

Algorithm Design and Analysis

CS 376, Spring 2023

Smaller Assignment 3

Due **BY THE BEGINNING OF CLASS** Wednesday, February 15

- For this Smaller Assignment, the standard conventions apply: Please submit your typewritten answers in a PDF file named

CS376_SA3_<userid>.pdf

where <userid> is replaced by your Colby userid. **Please be sure to give your full userid, including the numbers representing your class year.**

To submit this assignment, place the properly named file in your `Private` directory in your CS376 filespace on `filer`. **Please make sure all Problem Set and Smaller Assignment submissions are to your Private folders!**

- *A general note for CS376:* When writing up your homework, **explain your answers clearly and thoroughly**, giving all details needed to make your answers easy to understand. Clear communication *is* the point, on every assignment.

Unless explicitly specified otherwise, answers in CS376 should be accompanied by explanations; as part of this, in general, when you present an algorithm, give an accompanying correctness argument. Answers without convincing explanations that show command of the relevant concepts may not receive full credit. Please feel free to ask me any questions about explanations that might come up!

Exercises

Note: These exercises are asking you to redo (all or parts of) exercises on SA1, focusing on the proofs of correctness. Please be sure to follow guidelines presented in class for proving recursive algorithms correct: refer to details of the algorithms and specifications; inductive proofs should follow the inductive structure of the algorithms; include termination arguments; etc.

1. Prove that our recursive Sum algorithm from last week's lecture returns $\frac{n(n+1)}{2}$ on all inputs n that meet its input specifications.
2. Using our *IntBinTree* data structure (*IBT*, for short) from our lecture notes, write a recursive algorithm that takes as input an *IBT* T known to contain no negative values, and return the maximum value in T .

Input: IntBinTree T known to contain no negative values
Output: The maximum value in T

As usual, explain your algorithm and give a proof of correctness.