Object-oriented Programming

Mini Project Report

Demonstration of sorting algorithms on an array

Group 25:
Phan Thai Viet,
Le Thanh Vinh,
Doan Ngoc Vinh,
Dinh Cong Vu

Contents

1 Group members and assigment **3**

2 Project Description **4**

2.1 Overview 4

2.2 Design requirement 4

2.3 Usecase diagram 4

3 Project design **6**

3.1 General class diagram 6

3.2 Packages 6

4 OOP Technique and Exception **13**

1 Group members and assigment

- Phan Thai Viet 20204895 : controller package, elements, package, screens package, sortingalgorithms package : mergesort, handling exception and testing, video demo.
- Le Thanh Vinh 20200668 : initally, he assigned to create the count sort algorithm, but the day before the deadline then he got trouble and did not finish the assignment. Phan Thai Viet and Đoàn Ngoc Vinh have to cover his part.
- Doan Ngoc Vinh 20204935 : controller package, elements package, screens package, sortingalgorithms package : radixsort, slides and report.
- Dinh Cong Vu 20204896 : usecase diagram, general diagram, class diagram for each class, handling exception and testing.

2 Project Description

2.1 Overview

Our project " Demonstration of sorting algorithms on an array " aims to visualize the sorting process of three algorithms: merge sort, radix sort and counting sort. For simplicity we just use non-negative numbers with array length of 8.

2.2 Design requirement

On the main menu: title of the application, 3 types of sorting algorithms for the user to choose, help menu, quit.

- User must select a sort type to start the demonstration.
- Help menu shows the basic usage and aim of the program.
- Quit option exits the program. Remember to ask for confirmation.
- A button for creating the array: The user can choose to randomly create an array or input an array for the program.
- A button for starting the algorithm with the created array. Remember to show clearly each step of the sorting.
- A back button for the user to return to the main menu at any time.

2.3 Usecase diagram

In figure 1, user can choose one of the three sorting algorithms to visualize. The user can choose to create their array or array can be randomly created by the application. User can also view the help menu or exiting the application at anytime.

