



Universidade Paranaense - UNIPAR

GABRIELLA CRISTINI GALVAO - 00240549

REGIANE CARLA RODRIGUES LIMA - 00220666

**TRABALHO DE PESQUISA E DESENVOLVIMENTO DO**  
**2º BIMESTRE**  
Estrutura De Dados

Cascavel  
2023

GABRIELLA CRISTINI GALVAO  
REGIANE CARLA RODRIGUES LIMA

## **TRABALHO DE PESQUISA E DESENVOLVIMENTO DO 2º BIMESTRE**

Trabalho apresentado para obtenção da nota  
parcial da disciplina de Estrutura de Dados.

Prof.<sup>a</sup> Paulo H. P. Santos

Cascavel  
2023

## **SUMÁRIO**

Adicione títulos (Formatar > Estilos de parágrafo) e eles serão exibidos no seu sumário.

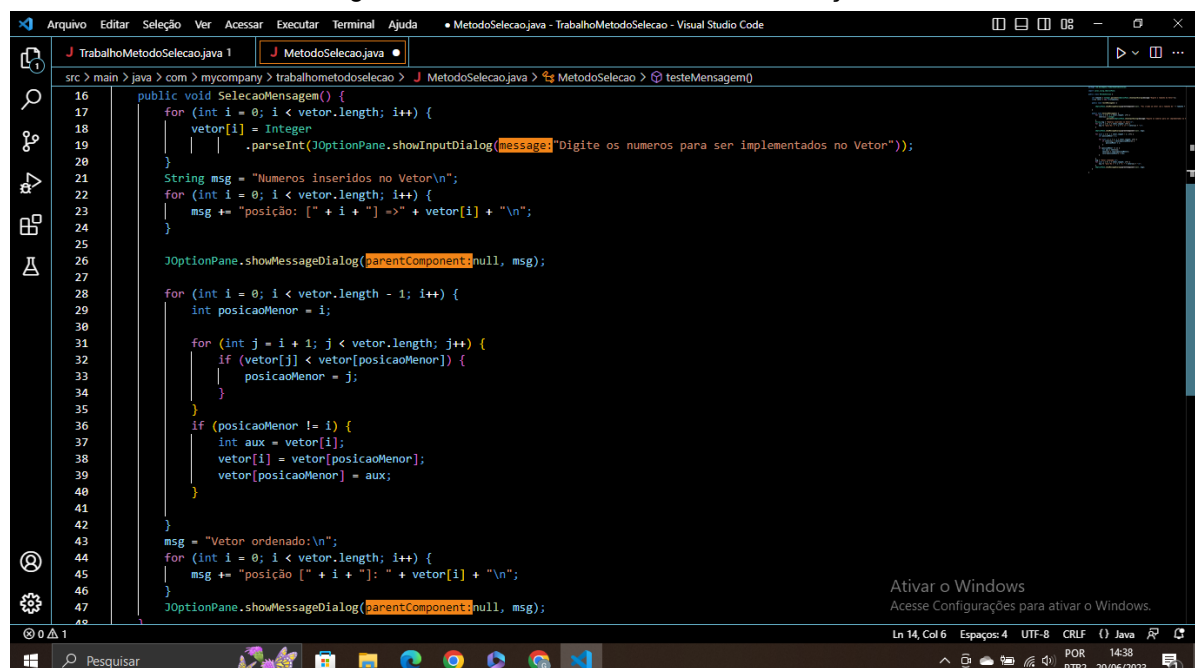
O início desse trabalho foi marcado com o estudo de todos os requisitos dos quais foram informados pelo nosso cliente, com os requisitos bem estruturados começamos a desenvolver o código.

Para realização do entendimento e desenvolvimento das aplicações, os exercícios em sala de aula e com a revisão nas classes fornecidas pelo professor no github foram essenciais nesse momento, pois nos auxiliou no desenvolvimento de nossas classes. Para o desenvolvimento do código optamos pelo uso da ferramenta VsCode ao invés do netbeans pois o mesmo se mostrou mais fácil e eficiente em sua utilização.

Também no início da criação de classes discutimos com quais cada uma iria ficar pois assim desenvolvermos uma classe que mais tivéssemos afinidade, também decidimos fazer uma classe separada para o Bubble Sort, Insertion Sort e Selection Sort sendo estas executadas em uma classe chamada ordenação, assim conseguiríamos compreender melhor o que cada ordenação faria e o código estaria mais limpo e claro para a compreensão.

