Assignment 8 - Loops and Recurrences

CS 234

due April 14th, 11:59pm

0 Introduction

This assignment is to be completed individually, but feel free to collaborate according to the course's external collaboration policy (which can be found in the syllabus). *Generative AI usage must follow course guidelines to be eligible for points.*

The deliverables consist of one `.pdf` file. The deliverables should be submitted electronically to by the deadline. Put any attribution text in the `.pdf` file. You may also consider adding an experience report to the `.pdf` describing your experience with the assignment: how long did it take, how hard/fulfilling was it, etc.

Your `.pdf` file should be named like `FLast_cs234_aX.ext` where `F` is your first initial, `Last` is your last name, `X` is the assignment number, and `ext` is the appropriate file extension. For example, Éva Tardos's `.pdf` file should be given the name `ETardos_cs234_a8.pdf`. (Éva Tardos is an award-winning algorithms researcher. She taught my algorithms course when I was an undergrad!)

1 The Only Part – Proofs on Paper

Please complete the following exercises from the textbook in your `.pdf` submission. Clearly label your responses with the exercise number.

- F.1 (prove your function returns the maximum element given a non-empty list of integers as input)
- F.3 (prove your function returns $\frac{n(n+1)}{2}$ given input $n \in \mathbb{N}$)
- F.5 (prove your function returns $n!$ given input $n \in \mathbb{N}$)[1]
- F.6 (prove your function returns F_n given input $n \in \mathbb{N}$ where

$$F_n = \begin{cases} n & n \leq 1 \\ F_{n-1} + F_{n-2} & n > 1 \end{cases}$$

)
- F.9 (prove your function mutates the input list to be reversed)[2]
- F.12 (prove your function returns a Boolean indicating whether the input integer is in the ascendingly sorted input list of integers)[3]

These tasks ask you to supply code. Code can be expressed in pseudocode, sreenshots, or similar. For this assignment, each piece of code must accomplish its goal using some form of while-loop or for-loop in a meaningful way. Make sure you actually implement what the textbook asks you to, and don't forget to supply the proof of the property written above too.

[1] Recall the definition of $n!$.

[2] To be "in-place" as the textbook requires, you cannot create any new lists in this function, nor can you extend/shrink the given input list.

[3] Make sure you actually implement binary search and not some other search.