
INSTITUTO DE COMPUTAÇÃO - UNICAMP

Especialização em Engenharia de Software

INF335 - Análise Orientada a Objetos e Projeto Arquitetural

Prof. Leonardo Montecchi

Trabalho 1

Debugger

Felipe Emygdio de Salles - RG: 47.061.238-1

Luan Silva - RG: 36.001.085-4

IC/UNICAMP

2020

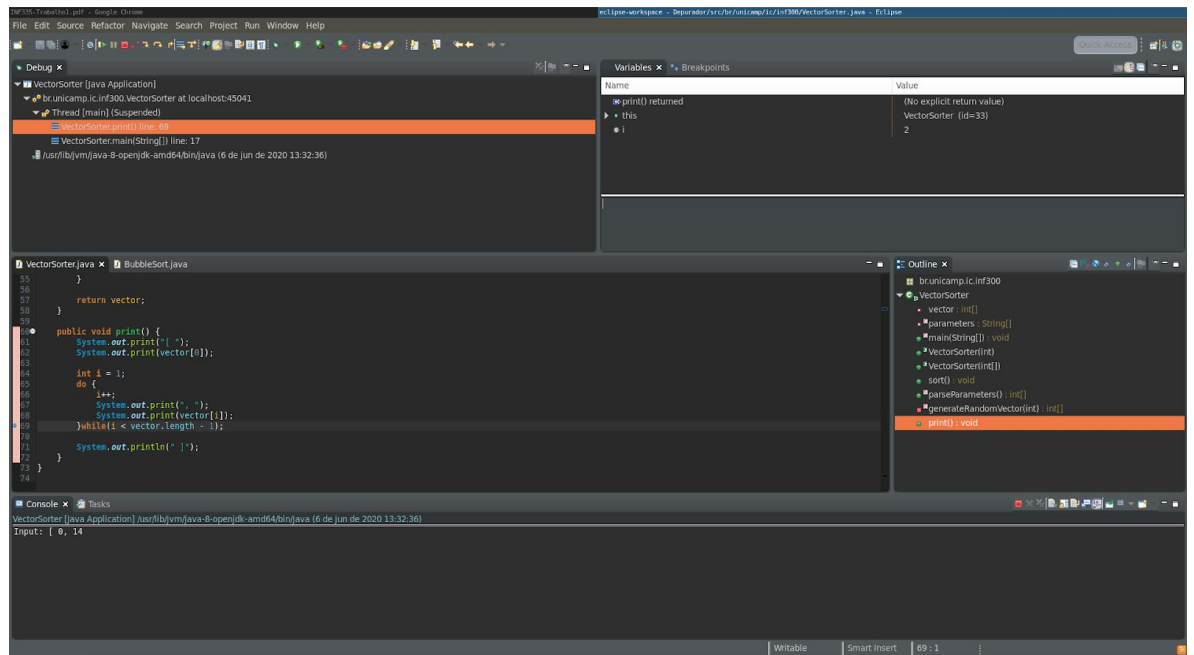
1. Quais defeitos contém o código original? Considere ambas as classes. Responda à pergunta da forma “Bug #1: Linha XX, o código [aaa] deveria ser substituído por [bbb]”

