**OOP Mini-Project Report**

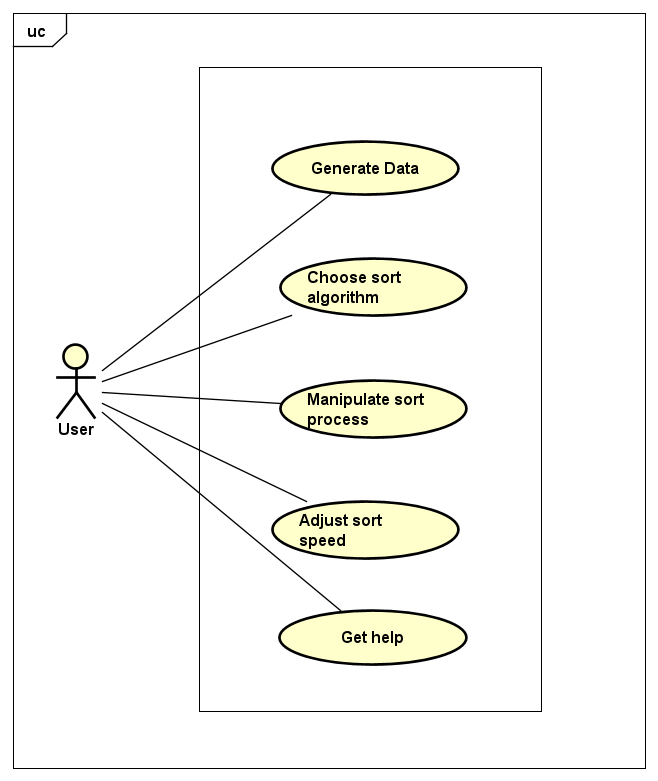
**Team: OOLT.ICT.20202.Team10**

**I. Topic:** 4. Demonstration of sorting algorithms on array (2)

Array is the most basic structure of computer science. Most operations as well as other data structures are built and performed on array. In this project, you will make an application in order to explain three sorting algorithms on array: bubble sort, quick sort, and insertion sort.

**II. Mini-project description:**

* Project Sorting Visualizer help users to visualize process of sorting algorithm for deeply understand how sorting algorithm work step by step.
* Use case:



**III. Assignment of members:**

**1. Ngo Viet Tung (Email:**[**tung.nv184326@sis.hust.edu.vn**](mailto:tung.nv184326@sis.hust.edu.vn)**)**

* Analyze and design use case and class diagram.
* Develop features:
  + Sort algorithm: class SortingAlgorithms, Sort Algorithm info in class ManipulateVisualizer.
  + Statistics info: label comparisons, array accessed in class ManipulateVisualizer.
  + Manipulate sorting process: add actions resume, pause, stop in both class ManipulateVisualizer and ManipulateSortingProcess.
* Contribute to write report and slide making.

**2. Nguyen Phu Truong (**[**truong.np184319@sis.hust.edu.vn**](mailto:truong.np184319@sis.hust.edu.vn)**)**

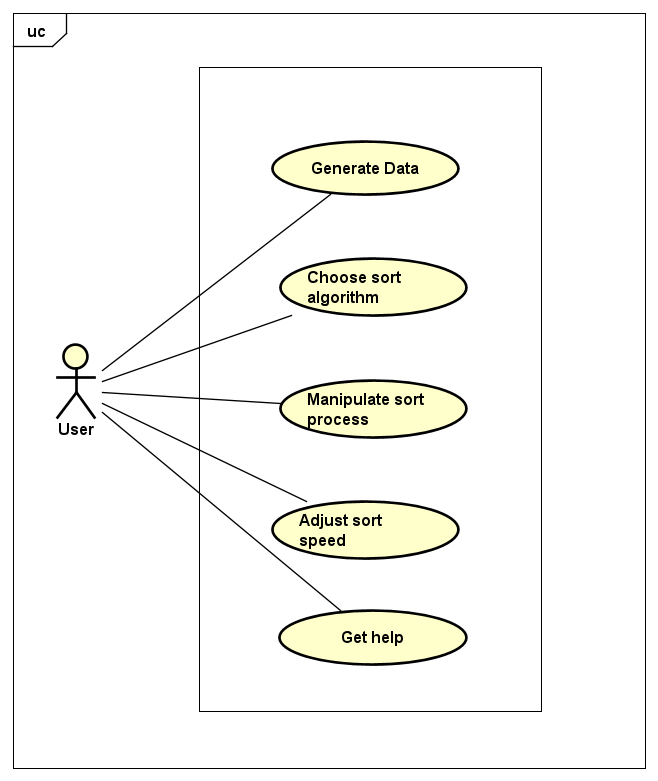
* Analyze and design use case and class diagram.
* Develop features:
  + Sort algorithm: class SortingAlgorithms
  + Manipulate sorting process: sorting(), pause(), reset(), delay(), update() in class ManipulateSortingProcess
* Contribute to write report and slide making.