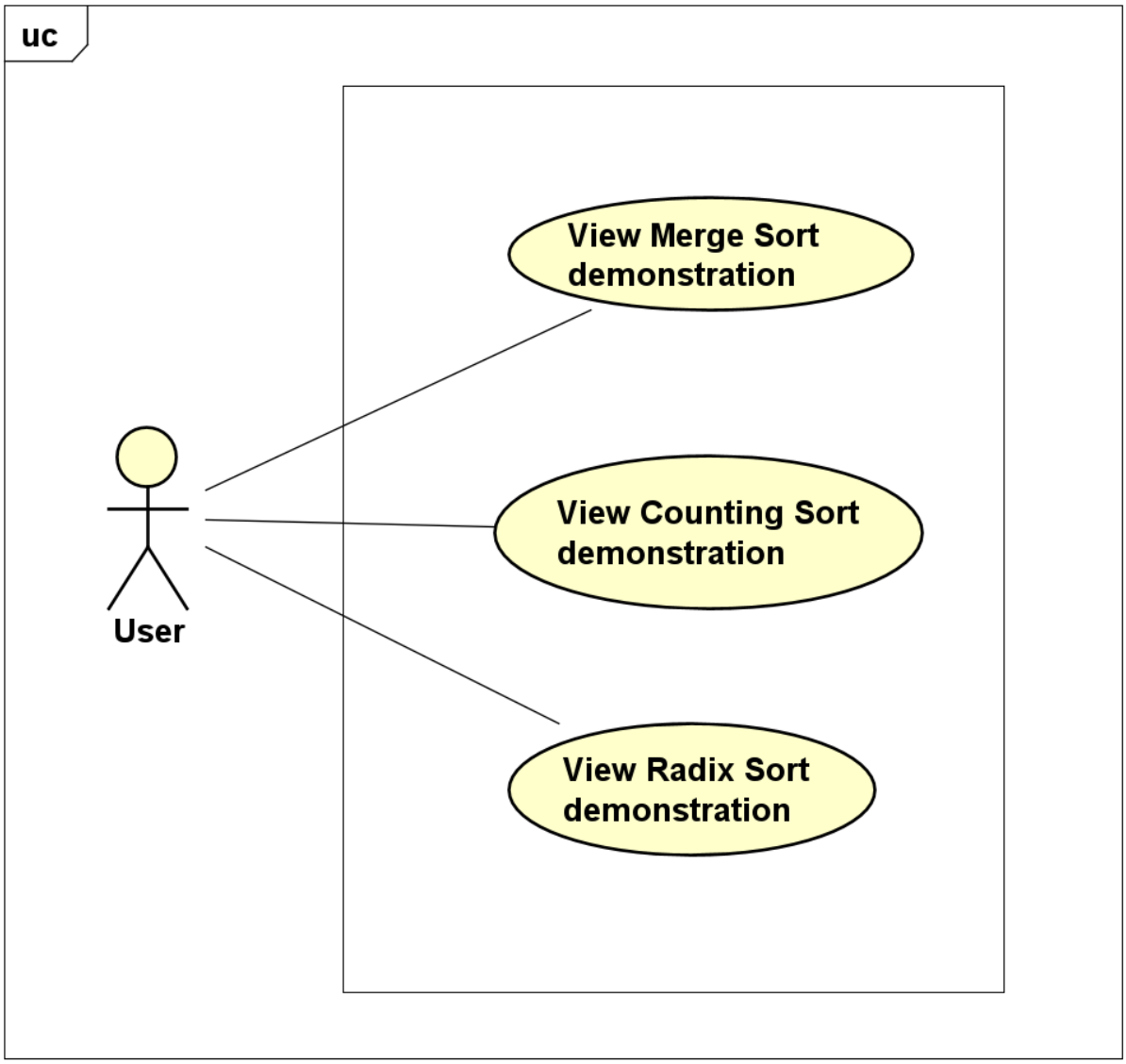

Figure 1: Usecase diagram

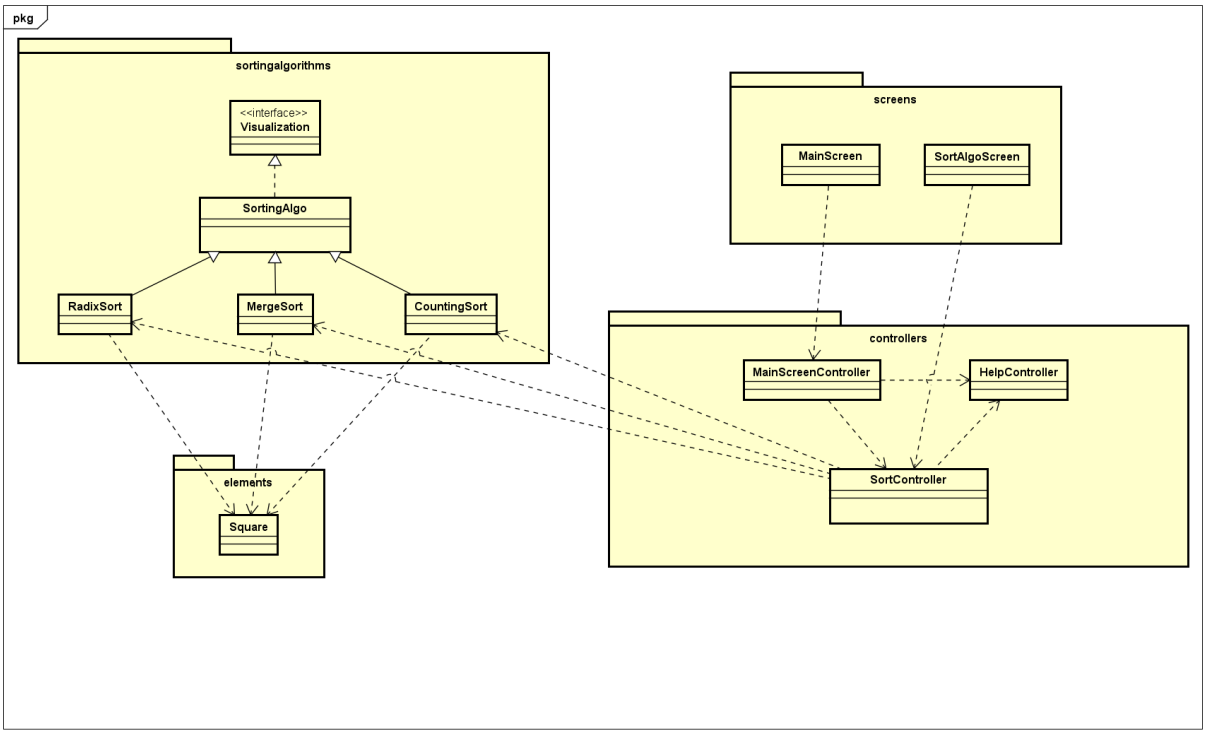

Figure 2: General class diagram

3 Project design

3.1 General class diagram

The general class diagram has 4 packages: ‘algorithms’, ‘controllers’, ’elements’, ‘screens’:

- "algorithms": contains 3 sorting algorithms and an abstract class "SortingAlgo".
- "controllers": contains main screen controller, sorting algorithm controller, and help menu.
- "screens": contains screen classes.
- "elements": contains element displayed on the screen.

3.2 Packages

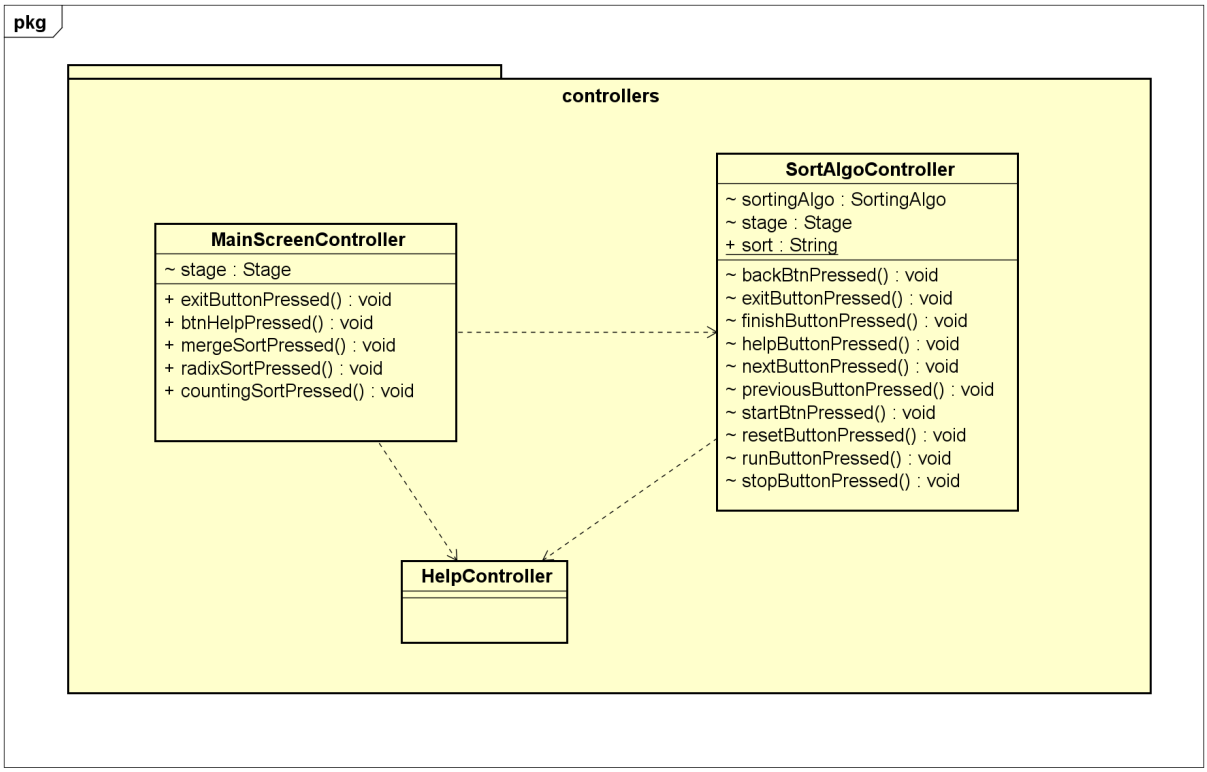

Figure 3: Controllers

The main screen controller manages the main screen of the application which contains 3 buttons for 3 algorithms, a help button and a exit button. The sort algorithm controller manages the sorting screen of the application, it has 1 attribute "sort" and 5 methods. In the sort algorithm controller, "logoutPressed" means exit the application, "backmenuPressed" means return to the main screen, "startButtonPressed" means start the sorting process, "prevPush" and "nextPush" means go to previous or next step of the sorting process. The help controller when called will open a text file which will be the instruction of the application.

