela é menor? Não. Seguiremos para a Ana, ela se saiu pior? Não. E Jonas, ele tirou uma nota pior? Sim. Vamos somar +1 na variável `menores`.



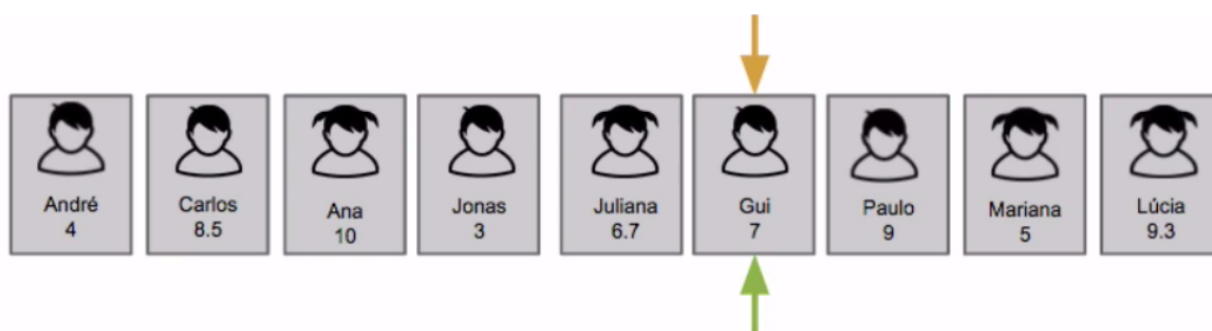
2
menores

A Juliana tirou uma nota menor? Sim. Somamos +1 na variável `menores`.



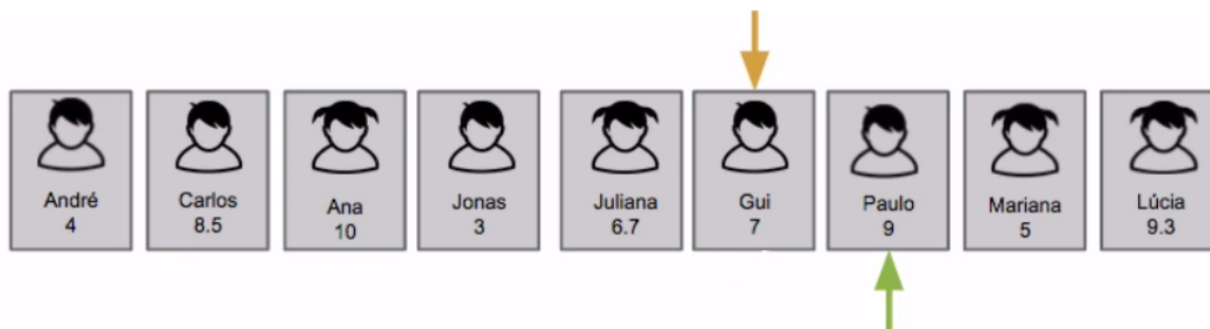
3
menores

Chegamos na minha posição, eu tirei uma nota menor? Além de não ter um resultado pior, devemos ignorar minha própria nota.



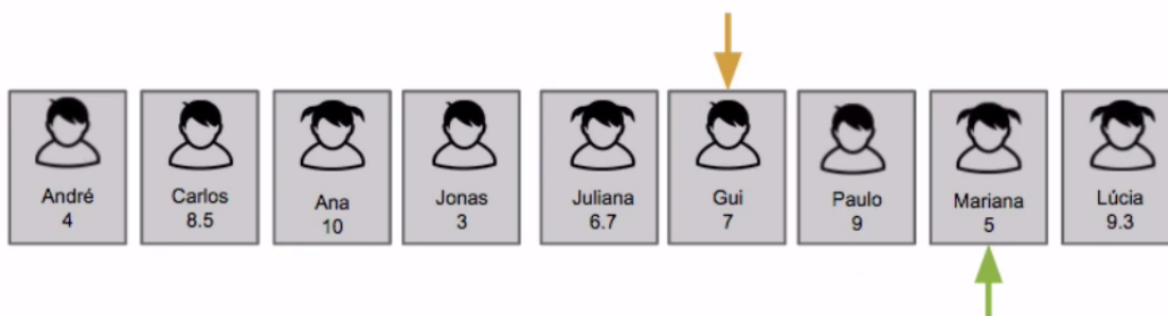
3
menores

O Paulo tirou uma nota menor? Não.



