BU CS 332 – Theory of Computation

https://forms.gle/VQ9JyAvixkrqfUbHA

Lecture 17:

- Mapping Reductions

Reading:

Sipser Ch 5.3

Mark Bun

November 9, 2021

Take-home part of test 2 due Wednesday, 11:59 PM

Reductions

A reduction from problem A to problem B is an algorithm for problem A which uses an algorithm for problem B as a subroutine

If such a reduction exists, we say "A reduces to B"

Positive uses: If A reduces to B and B is decidable, then A is also decidable

Ex. E_{DFA} is decidable $\Rightarrow$ EQ_{DFA} is decidable

Negative uses: If A reduces to B and A is undecidable, then B is also undecidable

Ex. E_{TM} is undecidable $\Rightarrow$ EQ_{TM} is undecidable

🚩🚩 Warning 🚩🚩

$\{\langle M, w\rangle \mid$ TM M accepts $w\}$

What's wrong with the following "proof"?

Bogus "Theorem": A_{TM} is not Turing-recognizable

Bogus "Proof": Let R be an alleged recognizer for A_{TM}. We construct a recognizer S for unrecognizable language $\overline{A_{\text{TM}}}$:

TM S

On input $\langle M, w\rangle$:

1. Run R on input $\langle M, w\rangle$
2. If R accepts, reject. Otherwise, accept.

$\overline{A_{\text{TM}}}$ is unrecognizable

Error: S is not a recognizer for $\overline{A_{\text{TM}}}$

If M loops on w, then $\langle M, w\rangle \in \overline{A_{\text{TM}}}$

But, $S(\langle M, w\rangle)$ loops forever, so behavior of S is not correct.

This sure looks like a reduction from $\overline{A_{\text{TM}}}$ to A_{TM}

Mapping Reductions: Motivation

1. How do we formalize the notion of a reduction?
2. How do we use reductions to show that languages are unrecognizable?
3. How do we protect ourselves from accidentally "proving" bogus statements about recognizability?

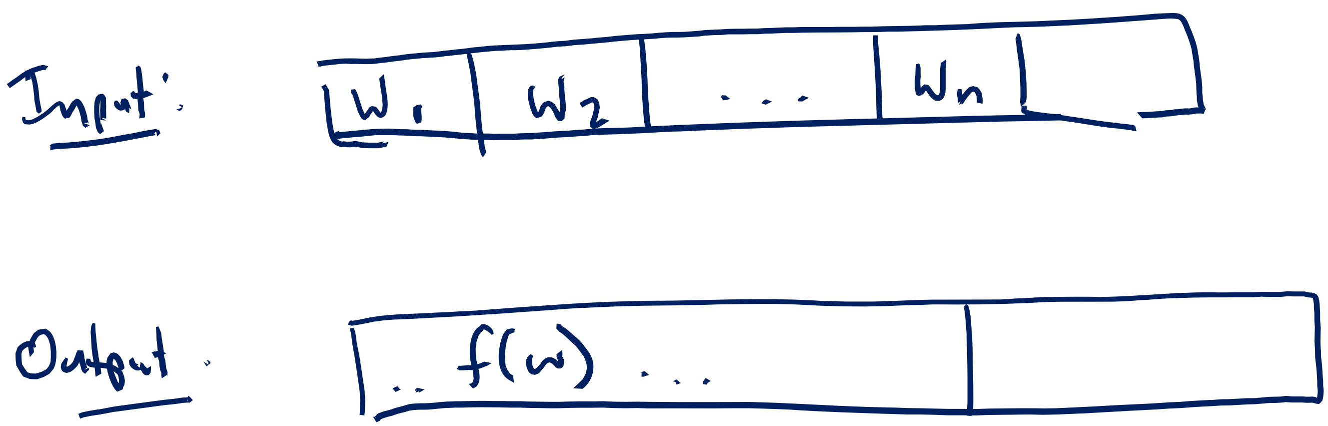
Computable Functions

So far, TMs solve "decision problems" (yes/no)

Now, we want TMs to compute more interesting functions

Definition:

A function $f: \Sigma^* \to \Sigma^*$ is computable if there is a TM M which, given as input any $w \in \Sigma^*$, halts with only $f(w)$ on its tape. ("Outputs $f(w)$")



Computable Functions

Definition:

A function $f: \Sigma^* \to \Sigma^*$ is computable if there is a TM M which, given as input any $w \in \Sigma^*$, halts with only $f(w)$ on its tape. ("Outputs $f(w)$")

Input: | x_1 | ... | x_m | y_1 | ... | y_n |
(x, y are binary numbers)

Example 1: $f(\langle x, y \rangle) = x + y$

Output: | z_1 | ... | z_k | |
$z = x + y$

Example 2: $f(\langle M, w \rangle) = \langle M' \rangle$ where M is a TM, w is a string, and M' is a TM that ignores its input and simulates running M on w

On input $\langle M, w \rangle$
Construct M':
"On input x:
1. Ignore x
2. Run M on input w. Accept if accepts, Reject if rejects"

Output $\langle M' \rangle$

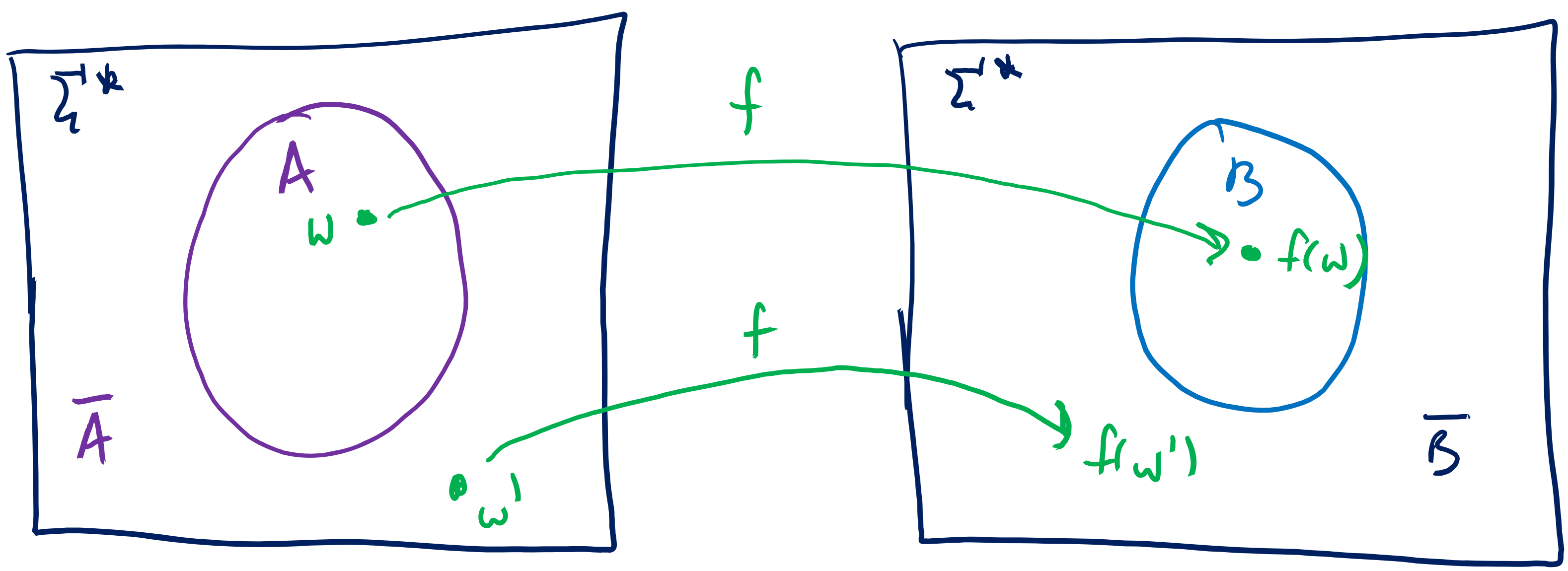
Mapping Reductions

Definition: $A, B \subseteq \Sigma^*$

Language A is mapping reducible to language B, written

$$A \leq_m B$$

if there is a computable function $f: \Sigma^* \rightarrow \Sigma^*$ such that for all strings $w \in \Sigma^*$, we have $w \in A \Leftrightarrow f(w) \in B$



Mapping Reductions

Definition:

Language A is mapping reducible to language B, written

$$A \leq_{\mathrm{m}} B$$

if there is a computable function $f: \Sigma^* \rightarrow \Sigma^*$ such that for all strings $w \in \Sigma^*$, we have $w \in A \Leftrightarrow f(w) \in B$

If $A \leq_{\mathrm{m}} B$, which of the following is true?