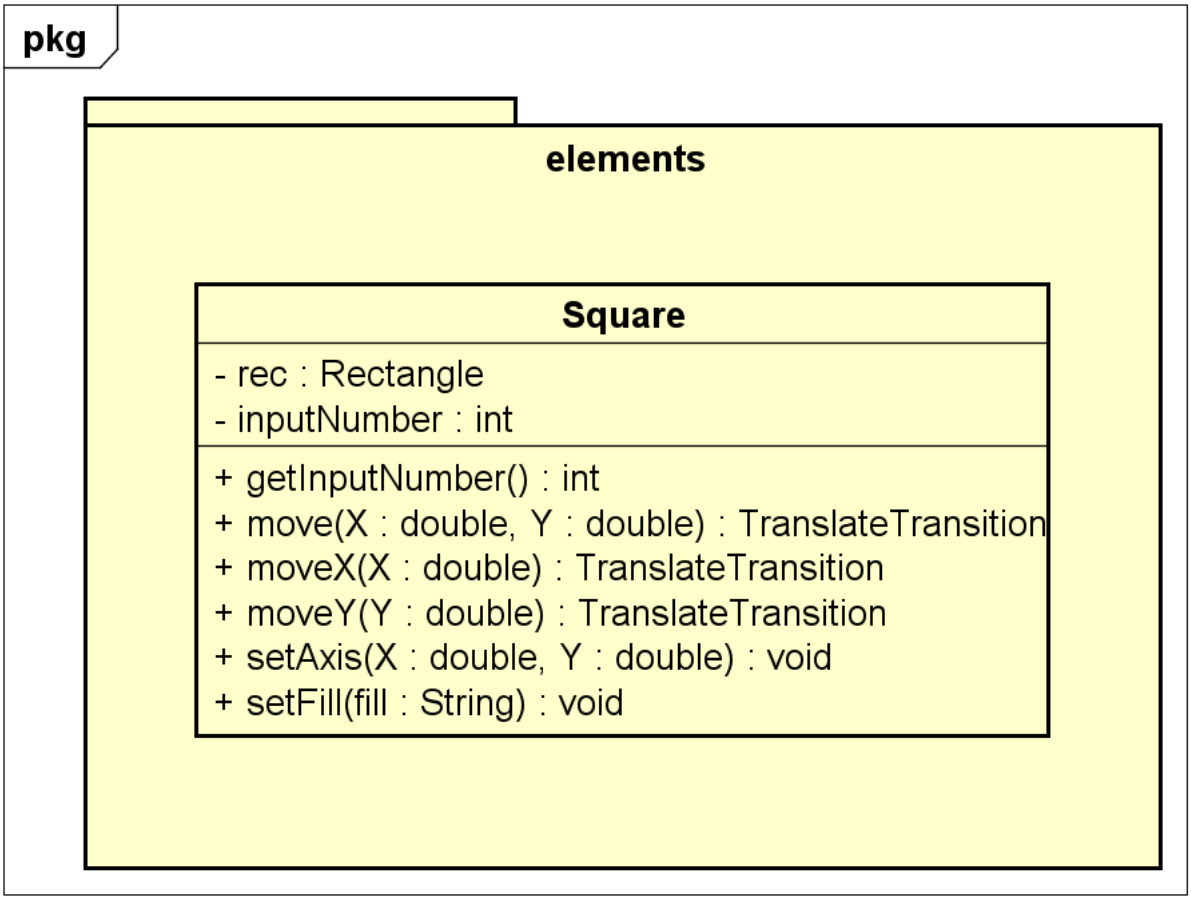

Figure 4: Elements

The "elements" packages contains square with 3 methods . The square will be displayed on the sorting screen to illustrate the sorting process.