**3. Phan Huy Thang (Email:**[**thang.ph184305@sis.hust.edu.vn**](mailto:thang.ph184305@sis.hust.edu.vn)**)**

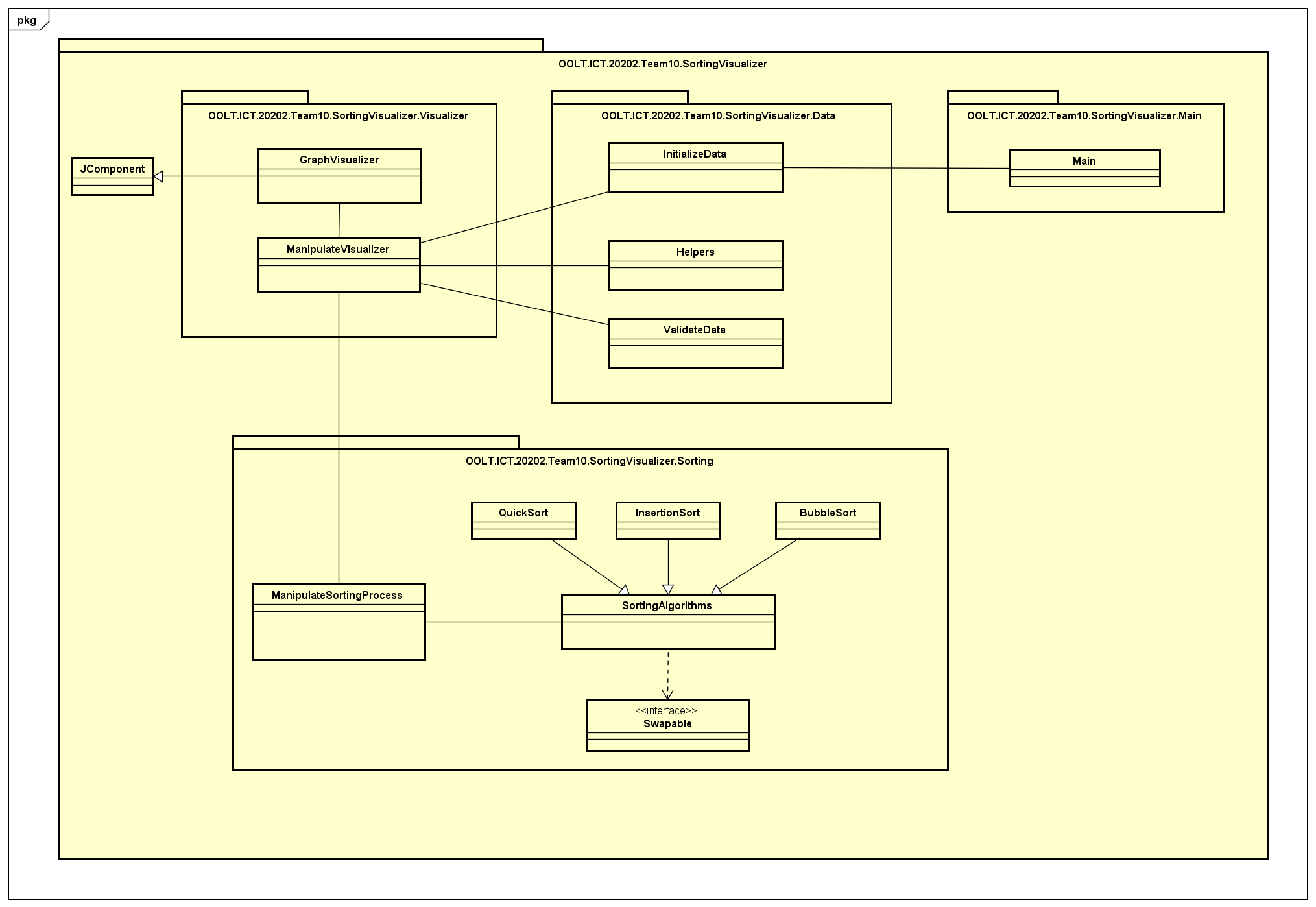
* Analyze and design use case and class diagram.
* Develop features:
  + Data input: generate data in class InitializeData, UI to input data in class SortingAlgorithms.
  + Help dialogs: “?” dialogs and GetHelp dialog.
  + Review PR.
* Contribute to write report and slide making.

