Assignment 1

SFWRENG 2CO3: Data Structures and Algorithms–Winter 2024

Deadline: January 21, 2024

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.

P1.1. Consider the following functions of n:

$$\begin{array}{ccccccc} n^2 & \sum_{i=0}^{n} 5 \cdot i & n^3 \cdot \sqrt{\frac{1}{n^3}} & n^2 + 2^n & \left(\prod_{i=1}^{9} i\right) & \left(\sum_{i=0}^{\log_2(n)} 2^i\right) + 1 & 7^{\ln(n)} \\ -\ln\left(\frac{1}{n}\right) & \ln(2^n) & 10 & n \log_2(n^7) & \sqrt{n^4} & n^n & 5n \end{array}$$

Group the above functions that have identical growth rate and order these groups on increasing growth. Hence,

(a) if you place functions $f_1(n)$ and $f_2(n)$ in the same group, then we must have $f_1(n) = \Theta(f_2(n))$;

(b) if you place function $f_1(n)$ in a group ordered before the group in which you place function $f_2(n)$, then we must have $f_1(n) = O(f_2(n))$ but *not* $f_1(n) = \Omega(f_2(n))$.

Explain your answers.

Hint: You do not have to explain any of the well-known identities of Chapter 2 of the course notes or any of the comparisons of the order-of-growth of two functions that is already explained in Example 3.25 of the course notes (if you use results from Example 3.25, then do make sure that it is clear which result you are using and why you can use that result).

P1.2. Consider the recurrence

$$T(n) = \begin{cases} 7 & \text{if } n \leq 1; \\ 3T(n-2) & \text{if } n > 1. \end{cases}$$

Use induction to prove that $T(n) = f(n)$ with $f(n) = 7 \cdot 3^{\frac{n}{2}}$.

Problem 2. Consider the following COUNT algorithm.

Algorithm **COUNT**(L, v) **:**

Pre: L is an *array*, v a value.

1: $i, c := 0, 0$.
2: **while** $i \neq |L|$ **do**
3:     **if** $L[i] = v$ **then**
4:         $c := c + 1$.
5:     **end if**
6:     $i := i + 1$.
7: **end while**
8: **return** c.

Post: return the number of copies of v in L.

P2.1. Provide an invariant for the while loop at Line 2 of the COUNT algorithm..

P2.2. Provide a bound function for the while loop at Line 2 of the COUNT algorithm.

P2.3. Prove that the COUNT algorithm is correct. Use the invariant and bound function of your answers to P2.1 and P2.2.

P2.4. What is the runtime complexity of COUNT? What is the message complexity of COUNT?

P2.5. Provide an algorithm FASTCOUNT(L, v) that operates on *ordered lists* L and computes the same result as COUNT(L, v) with a running time of $\mathcal{O}(\log_2(|L|))$.

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 exported 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.