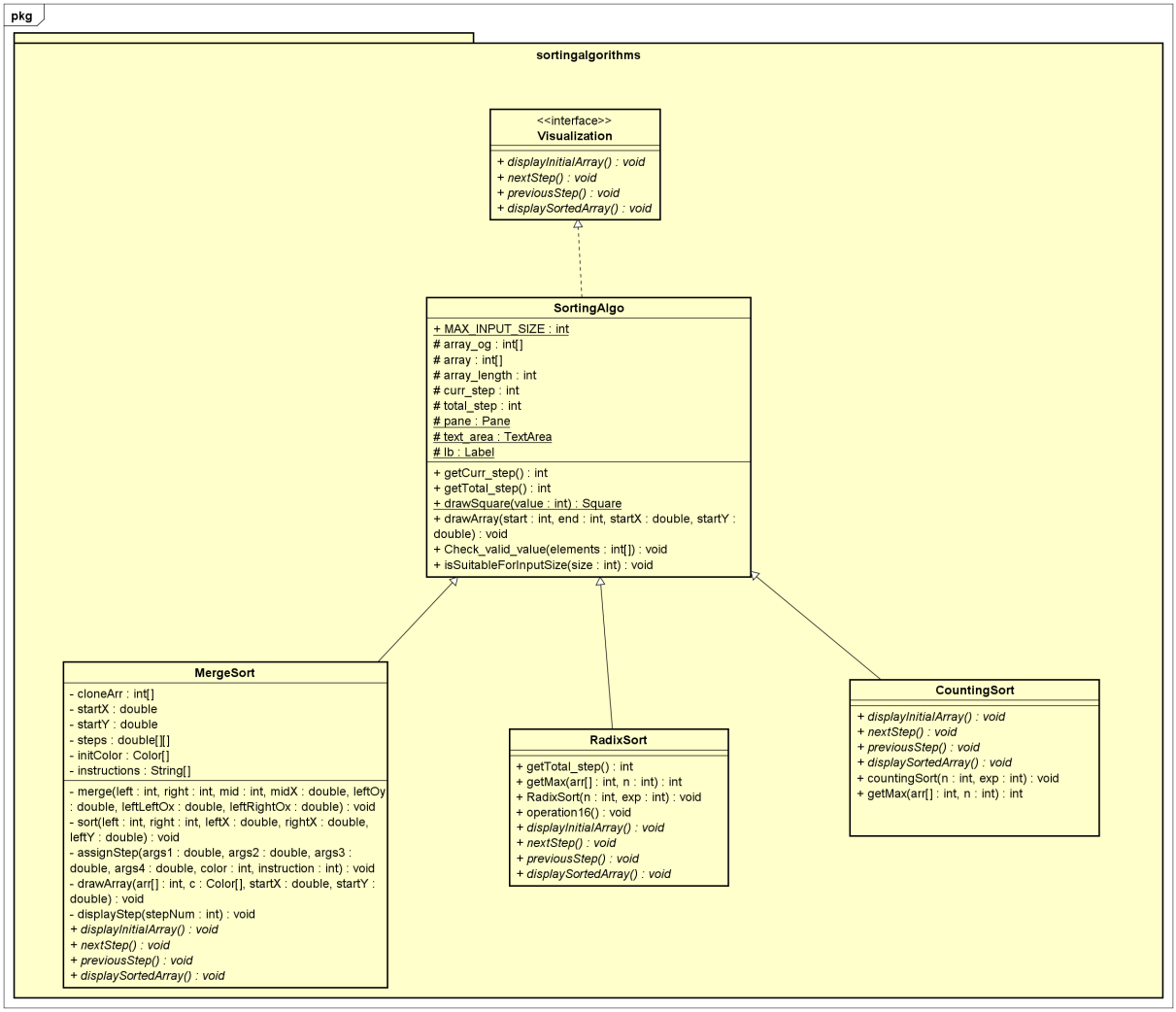

Figure 5: Sorting algorithms

We create an abstract class named "SortingAlgo" which has 5 attributes and 5 methods. The three algorithms inherit from "SortingAlgo" but each has its own unique methods and attributes.