1.1. *BUG #1: BubbleSort Linha 11. O código deveria ser substituído por:*

if (vector.length > i + 1 && vector[i] > vector[i + 1]) {...}

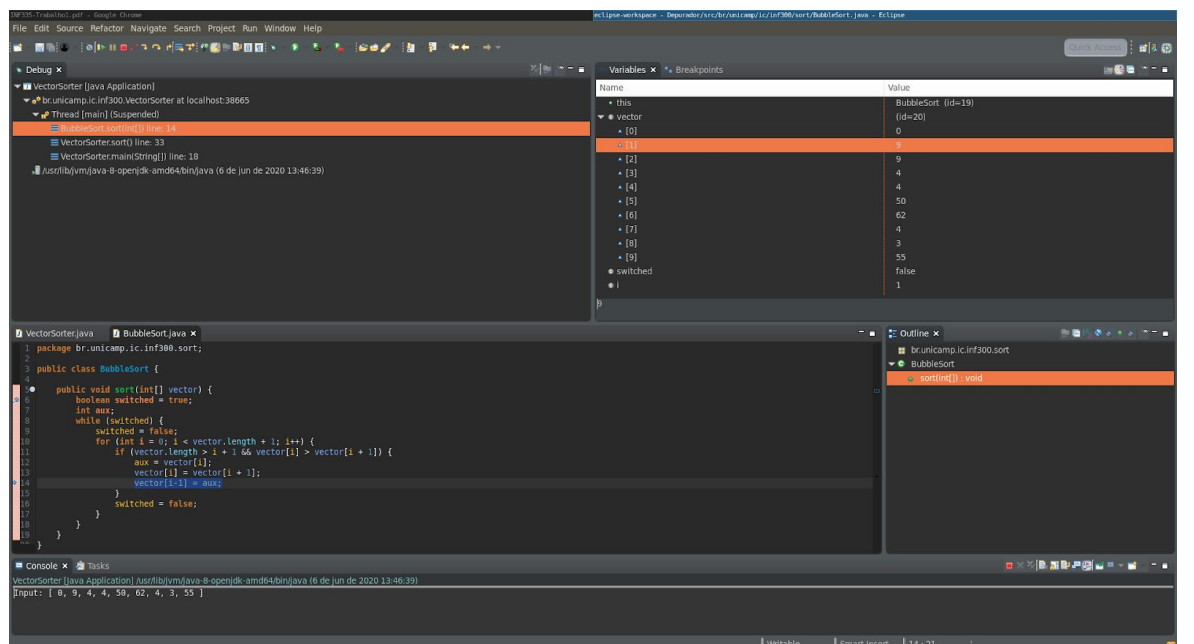
1.2. *BUG #2: VectorSorter Linha 64. Na primeira iteração, o sistema pula um elemento do vetor. O código deveria ser substituído por:*

int i = 0;

