Computer Science CSC373H
St. George Campus

September 14, 2022
University of Toronto

Homework Assignment #1

Due: September 21, 2022, by 11:00 am

- ***You must submit your assignment through the Crowdmark system.*** You will receive by email an invitation through which you can submit your work. If you haven't used Crowdmark before, give yourself plenty of time to figure it out!
- You must submit a ***separate*** PDF document with for **each** question of the assignment.
- To work with one or two partners, you and your partner(s) must form a ***group*** on Crowdmark (one submission only per group). We allow groups of ***up to three*** students. Submissions by groups of more than three students will not be graded.
- The PDF file that you submit for each question must be typeset (***not*** handwritten) and clearly legible. To this end, we encourage you to learn and use the LaTeX typesetting system, which is designed to produce high-quality documents that contain mathematical notation. You can use other typesetting systems if you prefer, but handwritten documents are not accepted.
- If this assignment is submitted by a group of two or three students, for each assignment question the PDF file that you submit should contain:
  1. The name(s) of the student(s) who ***wrote*** the solution to this question, and
  2. The name(s) of the student(s) who ***read*** this solution to verify its clarity and correctness.
- By virtue of submitting this assignment you (and your partners, if you have any) acknowledge that you are aware of the homework collaboration policy for this course, as stated here.
- For any question, you may use data structures and algorithms previously described in class, or in prerequisites of this course, without describing them. You may also use any result that we covered in class (in lectures or tutorials) by referring to it.
- Unless we explicitly state otherwise, you should justify your answers. Your paper will be marked based on the correctness and efficiency of your answers, and the clarity, precision, and conciseness of your presentation.
- The total length of your pdf submission should be no more than 4 pages long in a 11pt font.

Question 1. (10 marks)

For each $k \in \mathbb{N}$, we recursively define the ***DeSica*** matrix M_k to be the following $2^k \times 2^k$ matrix:

- For $k = 0$: $M_0 = [1]$.
- For $k > 0$: $M_k = \left[\begin{array}{c|c} M_{k-1} & M_{k-1} \\ \hline M_{k-1} & -M_{k-1} \end{array}\right]$

a. Write the matrix M_2.

b. Give a divide-and-conquer algorithm DS-MULT($\vec{v}$) that, given a column vector $\vec{v}$ of length $n = 2^k$, computes the product $M_k\,\vec{v}$ in $O(n \log n)$ time in the worst-case, assuming each arithmetic operation (addition, subtraction, multiplication, or division) takes $O(1)$ time in the worst-case.

NOTE: a column vector $\vec{v}$ of length $n = 2^k$ is a matrix of dimension $2^k \times 1$. The product $M_k\,\vec{v}$ is also a column vector $\vec{u}$ of length $n = 2^k$.

Explain how DS-MULT($\vec{v}$) computes $M_k\,\vec{v}$ briefly in English and give its pseudocode.

c. Use the Master Theorem to justify why your DS-MULT($\vec{v}$) runs in $O(n \log n)$ time in the worst-case.

Question 2. (20 marks) Claudia Cardinale trades Bitcoins for n days. For each day i, $1 \leq i \leq n$, she records her net gain or loss in slot i of an array $G[1..n]$: for example, $G[4] = 1200$ indicates that she gained 1200 CAD on day 4, $G[5] = -325$ indicates that she lost 325 CAD on day 5, and $G[6] = 0$ means she did not gain or lose any money on day 6.

Claudia wants to determine what was her *worst total loss* over any *continuous* period of days during her n days of trade; that is, she wants to find the *total amount* that she lost over any period of *contiguous* days $[i, i+1, \ldots, j]$ *such that this amount is the worst loss among all contiguous periods of days during the n days that she traded.*

For simplicity, assume that: (a) there is at least one day when Claudia had a gain, and at least one day when Claudia had a loss.[1], and (b) $n \geq 2$ and it is an exact power of 2.

a. Formulate Claudia's problem in a simple and precise mathematical way with a concise notation.

b. Give a simple divide-and-conquer algorithm that solves this problem in $O(n \log n)$ worst-case time.

c. Give a divide-and-conquer algorithm that solves this problem in $O(n)$ worst-case time.

HINT: You may be able to achieve an $O(n)$ algorithm by improving the efficiency of the "combine" part of your $O(n \log n)$ algorithm as follows: make the recursive call to each subproblem return more information than just the optimal solution of the subproblem.

NOTE: For parts **(b)** and **(c)** above, first describe your algorithm *clearly and concisely* in English and then give its pseudo-code. You should also give the recurrence relation for the worst-case running time, and use the Master Theorem to justify the algorithm's asymptotic time complexity.

[1] This assumption is not necessary, it just simplifies the exposition of the problem here.