Problem 1. Suppose that $A \leq_m B$ and $A \notin RE$ and $A \notin co-RE$.

a) What can you conclude about $B$?

b) Is it necessarily the case that $B \leq_m UHALT_{TM}$? Why?

Problem 2. Define the language

$$SS_{TM} = \{\langle M_1, M_2 \rangle \mid L(M_1) \subseteq L(M_2)\}.$$

Prove that $SS_{TM}$ is neither in $RE$ nor $co-RE$.

(Hint: Lab 1 has a useful fact)

Problem 3. Recall that we reduced $\overline{HALT_{TM}}$ to $UHALT_{TM}$ by constructing the utility Turing machine

\begin{algorithm}
\textbf{Algorithm 1: $U_{M,\omega}$} = On input $\nu$
\begin{itemize}
  \item Run $M(\omega)$ for $|\nu|$ steps
  \item Accept
\end{itemize}
\end{algorithm}

$U_{M,\omega}$ halts on everything if $M(\omega)$ loops forever and does not otherwise. From this, we concluded that $UHALT_{TM}$ is not in $RE$.

Show that $HALT_{TM} \leq_m FIN$.

Problem 4. Define the following language

$$INFIN = \{\langle M \rangle \mid |L(M)| = \infty\}.$$

Show that $INFIN$ is not in $RE$ or $co-RE$. 
**Problem 5.** The set of all computable functions is countable, since each corresponds to a Turing machine, so we can enumerate them as $f_1, f_2, f_3, \ldots$. However the function

$$f(n) = \min_{1 \leq i \leq n} f_i(n) - 1$$

is not computable despite the obvious seemingly reasonable algorithm to compute $f$ given below.

**Algorithm 2: $f(n)$**

1. Let $m = \infty$
2. for $i = 1$ to $n$
   - $m = \min(m, f_i(n))$
3. return $m - 1$

While we can give a diagonalization argument to show that $f$ is uncomputable, there is a more fundamental question to answer. What did we do in the construction of $f$ that is uncomputable? In other words, what hidden assumption did we make that is incorrect? Explain without proof why it was incorrect.