Define the set of all Turing machines to be $\mathcal{M}$. For a property $P : \mathcal{M} \rightarrow \{\text{true}, \text{false}\}$, define the language

$$L_P = \{\langle M \rangle \mid M \in \mathcal{M} \text{ and } P(M)\}.$$ 

Rice’s Theorem requires two facts to hold.

1. $P$ is nontrivial (i.e. $P$ is not always true or always false), and
2. $P$ is a property of the Turing machine’s language, that is for every pair of Turing machines $A$ and $B$ such that $L(A) = L(B)$, $P(A)$ if and only if $P(B)$ (i.e. $\langle A \rangle \in L_P$ if and only if $\langle B \rangle \in L_P$).

If both of these are true, then Rice’s Theorem tells us that $L_P \notin \text{DEC}$.

**Problem 1.** Prove that the following language is not decidable using Rice’s Theorem.

$$\text{FIN} = \{\langle M \rangle \mid |L(M)| < \infty\}$$

Here, $P(M) := |L(M)| < \infty$.

**Solution 1.** The Turing machine that accepts everything clearly does not belong to FIN, and the Turing machine that accepts nothing clearly does, so $P$ is nontrivial.

Now suppose that $A$ and $B$ are Turing machines such that $L(A) = L(B)$. Then if $\langle A \rangle \in \text{FIN}$, $|L(A)| < \infty$. But this implies that $|L(B)| < \infty$ since $L(A) = L(B)$, hence $|L(A)| = |L(B)|$. An identical argument applies in reverse, so $P$ is a property of the Turing machine’s language.

Thus by Rice’s Theorem, $\text{FIN} \notin \text{DEC}$. 

Recall that a language $A$ mapping reduces to a language $B$, written $A \leq_m B$, if there is a computable function $f$ such that for every $\omega \in \Sigma^*$, $\omega \in A$ if and only if $f(\omega) \in B$.

A mapping reduction transforms an input to $A$ into an input to $B$. In other words, $f$ makes $B$ do $A$’s job. In general, you can think of $A \leq_m B$ as meaning “$A$ is no harder than $B$” or “$B$ is at least as hard as $A$”. Put plainly, if you can “solve” $A$, then solving $B$ would solve $A$, so you can’t solve $B$ either.

This results in the following useful theorems when $A \leq_m B$.

- $B \in \text{RE} \implies A \in \text{RE}$
- $A \notin \text{RE} \implies B \notin \text{RE}$
Problem 2. In the chart below, the first two rows tell you if a language is in \( RE \) or \( co-RE \). Then in each cell below that, place an \( X \) if the row’s language mapping reduces to the column’s language. No proof is required.

<table>
<thead>
<tr>
<th></th>
<th>( A_{TM} )</th>
<th>( A_{TM} )</th>
<th>( HALT_{TM} )</th>
<th>( E_{TM} )</th>
<th>( ALL_{TM} )</th>
<th>( EQ_{TM} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( RE )</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( co-RE )</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( A_{TM} )</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \overline{A}_{TM} )</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( HALT_{TM} )</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( E_{TM} )</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( ALL_{TM} )</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>( EQ_{TM} )</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Solution 2.

* These two problems lie on the same level of the arithmetic hierarchy, so it is likely they reduce to each other, but the reduction \( EQ_{TM} \leq_m ALL_{TM} \) eludes me.

One lesson to take away from the previous problem is that like reduces to like (observe the \( RE \) rows and the \( co-RE \) rows). The other lesson is that there are problems that are simply so hard that they can solve every \( RE \) and \( co-RE \) problem (observe the \( EQ_{TM} \) column).

Problem 3.

a) Give a utility Turing machine \( U_{M,\omega} \) so that

\[
L(U_{M,\omega}) = \begin{cases} 
\Sigma^* & M(\omega) \text{ halts} \\
\emptyset & M(\omega) \text{ does not halt.} 
\end{cases}
\]
b) Show that $\overline{HALT_{TM}} \leq_m FIN$.

c) Prove that $FIN \not\in RE$.

**Solution 3.**

a)

Algorithm 1: $U_{M,\omega} = \text{On input } \nu$

- Run $M(\omega)$
- Accept

Clearly, $U_{M,\omega}$ accepts everything if $M(\omega)$ halts and loops forever on everything otherwise, so

$L(U_{M,\omega}) = \begin{cases} 
\Sigma^* & \text{if } M(\omega) \text{ halts} \\
\emptyset & \text{if } M(\omega) \text{ does not halt.} 
\end{cases}$

b) Define the function $f(\langle M, \omega \rangle) = \langle U_{M,\omega} \rangle$.

Then if $\langle M, \omega \rangle \in \overline{HALT_{TM}}$, then $M(\omega)$ does not halt. But then $L(f(\langle M, \omega \rangle)) = L(U_{M,\omega}) = \emptyset$, so $f(\langle M, \omega \rangle) \in FIN$.

On the other hand, if $\langle M, \omega \rangle \notin \overline{HALT_{TM}}$, then $M(\omega)$ halts. But then $L(f(\langle M, \omega \rangle)) = L(U_{M,\omega}) = \Sigma^*$, so $f(\langle M, \omega \rangle) \notin FIN$.

Thus $f$ is a mapping reduction from $\overline{HALT_{TM}}$ to $FIN$, so $\overline{HALT_{TM}} \leq_m FIN$. □

c) Since $\overline{HALT_{TM}} \leq_m FIN$ and $\overline{HALT_{TM}} \notin RE$, it follows that $FIN \notin RE$. □

**Problem 4.** Prove that if $A \in RE$ and $A \leq_M \overline{A}$, then $A \in DEC$.

**Solution 4.** Since $A \leq_m \overline{A}$, it follows that $\overline{A} \leq_m A$. But since $A \in RE$, it follows that $\overline{A} \in RE$. In other words, $A \in co-RE$. But if $A \in RE$ and $A \in co-RE$, then $A \in DEC$. □