**OOADJ Lab 6 Hackerrank challenge**

//Assigned by Deepthi

1. **Inheritance:**

Create a program to manage a ramen shop's menu, where various types of ramen have different names, prices, descriptions, and flavours. The system should support ramen types such as tonkotsu, shoyu, miso, and vegetarian, each with distinct ingredients and flavors. Utilize inheritance to design a base class called "Ramen'' with common attributes shared by all ramen types, such as name, price, and description. Then, create specific subclasses for each type of ramen, such as "TonkotsuRamen" (Chasu flavoured), "ShoyuRamen''(Soy Sauce flavoured), "MisoRamen" (Miso Paste flavoured), and "VegetarianRamen", inheriting from the base class "Ramen" and adding style-specific ingredients and flavors.

The system should demonstrate the use of inheritance. Additionally, you could use method overriding, and polymorphism to write better and cleaner code.

**Input Format**

Input is given in the form of one of the following values.

* Tonkotsu Ramen
* Shoyu Ramen
* Miso Ramen
* Vegetarian Ramen

Other inputs are not to be accepted, and must be handled as an invalid choice.

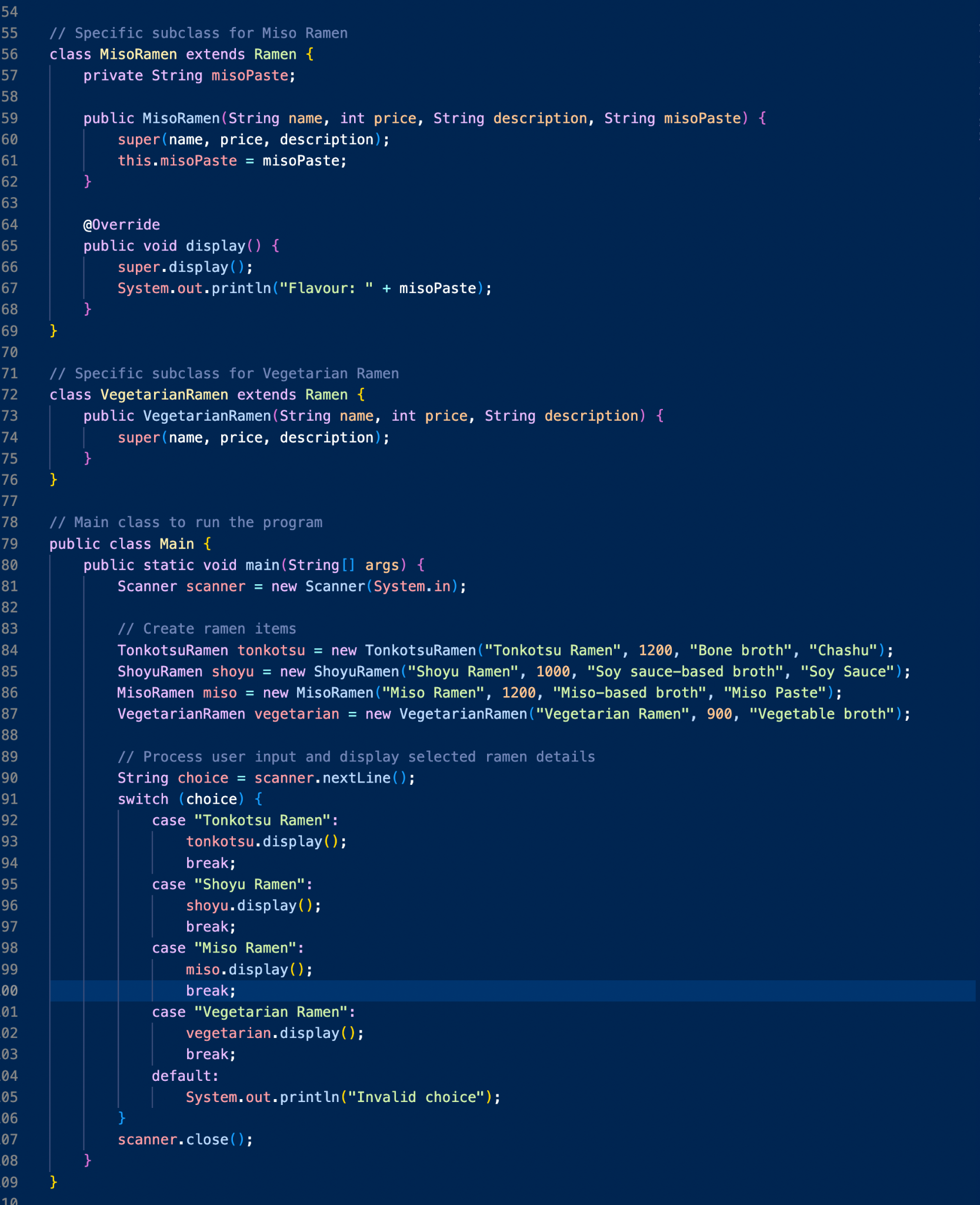
**Constraints**

Use the following code segment to create ramen items in line with output format requirements. (The name, price, description, and flavours are fixed and must be the same as mentioned below to clear test cases.)

