# Assignment 3

## COMPSCI 2CO3: Data Structures and Algorithms–Fall 2025

Deadline: October 10, 2025

Department of Computing and Software
McMaster University

Please read the *Course Outline* for the general policies related to assignments.

**Plagiarism is a *serious academic offense* and will be handled accordingly.**
**All suspicions will be reported to the *Office of Academic Integrity***
**(in accordance with the Academic Integrity Policy).**

This assignment is an *individual* assignment: do not submit work of others. All parts of your submission *must* be your own work and be based on your own ideas and conclusions. Only *discuss or share* any parts of your submissions with your TA or instructor. You are *responsible for protecting* your work: you are strongly advised to password-protect and lock your electronic devices (e.g., laptop) and to not share your logins with partners or friends!

If you *submit* work, then you are certifying that you are aware of the *Plagiarism and Academic Dishonesty* policy of this course outlined in this section, that you are aware of the Academic Integrity Policy, and that you have completed the submitted work entirely yourself. Furthermore, by submitting work, you agree to automated and manual plagiarism checking of all submitted work.

*Late submission policy.* Late submissions will receive a late penalty of 20% on the score per day late (with a five hour grace period on the first day, e.g., to deal with technical issues) and submissions five days (or more) past the due date are not accepted. In case of technical issues while submitting, contact the instructor *before* the deadline.

**Problem 1.** Consider the following algorithm

**Algorithm** SORT$(L[0 \ldots N))$ **:**
**Pre:** $L$ is an *array*.

```
1: for i := 0 to N (excluding N) do
2:   for j := N − 1 to i (excluding i) do
3:     if L[j] < L[j − 1] then
4:       Exchange L[j] and L[j − 1].
5:     end if
6:   end for
7: end for
```

**Post:** $L$ is sorted.

P1.1. Does the SORT algorithm sort correctly? If yes, then provide invariants for the two loops and provide a bound function that can be used to prove the correctness of the algorithm. If no, show how to correct the algorithm.

P1.2. Assume the algorithm SORT is correct. Is the algorithm stable? If yes, then explain why. Otherwise, show how to correct the algorithm such that the result is stable.

P1.3. What is the worst-case running time of this algorithm? Is this algorithm optimal? Explain your arguments.

P1.4. What is the worst-case memory usage of this algorithm? Explain your answer.

**Problem 2.** Some lists are sorted, while others are far away from sorted. To measure how close to a *sorted* list a list $L$ is, we will count the number of *sort violations*. Let $0 \leq i < j < |L|$ be two positions in list $L$. The pair $(i, j)$ is a sort violation if $L[j] < L[i]$.

P2.1. Consider the list $[5, 2, 3, 1]$. List all sort violations in this list.

P2.2. Show that in the worst case, a list has $\frac{|L| \cdot (|L|-1)}{2}$ sort violations.

P2.3. Provide an algorithm that efficiently computes the number of sort violations in a list $L$ with a worst-case running time of $O(|L| \log_2(|L|))$.

**Hint:** How can you compute the number of sort violations for $[5, 2, 3, 1]$ by combining the number of sort violations in $[5, 2]$ and in $[3, 1]$?

## Assignment Details

Write a report in which you solve each of the above problems. Your submission:

1. must start with your name, student number, and MacID;
2. must be a PDF file;
3. must have clearly labeled solutions to each of the stated problems;
4. must be clearly presented;
5. must *not* be hand-written: prepare your report in LaTeX or in a word processor such as Microsoft Word (that can print or export to PDF).

**Submissions that do not follow the above requirements will get a grade of zero.**

## Grading

Each problem counts equally toward the final grade of this assignment.