LabExercise7—ArrayHelper

The problem is divided into six parts:

1. Lab Objectives
2. Description of the Problem
3. UML Diagram
4. Sample Output
5. Test Code Template
6. Problem-Solving Tips

The test program template represents a complete working C++ test program. Read the problem description and examine the sample output; then study the testcode. Using the problem-solving tips as a guide write your C++ code. Compile and execute the program. Compare your output with the sample output provided.

Lab Objectives

In this lab, you will practice:

- Creating a class usingtemplate to perform somebasic array operations.
- Using function templates to create function-template specializations.

Description of the Problem

Create a class called ArrayHelper and use templates to create a generic implementations. By the help of the templates, this class can handle the all of the primitive types and user defines types after some modifications on the user defines types. In the test code, you will see a integer case of the code. You have to write test code for the case of double.

UML Diagram

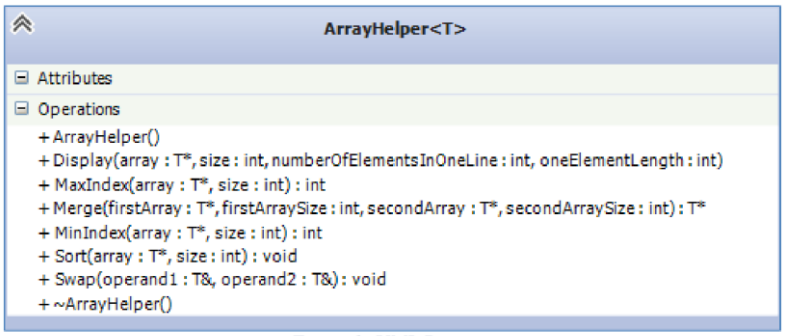
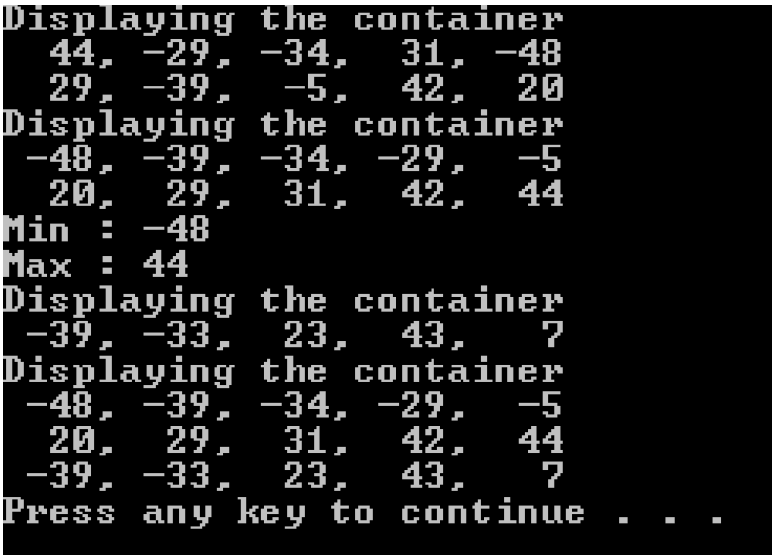


Figure 1: UML Diagram

Functions:

- Display(array, size, numberofElementsInOneLine, onElementLength) : Displays the given array according to the 3 rd and 4th argument.
- MaxIndex(array,size) : Finds the maximum element and returns the index of it.
- MinIndex(array,size) : Finds the minimum element and returns the index of it.
- Merge(firstArray, firstArraySize, secondArray, secondArraySize) : Merges the given arrays and returns a new array containing the merged elements.
- Sort(array, size) : Sorts the given array in ascending order.
- Swap(operand1,operand2) : Swaps the given elements.

Sample Output



Test Code Template

//Given with pdf file.

Problem Solving Tips

1- Use UML Diagram, function DEFINITIONS and sample output