Defn: Given two languages $A, B \subseteq \Sigma^*$, say that $A \overset{\text{many-one}}{\leq_m} B$ if there is a computable function $f : \Sigma^* \rightarrow \Sigma^*$ such that for all $x \in \Sigma^*$, $x \in A \iff f(x) \in B$. (*)

Proposition: $A \leq_m B$ if and only if $\bar{A} \leq_m B$.

because $x \in A \iff f(x) \in B$ if (*) holds.

Proposition: If $A \leq_m B$ via $f$ and $B \leq_m C$ via a function $g$, then $A \leq_m C$ via the function $g \circ f$.

Lemma: Suppose $A \leq_m B$. Then:

(a) If $B$ is decidable, then so is $A$.
(b) If $B$ is recognizable, then so is $A$.
(c) If $B$ is co-c.e. then so is $A$.

Proof: (a) By $B$ being decidable, we can take a total TM $M_B$ s.t. $L(M_B) = B$. Goal: build a total TM $M_A$ such that $L(M_A) = A$.

Then $M_A$ is total and $M_A$ accepts $x \iff M_B$ accepts $f(x)$. So $M_A$ accepts $x \iff x \in A$ and $M_A$ is total, so $A$ is decidable.
6) Suppose $B$ is merely c.e. Then we can take $M_B$ st. $L(M_B) = B$, but $M_B$ might not be total. Can diagram using "fuzzy box". Then $M_A$ might not be total either, but we still have $M_A \text{ accept } x \Rightarrow M_B \text{ accept } y \Rightarrow \text{yes } \Rightarrow x \in A$. So $L(M_A) = A$, so $A$ is recognizable/c.e./re./etc.

5) Suppose $B$ is c-o-c.e. This means $\overline{B}$ is c.e. By $A \leq_m B$, we also have $A \leq_m \overline{B}$. By part (b), $\overline{B}$ is c-e. Thus $A$ is co-c.e.

**Theorem:** If $A \leq_m B$ where $B$ is c.e. and $A \leq_m C$ where $C$ is co-c.e. then $A$ is decidable.

**Corollary:** If $A$ is c-e. and co-c-e. then $A$ is decidable.

**Proof of this corollary:** Take $B = A$, $\overline{B} = A$ and use identity reductions.

**Proof:** Take TMs $M_B$ st. $L(M_B) = B$ and $M_C$ st. $L(M_C) = \overline{C}$, and take functions $f, g$ st. $A \leq_m B$ via $f$ and $A \leq_m C$ via $g$. Build a TM $M_A$ that does step-by-step simulations of $M_B$ and $M_C$ in tandem. Then $M_A \leq_m B$ because $x \in A \Rightarrow \exists y \in B \Rightarrow M_B$ will eventually accept $y$, while $x \in \overline{A} \Rightarrow \exists \overline{C} \Rightarrow M_C$ will eventually accept $z$. And $L(M_A) = A$. 

\[\text{Do the next step of } M_B \text{ (if yes)} \]
\[\text{Did it accept } y \text{ in that step?} \]
\[\text{Do 1 more step of } M_C \text{ (if possible).} \]
Examples of Reductions

\[ D_{\text{TM}} = \{ \langle M \rangle : \langle M \rangle \notin L(M) \} \text{ not c.e.} \]
\[ K_{\text{TM}} = \{ \langle M \rangle : L(M) \notin \text{L(M)} \} \text{ c.e. but undecidable} \]
\[ A_{\text{TM}} = \{ \langle M, w \rangle : M \text{ accepts } w \} \text{ ditto} \]

- \( K_{\text{TM}} \leq_m A_{\text{TM}} \) via \( f(\langle M \rangle) = \langle M, \langle M \rangle \rangle \).
- Here \( f \) is computable because it just duplicates whatever code it is given, and \( f \) is correct because \( \langle M \rangle \in K_{\text{TM}} \equiv M \text{ accepts } \langle M \rangle \).

Per the diagram, \( D_{\text{TM}} \) does not \( \leq_m \) to \( K_{\text{TM}} \), which is good because \( K_{\text{TM}} \) is c.e. so \( D_{\text{TM}} \) would be c.e., which it is not.

\( \text{NE}_{\text{TM}} \): INST: A TM M

\[ \text{Guess: } L(M) \neq \emptyset ? \]

As a language, \( \text{NE}_{\text{TM}} = \{ \langle M \rangle : L(M) \neq \emptyset \} \).

\[ A_{\text{TM}} \leq_m \text{NE}_{\text{TM}} \text{: We need to build a computable mapping of code } \langle M, w \rangle \rightarrow f \rightarrow M' \text{ such that } L(M') \neq \emptyset \equiv M \text{ accepts } w.\]

\( f \) is computable because we can just drop a particular \( M \) and \( W \) into the slot in our code framework for \( M' \).

\( f \) is correct because \( \langle M, w \rangle \in A_{\text{TM}} \Rightarrow \text{M accepts } w \Rightarrow \forall x, M' \text{ accepts } x \Rightarrow L(M') = \Sigma^* \Rightarrow L(M') \neq L(M) \Rightarrow (M') \notin \text{NE}_{\text{TM}} \)

whereas \( \langle M, w \rangle \notin A_{\text{TM}} \Rightarrow \forall x, M'(x) \) does not escape the reject box \( \Rightarrow L(M') = \emptyset \).

\[ A_{\text{TM}} \leq_m \text{NE}_{\text{TM}}, \text{ and since } A_{\text{TM}} \text{ is undecidable, so is } \text{NE}_{\text{TM}}. \]

\[ f(\langle M, w \rangle) = L(M') \in \text{NE}_{\text{TM}} \]

\[ A_{\text{TM}} \leq_m \text{NE}_{\text{TM}}, \text{ and since } A_{\text{TM}} \text{ is undecidable, so is } \text{NE}_{\text{TM}}. \]

\[ A_{\text{TM}} \leq_m \text{ALL}_{\text{TM}} \equiv \{ \langle M \rangle : L(M) = \Sigma^* \} \text{ so } \text{ALL}_{\text{TM}} \text{ is undecidable too.} \Rightarrow \langle M' \rangle \notin \text{ALL}_{\text{TM}} \]