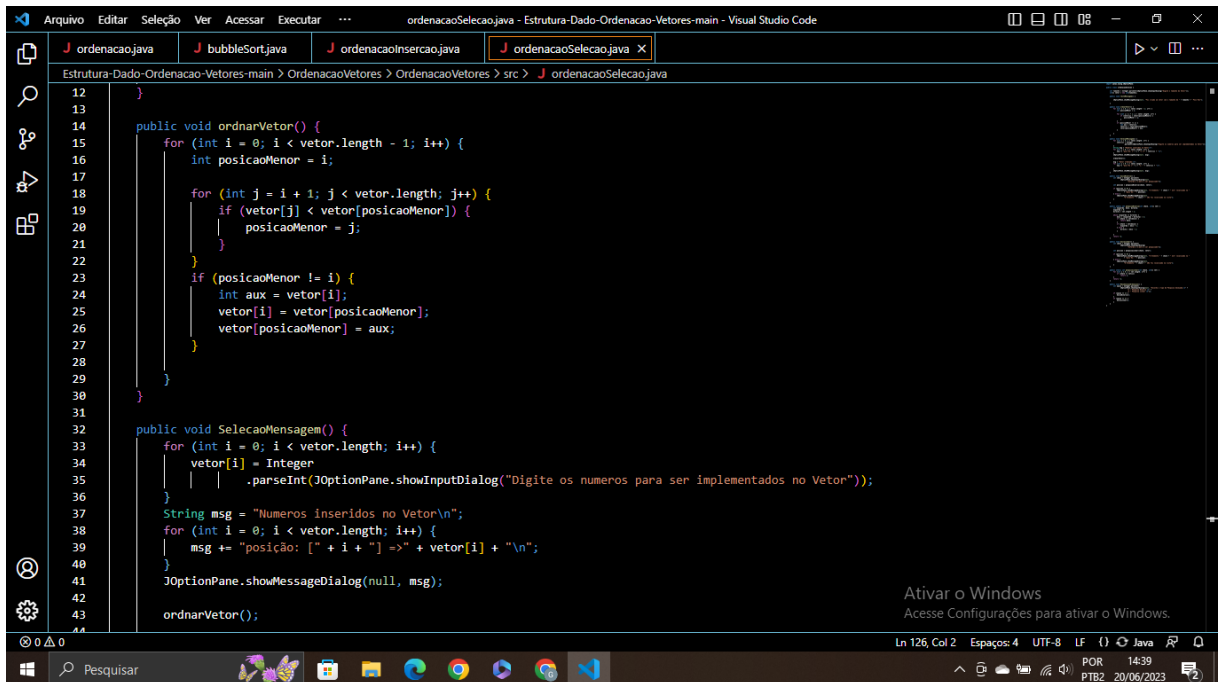
Entretanto a pesquisa linear e binária se mostrou um pouco complexa causando assim algumas dúvidas, como por exemplo de como implementá-la no código. Como o Selection Sort foi implementado primeiro com as duas pesquisas, está diferente dos outros dois pois quando estava sendo executado para retornar os resultados das pesquisas, analisamos erroneamente que estava retornando o resultado incorreto, então o mesmo foi reestruturado e dividindo para que a ordenação do vetor do método SeleçãoMensagem(), fosse separada da mesma criando assim o método ordnarVetor(), onde quando chamado na SeleçãoMensagem() ordenaria o vetor informado pelo usuário. Então foram criados mais três métodos, sendo eles buscaBinaria(), buscaLinear() e MenuSelecaoOrnenacao(), onde quando chamado o menu na classe principal ordenação, executaria as pesquisas Binária ou Linear, deixando também o usuário escolher qual seria a forma de busca utilizada.

Figura 1: Recorte tela antes da reestruturação.