**IV. Design:**

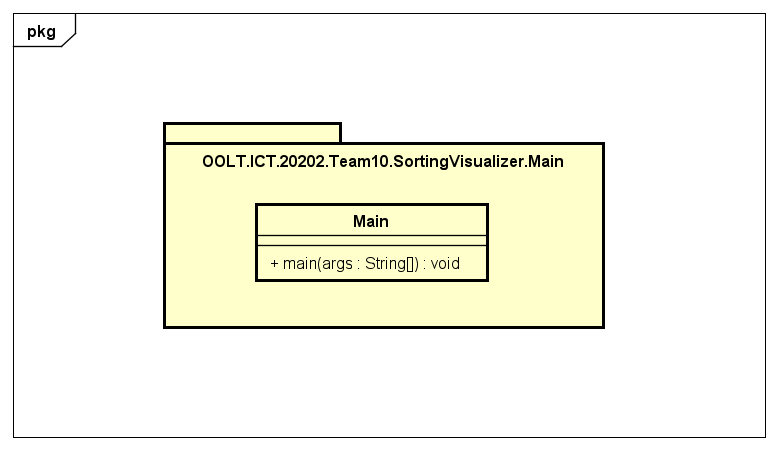
* Use case diagram:



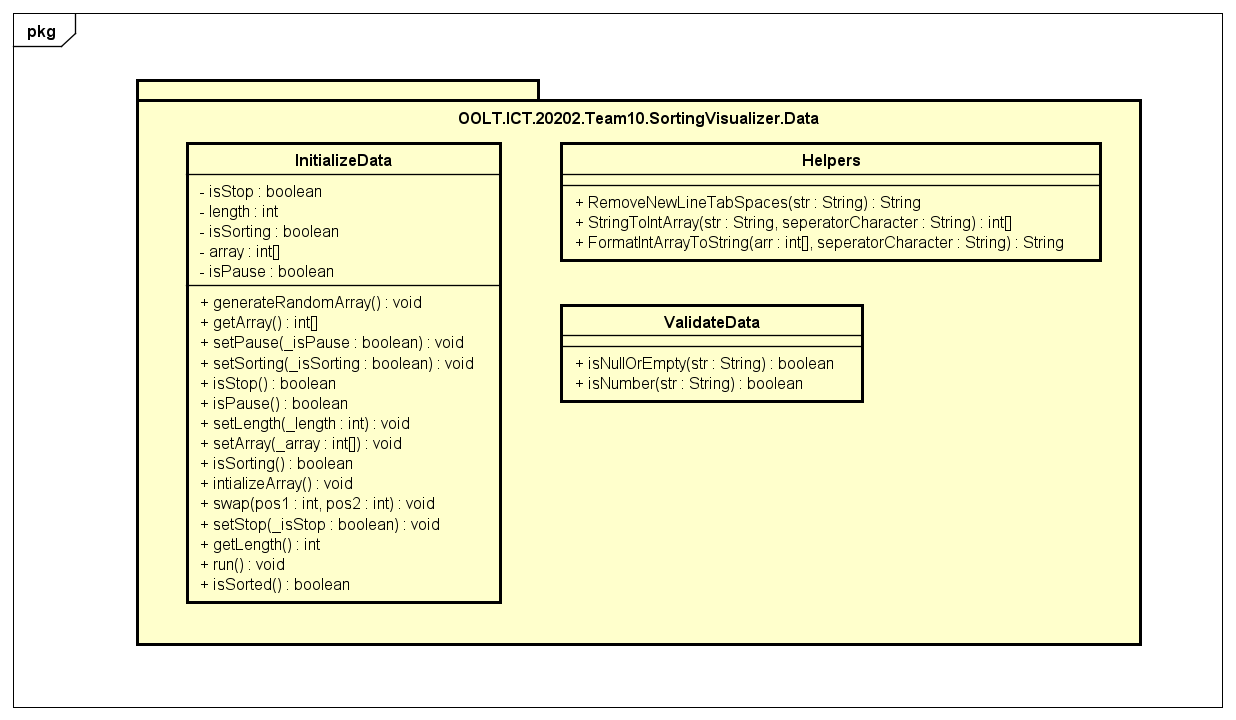
* Class diagram:
  + General overview:



