

# CS 469 Special Topics in Computer Science

## Assignment 2

### Exercise 1

Write an algorithm and a C++ program that reverse the order of the elements of an array that is assumed to have been previously entered ("the first will be the last ..."). The reversion must be performed without the usage of any intermediary array.

### Exercise 2

Write an algorithm and a C++ program that allow the user to enter the elements of an array, and that checks if they are all different. The algorithm will simply display "There are one or more duplicates" or "There are no duplicates" depending on the case.

### Exercise 3: Dynamic arrays

Write an algorithm and a C++ program that allows the user to delete a value from a previously entered array. The user will give the index of the value he wishes to delete. Be careful, it is not a question of setting a value to zero, but rather of removing it from the array itself! If the starting array was:

|    |   |   |    |    |   |   |
|----|---|---|----|----|---|---|
| 12 | 8 | 4 | 45 | 64 | 9 | 2 |
|----|---|---|----|----|---|---|

And that the user wants to delete the index value 4, the new table will be:

|    |   |   |    |   |   |
|----|---|---|----|---|---|
| 12 | 8 | 4 | 45 | 9 | 2 |
|----|---|---|----|---|---|

Perform the same principle of dynamic arrays for the case of the insertion of a new element in the array.

### Exercise 4: Sorting algorithms

Implement in C++ the following sorting algorithms we saw in class:

1. Insertion sort
2. Selection sort
3. Bubble sort

4. Merge sort
5. Quicksort

Test each of the above algorithms by varying the size of the array from 10 to 10000, and compute the elapsed time for each sorting algorithm according to the size of the array. Draw the associated curves. Explain briefly the obtained results.