CIS 1210 — Data Structures and Algorithms
Homework Assignment 2

Assigned: September 10, 2024 **Due:** September 16, 2024

Note: The homework is due **electronically on Gradescope** on September 16, 2024 by 11:59 pm ET. For late submissions, please refer to the Late Submission Policy on the course webpage. You may submit this assignment up to 2 days late.

A. **Gradescope**: You must select the appropriate pages on Gradescope. Gradescope makes this easy for you: before you submit, it asks you to associate pages with the homework questions. Forgetting to do so will incur a 5% penalty, which cannot be argued against after the fact.

B. **LaTeX**: You must use the LaTex template provided on the course website, or a 5% penalty will be incurred. Handwritten solutions or solutions not typeset in LaTex will not be accepted.

C. **Solutions**: Please write concise and clear solutions; you will get only a partial credit for correct solutions that are either unnecessarily long or not clear. Please refer to the Written Homework Guidelines for all the requirements.

D. **Algorithms**: Whenever you present an algorithm, your answer must include 3 separate sections.

1. A precise description of your algorithm in English. No pseudocode, no code.
2. Proof of correctness of your algorithm
3. Analysis of the running time complexity of your algorithm

E. **Collaboration**: You are allowed to discuss **ideas** for solving homework problems in groups of up to 3 people but *you must write your solutions independently.* Also, you must write on your homework the names of the people with whom you discussed. For more on the collaboration policy, please see the course webpage.

F. **Outside Resources**: Finally, you are not allowed to use *any* material outside of the class notes and the textbook. Any violation of this policy may seriously affect your grade in the class. If you're unsure if something violates our policy, please ask. If you would like to cite from CLRS in your proofs you must first ask for permission on ED.

1. [10 pts] 40 Crewmates, 1 Imposter

The 1210 Staff suddenly becomes a part of a real-life game of Among Us and must find the imposter. Working hard to win the game, Tommy goes to Weapons to clear the asteroids. However, two asteroids are quickly approaching and must be destroyed to save the ship! The first asteroid's acceleration speed can be modeled by a function $f(n)$, and the second asteroid's acceleration speed can be modeled by $g(n)$. When analyzing $f(n)$ and $g(n)$, which are asymptotically non-negative functions, Tommy must figure out if the relationship

$$\max\{f(n),\ g(n)\} = \Theta(f(n) + g(n))$$

always holds. Help Tommy clear the asteroids by either proving or disproving the relationship!

Note: $\max\{f(n),\ g(n)\}$ refers to the maximum value at any given point in time, meaning that for some given value of $n = i$, we take $\max\{f(i),\ g(i)\}$. Additionally, an asymptotically non-negative function $f(n)$ implies that there is some positive n_0 such that for all $n \geq n_0$, we have $f(n) \geq 0$.

2. [15 pts] Finding the Imposter Among Us

An emergency meeting has been called and the CIS 1210 TA's are in the middle of a heated argument on who the imposter is. A majority of the TA's think that Jesse is the traitor because of how sus he has been recently. However, Darsh feels pity for Jesse so he has offered Jesse the chance to clear his name if he can correctly prove or disprove the following three statements.

In case of a proof, use the definitions of O, Ω, Θ and give values of the constants in the definitions for which the conditions in the definition hold. You are also given three functions f, g, and h such that $f(n) = O(g(n))$ and $g(n) = O(h(n))$.

a. $\sqrt{f(n)} = O(\sqrt{g(n)})$

b. $\lg f(n) = O(\lg g(n))$

c. $h(n) = \Omega(f(n)))$

3. [20 pts] Clutch Chris Saves the Day!

Like the diligent Crewmate he is, Chris heads to the Shields room on The Skeld ship to conduct a routine check. Once there, he realizes that three different shields are severely damaged. Chris can repair one shield for each recurrence he solves. Help him maximize his chances of protecting the ship by solving the following recurrences **using the method of expansion**. Give your answer in Θ notation.

a. $T(n) = 3T(\frac{n}{3}) + 6n$. You may assume that $T(n) = 1$ for all $n \leq 3$.

b. $T(n) = T(n-1) + 5\text{lg}n$. You may assume that $T(1) = 1$.

c. $T(n) = \sqrt{n}T(\sqrt{n}) + 4n$. You may assume that $T(n) = 1$ for all $n \leq 16$.