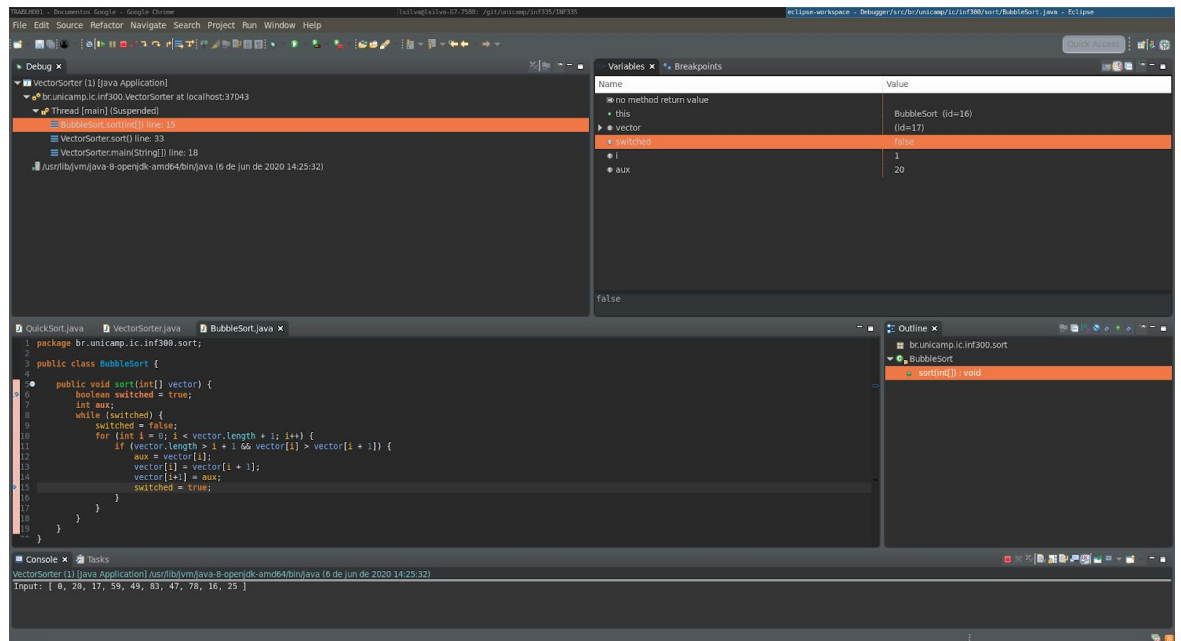


1.3. *BUG #3: BubbleSort Linha 14. O código deveria ser substituído por:*

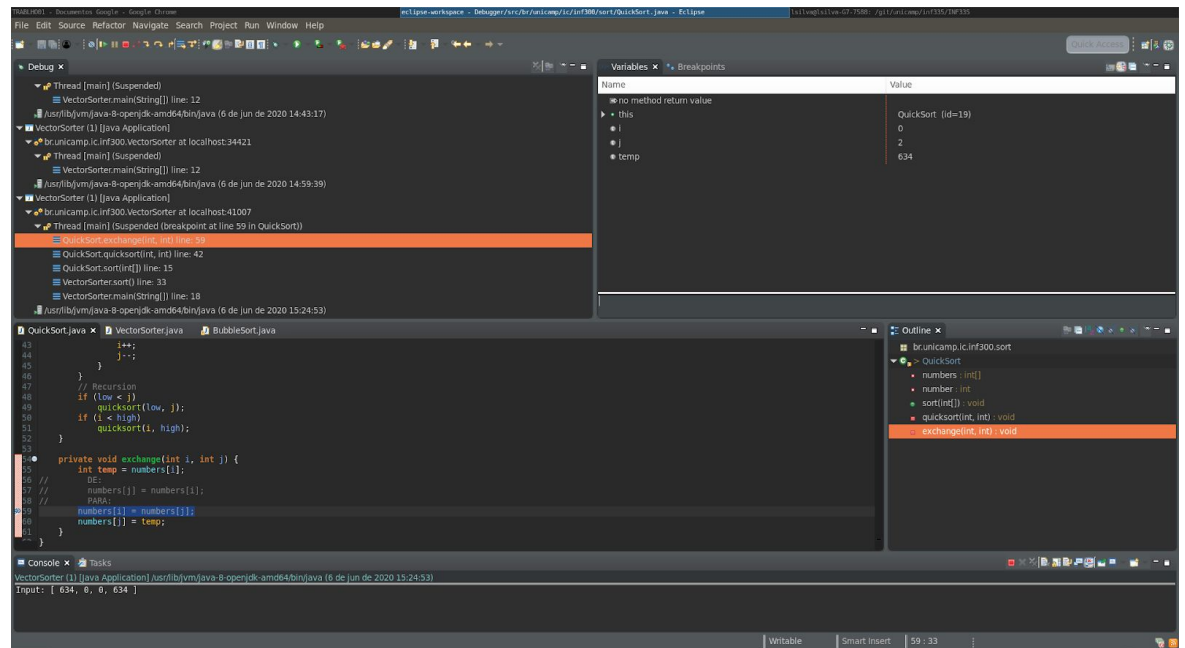
vector[i+1] = aux;



- 1.4. BUG #4: *BubleSort Linhas 15 e 16*. O código deveria ser substituído por:
- switched = true;* (dentro do laço de repetição *for*)



