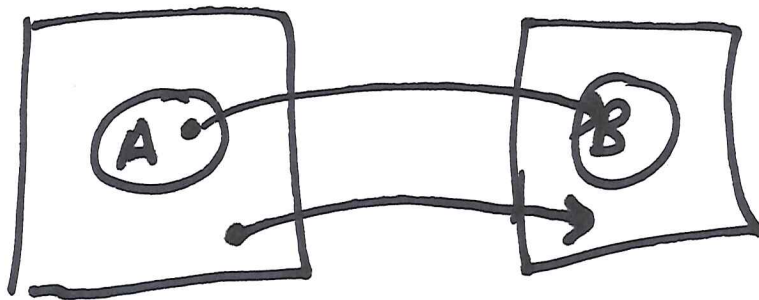


Reductions

A reduction from language A to language B is a ^{TM-}function f such that $f(w) \in B \iff w \in A$.



Written
 $A \leq B$

Fact:

If $A \leq B$ and B recursive, then

A recursive

If $A \leq B$ and B r.e., then A r.e.

— — —
We reduced ~~the~~ acceptance problem A_{TM} to halting problem.

Emptiness Problem

Define $NE_{TM} = \{ \langle M \rangle : TM \text{ } M \text{ non-empty accepts something} \}$

Claim: $A_{TM} \leq NE_{TM}$

Idea: For $\langle M, w \rangle$

build machine $N_{M,w}$ which has M & w hard-coded into it and is programmed to erase its input, write w on tape, and then simulate M , and accept $\Leftrightarrow M$ accepts.

This is a reduction.

If $\langle M, w \rangle$ in A_{TM} , $N_{M,w}$ accepts everything
If $\langle M, w \rangle$ not in A_{TM} , $N_{M,w}$ accepts nothing.