- "array-og" and "array": is the input array.
- "arr-length": is fixed to 8.
- "curr-step": is the current step.
- "total-step": is the total step.
- "getCurr-step" : to get the current step.
- "getTotal-step": to get the total step.
- "drawSquare" : to draw the square elements.
- "drawArray" : to draw the array to be illustrated.

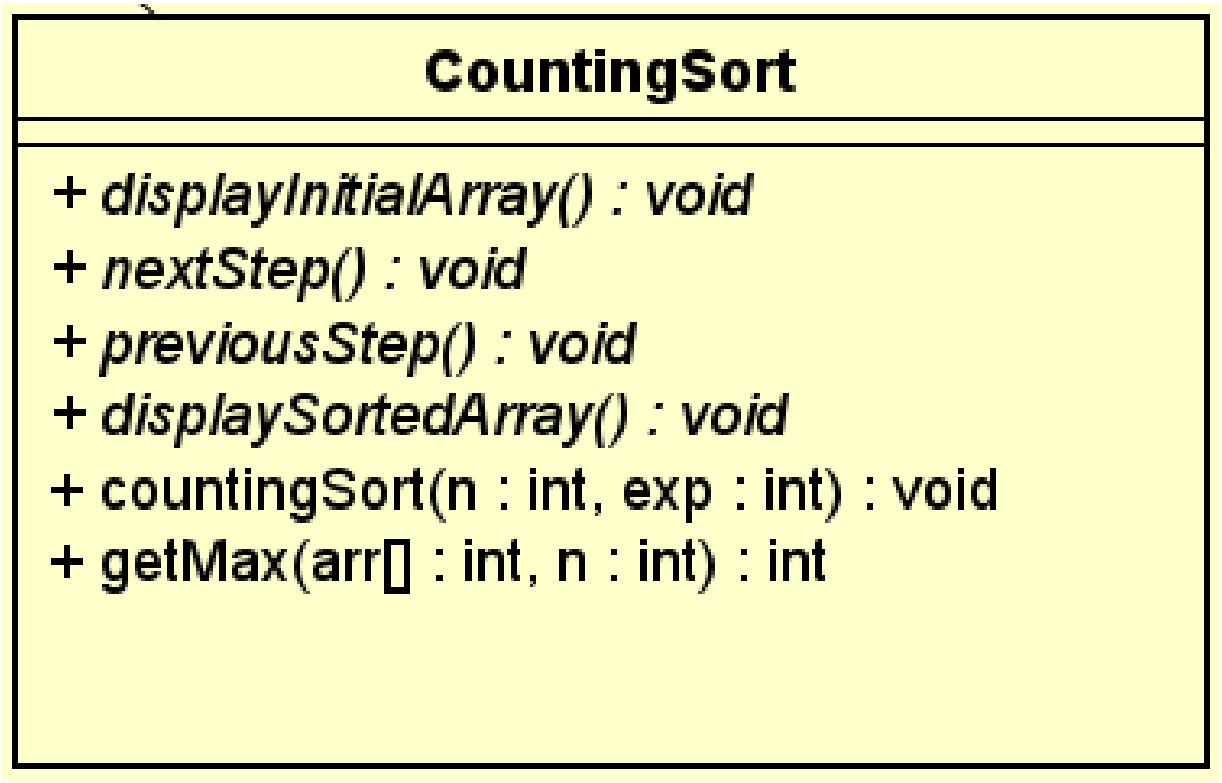

Figure 6: Counting sort

The Counting sort algorithm inherits attributes and methods from the abstract class "SortingAlgo" then having its own attribute, methods to display the whole counting process on the application screen. "nextStep" and "previouStep" help users to control the process.

