

Analysis of Algorithms
CS 375, Fall 2019

Homework 2

Due **AT THE BEGINNING OF CLASS** Monday, September 16

- In general, there may be multiple correct ways of presenting an algorithm, although excessively inefficient or inelegant solutions may not receive full credit. If you have questions about whether your proposed solution is excessively inefficient or inelegant, please ask your Prof.!
- *A general note:* When writing up your homework, please write neatly and **explain your answers clearly**, giving all details needed to make your answers easy to understand. Graders may not award credit to incomplete or illegible solutions. Clear communication *is* the point, on every assignment.

Exercises

1. Using our *IntBinTree* data structure from class (*IBT*, for short), come up with a recursive algorithm that returns the number of nodes in a tree.

Input: IntBinTree T
Output: The number of nodes in tree T (0 if it's empty)

Give an inductive explanation of the algorithm's correctness.

2. Using our *IBT* data structure, come up with a recursive algorithm that returns the sum of the elements in a tree.

Input: IntBinTree T
Output: The sum of all of the integers in tree T

What did you decide that the algorithm should return on an empty tree as input? Explain your reasoning for that decision (a sentence or so could be sufficient), and give an inductive explanation of the algorithm's correctness.

3. (Yet another classic problem solving puzzle. No bridge this time!) There are n hikers who need to cross a river. The river is too wide and deep to cross on foot, and there's no bridge in sight. As is common in these kinds of puzzles, however, there's a twist!

They notice two boys playing in a small rowboat by the river bank. The boat is so tiny, however, that it can only hold two boys or one hiker. How can all n of the hikers get across the river and leave the boys in joint possession of the boat? In your solution, how many times does the boat need to pass from one side of the river to the other?