Vanderbilt University – College of Engineering
CS 2212, SPRING 2023 - Sections 01/02/03/04
Homework Assignment #3B (maximum 30 points)

Name: Shiva Senthilkumar

Section: 02/Arena

Honor Statement:
By submitting this homework under your personal Gradescope account, you attest that you have neither given nor received unauthorized aid concerning this homework. You further acknowledge the instructors are the copyright owners of this HW. Any posting/uploading of HW questions for distribution (e.g., GitHub, Chegg) will be considered an honor code violation (even after finishing this class) and submitted to the honor council.

HW Philosophy:
The objective of HW in CS2212 is to expand on lecture topics and examine them in more detail and depth. For this reason, we typically give a couple of weeks to complete the HW. Keep in mind that this is not supposed to be an easy task. Expect HWs to take time, require thinking, and challenge you. The TAs and instructors are not here to provide you with an answer or tell you whether your answer is correct while you are working on the HW. We are here to help you think more critically and at a higher level than when you started this class.

Clear, Correct, and Concise:
One of the critical aspects of this course is learning to communicate your ideas effectively. This skill is not only necessary for this course (or CS in general) but also a skill that employers highly value. Answers on all HWs, exams, and other activities in this class are graded on correctness, clarity, and conciseness. Keeping this in mind, here are a couple of pieces of advice:

- **Know Your Audience** - For this class, you should imagine your audience are your fellow students. Keeping this in mind ensures you provide enough details to prove to the grader that you understand your solution while keeping your answer concise enough to avoid unnecessary information.
- **Reread, Rewrite and Refine** - As with any good piece of writing, a well-written proof or answer often goes through several revisions. Reread, rewrite, refine, and take the time to get it right.

You must type your answers (or you will lose 50% on the exercise). Each exercise should begin on a new page (you can put multiple parts to the same exercises on the same page). Save your file as a PDF and upload the PDF document in an electronic format to Gradescope (https://www.gradescope.com/).

You can use different colors to markup your solution but **avoid using a red font.** When submitting your work, designate the corresponding page(s) of your submission for the appropriate question (or risk losing 5 points). See the following video at the 0:46 mark: https://www.gradescope.com/get_started#student-submission

Questions:
Ask your instructor before/after class or your instructor or TAs during office hours.

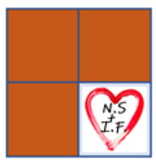
Due Date:
Your completed homework, including all the questions and answers, must be uploaded as a PDF to Gradescope no later than **Wednesday, April 5th, @ 9 AM CT (Nashville Time).**

Exercise 1: Inductive Reasoning [15 points]

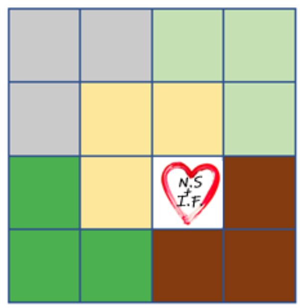
Skills: inductive reasoning, inductive proofs

A. Living Room Floor Troubles [6 points]. Nadha Skolar wishes to tile the 32x32 Living Room floor of his condo. Since he is a superfan of the game tetris, he wants to tile the floor using only tetris shapes. Hoping to convince his girlfriend to support his design idea, he creates a sample out of foam blocks (see image to the right).

Unfortunately, Nadha's girlfriend hates Nadha's idea and threatens to break up with him if he tiles the floor this way. They attempt to strike a compromise whereby Nadha agrees to only use the L-shaped puzzle piece provided he agrees to allow his girlfriend one single tile in the floor design where she can paint a heart containing their initials (see examples below). It turns out that as long as Nadha's floor is 2^nx2^n (n>=1), Nadha is guaranteed to have exactly one single tile left over for his girlfriend to paint a heart with their initials. Let's prove it.



(a) n =1



(b) n =2

Prove by induction that as long as Nadha's room is 2^nx2^n (n>=1), it is always possible to have exactly one single tile leftover for Nadha's girlfriend to paint a heart without ruining Nadha's design.

Proof by Induction: We will be using a standard induction proof technique to prove that as long as Nadha's room is P(n) = 2^nx2^n (n>=1), it is always possible to have exactly one single tile leftover for Nadha's girlfriend to paint a heart without ruining Nadha's design.

Base Case: n = 1, P(1) = $2^1 * 2^1$ = 4 is true since each L-shaped tile takes up three spots, this means that there is one remaining tile that Nadha's girlfriend can paint a heart over.

