## LAB 3 IntArrayPlay – playing with arrays

IntArrayPlay2023.cpp

### Instructions:

You will be building array functions to read an array, insert, delete, sort, reverse.

The array contains integers.

Please:

* Write one function at a time. Test it to make sure it **really works**, and then move on to next function.
* Write comments to:

1) document your algorithms and design, including input parameters, return values and known limitations

2) make your code readable to others

**Starter Code: Follow ToDo instructions, in order.**

#include <iostream>  
using namespace std;  
  
  
int main()  
{

const int CAPACITY=20;  
 int numArray[CAPACITY]; // an int array with a given CAPACITY

// numArray can be partially filled, (so use numArrayElems below)  
  
 int numArrayElems=0; // the array is initially empty, i.e., contains 0 elements  
 // Actual number of elements stored

// 1. ToDo: Build and call **fillArray** function

// fills an int array with values entered by the user.

// Stops reading when the user inputs -1 or CAPACITY is reached.

// Use this prompt: "**Enter non-negative integers. End with -1:** "

// 2. ToDo: Build and call **printArray** function.

// printArray should print elements separated by a space, ending with newline

// 3. ToDo: Build **IsSorted** function.

// return a Boolean, indicating if the array is sorted (low to high).

// 4. If the array is sorted, print "**Array is sorted\n**"

// otherwise print "**Array is not sorted\n**"

// 5. ToDo: Define and Build your own function to:

// Read a value from the user and **insert** it into the array

// Use this prompt: "**Enter a value to insert:** "  
 // If the array is sorted, put it in the proper location, keeping the order

// Consider making a function to determine the proper location

// The proper location has a value >= the new value.

// Push all the other numbers to the right to add this value

// If the array is not sorted, you may put it at the end.

// 6. ToDo: Call printArray

// 7. ToDo: Define and Build your own function to:

// Read a value from the user and **delete** it from the array, if it is there.

// Use this prompt: "**Enter a value to delete:** "

// If it is not there, print "**Not found\n**", and return (don’t change the array)

// Consider making a function **findInArray** which returns the

// index of the given value or -1 if not found.

// If the array is sorted, be sure to maintain the order.

// If the array is not sorted, replace the found element with the last element.

// 8. ToDo: Call printArray

// 9. ToDo: Define and Build your own **Sort** function.  
 // Use Bubble Sort or Selection Sort from PowerPoint 2 (Arrays)

// 10. ONLY if the array is not sorted:

// Sort it

// Call printArray

// Verify that it is sorted by using isSorted

// If the array is sorted, print "Array is sorted\n"

// otherwise print "Array is not sorted\n"

// 11. ToDo: Define and Build your own function to **Reverse**

// the elements array.

// 12. ToDo: call your Reverse function with the array.

cout << "Reversing the Array produces: " << endl;

// 13. ToDo: Call printArray

return 0;

}

//TODO: Implement all functions declared above.

## Test Cases

1. Not Sorted Add 5 Remove 52

Enter non-negative integers. End with -1: 11 15 68 52 65 54 -1

11 15 68 52 65 54

Array is not sorted

Enter a value to insert: 5

11 15 68 52 65 54 5

Enter a value to delete: 52

11 15 68 5 65 54

5 11 15 54 65 68

Array is sorted

Reversing the Array is:

68 65 54 15 11 5

Input 1:

11 15 68 52 65 54 -1

5

52

1. Sorted Add 31 Remove 62

Enter non-negative integers. End with -1:

5 10 15 20 25 30 35 40 45 50 55 62 65 -1

5 10 15 20 25 30 35 40 45 50 55 62 65

Array is sorted

Enter a value to insert: 31

5 10 15 20 25 30 31 35 40 45 50 55 62 65

Enter a value to delete: 62

5 10 15 20 25 30 31 35 40 45 50 55 65

Reversing the Array is:

65 55 50 45 40 35 31 30 25 20 15 10 5

Input 2:

5 10 15 20 25 30 35 40 45 50 55 62 65 -1

31

62

1. Not Sorted Add 43 Remove 17 (not there)

Enter non-negative integers. End with -1: 31 25 78 62 65 14

Array is not sorted

Enter a value to insert: 31 25 78 62 65 14 43

Enter a value to delete: Not found

31 25 78 62 65 14 43

14 25 31 43 62 65 78

Array is sorted

Reversing the Array is:

78 65 62 43 31 25 14

Input 3:

5 10 15 20 25 30 35 40 45 50 55 62 65 -1

31

62