ASSIGNMENT 2B

Assignment 2B tests your knowledge of sorting algorithms (Chapter 23).

Part 1. Design a Java **driver class** (called **_YourName_Assignment2B**; replace **_YourName_** with your actual name), without any fields, just the exact methods (with exact[1] names, parameters, return values, functionality, and order):

Method Name	Description
`YourNameGetArray`	A method that reads 10 integral numbers from user, stores them into an array `A`, and returns the array `A`.
`YourNameCopyArray`	A method with an integral array `A` parameter that create a copy of the array A meaning creates another array `B` with the same integral numbers as A and returns it.
`YourNamePrintArray`	A method with an integral array `A` parameter that prints the array A to the console on the same line, values separated by tabs: `A0        A1    A2    …    An-1`.
`YourNameInsertionSort`	A method with an integral array `A` parameter that sorts `A` using insertion sort algorithm.
`YourNameBubbleSort`	A method with an integral array `A` parameter that sorts `A` using bubble sort algorithm.
`YourNameMergeSort`	A method with an integral array `A` parameter that sorts `A` using merge sort algorithm.
`YourNameQuickSort`	A method with an integral array `A` parameter that sorts `A` using quick sort algorithm.
`main`	- Use the `YourNameGetArray` method to read the values into the `YourNameArray` array. - Output "`INSERTION SORT`" on a separate line. - Use the `YourNameCopyArray` method to create a copy of the `YourNameArray` array into `YourNameArrayI` , output "`BEFORE      `" and then `YourNamePrintArray` method to print `YourNameArrayI`. - Use the `YourNameInsertionSort` method to sort the array `YourNameArrayI` , output "`AFTER: `" and the `YourNamePrintArray` method to print it. - Output "`BUBBLE SORT`" on a separate line. - Use the `YourNameCopyArray` method to create a copy of the `YourNameArray` array into `YourNameArrayB`, output "`BEFORE: `" and then `YourNamePrintArray` method to print `YourNameArrayB`. - Use the `YourNameBubbleSort` method to sort the array `YourNameArrayB` , output "`AFTER: `" and the `YourNamePrintArray` method to print it. - Output "`MERGE SORT`" on a separate line. - Use the `YourNameCopyArray` method to create a copy of the `YourNameArray` array into `YourNameArrayM`, output "`BEFORE: `" and then `YourNamePrintArray` method to print `YourNameArrayM`. - Use the `YourNameMergeSort` method to sort the array `YourNameArrayM` , output "`AFTER: `" and the `YourNamePrintArray` method to print it. - Output "`QUICK SORT`" on a separate line. - Use the `YourNameCopyArray` method to create a copy of the `YourNameArray` array into `YourNameArrayQ`, output "`BEFORE: `" and then `YourNamePrintArray` method to print `YourNameArrayQ`. - Use the `YourNameQuickSort` method to sort the array `YourNameArrayQ` , output "`AFTER: `" and the `YourNamePrintArray` method to print it.

Part 2. Implement the class in NetBeans IDE and JAVA: Create a JAVA project called **_YourNameAssignment2B_** (same name as the driver class) and add your code to **_YourNameAssignmen2B_** project/driver class. Your program should be **user-friendly** (prompt the user for the input using a descriptive message) and be **well-documented/commented** (have comments for every line of code).

Part 3. Create the screenshot document for your code and output: Create a Microsoft Word document from the **_YourName-Assignment1B.docx_** template and call it **_YourName-Assignment_.docx** (replace **_YourName_** with your actual name) Add to the document your **complete screenshots** of the NetBeans editor window showing the **complete JAVA source code** for **_YourNameAssignmen2B_** class and **complete screenshots** the **complete output**. If the entire class JAVA source code or the output does not fit in one screenshot or the screenshots cannot be easily read, create multiple screenshots and add multiple screenshots to the screenshot document. Please keep the screenshots in order (look at the line numbers). If your output is longer than a line and does not fit on one screen, Wrap Text in your output panel. See the instructions on how to use the template.

Part 4. Submit your work: Submit **_YourName-Assignment2B.docx_** on eCampus under the **Assignment 2B.** Do not archive the files (e.g. no RAR, ZIP, etc.) or submit other types/formats of files (e.g. no CLASS, PDF, JAVA, etc.). Please take time after submitting to review the file and check it is correct and resubmit it if there are any errors.

---

[1] Use the exact/precise names (spelling, caps), parameters, returned values, functionality, and do not add or remove fields or methods.