Inductive Hypothesis: Assume our base case holds true for k = n, where n>=1. We will show that if this is the true, then P(k+1) is also true.

Inductive Step: $P(k+1) = 2^{(k+1)} * 2^{(k+1)} = 2 * 2^k * 2 * 2^k = 4 * \ 2^k * 2^k = 4 * P(k)$. The result of $P(k+1)$ is four quadrants of size $P(k)$ – we are taking a larger grid and breaking it down into 4 smaller grids of equal size. Since we know from our inductive hypothesis that each P(k) has an empty spot, when four of these are combined together, this signifies that there is a total of four empty spots (i.e. each smaller grid has one empty spot and adding all four of them together gives us 4 empty spots).

Conclusion: Due to this, three of the empty spots will be filled with a L-shaped puzzle piece, and the last leftover empty spot will be for Nadha's girlfriend to paint a heart on it. QED.

B. Let's Play The Margarita Game [9 points]. There is a trendy resort on a small island in Mexico known as Isla Mujeres. Upon arriving, guests (18 or older) are handed an 8x8 map of the resort to play the hotel's famous "Margarita game." A colorful loop adorns the resort map containing n pictures of margaritas and n pictures of the Mexican flag. The game goes as follows. The guest may pick any place to start on the map and traces a clockwise path around the colorful loop. The watchful employee counts how many Mexican flags the guest has passed versus how many pictures of margaritas the guest has passed. During the trace, the house rules state that the number of margaritas passed must be at least the same as the number of Mexican flags passed (at all times). If the guest successfully finishes tracing the loop according to the house rules, they win a complimentary margarita! If not, they receive a complimentary lemonade.

Write a well-structured inductive proof to demonstrate that no matter how the margaritas and flags are arranged on the colorful loop adorning the map, it is always possible for the guest to win a complimentary margarita.

Proof by Induction: We will be using a standard induction proof technique to prove that no matter how the margaritas and flags are arranged on the colorful loop adorning the map, it is always possible for the guest to win a complimentary margarita.

Base Case: When n = 1, there is 1 picture of a margarita and 1 picture of the Mexican flag. There is only one orientation in which the pictures can be placed when reduced by symmetry. Furthermore, since you can pick any place to start on the map, you will start where the picture of the margarita is, to assure that the # of margarita pictures ≥ # of pictures of Mexican flags when going in a clockwise direction around the loop. P(n) is an arrangement where there are n pictures of margaritas and n pictures of mexican flags.

Inductive Hypothesis: Assume our base case holds true for k = n, where n>=1. We will show that if this is true, then P(k+1) is also true through the "build-down, build-up" method.

Inductive Step: Starting with P(k+1), let's build down to P(k). To assure that the house rules are maintained, a consecutive pair of pictures, with a margarita followed by a mexican flag, should be removed from the P(k+1). By removing one picture of each, we are back down to our P(k) case, which we know we can win. Since we have proven that P(k) is true based on our base case and inductive hypothesis, if we put these consecutive pair of pictures, with a margarita followed by a mexican flag, back into the location that the pictures were removed, this path that traverses through P(k+1) holds true.

Conclusion: Due to this, no matter how the margaritas and flags are arranged on the colorful loop adorning the map, it is always possible for the guest to win a complimentary margarita. QED.

Exercise 2: NadhaSort [15 points]

Skills: algorithm and asymptotic analysis, sequences, proof techniques, recurrence relations.

There are (3) parts to this problem A, B, C.

Our favorite computer programmer, Nadha Skolar, picked up a new hobby during the pandemic – making mixed drinks. While watching a TikTok video about making the best Cosmopolitan, Nadha has a clever idea for a new sort he dubbed, **NadhaSort**. The sort is similar to bubble sort, but "bubbles" in both directions of the array. Look at the example below. NadhaSort first bubbles left to right below. Then it reverses direction and bubbles right to left. The sorting continues in this fashion until no swaps are made during a traversal (either left or right). Here's an illustration of what a single pass of NadhaSort idea looks like:

Original list:

43	18	10	23	7

Pass #1:

After traversing through the list left-to-right, notice the value 43 is in the correct spot:

18	10	23	7	43

After traversing back through the list (right-to-left), notice the value 7 is in the correct spot:

7	18	10	23	43

A. **[3 points]** Nadha excitedly informs the Skolar family at dinner that he plans to submit a paper to the ACM Journal proving beyond any doubt his algorithm, NadhaSort is $\Omega(1)$ in both the best and the worst-case scenarios. Nadha says this realization will make him as famous as the developers of ChatGpt and remove any doubt about his ability to perform at a high-level as a research scholar. Is Nadha correct about the analysis of his algorithm? Is Nadha's result like to make him as famous as the developers of ChatGPT? Explain briefly.