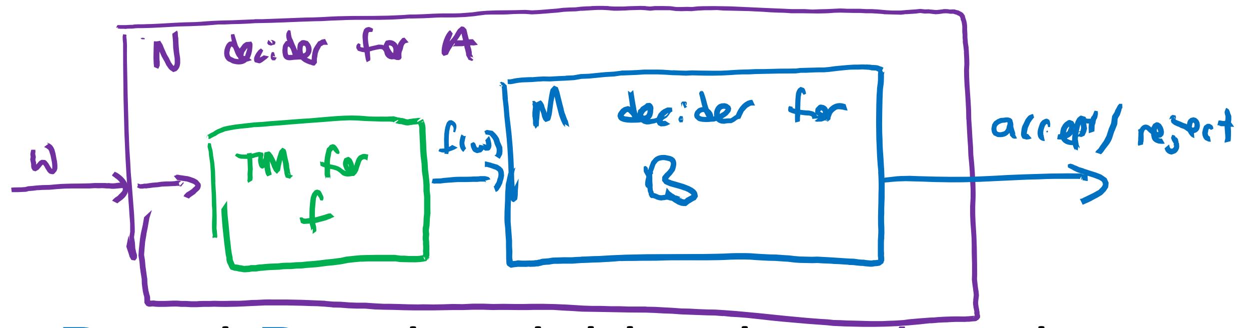
a) $\bar{A} \leq_{\mathrm{m}} B$

b) $A \leq_{\mathrm{m}} \bar{B}$

c) $\bar{A} \leq_{\mathrm{m}} \bar{B}$

d) $\bar{B} \leq_{\mathrm{m}} \bar{A}$

Decidability



Theorem: If $A \leq_{\mathrm{m}} B$ and B is decidable, then A is also decidable

Proof: Let M be a decider for B and let $f: \Sigma^* \to \Sigma^*$ be a mapping reduction from A to B. Construct a decider for A as follows:

TM N:

On input w:

1. Compute $f(w)$
2. Run M on input $f(w)$
3. If M accepts, accept. If it rejects, reject.

Proof of correctness (N decides A):

1) If $w \in A$, then $f(w) \in B$ [defn of mapping red.]
$\Rightarrow$ M accepts f(w) [M decides B]
$\Rightarrow$ N accepts

2) If $w \notin A$, then $f(w) \notin B$ [def of map. red.]
$\Rightarrow$ M rejects f(w) [M decides B]
$\Rightarrow$ N rejects

Undecidability

Theorem: If $A \leq_{\mathrm{m}} B$ and B is decidable, then A is also decidable

(contrapositive of Thm)

Corollary: If $A \leq_{\mathrm{m}} B$ and A is undecidable, then B is also undecidable

Old Proof: Equality Testing for TMs

$$EQ_{\text{TM}} = \{\langle M_1, M_2\rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2)\}$$

Theorem: EQ_{TM} is undecidable

know: E_{TM} undecidable $= \{\langle M\rangle \mid M \text{ is a TM}, L(M) = \emptyset\}$

Proof: Suppose for contradiction that there exists a decider R for EQ_{TM}. We construct a decider for E_{TM} as follows:

On input $\langle M\rangle$:

1. Construct TMs M_1, M_2 as follows:

   $M_1 = M$

   $M_2 =$ "On input x,
   1. Ignore x and reject"

2. Run R on input $\langle M_1, M_2\rangle$

3. If R accepts, accept. Otherwise, reject.

If $\langle M\rangle \in E_{TM}$, then $L(M_1) = L(M) = \emptyset$, $L(M_2) = \emptyset$ $\Rightarrow \langle M_1, M_2\rangle \in EQ_{TM}$ $\Rightarrow R$ accepts.

