CS41 lab 1

In typical labs this semester, you'll be working on a number of problems in groups of 3-4 students each. You will not be handing in solutions. The primary purpose of these labs are to have a low-pressure space to discuss algorithm design, and to gain experience collaborating with others on algorithm design and analysis. However, it will be common to have some overlap between lab exercises and homework sets. There will typically be many more problems than you have time to complete during the lab. I encourage you to work on any problems your group wants, in any order.

1. First, get git set up. See the "set up git" help page linked off the course website.

2. Clone the **examples** repo from the **CS41-F24** github org.

   ```
   $ git clone git@github.swarthmore.edu:CS41-F22/examples.git
   ```

3. Using `LearningLaTeX.tex` as a guide, provide a writeup of the analysis of one of the Hiking Problem algorithms from class.

4. **Induction**

   Using induction, show that the following summations hold for all $n \geq 0$.

   - $\sum_{k=0}^{n} k = \frac{n(n+1)}{2}$.
   - $\sum_{k=0}^{n} 2^k = 2^{n+1} - 1$.
   - For all positive $c \neq 1$, $\sum_{k=0}^{n} c^k = \frac{c^{n+1} - 1}{c - 1}$.

5. **Logarithmic Properties**

   $\log_2(n)$ is the unique real number x such that $2^x = n$. Using direct proof, show that the following properties hold for all positive real numbers a, b.

   - $\log_2(ab) = \log_2(a) + \log_2(b)$.
   - $-\log_2(a) = \log_2(1/a)$.
   - $\log_2(a^b) = b \log_2(a)$.
   - $a^{\log_2(b)} = b^{\log_2(a)}$. *(really)*

6. **Analyzing Chat GPT responses.** Select an exercise from this week's lab problems, and get a "solution" using Chat-GPT. Is the solution correct? Answer YES or NO. Justify your response.

7. **Sorting to Half-Sorting**. In the HALF-SORT problem, you're given an array of n integers and must return an array that has the first $\lceil n/2 \rceil$ integers in sorted order. For example, if your array is $A = [5, 9, 1, 2, 6, 3]$, then a valid output of HALF-SORT(A) might be $[1, 2, 3, 9, 5, 6]$ since $1, 2, 3$ are the least elements of A.

(a) Reduce the sorting problem to HALF-SORT. i.e., imagine you have an algorithm $\mathcal{A}$ for HALF-SORT, and use it to design a sorting algorithm.

(b) Reduce HALF-SORT to the sorting problem. i.e., imagine you have a sorting algorithm $\mathcal{B}$, and use it to design an algorithm for HALF-SORT.

(c) Now, suppose that your friend claims to have an algorithm for HALF-SORT that runs in $10n$ time in the worst case. What is the runtime of your sorting algorithm? Is $10n$ a reasonable running time for HALF-SORT?