The goal of this lab is to help you get familiar with arrays.

Due Date

You must demonstrate the solution to this lab exercise to the instructor by **Wednesday, September 25, 2019**, in order to receive full credit for this work.

Lab Setup

1. Create the project using Visual Studio.
2. Download the ZIP file for **Lab 7.1** from Moodle.
3. Copy the sample input files, **`numbers.txt`**, and **`fiveNumbers.txt`** from the ZIP file to the same folder where your source file is located.

Programming Exercise

This lab exercise involves writing a program with several functions:

The "**main**" function must:

- Define an **array** of 12 integers:

```
const int ARRAY_SIZE = 12;
int numbers[ARRAY_SIZE];
```

- Prompt the user for an input file name. Open this file for input. The file contains integers (one per line).
- Read the contents of the file into the array.
- Call four *separate* functions: **getLowest**, **getHighest**, **getSum**, and **getAverage,** and display the value returned by each function.

Each of the other functions must contain a **loop** that scans the data in the array and returns the appropriate result:

The **getLowest** function returns the lowest number in the array.
The **getHighest** function returns the highest number in the array.
The **getSum** function returns the total of the numbers in the array.
The **getAverage** function returns the average of the numbers in the array.

Design Restriction

For this lab exercise use an **array** of integers, ***not*** a vector of integers. We will use vectors later in the course, but for now we need to use only arrays.

Sample Data Files and Sample Program Output

The ZIP file, which can be downloaded from Moodle, contains two text files. These should be used to test your program. (They are also shown on the next page.)

The sample data and sample program output are provided below. (In the sample program output shown in this document, text typed by the user is shown in **BOLD** font. In actuality, all text will be displayed in the same font)

Input file: numbers.txt

```
53
22
87
103
-3
75
220
1
64
543
98
44
```

Output from processing the numbers.txt file

```
Enter name of input file: numbers.txt
12 numbers read from input file.
The highest value is 543
The lowest value is -3
The sum of the numbers is 1307
The average of the numbers is 108.917
Press any key to continue . . .
```

Input file: fiveNumbers.txt

```
-53
-22
-87
-103
-3
```

Output from processing the fiveNumbers.txt file

```
Enter name of input file: fiveNumbers.txt
5 numbers read from input file.
The highest value is -3
The lowest value is -103
The sum of the numbers is -268
The average of the numbers is -53.6
Press any key to continue . . .
```