Nadha is stating that the asymptotic lower bound for the best and worst-case scenarios is 1 – meaning that in both cases, NadhaSort has to do at least one operation when called upon. Nadha is correct about the analysis of his algorithm, but it won't make him as famous as the developers of ChatGPT because every algorithm has to do at least operation, otherwise it is not an algorithm. Nadha should use a tighter lower bound for the best case scenario. In the best case scenario, all n elements are in order, and the algorithm has to do only n-1 comparisons since no elements are being swapped. The lower bound for the best case scenario will be $\Omega(n)$. The lower bound for the worst-case scenario is explained in Part B.

B. **[6 points]** Prove the worst-case running time of NadhaSort is $\theta(n^2)$ by demonstrating both the big-oh and the omega for the worst-case of NadhaSort is n^2.

Proof: We can prove by **direct proof** that NadhaSort is $\theta(n^2)$ by demonstrating that O and Ω for the worst-case of NadhaSort is n^2

Determine the Worse-Case Scenario for NadhaSort:

- During the first traversal of n elements, there will be n-1 swaps to put the biggest element in the last position
- During the second traversal of n elements, there will be n-2 swaps to put the smallest element in the first position
- This pattern will keep occurring, and this provides us with a summation of (n-1) + (n-2) + (n-3) + … + n-(n-3) + n-(n-2) + n-(n-1) + n-(n-0).
- This can be expressed as $\sum_1^{n-1} i$
- The summation above equals $\frac{n(n-1)}{2}$

Big-Oh for the Worst-Case Scenario for NadhaSort:

- Big-Oh for the Worst-Case Scenario is n^2 because the leading term in the expression $\frac{n(n-1)}{2}$ is n^2

Omega for the Worst-Case Scenario for NadhaSort:

- Omega for the Worst-Case Scenario is n^2 because the leading term in the expression $\frac{n(n-1)}{2}$ is n^2

Conclusion: Since both Big-Oh and Omega are the same for the worst-case scenario, this means that the theta bound must also equal n^2. QED.

C. [6 points – 2, 4] Nadha's little sister, 5-year old Stellar Skolar, designs a divide-and-conquer algorithm that solves problems of size n by recursively solving one subproblem of n/2 and performing linear processing time of cn to divide/combine subproblems. The base case for Stellar's algorithm is T(1) = 1.

i. **[2 points] List the recurrence.** Write the recurrence relation for T(n), the running time on an instance of size n.

T(n) = T(n/2) + cn

ii. **[4 points] Solve your recurrence** from (i) above using either the substitution method or the cancellation method and provide an asymptotic bound for the amount of work done for Stella's algorithm. **Tip:** You may assume n is a power of 2 if it helps make your analysis easier.

$T(1) = 1$

$$T(n) = T\left(\frac{n}{2}\right) + cn$$
$$T\left(\frac{n}{2}\right) = T\left(\frac{n}{4}\right) + c\left(\frac{n}{2}\right)$$
$$T\left(\frac{n}{4}\right) = T\left(\frac{n}{8}\right) + c\left(\frac{n}{4}\right)$$

Substitute

$T(n) = T\left(\frac{n}{2}\right) + cn$

$T(n) = T\left(\frac{n}{4}\right) + c\left(\frac{n}{2}\right) + cn$

$T(n) = T\left(\frac{n}{8}\right) + c\left(\frac{n}{4}\right) + c\left(\frac{n}{2}\right) + cn$

General Formula

$T(n) = T\left(\frac{n}{2^i}\right) + \frac{cn}{2^{i-1}} + \frac{cn}{2^{i-2}} + \cdots + cn$

Summation

Assume that $k = log_2 n$

$$(Step\ 1) \sum_{i=0}^{k} \frac{cn}{2^i} \rightarrow (Step\ 2)\ cn * \sum_{i=0}^{k} \frac{1}{2^i} \rightarrow (Step\ 3)\ 2 * cn$$

To go from the first step to the second step, you need to pull out *cn* since it is a constant.
To go from the second step to the third step, you are left with *cn* multiplying a summation of a geometric series that converges to 2.
You are left with *2cn*. Since T(1) = 1, you need to add 1 to the result of the summation.
The total therefore is ***2cn + 1***.

The asymptotic bound for the amount of work done for Stella's algorithm is **O(n)** since the highest degree term in the expression – *2cn + 1* – is n.