# HW6? - Reductions and (un-)Decidability

COMP361 — Suhas Arehalli

Spring 2025

## Instructions

Note that unlike other assignments, there are **no graded problems** on this assignment. Think of it as a set of practice problems to guide your preparation for the next exam. These are organized by topic, covering the relevant topics we haven't seen a HW problem on.

My advice is to know the relevant proof strategies, here suggested through the ***HINT***s. Though many of these are most quickly solved using the language of mapping reducibility, that content is considered optional for this unit, and every problem is solvable by an explicit reduction. For instance, proving undecidability is entirely possible through proof by contradiction: assume the language is decidable and show you can build a decider for a language we already know is undecidable, a contradiction.

## Questions

### Decidability

1. *(Based on Sipser 4.14)* Let $C = \{\langle G, x\rangle \mid G \text{ is a CFG and } x \text{ is a substring of } y \in L(G)\}$. Prove that $C$ is decidable using a reduction to $E_{CFG}$.

   ***HINT**: Recall from your Exam 1 that the intersection of a CFL and a Regular Language is Context-Free.*

### Undecidability

1. *(Based on Sipser 5.30)* Consider the language

   $$LEFT_{TM} = \{\langle M, w\rangle \mid M \text{ attempts to move left from the leftmost cell when run on } w\}.$$

   Prove $LEFT_{TM}$ is undecidable.

   ***HINT**: Construct a machine $M'$ such that $\langle M', w\rangle \in LEFT_{TM}$ iff $M$ accepts $w$.*

2. *(Based on Sipser 5.29)* A useless state in a TM is a state that is never visited during the computation history corresponding to any input $w$. Let

   $$USELESS_{TM} = \{\langle M, q\rangle \mid M \text{ is a TM, and } q \text{ is a useless state of } M\}$$

Prove $USELESS_{TM}$ is undecidable.

***HINT**: Recall that $E_{TM}$ is undecidable. Consider the relationship between $USELESS_{TM}$ and $E_{TM}$.*

3. Prove that $EQ_{CFG}$ is undecidable by...
   (a) ...a reduction using $ALL_{CFG}$.
   (b) ...a reduction using computation histories.
   ***HINT**: Revisit the proof for $ALL_{CFG}$ and attempt to modify it to use $EQ_{CFG}$ instead of $ALL_CFG$. This shouldn't be a large modification, since a decider for $ALL_{CFG}$ would be a special case of $EQ_{CFG}$.*

## (non-)RE-ness

1. *(Based on Sipser 4.12)* Consider a Recursively Enumerable (RE) language $A = \{\langle M_1\rangle, \langle M_2\rangle, \dots\}$ where each $M_i$ is a decider. Prove that there exists a decidable language $D$ such that no $M_i \in A$ decides $D$ (i.e., $\forall M_i \in A$, $L(M_i) \neq D$). Conclude that $B = \{\langle M\rangle \mid M \text{ is a decider}\}$ is not RE.

   ***HINT**: Sipser claims thinking about an Enumerator of $A$ is helpful (which must exist, since $A$ is RE. I personally don't find it necessary, or particularly helpful, but your mileage may vary!*