If $\langle M\rangle \notin E_{TM}$ then $L(M_1) = L(M) \neq \emptyset$; $L(M_2) = \emptyset$ $\Rightarrow \langle M_1, M_2\rangle \notin EQ_{TM} \Rightarrow R$ rejects

This is a reduction from E_{TM} to EQ_{TM}

New Proof: Equality Testing for TMs

$EQ_{\mathrm{TM}} = \{\langle M_1, M_2\rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2)\}$

Theorem: $E_{\mathrm{TM}} \leq_{\mathrm{m}} EQ_{\mathrm{TM}}$ hence EQ_{TM} is undecidable

Proof: The following TM N computes the reduction f:

$f : \Sigma^* \to \Sigma^*$ Input: $\langle M\rangle$ Output: $\langle M_1, M_2\rangle$ $\langle M\rangle \in E_{TM} \iff \langle M_1, M_2\rangle \in EQ_{TM}$

On input $\langle M\rangle$:

1. Construct TMs M_1, M_2 as follows:

   $M_1 = M$

   $M_2 =$ "On input x,
   1. Ignore x and reject"

2. Output $\langle M_1, M_2\rangle$

Mapping Reductions: Recognizability

Theorem: If $A \leq_m B$ and B is recognizable, then A is also recognizable

Proof: Let M be a recognizer for B and let $f: \Sigma^* \to \Sigma^*$ be a mapping reduction from A to B. Construct a recognizer for A as follows:

TM N:

On input w:

1. Compute $f(w)$
2. Run M on input $f(w)$
3. If M accepts, accept. Otherwise, reject.

Correctness:

1) If $w \in A \Rightarrow f(w) \in B$ [f is a map. red.]
$\Rightarrow$ M accepts [M recognizes B]
$\Rightarrow$ N accepts w.

2) If $w \notin A \Rightarrow f(w) \notin B$ [f is a mapping red]
$\Rightarrow$ M either rejects or loops [M recognizes B]
$\Rightarrow$ N either rejects or loops on w

Unrecognizability

Theorem: If $A \leq_m B$ and B is recognizable, then A is also recognizable

Corollary: If $A \leq_m B$ and A is unrecognizable, then B is also unrecognizable

We know $\overline{A_{TM}}$ is unrecognizable

Corollary: If $\overline{A_{TM}} \leq_m B$, then B is unrecognizable

Corollary: If $A_{TM} \leq_m \overline{B}$ then B is unrecognizable

Recognizability and A_{TM}

Let L be a language. Which of the following is true?

A_{TM} is recognizable + Theorem

a) If $L \leq_{\mathrm{m}} A_{\mathrm{TM}}$, then L is recognizable
b) If $A_{\mathrm{TM}} \leq_{\mathrm{m}} L$, then L is recognizable
c) If L is recognizable, then $L \leq_{\mathrm{m}} A_{\mathrm{TM}}$ ← also true!
d) If L is recognizable, then $A_{\mathrm{TM}} \leq_{\mathrm{m}} L$

Theorem: L is recognizable *if and only* if $L \leq_{\mathrm{m}} A_{\mathrm{TM}}$

Recognizability and A_{TM}

"A_{TM} is the hardest recognizable language"
i.e. "A_{TM} is complete for RE = {recognizable lang.} under mapping reductions"

Theorem: L is recognizable *if and only if* $L \leq_{\mathrm{m}} A_{\mathrm{TM}}$

Proof: $\Leftarrow$) If $L \leq_m A_{TM}$, then by Thm + fact that A_{TM} is recognizable, L is recognizable

$\Rightarrow$) Suppose L is recognizable by TM M.