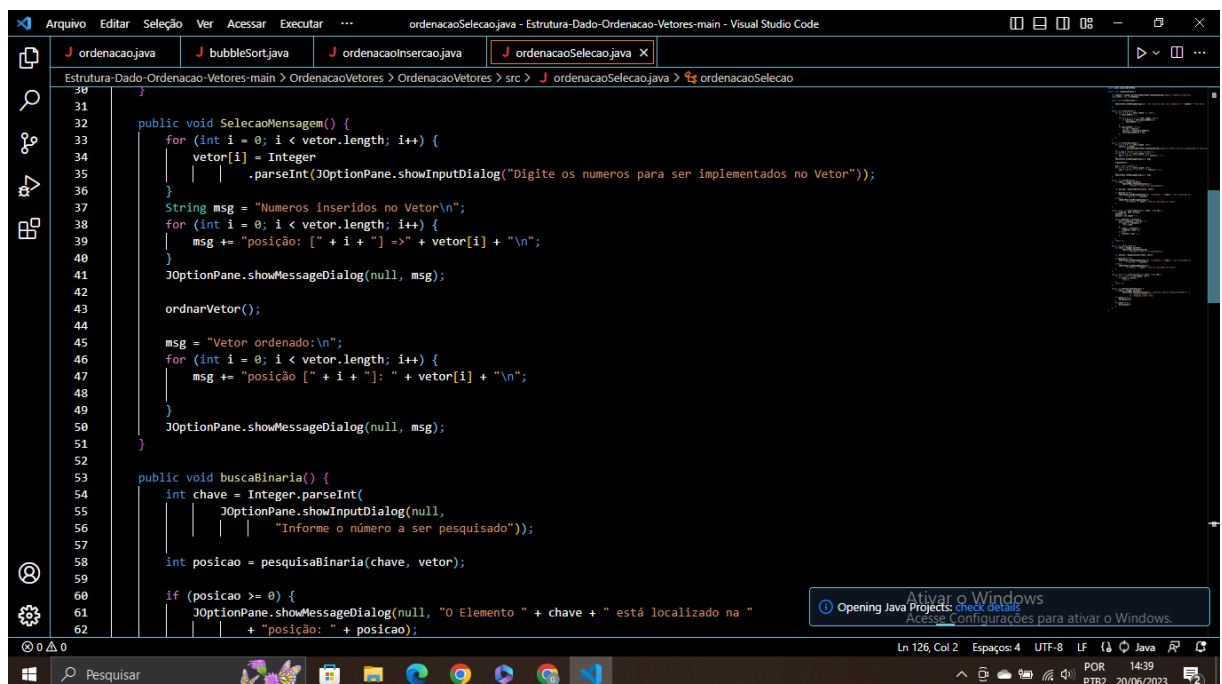


```
16 public void SelecaoMensagem() {
17     for (int i = 0; i < vetor.length; i++) {
18         vetor[i] = Integer
19             .parseInt(JOptionPane.showInputDialog(message:"Digite os numeros para ser implementados no Vetor"));
20     }
21     String msg = "Numeros inseridos no Vetor\n";
22     for (int i = 0; i < vetor.length; i++) {
23         msg += "posição: [" + i + "] -> " + vetor[i] + "\n";
24     }
25
26     JOptionPane.showMessageDialog(parentComponent:null, msg);
27
28     for (int i = 0; i < vetor.length - 1; i++) {
29         int posicaoMenor = i;
30
31         for (int j = i + 1; j < vetor.length; j++) {
32             if (vetor[j] < vetor[posicaoMenor]) {
33                 posicaoMenor = j;
34             }
35         }
36         if (posicaoMenor != i) {
37             int aux = vetor[i];
38             vetor[i] = vetor[posicaoMenor];
39             vetor[posicaoMenor] = aux;
40         }
41     }
42
43     msg = "Vetor ordenado:\n";
44     for (int i = 0; i < vetor.length; i++) {
45         msg += "posição [" + i + "] : " + vetor[i] + "\n";
46     }
47     JOptionPane.showMessageDialog(parentComponent:null, msg);
48 }
```

Figura 2 e 3 : Recorte tela depois da reestruturação.



```
12 }
13
14 public void ordnarVetor() {
15     for (int i = 0; i < vetor.length - 1; i++) {
16         int posicaoMenor = i;
17
18         for (int j = i + 1; j < vetor.length; j++) {
19             if (vetor[j] < vetor[posicaoMenor]) {
20                 posicaoMenor = j;
21             }
22         }
23         if (posicaoMenor != i) {
24             int aux = vetor[i];
25             vetor[i] = vetor[posicaoMenor];
26             vetor[posicaoMenor] = aux;
27         }
28     }
29 }
30
31
32 public void SelecaoMensagem() {
33     for (int i = 0; i < vetor.length; i++) {
34         vetor[i] = Integer
35             .parseInt(JOptionPane.showInputDialog("Digite os numeros para ser implementados no Vetor"));
36     }
37     String msg = "Numeros inseridos no Vetor\n";
38     for (int i = 0; i < vetor.length; i++) {
39         msg += "posição: [" + i + "] => " + vetor[i] + "\n";
40     }
41     JOptionPane.showMessageDialog(null, msg);
42
43     ordnarVetor();
44 }
```



```
30 }
31
32 public void SelecaoMensagem() {
33     for (int i = 0; i < vetor.length; i++) {
34         vetor[i] = Integer
35             .parseInt(JOptionPane.showInputDialog("Digite os numeros para ser implementados no Vetor"));
36     }
37     String msg = "Numeros inseridos no Vetor\n";
38     for (int i = 0; i < vetor.length; i++) {
39         msg += "posição: [" + i + "] => " + vetor[i] + "\n";
40     }
41     JOptionPane.showMessageDialog(null, msg);
42
43     ordnarVetor();
44
45     msg = "Vetor ordenado:\n";
46     for (int i = 0; i < vetor.length; i++) {
47         msg += "posição [" + i + "]: " + vetor[i] + "\n";
48     }
49     JOptionPane.showMessageDialog(null, msg);
50 }
51
52
53 public void buscaBinaria() {
54     int chave = Integer.parseInt(
55         JOptionPane.showInputDialog(null,
56             "Informe o número a ser pesquisado"));
57
58     int posicao = pesquisaBinaria(chave, vetor);
59
60     if (posicao >= 0) {
61         JOptionPane.showMessageDialog(null, "O Elemento " + chave + " está localizado na "
62             + "posição: " + posicao);
63     }
```

Fonte: Autoria própria

Depois que constatamos nosso erro, implementamos as duas pesquisas nas outras duas classes sem muitos problemas, o unico problema quando fomos implementá-las foi que tivemos que retirar a chamada do vetor de dentro dos métodos pois quando o mesmo estava por exemplo dentro do método `ordenacaoInsercao()` não poderia ser acessada pelo método `buscaBinaria()`.