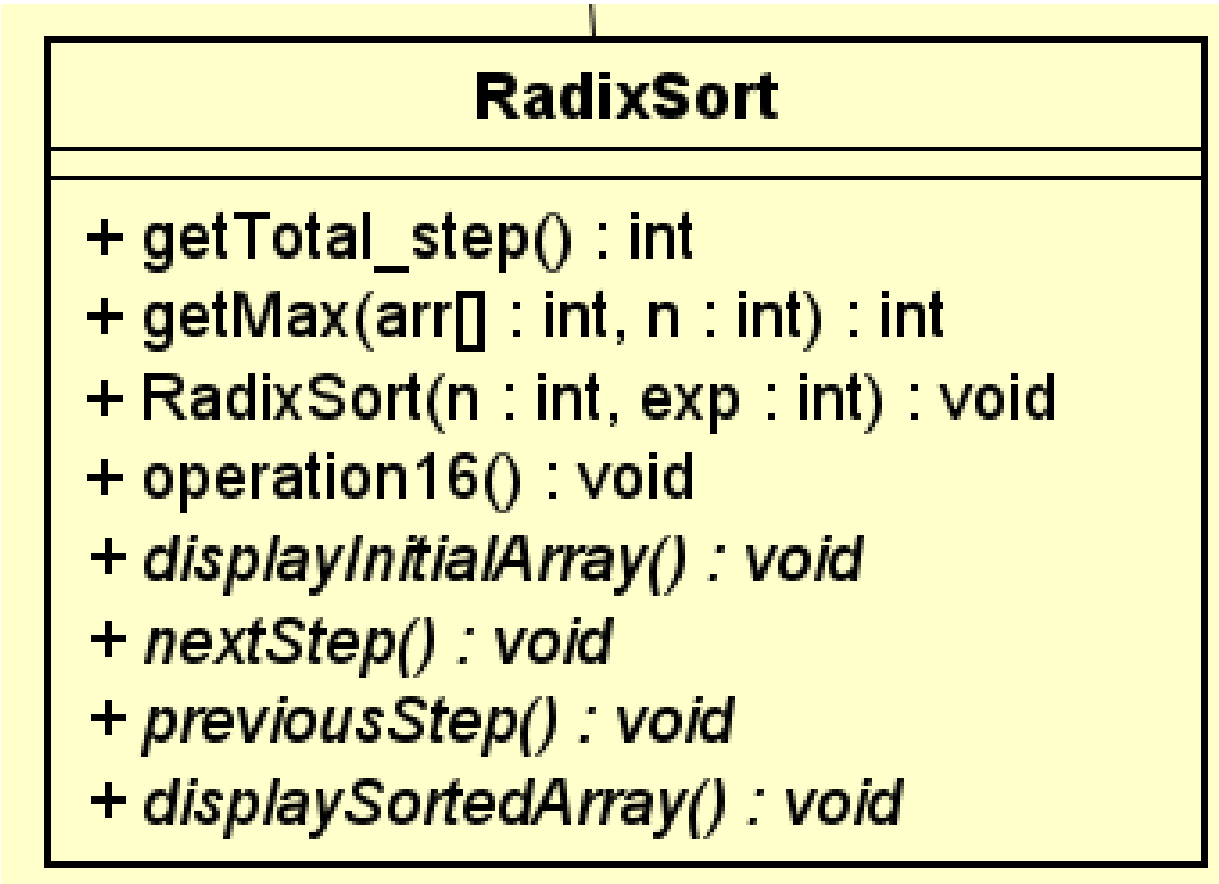

Figure 7: Radix sort

The Radix sort algorithm inherits attributes and methods from the abstract class "SortingAlgo" then having its own attribute, methods to display the whole counting process on the application screen. "nextStep" and "previouStep" help users to control the sorting process.