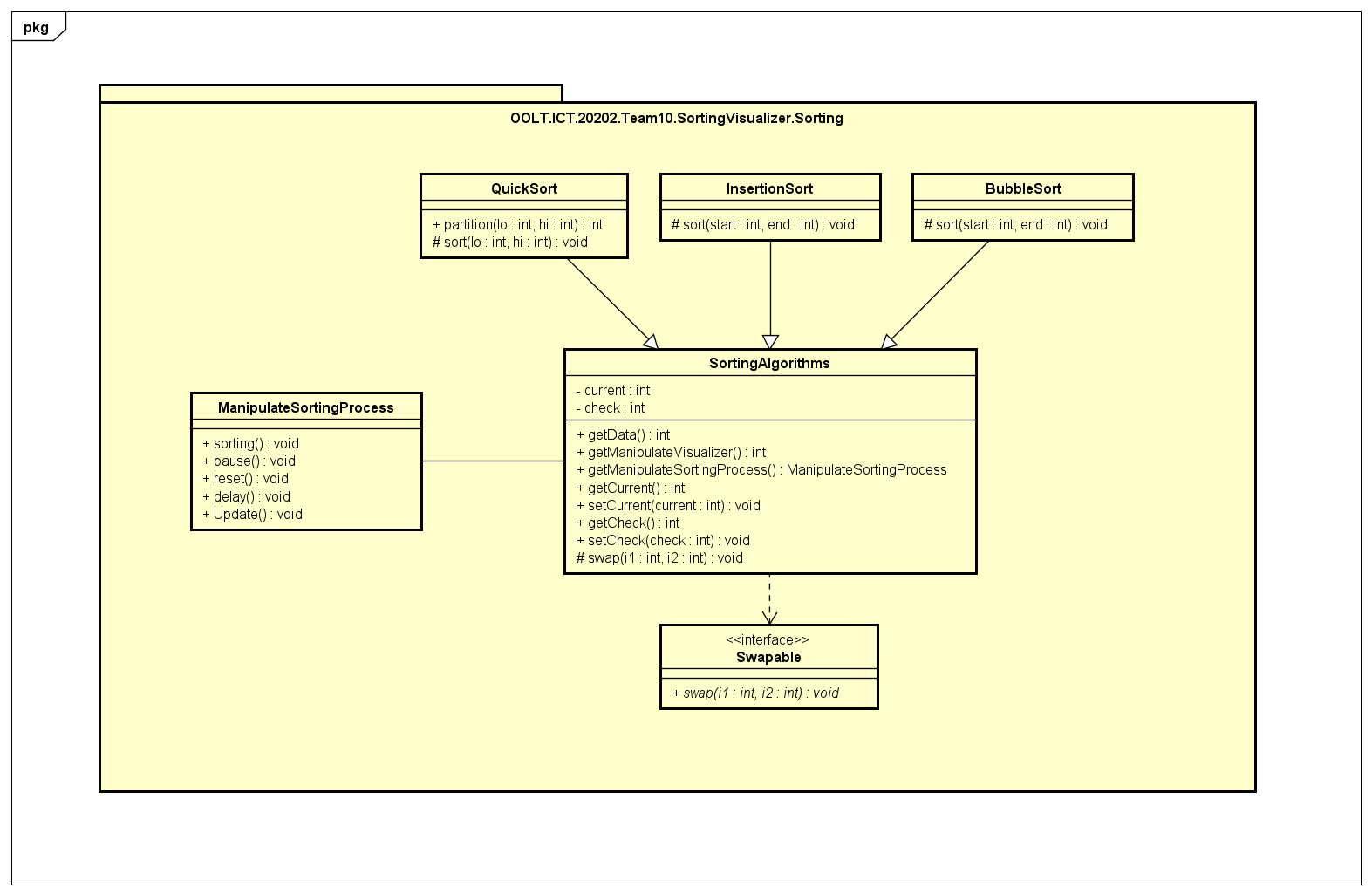
* + Package Main:



