COMP 116 Data Structures

Lab #2

In this lab, we will practice the use of functions, while continuing to use arrays and control structures, by implementing sorting algorithms for arrays. You can do all the exercises in a project called Lab02.

1. First, we need functions to fill and print the array of integers.

   (a) Write a function `getInt` that takes no parameters, asks the user to enter an integer and returns it.

   (b) Write a function `fillArray` that takes as parameters an array of `int` and its size (another `int`) as parameters, and uses the previous function to successively ask the user to fill the array with integers. The function does not return anything.

   (c) Write a function `printArray` that takes an array of `int` and its size as parameters, and prints the elements of the array on a single line, separated by a space. The function does not return anything.

   Finally, in the function `main`, declare an array of `int` of size 10, fill it and print its content using the functions above.

   _____ Show me the result when you are done.

2. Our first sorting algorithm is called **Selection Sort**.

   (a) Write a function `findMaxIndex` that takes an array and its size as parameters, and returns the *index* of its maximal element.

   (b) Write a function `selectionSort` that takes an array and its size as parameters. The function should do as follows:

   - Find the index of the maximal element of the array using the function `findMaxIndex`.
   - Swap the element at that index with the last element of the array. So the maximal element is now in the correct position for the array to be sorted.
   - Find the index of the maximal element among the remaining elements of the array using the function `findMaxIndex`. You can do this by *pretending* that the array has size one less than the first time.
   - Swap the element at that index with the second to last element of the array. So the last two elements are now in the correct position for the array to be sorted.
   - Find the index of the maximal element among the remaining elements of the array using the function `findMaxIndex`. You can do this by *pretending* that the array has size two less than the first time.

- Swap the element at that index with the third to last element of the array. So the last three elements are now in the correct position for the array to be sorted.
- etc, all the way until all the elements are in the correct position

The function does not return anything.

Finally, in the function `main`, make a call to selectionSort between the call to fill the array and the call to print it. Confirm that the elements are now in sorted order.

_____ Show me the result when you are done.

3. If time allows, you can try to implement a second sorting algorithm called **Insertion Sort**.

   (a) Write a function `insert` that takes an array and an index `ind` (an `int`) as parameters. The function assumes that the first `ind` numbers of the array are in sorted order and tries to insert the element at index `ind` in its correct position in the array.

   - Store the element at index `ind` in a temporary variable.
   - Start a counter variable `current` at index `ind` - 1.
   - While `current` is greater than or equal to zero and the element at index `current` is greater than the temporary variable
     - move the element at index `current` one position to the right in the array
     - decrement `current` by 1.
   - Once you are out of the while loop, plop the element from the temporary variable into index `current` in the array.

   Stare at this code for a few minutes and convince yourself that at the end, the first `ind` + 1 elements of the array are in sorted order.

   (b) Write a function `insertionSort` that takes an array and its size as input. The function should do as follows:

   - insert the second element
   - insert the third element
   - insert the fourth element
   - etc, until all the elements have been inserted.

   The function does not return anything

   Finally, replace the call to `selectionSort` from main by a call to `insertionSort`

   _____ Show me the result when you are done.

When you are done, write your name on the sheet and hand it to the lab instructor.

Name: _____