Problem 1. Prove $\log_2(n!) = \Theta(n \log n)$.

Problem 2. Let $f(n)$ be a function of positive integer $n$. We know:

\[
\begin{align*}
    f(1) & = 1 \\
    f(n) & = 2 + f(\lceil n/10 \rceil).
\end{align*}
\]

Prove $f(n) = O(\log n)$. Recall that $\lceil x \rceil$ is the ceiling operator that returns the smallest integer at least $x$.

If necessary, you can use without a proof the fact that $f(n)$ is monotone, namely, $f(n_1) \leq f(n_2)$ for any $n_1 < n_2$.

Problem 3. Let $f(n)$ be a function of positive integer $n$. We know:

\[
\begin{align*}
    f(1) & = 1 \\
    f(n) & = 2 + f(\lceil 3n/10 \rceil).
\end{align*}
\]

Prove $f(n) = O(\log n)$. Recall that $\lceil x \rceil$ is the ceiling operator that returns the smallest integer at least $x$.

Problem 4. Let $f(n)$ be a function of positive integer $n$. We know:

\[
\begin{align*}
    f(1) & = 1 \\
    f(n) & = 2n + 4f(\lceil n/4 \rceil).
\end{align*}
\]

Prove $f(n) = O(n \log n)$. If necessary, you can use without a proof the fact that $f(n)$ is monotone.

Problem 5 (Bubble Sort). Let us re-visit the sorting problem. Recall that, in this problem, we are given an array $A$ of $n$ integers, and need to re-arrange them in ascending order. Consider the following bubble sort algorithm:

1. If $n = 1$, nothing to sort; return.


Prove that the algorithm terminates in $O(n^2)$ time.

As an example, support that $A$ contains the sequence of integers $(10, 15, 8, 29, 13)$. After Step 2 has been executed once, array $A$ becomes $(10, 8, 15, 13, 29)$. 