4. [15 pts] Selina's Sus-Score Sorting Scenario

A meeting has just been called, and Selina has some difficult decisions to make. There are n other crewmates, and Selina thinks they are all super sus. In order to organize her thoughts, she assigns each crewmate a sus-score, and now needs a fast way to sort the crewmates based on how sus they currently are. She puts the n crewmate sus-scores into an array and applies her favorite sorting algorithm: $fiveSort$. $fiveSort(A)$ takes as input an array A of integers. The algorithm works by calling $five(0, n, A)$ where

n is the length of A and where $five(lo, hi, A)$ is a recursive algorithm. The arguments lo and hi of $five$ delimit the portion of the array A that $five$ sorts, namely $A[lo], A[lo+1], \ldots, A[hi-1]$.

The recursive function $five(lo, hi, A)$ works as follows:

1. If $hi - lo$ is 0 or 1, return. Otherwise go to the next step.
2. If $hi - lo$ is 2, 3, or 4, put the elements between and including $A[lo]$ and $A[hi-1]$ in order (using *insertion-sort*) then return. Otherwise go to the next step.
3. Divide the array portion between lo and hi into five (approximately) equal parts. Call *insertion-sort* to order the middle fifth, then recursively call $five$ on the first, second, fourth, and fifth quin-tiles of the (sub)array.
4. Merge all five sorted parts.

However, Selina does not have a lot of time. Voting is commencing soon! She needs your help determining how long her $fiveSort$ algorithm will take.

Let $T(n)$ be the worst-case running time for $fiveSort$ on an array of length n (assume that n is an exact power of 5). Write a recurrence relation for $T(n)$. Only include the term of highest degree for terms of the form cn^p where $p \geq 0$. Explain your answer briefly.

Analyze the running time of the algorithm. Prove your answer.

5. [20 pts] A Mysterious Trial

Players on The Airship are suddenly bombarded with sabotages! The Crewmates decide to hold a trial in which Angie has been thrust into the spotlight as a likely Imposter. To prove her innocence, she has been given the following task: for each of the code fragments given, provide a bound of the form $\Theta(f(n))$ on its running time on an input size n. You may assume that the print function takes constant time. Remember to show your work and justify your answer. Assume that the code snippets below use integer division.

(a)

```
for (i = 1; i ≤ 6n; i = i + 1) do
    for (j = 1; j ≤ i/2; j = j + 1) do
        print ("Three tasks away from freedom!")
```

(b)

```
for (i = 1; i ≤ n; i = i * 2) do
    for (j = i; j ≤ 9 * i + 7; j = j + 4) do
        print ("Two tasks away from freedom!")
```

(c)

```
for (i = 2; i ≤ n; i = i + 4) do
    for (j = 1; j < i; j = j * 7) do
        print ("One task away from freedom!")
```

6. [20 pts] Katherine's Wire Fiasco

Katherine starts to think some of the crewmates may be onto her Imposter-ish behavior, and obtains wires of n different colors to practice simulating tasks in electrical. She sorts the wires by color, creating an n-length array A, where $A[i]$ denotes the number of wires of color i. Being the smart, algorithmic Imposter she is, Katherine implements the following method, which takes in A and outputs an n-by-n array B, where $B[i, j]$ equals the number of wires with colors between i and j. That is, for $i < j$, $B[i, j]$ contains the sum $A[i] + A[i + 1] + \cdots + A[j]$. Note that the value of $B[i, j]$ is left unspecified whenever $i \geq j$, so its value does not matter.

Hearing footsteps approaching her hiding spot, Katherine only has time to put together a rough outline of her method:

```
for (i = 1; i ≤ n; i++)
    for (j = i + 1; j ≤ n; j++)
        Add up array entries A[i] through A[j]
        Store the result in B[i, j]
```

a. For the above code fragment, give a bound of the form $\Theta(f(n))$ on its running time on an input of size n by giving both a $\mathcal{O}$ and Ω bound. That is, show that the runtime is both $O(f(n))$ and $\Omega(f(n))$ for some function $f(n)$. Justify your answer.

b. Although the code she came up with might be the most natural solution to this problem - after all, it just iterates through the relevant entries of the array B, filling in a value for each - Katherine knows if she wants fool the other crewmates, she'll have to find a more efficient solution. Give a different algorithm to solve this problem, with an asymptotically better running time than Katherine's. In other words, you should design an algorithm with running time $O(g(n))$, where $\lim_{n\to\infty} g(n)/f(n) = 0$.

   For this problem only, you may give us properly formatted pseudocode (as above) to supplement your English explanation of your algorithm. **Pseudocode alone will receive no credit.**