Claim: $\exists$ a mapping reduction f from L to A_{TM}

Want $w \in L \iff f(w) \in A_{TM}$. Compute f using the following TM R:

"On input w:
Output $\langle M, w \rangle$"

Correctness: $w \in L \Rightarrow M$ accepts $w \Rightarrow \langle M, w \rangle \in A_{TM}$
$w \notin L \Rightarrow M$ does not accept $w \Rightarrow \langle M, w \rangle \notin A_{TM}$

Example: Another reduction to EQ_{TM}

$$EQ_{\mathrm{TM}} = \{\langle M_1, M_2\rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2)\}$$

Theorem: $A_{\mathrm{TM}} \leq_{\mathrm{m}} EQ_{\mathrm{TM}}$ $\quad A_{TM} = \{\langle M, w\rangle \mid \text{TM } M \text{ accepts } w\}$

Proof: The following TM N computes the reduction f:

$\langle M, w\rangle \in A_{TM} \Rightarrow f(\langle M, w\rangle) = \langle M_1, M_2\rangle \in EQ_{TM}$

$\langle M, w\rangle \notin A_{TM} \Rightarrow f(\langle M, w\rangle) = \langle M_1, M_2\rangle \notin EQ_{TM}$

What should the inputs and outputs to f be?

a) f should take as input a pair $\langle M_1, M_2\rangle$ and output a pair $\langle M, w\rangle$

b) f should take as input a pair $\langle M, w\rangle$ and output a pair $\langle M_1, M_2\rangle$

c) f should take as input a pair $\langle M_1, M_2\rangle$ and either accept or reject

d) f should take as input a pair $\langle M, w\rangle$ and either accept or reject

Example: Another reduction to EQ_{TM}

$$EQ_{\mathrm{TM}} = \{\langle M_1, M_2\rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2)\}$$

Theorem: $A_{\mathrm{TM}} \leq_{\mathrm{m}} EQ_{\mathrm{TM}}$

Proof: The following TM computes the reduction:

$\langle M, w\rangle \in A_{TM} \Leftrightarrow \langle M_1, M_2\rangle \in EQ_{TM}$

M accepts w $\Leftrightarrow L(M_1) = L(M_2)$

$L(M_1)$: Σ^* if M accepts w; $\emptyset$ if M does not accept w

$L(M_2)$: Σ^*

On input $\langle M, w\rangle$:

1. Construct TMs M_1, M_2 as follows:

   M_1 = "On input x,
   1. Ignore x
   2. Run M on w. If accepts, accept. If rejects, reject."

   M_2 = "On input x,
   1. Ignore x
   2. Accept"

2. Output $\langle M_1, M_2\rangle$

Correctness of reduction:

1) If $\langle M, w\rangle \in A_{TM} \Rightarrow L(M_1) = L(M_2) = \Sigma^*$ ✓
2) If $\langle M, w\rangle \notin A_{TM} \Rightarrow L(M_1) = \emptyset \neq \Sigma^* = L(M_2)$ ✓

1) + 2) means $\langle M, w\rangle \in A_{TM} \Leftrightarrow \langle M_1, M_2\rangle \in EQ_{TM}$

Consequences of $A_{\mathrm{TM}} \leq_{\mathrm{m}} EQ_{\mathrm{TM}}$

1. Since A_{TM} is undecidable, EQ_{TM} is also undecidable

2. $A_{\mathrm{TM}} \leq_{\mathrm{m}} EQ_{\mathrm{TM}}$ implies $\overline{A_{\mathrm{TM}}} \leq_{\mathrm{m}} \overline{EQ_{\mathrm{TM}}}$
   Since $\overline{A_{\mathrm{TM}}}$ is unrecognizable, $\overline{EQ_{\mathrm{TM}}}$ is unrecognizable

EQ_{TM} itself is also unrecognizable

$EQ_{\mathrm{TM}} = \{\langle M_1, M_2\rangle \mid M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2)\}$

Theorem: $\overline{A_{\mathrm{TM}}} \leq_{\mathrm{m}} EQ_{\mathrm{TM}}$ hence EQ_{TM} is unrecognizable

Proof: The following TM computes the reduction:

On input $\langle M, w\rangle$:

1. Construct TMs M_1, M_2 as follows:

M_1 = "On input x,
1. Ignore x
2. Run M on input w
3. If M accepts, accept. Otherwise, reject."

M_2 = "On input x,
1. Ignore x and reject"

To show Correctness:

$\langle M, w\rangle \in \overline{A_{TM}} \iff \langle M_1, M_2\rangle \in EQ_{TM}$.

2. Output $\langle M_1, M_2\rangle$