**Output Format**

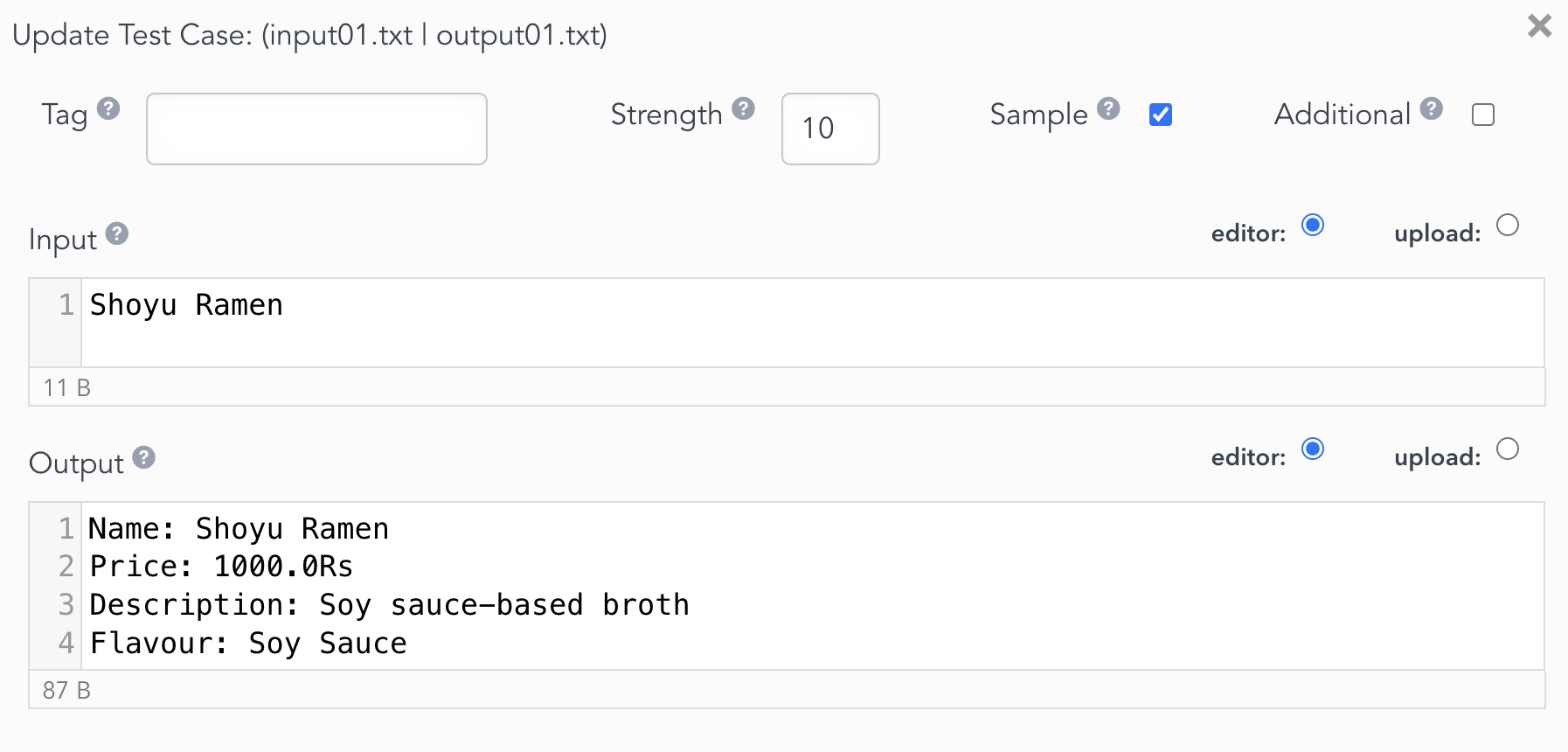
* The output must display the Name, Price, Description and Flavour (if applicable) of the given input.
* If the input isn't one of the four mentioned inputs, default to "Invalid Choice"

**Code:**



**Test Cases:**

Test cases check whether the right input is mapped to the right output and specific attributes of the given input.

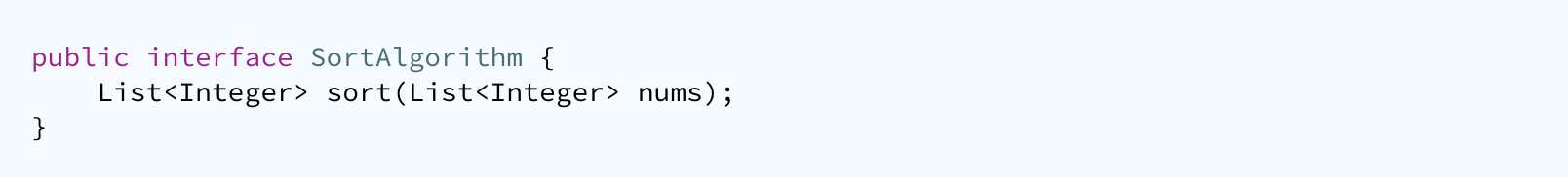




1. **Interfaces**

You are tasked with implementing a program that performs sorting on a list of integers. The program should allow the user to input a list of integers, choose a sorting algorithm, and then display the sorted list. You should utilize interface concepts to define the structure of the sorting algorithms.

