Mapping Reduction

CSE 105 Week 9 Discussion

Deadlines and Logistics

- Test 2 next week (week 9)
- Do review quizzes on PrairieLearn
- HW 6 due 12/3/24 at 5pm (week 10)
- Project due 12/11/24 11am (final week)

Motivation

- We want to leverage our previous results of language properties
- Thus, we want to relate the “difficulty level” of one problem to another

If problem X is no harder than problem Y
…and if Y is **decidable**
…then X must also be **decidable**

If problem X is no harder than problem Y
…and if X is **undecidable**
…then Y must also be **undecidable**

“Problem X is no harder than problem Y” means
“Can convert questions about membership in X to questions about membership in Y”

Mapping reduction & computable functions

Definition: A is **mapping reducible to** B means there is a computable function $f : \Sigma^* \to \Sigma^*$ such that *for all* strings x in Σ^*,

$$x \in A \qquad \text{if and only if} \qquad f(x) \in B.$$

Notation: when A is mapping reducible to B, we write $A \leq_m B$.

Intuition: $A \leq_m B$ means A is no harder than B, i.e. that the level of difficulty of A is less than or equal the level of difficulty of B. **"Can convert questions about membership in A to questions about membership in B"**

Computable functions

Definition: A function $f : \Sigma^* \to \Sigma^*$ is a **computable function** means there is some Turing machine such that, for each x, on input x the Turing machine halts with exactly $f(x)$ followed by all blanks on the tape

Warm up: If A is mapping reducible to B then the complement of A is mapping reducible to the complement of B.

Theorems 5.22, 5.28: If A is mapping reducible to B…
… and B is decidable, then A is decidable.
… and A is undecidable, then B is undecidable.
… and B is recognizable, then A is recognizable.
… and A is unrecognizable, then B is unrecognizable.

Mapping reduction practice

RQ8.10. Properties of mapping reductions

Recall that **mapping reduction** is defined in section 5.3: For languages A and B over Σ, we say that the problem A mapping reduces to B means there is a computable function $f : \Sigma^* \to \Sigma^*$ such that for all $x \in \Sigma^*$, $x \in A$ iff $f(x) \in B$. A computable function that makes the iff true is said to witness the mapping reduction from A to B.

Select all and only the true statements below.

- [ ] For all languages A and B, if A mapping reduces to B then B mapping reduces to A.
- [ ] Every language mapping reduces to its complement.
- [ ] Σ^* mapping reduces to every nonempty language over Σ.
- [ ] Every decidable language mapping reduces to $\emptyset$.
- [ ] $\emptyset$ mapping reduces to every nonempty language over Σ.
- [ ] For all languages A and B and C, if A mapping reduces to B and B mapping reduces to C then A mapping reduces to C.

Recall that **mapping reduction** is defined in section 5.3: For languages A and B over Σ, we say that the problem A mapping reduces to B means there is a computable function $f : \Sigma^* \to \Sigma^*$ such that for all $x \in \Sigma^*$, $x \in A$ iff $f(x) \in B$. A computable function that makes the iff true is said to witness the mapping reduction from A to B.

Fix $\Sigma = \{0, 1\}$ throughout this question.

Is each of the stated mapping reductions witnessed by the given function?

$\{0, 1\} \leq_m \{00, 10\}$ is witnessed by the computable function $g : \Sigma^* \to \Sigma^*$ given by $g(x) = x0$ for all x

$\{00, 10\} \leq_m \{0, 1\}$ is witnessed by the computable function $f : \Sigma^* \to \Sigma^*$ given by

$$f(x) = \begin{cases} 0 & \text{if } x = y0 \text{ for some } y \in \{0,1\} \\ 00 & \text{otherwise} \end{cases}$$

$\{0, 1\} \leq_m \{00, 10\}$ is witnessed by the computable function $f : \Sigma^* \to \Sigma^*$ given by

$$f(x) = \begin{cases} 0 & \text{if } x = y0 \text{ for some } y \in \{0,1\} \\ 00 & \text{otherwise} \end{cases}$$

1. The mapping reduction relationship is not true.
2. The mapping reduction relationship is true but the given function does not witness this mapping reduction.
3. This mapping reduction is witnessed by this computable function.

Halting Problem

Halting problem

$$HALT_{TM} = \{\langle M, w\rangle \mid M \text{ is a Turing machine, } w \text{ is a string, and } M \text{ halts on } w\}$$

Prove : $A_{TM} \leq_m HALT_{TM}$

$$HALT_{TM} \leq_m A_{TM}$$

ᴬ**5.7** Show that if A is Turing-recognizable and $A \leq_m \overline{A}$, then A is decidable.

Fix $\Sigma = \{0, 1\}$ and define $const_{out} \in \Sigma^*$ to be a string constant that is not the code of any pair of the form $\langle M, w \rangle$, where M is a Turing machine and w is a string.

Consider the computable function defined by the high-level description of the TM computing it:

$F =$ "On input x:

1. If $x \neq \langle M, w \rangle$ for any Turing machine M and string w, output $const_{out}$.
2. Otherwise, let M be the Turing machine and w the string such that $x = \langle M, w \rangle$.
3. Define the Turing machine M' as :
   "On input y,
   1. Run M on y^R. If it accepts, accept. If it rejects, reject."
4. Output $\langle M', w^R \rangle$."

- [ ] There is a string x for which $x \neq F(x)$
- [ ] For all strings x, if $x \in A_{TM}$ then $F(x) \in HALT_{TM}$
- [ ] For all strings x, if $x \in HALT_{TM}$ then $F(x) \in HALT_{TM}$
- [ ] For all strings x, if $x \in HALT_{TM}$ then $F(x) \in A_{TM}$
- [ ] There is a string x for which $x = F(x)$

RQ9.1. Complementation and mapping reduction

Recall that **mapping reduction** is defined in section 5.3: For languages A and B over Σ, we say that the problem A mapping reduces to B means there is a computable function $f : \Sigma^* \to \Sigma^*$ such that for all $x \in \Sigma^*$, $x \in A$ iff $f(x) \in B$.

- For any languages X and Y, if $X \leq_m Y$ then $\overline{X} \leq_m \overline{Y}$.
- For each language X, $X \leq_m \overline{X}$.
- For any languages X and Y, if $X \leq_m Y$ then $\overline{Y} \leq_m \overline{X}$.
- None of the above

Recall that **mapping reduction** is defined in section 5.3: For languages A and B over Σ, we say that the problem A mapping reduces to B means there is a computable function $f : \Sigma^* \rightarrow \Sigma^*$ such that for all $x \in \Sigma^*$, $x \in A$ iff $f(x) \in B$.

Also recall that

$$A_{TM} = \{\langle M, w\rangle \mid M \text{ is a Turing machine, } w \text{ is a string and } M \text{ accepts } w\}$$

What can we conclude from knowing that A_{TM} mapping reduces to a language L? (Select all and only that apply)

- [ ] L is infinite.
- [ ] L doesn't equal Σ^*.
- [ ] L is undecidable.

Select all possible options that apply.