ASSIGNMENT 2

Assignment 2 tests your knowledge of program complexity (Chapter 22), sorting methods (Chapter 23), and user-defined lists, stacks, queues, and priority queues (Chapter 24).

Design a **program/project/driver class YourNameAssignment2** and the following **classes** (with exact[1] names, replace **YourName** with your actual first name or the name you go by):

Class	**Description**
YourNameArray	A new class that contains an array **A** of N integers, a constructor that takes an array as parameter to create the array A, and the following sorting methods (see Chapter 23): ▪ **InsertionSort** that sorts the array A of N integers using insertion sort. ▪ **BubbleSort** that sorts the array A of N integers using bubble sort. ▪ **MergeSort** that sorts the array A of N integers using merge sort. ▪ **QuickSort** that sorts the array A of N integers using quick sort. ▪ **SortAlgorithmComplexity** that prints out to the console for each of the 4 sorting algorithms above, their complexity (see Chapter 22) and an explanation of which one of the sorting methods is better and why. Make sure the entire text shows in screenshots. ▪ **SortArray** that uses the best sorting method from above (that you determined and explained in the SortAlgorithmComplexity) to sort the array A and print the array.
YourNameList	A complete version of user-defined MyList class from Chapter 24 (meaning you need to add the code for the method that were not coded in the book and have a "Left as an exercise" comment). Add an additional output method that outputs the list in the format: [position0, position1, …, positionN]
YourNameArrayList	A complete version of the user-defined MyArrayList class from Chapter 24 that code all the methods (including the "Left as an exercise" methods and the additional output one and uses the YourNameList instead of MyList.
YourNameLinkedList	A complete version of the user-defined MyLinkedList class from Chapter 24 that code all the methods (including the "Left as an exercise" methods and the additional output one and uses the YourNameList instead of MyList.
YourNameStack	A complete version of the user-defined GenericStack class from Chapter 24 that code all the methods (including the "Left as an exercise" methods and the additional output one) and uses the YourNameArrayList instead of the pre-defined ArrayList used in textbook.
YourNameQueue	A complete version of the user-defined GenericQueue class from Chapter 24 that code all the methods (including the "Left as an exercise" methods and the additional output one) and uses a YourNameLinkedList instead of the pre-defined LinkedList used in textbook.
YourNameAssignment2 Driver class main	Creates an array of 10 integers, read the values from the user and build instances of the user-defined lists above (YourNameArrayList, YourNameLinkedList, YourNameStack, and YourNameQueue, and YourNameArray) and test/demonstrate their functionality (e.g. output, add, remove, delete, etc) including the SortArray and SortAlgorithmComplexity.

You can and should start with the examples from the textbook/presentation and adapt them to the assignment at hand and use the appropriate names and add the code and requirements above.

Create a Microsoft Word **screenshots document** called YourNameAssignment2-Screenshot.docx (replace YourName with your actual name) that contains screenshots of the entire JAVA source code in the editor window, (**YourNameAssignment2**) and the entire output (from the driver class). If the entire class JAVA source code or the output does not fit in one screenshot, create multiple screenshots and add them to the document.

Submit the following **8** files: **YourNameAssignment2.java**, **YourNameArray.java**, **YourNameList.java**, **YourNameArrayList.java**, **YourNameLinkedList.java**, **YourNameStack.java**, **YourNameQueue.java** Java source code files and **YourNameAssignment2-Screenshots.docx** screenshots document on eCampus under **Assignment 2.** Do not archive the files or submit other file formats. If you choose add additional classes, please make sure you explain why at the beginning of the screenshots document, attach screenshots to the document and attach the JAVA files to your submission.

[1] Use the exact names (spelling, caps), parameters, returned values, functionality, and do not add or remove fields or methods. Yes, you may find examples in the textbook with different names and cases and with other methods, but you will need to adapt them to have this exact names and cases, to earn credit for the assignment.