

# CSC 2720: Data Structures

## Lab 5

Instructor: Shiraj Pokharel

Due : @ 11:00 PM ET , Next Day after release

Answer the below questions. You may use whatever IDEs / editors you like, but you must submit your responses on iCollege as .java files. Failure to comply with this simple requirement will result in a score of Zero. Please, be careful not to be assigned a Zero score this way.

*Few Rules to be followed, else will receive a score of ZERO*

- (1) Your submissions will work exactly as required.
- (2) Your files shall not be incomplete or worse corrupted such that the file does not compile at all. Make sure you submit a file that compiles.
- (3) Your submission will show an output. Should you receive a Zero for no output shown do not bother to email me with "but the logic is perfect" !

Note that your program's output must **exactly** match the specs(design , style) given here for each problem to pass the instructor's test cases .

*Design* refers to how well your code is written (i.e. is it clear, efficient, and elegant), while *Style* refers to the readability of your code (commented, correct indentation, good variable names).

**ATTN : *This week I have "slightly modified" the problem we solved last week. Here, we remove the fundamental assumption we made about input arrays being always sorted :)***

In user content generated web-services - lets say YouTube - the process of de-duplication is of serious importance. One straight forward reason is same video by different names is just an extra cost in data-storage. So getting rid of exact duplicate content makes financial sense. Please be reminded that exact same content with different video qualities may not be candidates for removal via de-duplication.

In today's Lab we will explore on ways to do a de-duplication of videos where video filenames are presented as integers. For the purposes of our task, we will

set a very narrow criterion for de-duplication : just the filenames.

Below is how the filenames are represented  
ARRAY[] = [50, 11, 33, 21, 40, 50, 40, 40, 21]

Below is the expected output after de-duplication  
ARRAY[] = [11, 21, 33, 40, 50]

ATTN : Notice the reduced size of the output array.  
Also, take a look again at the input array and be reminded that the input array is not sorted !

Further, please be reminded that you cannot use library functions to either sort and or perform the de-duplication operation. Doing so would straight up result in a score of Zero !

You will solve the problem in two ways:-

- (1) [50 points] Implement the function in such a way that your solution solves the problem with  $O(n^2)$  time complexity overall and  $O(1)$  space complexity. Here,  $n$  is the length of the list of input integers (array). In this part of the problem you will be using the **Selection Sort Algorithm** that we covered in class. Please sort the array with that algorithm and solve the de-duplication problem thereafter. De-duplication part of the solution in itself must adhere to  $O(n)$  time and  $O(1)$  space bounds.
- (2) [50 points] In a separate implementation, code up a solution in such a way that your solution solves the problem with  $O(n^2)$  time complexity overall and stays within the bounds of  $O(1)$  space complexity. Here,  $n$  is the length of the list of input integers (array). In this part of the problem you will be using the **Bubble Sort Algorithm** that we covered in class. Please sort the array with that algorithm and solve the de-duplication problem thereafter. De-duplication part of the solution in itself must adhere to  $O(n)$  time and  $O(1)$  space bounds.

Very Very Important :

- (1) Your code should be well commented which explains all the steps you are performing to solve the problem. A submission without code comments will immediately be deducted 15 points !

- (2) As a comment in your code, please write your test-cases on how you would test your solution assumptions and hence your code.

**A submission without test cases will immediately be deducted 15 points !**

Example of cases to be tested for are like : What if the array input which is expected does not exist - that is , input is a null. How should your code handle such a situation ? Maybe output some message like "Null input case, so no output"? What if the length of the array is one ?....so on and so forth.

Please Remember : Although, written as comments - You will address your test cases in the form of code and not prose :)