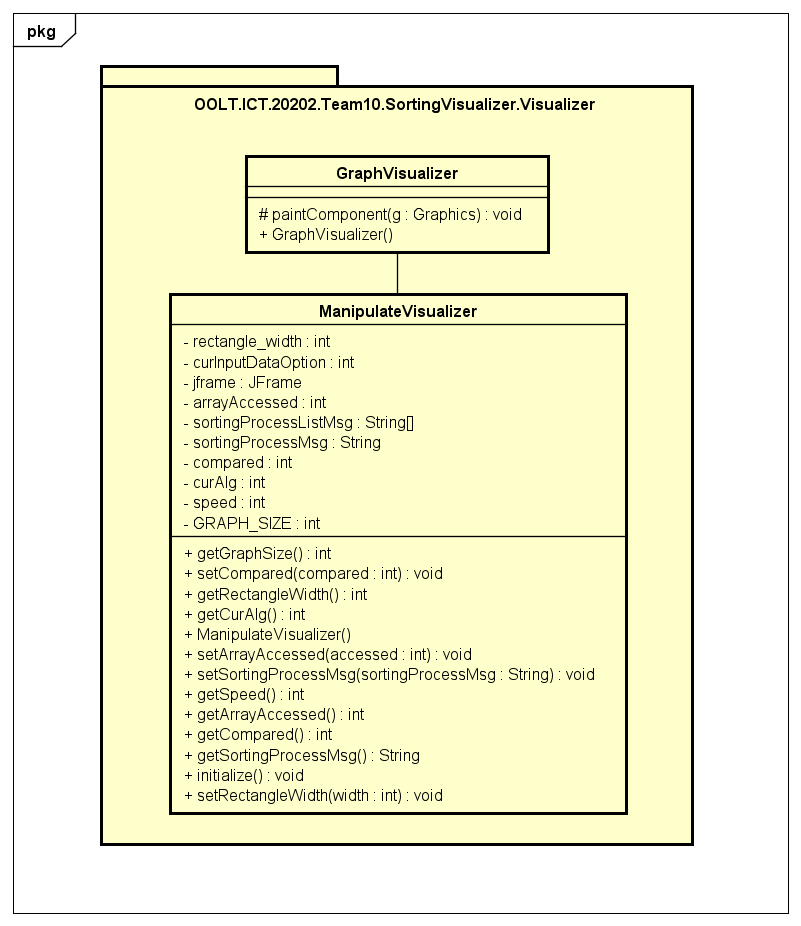
* + Package Data:

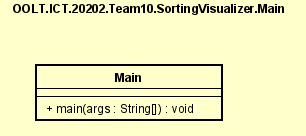


* + Package Sorting:



* + Package Visualizer:





* OOP techniques:
  + Inheritance:
    - Class SortingAlgorithms inherit from interface Swapable.
  + Polymophism:
    - Classes QuickSort, InsertionSort, BubbleSort extend class SortingAlgorithms override method sort.
    - Class GraphVisualizer override method paintComponent in class JComponent.
* Important methods:
  + paintComponent in class GraphVisualizer: for draw graphs.
  + Implementations of sorting algorithms in class SortingAlgorithm: also contains state like pause, stop, sorting,… for manipulating sort process