You are required to implement the following interface for sorting:



Your task is to implement the SortAlgorithm interface with two sorting algorithms: Bubble Sort and Merge Sort.

Provide your implementation of the SortAlgorithm interface with Bubble Sort and Merge Sort.

**Input Format**

The input is in the form of an array of integers, followed by the name of the sorting method (BubbleSort or MergeSort)

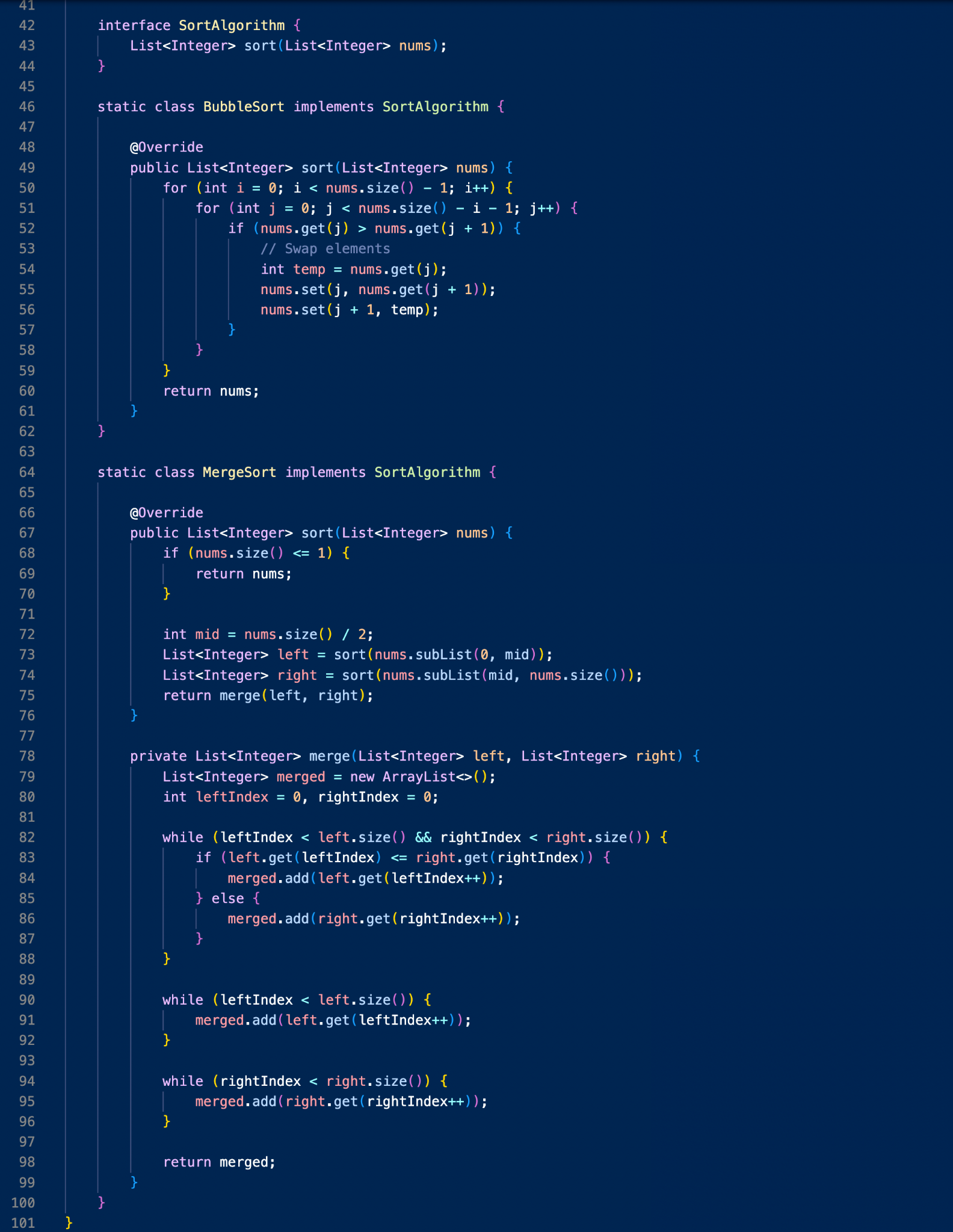
**Constraints**

Follow input and output formatting to clear all test cases.

**Output Format**

The output consists of the sorted list.

**Code**:



**Test cases:**

The program is tested for varying numbers of parameters as well as null values.

****

****