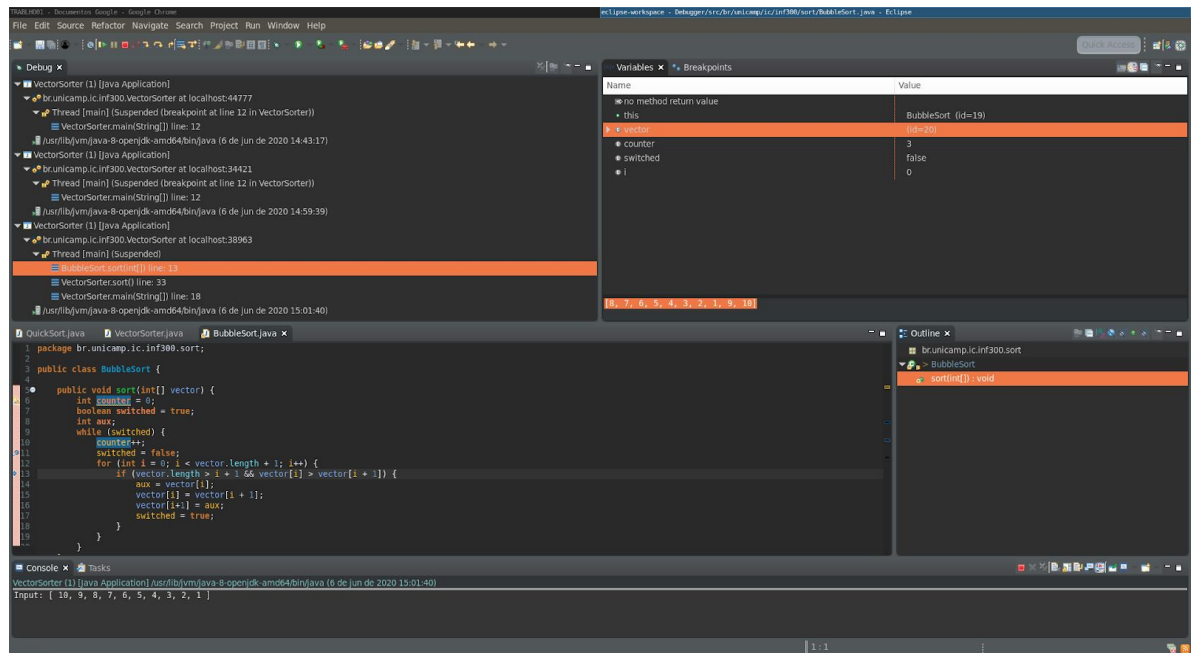
- 1.5. BUG #5. QuickSort Linha 56. As variáveis j e i estão invertidas, o código deve ser substituído por:
 $numbers[i] = numbers[j];$ (dentro do método *exchange*)



2. Considere o vetor [10, 9, 8, 7, 6, 5, 4, 3, 2, 1] como entrada da função sort:

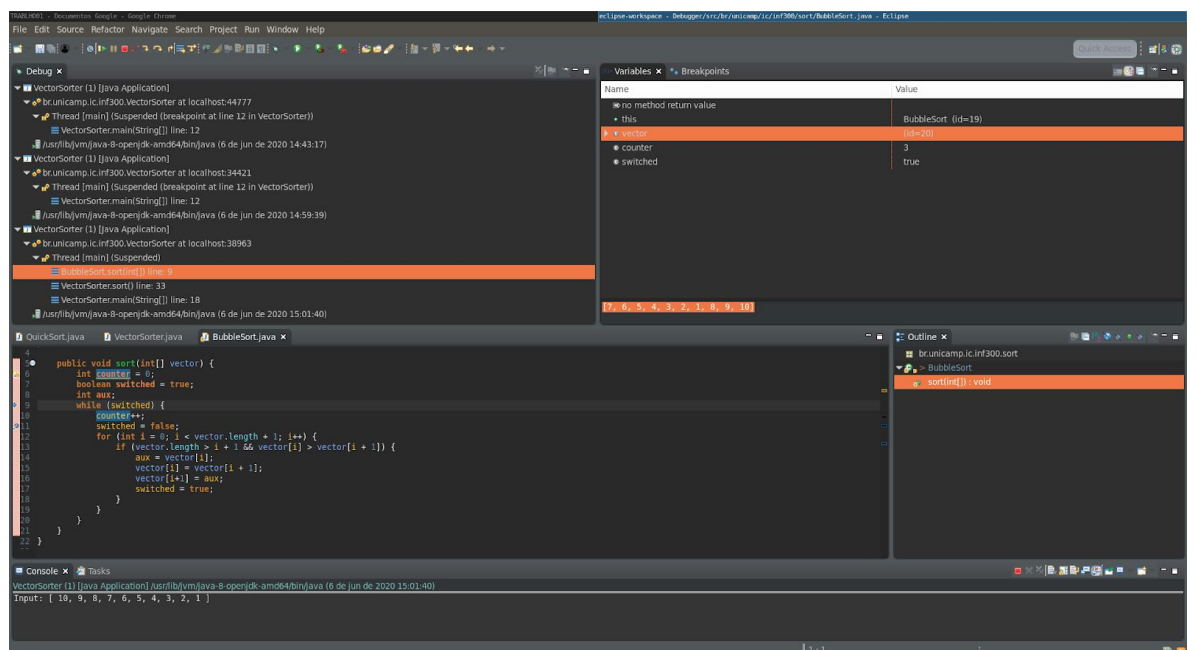
2.1. Qual é o conteúdo do vetor a terceira vez que o programa entrar no ciclo for?