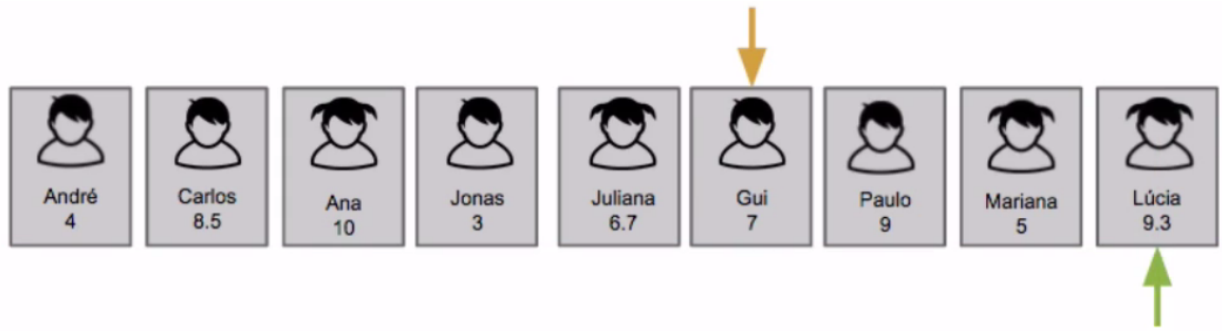
3
menores

A Mariana tirou uma nota menor? Sim. Vamos somar +1 na variável `menores`.



4
menores

Seguimos para a Lúcia, ela é menor? Não. Então, terminamos.



4
menores

Quantas pessoas tiraram uma nota menor do que a minha? **Quatro** pessoas. Então, quando chegar em casa, poderei dizer aos meus pais que tirei uma nota 7, mas quatro pessoas se saíram pior do que eu. Posso usar como argumento, a quantidade de pessoas que tiraram notas maiores ou menores. Como fizemos isto? Fomos comparando cada um dos elementos com a nossa nota e verificando qual era a menor. À medida que fomos passando por cada um, identificamos quantas pessoas tiveram resultados piores. Em seguida, iremos implementar o algoritmo que identifica quantas pessoas no *array tiraram notas piores do que a nossa.

Implementando o encontra menores

Dentro do nosso projeto, nós iremos criar um novo teste e o chamaremos de `TestaEncontraMenores`.

Nós queremos descobrir quantas pessoas tiraram uma nota menor do que a minha. Primeiro, vamos criar o método `main()`. Depois vou abrir o `TestaOrdenacaoAoIntercalar` vou copiar as notas, e trabalhar com elas.

```
public static void main(String[] args) {  
    Nota[] notas = {  
        new Nota("andre", 4),  
        new Nota("carlos", 8.5),  
        new Nota("ana", 10),  
        new Nota("jonas", 3),  
        new Nota("juliana", 6.7),  
        new Nota("guilherme", 7),  
        new Nota("paulo", 9),  
        new Nota("mariana", 5),  
        new Nota("lucia", 9.3),  
    };  
}
```

Estas são as notas que tínhamos no nosso sistema. Agora queremos encontrar todas as notas que tiraram uma nota menor do que a do Guilherme. Vamos extrair o Guilherme, considerando-o como uma única variável.

```
public static void main(String[] args) {  
    Nota guilherme = new Nota("guilherme", 7);  
    Nota[] notas = {  
        new Nota("andre", 4),  
        new Nota("carlos", 8.5),  
        new Nota("ana", 10),  
        new Nota("jonas", 3),  
        new Nota("juliana", 6.7),  
        guilherme,  
        new Nota("paulo", 9),  
        new Nota("mariana", 5),  
        new Nota("lucia", 9.3),  
    };  
}
```