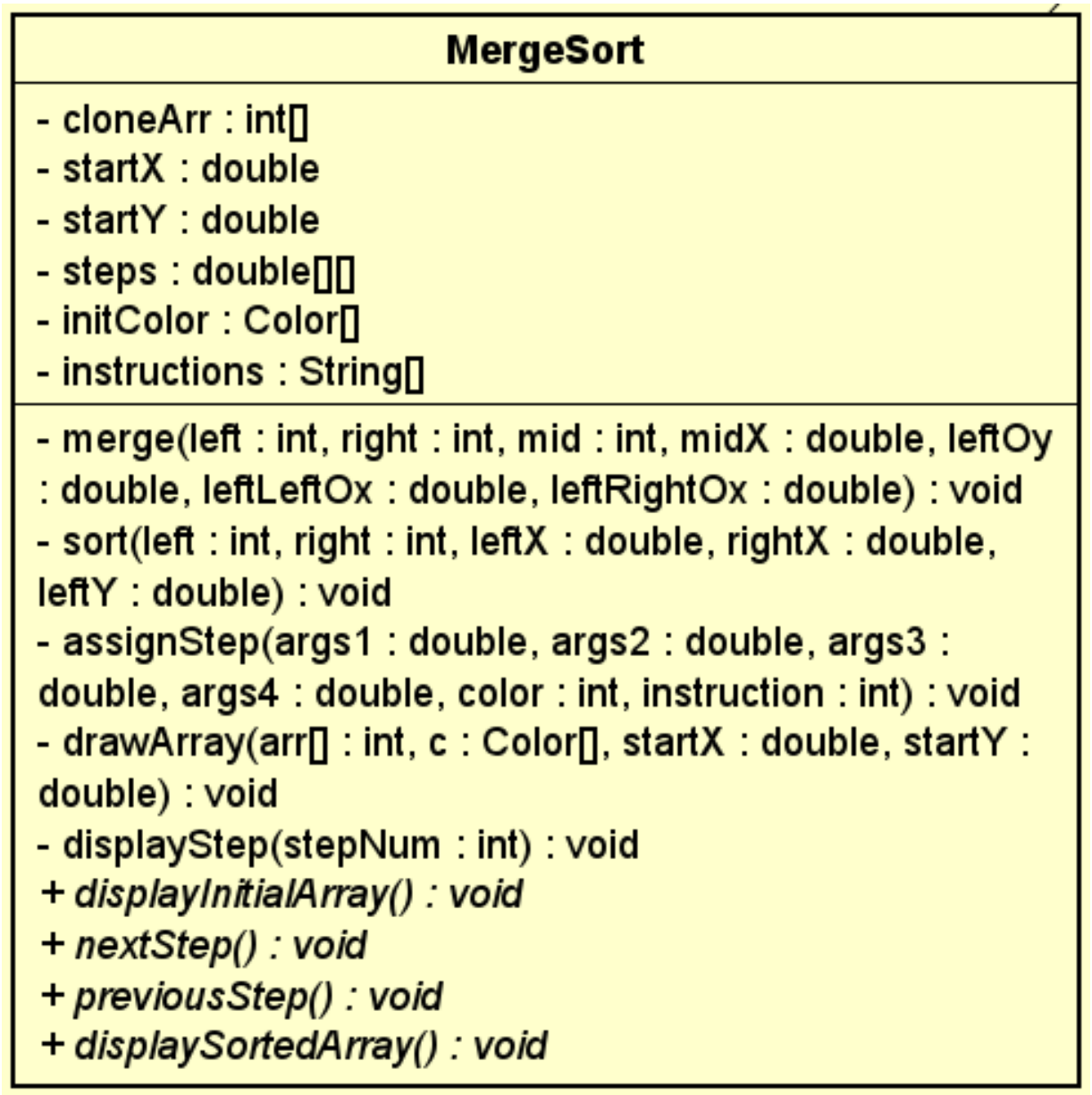

Figure 8: Merge sort

The Merge sort algorithm inherits attributes and methods from the abstract class "SortingAlgo" then having its own attribute, methods to display the whole counting proccess on the application screen. "nextStep" and "previouStep" help users to control the proccess. "merge", "sort", "assignStep" are methods helping the sorting process.

4 OOP Technique and Exception

In figure 5, we use an abstract class called "SortingAlgo", encapsulated it with attributes and methods, hide the values or state of a structured data object inside a class, therefor, preventing

direct access to them by clients in a way that could expose hidden implementation details or violate state invariance maintained by the methods. Moreover, the three classes: "RadixSort", "CountingSort", "MergeSort" then inherit attributes and methods from the abstract class "SortingAlgo", reuse the code from the abstract class. We also use an Interface named "Displayable" with 4 public methods: "displayInitialArray", "nextStep","previousStep", "displaySortedArray".

We use an exception called "IllegalArgumentException" to check whether the input array is good or not. Specifically, we check if numbers in the array are in good value or not and check if the input size is valid or not. The application has the help text, we recommend users to read it before using the application.