R: [8, 7, 6, 5, 4, 3, 2, 1, 9, 10]



2.2. Qual é o conteúdo do vetor a terceira vez que o programa sair do ciclo for?

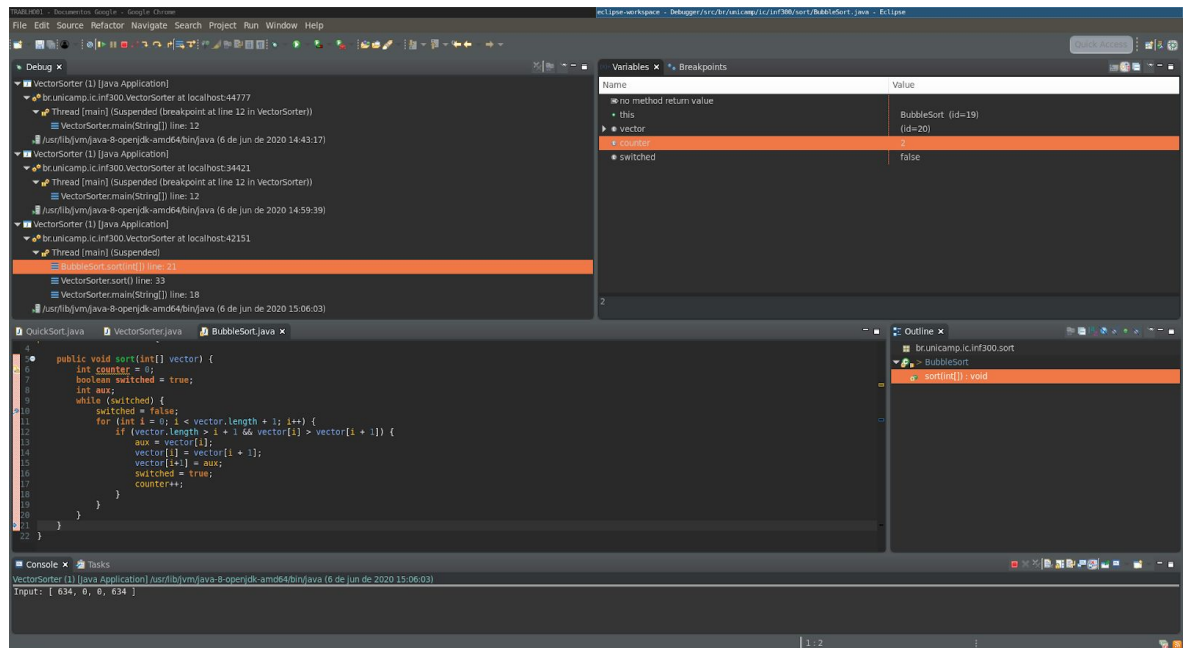
R: [7, 6, 5, 4, 3, 2, 1, 8, 9, 10]



