This assignment is due on March 22. Submit your assignment via Gradescope. All submitted work must be done individually without consulting someone else's solutions in accordance with the University's "Academic Dishonesty and Plagiarism" policies.

As a first step, go to the last page and read the section: Advice on how to do the homework.

Problem 1 (10 points)

Consider the following snippet of pseudocode that takes as input an array of n integers.

```
function ALGORITHM(A)
    n ← length of A
    num_matches ← 0
    for i ∈ [0 : n] do                          ▷ i ranges from 0 to n − 1
        for j ∈ [i + 1 : n] do                  ▷ j ranges from i + 1 to n − 1
            if A[i] == A[j] then
                num_matches ← num_matches + 1
    return num_matches
```

Your task is to:

a) upperbound the running time of the algorithm in terms of n using O-notation.

b) lowerbound the running time of the algorithm in terms of n Using Ω-notation.

Problem 2 (25 points)

Consider a stack where each element stores an integer value. We want to extend the stack ADT with the additional operations listed below. The operations should run in $O(1)$ amortized time and the running time of the other stack operations should remain the same as those of a regular stack:

Sum() returns the sum of all the elements on the stack

Min() returns the smallest value of all the elements on the stack

PopSmaller(e) pop the stack until the top element stores a value that is bigger than the value stored in e, then *push(e)*. For example, if the stack contains elements with values $[4, 7, 9]$ then operation *PopSmaller(e)*, where e stores the value 8, will result in the stack $[8, 9]$.

For each operation, describe their implementation in English (pseudocode is optional), argue the correctness, and the running time.

Problem 3 (25 points)

As input we are given a *sorted* array B containing n positive integers, together with an integer m. The aim is to compute how many indices i and j (with $i < j$) there are such that the sum of the i th and the j th elements of B is at least m, that is, $A[i] + A[j] \geq m$. For full marks, your algorithm needs to run in $O(n)$ time.

Example:
$B = [1, 4, 4, 6]$, $m = 7 \rightarrow$ return 4

a) Design an algorithm that solves the problem.

b) Argue the correctness of your algorithm.

c) Analyze the running time of your algorithm.

Advice on how to do the home work

- Assignments should be typed and submitted as pdf (no handwriting)
- When designing an algorithm or data structure, it might help you (and us) if you briefly describe your general idea, and after that you might want to develop and elaborate on details. If we don't see/understand your general idea, we cannot give you points for it.
- Be careful with giving multiple or alternative answers. If you give multiple answers, then we will give you marks only for "your worst answer", as this indicates how well you understood the question.
- Some of the questions are very easy (with the help of the lecture notes or book). You can use the material presented in the lecture or book without proving it. You do not need to write more than necessary.
- When giving answers to questions, always prove/explain/motivate your answers.
- When giving an algorithm as an answer, the algorithm does not have to be given as (pseudo-)code.
- If you do give (pseudo-)code, then you still have to explain your code and your ideas in plain English.
- Unless otherwise stated, we always ask about worst-case analysis, worst-case running times, etc.
- As in the lecture, and as is typical for an algorithms course, we are interested in the most efficient algorithms and data structures.
- If you use additional resources (books, scientific papers, the internet, etc.) to formulate your answers, then add references to your sources.
- If you refer to a result in a scientific paper or on the Web, you need to explain the results to show that you understand the results and how it was proven.