Especificaremos que queremos encontrar as pessoas com notas menores do que o Guilherme (`encontraMenores(guilherme)`).

```
encontraMenores(guilherme);
```

Para isto teremos que incluir também o *array*.

```
encontraMenores(guilherme, notas);
```

Vamos criar a função?

```
private static void encontraMenores(guilherme, Nota[] notas) {  
  
}
```

Agora começou o nosso trabalho: iremos procurar quem tem a nota menor do que o Guilherme dentro do *array*. Qual foi o algoritmo que usamos mentalmente? Quando observamos todas as notas, analisamos cada uma por vez. Por isto, no nosso código precisamos criar um `for` .

```
for(int atual = 0; atual < notas.length; atual++) {  
  
}
```

Analisaremos as notas, se elas forem menores do que a do Guilherme, teremos encontrado o elemento!

```
Nota nota = notas[atual];
```


Se (`if`) a nota (`nota.getValor()`) for menor do que a do Guilherme (`guilherme.getValor()`) daremos um "joinha, achei!"

```
if(nota.getValor() < guilherme.getValor()) {  
    // joinha, achei!  
}
```

Quando calculamos mentalmente, o que faríamos em seguida? Somaríamos +1 na variável. Então, precisávamos em alguma parte, começar do 0. Vamos começar fora do laço.

```
int menores = 0;
```

E o trecho do código ficará assim:

```
private static void encontraMenores(Nota guilherme, Nota[] notas) {  
    int menores = 0;  
    for(int atual = 0; atual < notas.length; atual++) {  
        Nota nota = notas[atual];
```

Cada vez que encontrarmos um nota menor, iremos somar +1 na variável `menores` .

```
if(nota.getValor() < guilherme.getValor()) {  
    menores = menores + 1;  
}
```

Que podemos traduzir para `menores ++` .

```
if(nota.getValor() < guilherme.getValor()) {  
    menores = menores++;  
}
```

Ao somarmos um ao número de notas menores, o algoritmo irá passar para a próxima nota continuamente.

No fim, pediremos para ele retornar (`return`) o número de notas menores.

```
return menores;
```

Precisamos que o método retorne `int` :

```
private static int encontraMenores(guilherme, Nota[] notas) {  
  
}
```

Iremos encontrar a quantidade de menores. Vamos acrescentar isto no código...

```
int menores = encontraMenores(Nota guilherme, notas);
```

E depois, imprimir o Número de menores é o valor de menores .

```
int menores = encontraMenores(guilherme, notas);  
System.out.println("Número de menores: " + menores);
```

Será que o programa irá imprimir 4? Rodaremos o programa, clicamos no botão à direita, depois em *Run As*, em seguida, em *Java Application*. O resultado será:

Número de menores: 4

O que o algoritmo fez? Ele passou por cada elemento, verificando quantas notas menores do que a minha tinham dentro do *array*.

Podemos alterar o nome da variável *guilherme* , para *busca* , porque nem sempre estou encontrando as menores baseadas no Guilherme. Iremos alterar a seguinte linha:

```
private static void encontraMenores(Nota guilherme, Nota[] notas) {
```

Que receberá o novo nome da variável, *busca* .

```
private static void encontraMenores(Nota busca, Nota[] notas) {
```

Faremos o mesmo com a linha:

```
if(nota.getValor() < guilherme.getValor()) {
```

Com a alteração será:

```
if(nota.getValor() < busca.getValor()) {
```

Após as modificações, nosso código ficará assim:

```
private static void encontraMenores(Nota busca, Nota[] notas) {  
    int menores = 0;  
    for(int atual = 0; atual < notas.length; atual++) {  
        Nota nota = notas[atual];  
        if(nota.getValor() < busca.getValor()) {  
            menores = menores ++;  
        }  
    }  
    return menores;  
}  
}
```