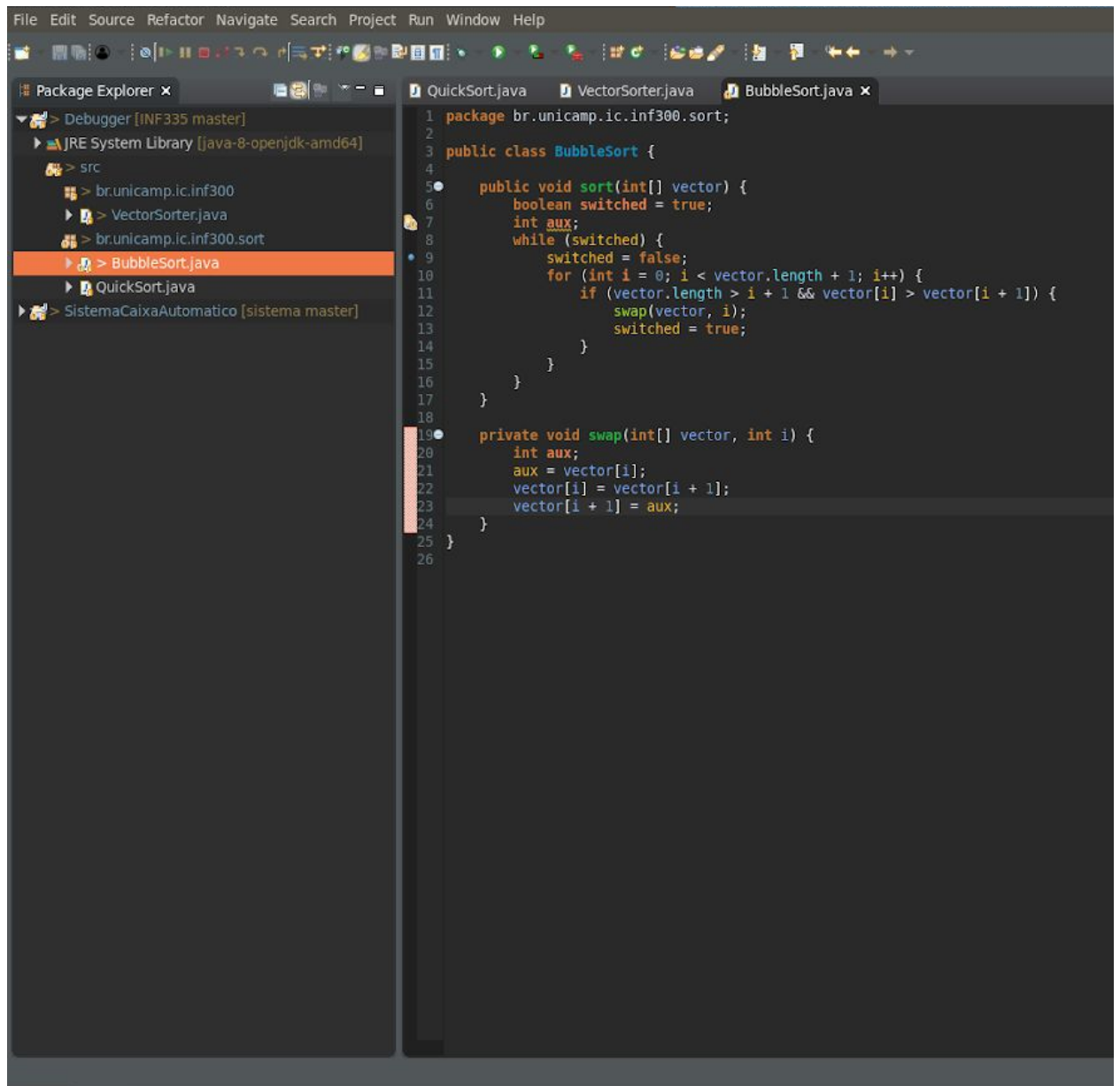
3. Considere o vetor [634, 0, 0, 634] como entrada da função sort:

3.1. Quantas trocas de posição são feitas até obter um vetor ordenado?

R: 2 trocas

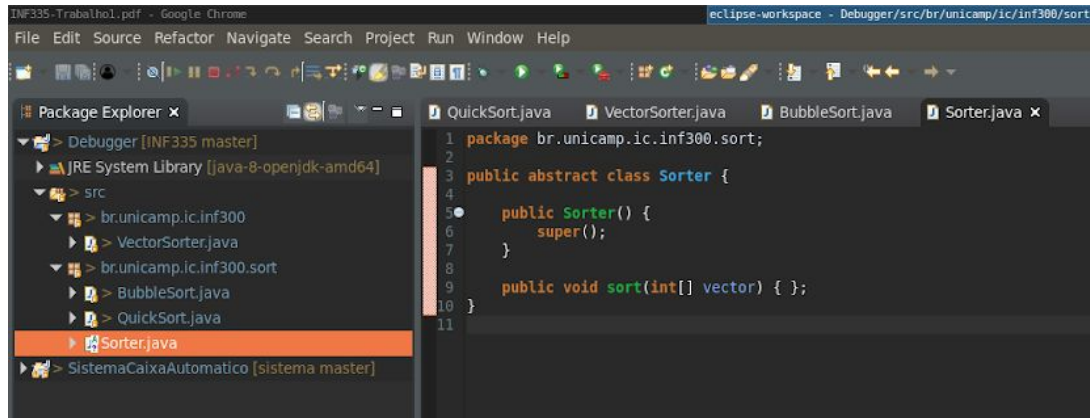


4. Extract Method

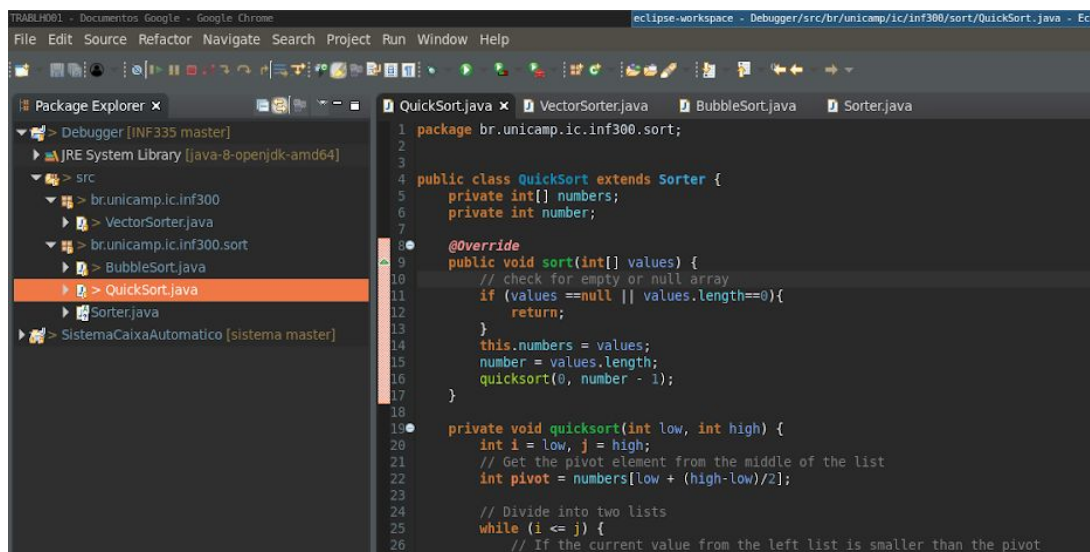


5. Extract Superclass

Sorter.java



QuickSort.java



BubbleSort.java

