Today’s lecture

• Mapping reductions for RE, coRE
• More examples of decidable/RE/coRE languages
• More examples of undecidable/unRE,unCoRE languages
Mapping reducibility

- Let $A, B \subseteq \Sigma^*$ be any two languages.
- Definition: $A$ is map-reducible to $B$ (written “$A <_m B$”) if there is a function $f$ such that:
  1) The function $f$ is computable (by a TM $M$).
  2) For all $w \in A$, we have $f(w) \in B$.
  3) For all $w$ not in $A$, we have that $f(w)$ is not in $B$.
- Equivalently: $(w \in A) \iff (f(w) \in B)$.
Reductions and undecidability

What we proved:

• Theorem: if $A \lessdot_{m} B$, and $A$ is undecidable, then $B$ is undecidable.

• Equivalently, if $A \lessdot_{m} B$ (i.e., $A$ map reduces to $B$), then
  - $(A$ is undecidable) $\rightarrow$ $(B$ is undecidable)

• Equivalently, if $A \lessdot_{m} B$ (i.e., $A$ map reduces to $B$), then
  - $\neg(B$ is undecidable) $\rightarrow$ $\neg(A$ is undecidable)
  - $(B$ is decidable) $\rightarrow$ $(A$ is decidable)
Reductions and undecidability

- Let $F$ be a TM computing a map reduction from $A$ to $B$
- What we proved: if $B$ is decidable, then $A$ is decidable
  - Let $M$ is a decider for $B$
  - Goal: build a decider $M'$ for $A$
- $M'(w)$:
  1) Compute $w' = F(w)$
  2) Run $M(w')$
  3) If $M(w')$ accepts, then accept $w$, otherwise reject $w$
Reductions and recognizability

- Let $F$ be a TM computing a map reduction from $A$ to $B$
- WTS: if $B$ is in $\text{RE}$, then $A$ is in $\text{RE}$
  - Let $M$ be a TM such that $L(M)=B$
  - Goal: build a TM $M'$ such that $L(M')=A$
- $M'(w)$:
  1) Compute $w' = F(w)$
  2) Run $M(w')$
  3) If $M(w')$ accepts, then accept $w$, otherwise reject $w$
Reductions and undecidability

• Let $F$ be a TM computing a map reduction from $A$ to $B$
• WTS: if $B$ is in $\text{RE}$, then $A$ is in $\text{RE}$
  – Let $M$ be a TM such that $L(M)=B$
  – Goal: build a TM $M'$ such that $L(M')=A$

• $M'(w)$:
  1) Compute $w' = F(w)$
  2) Run $M(w')$
  3) If $M(w')$ accepts, then accept $w$, otherwise reject $w$

$M'$ is a decider

A) Yes
B) No, because it may loop in step 1
C) No, because it may loop in step 2
D) It depends on $w$
E) I don’t know
Reductions and undecidability

- Let $F$ be a TM computing a map reduction from $A$ to $B$.
- WTS: if $B$ is in RE, then $A$ is in RE.
  - Let $M$ be a TM such that $L(M) = B$.
  - Goal: build a TM $M'$ such that $L(M') = A$.

- $M'(w)$:
  1) Compute $w' = F(w)$
  2) Run $M(w')$
  3) If $M(w')$ accepts, then accept $w$, otherwise reject $w$

What can you say about $L(M')$?

A) $L(M') = A$
B) $L(M') \subset A$
C) $L(M') \supset A$
D) None of the above
E) I don’t know
Proof of $A \subseteq L(M')$

- Let $F: A \leq_m B$, and $L(M) = B$

- $M'(w)$:
  
  1) Compute $w' = F(w)$
  
  2) Run $M(w')$, and accept iff $M(w')$ accept

- Assume $w \in A$, then
  
  - $F(w) \in B$ (by definition of reduction)
  
  - $M(w')$ accepts
  
  - $M'(w)$ accepts
Proof of $A \supseteq L(M')$

- Let $F: A \prec B$, and $L(M) = B$
- $M'(w)$:
  1) Compute $w' = F(w)$
  2) Run $M(w')$, and accept iff $M(w')$ accept
- Assume $w$ is not in $A$, then
  - $F(w)$ is not in $B$ (by definition of reduction)
  - $M(w')$ rejects or loops
  - $M'(w)$ rejects or loops
... therefore

- If $A \leq_m B$ and $B$ is RE, then $A$ is RE
- What about coRE?
- Claim: if $F$ is a reduction from $A$ to $B$, then $F$ is also a reduction

A) From $B$ to $A$
B) From $B$ to $A$
C) From $A$ to $B$
D) I don't know
... therefore

- If $A \leq_m B$ and $B$ is RE, then $A$ is RE
- What about coRE?
- Theorem: if $F$ is a reduction from $A$ to $B$, then $F$ is also a reduction from $A$ to $B$
  - $(w \in A) \leftrightarrow (F(w) \in B)$
  - $\neg(w \in A) \leftrightarrow \neg(F(w) \in B)$
  - $(w \in A) \leftrightarrow (F(w) \in B)$
- Corollary: if $A \leq_m B$ and $B$ is coRE, then $A$ is coRE
Mapping reducibility Summary

- Assume $A \leq_m B$, i.e., there is a map reduction from $A$ to $B$
- Then, we have
  - If $B$ is RE, then $A$ is RE
  - If $B$ is coRE, then $A$ is coRE
  - If $B$ is decidable, then $A$ is decidable
  - If $A$ is undecidable, then $B$ is undecidable
  - If $A$ is not in RE, then $B$ is not in RE
  - If $A$ is not in coRE, then $B$ is not in coRE
More problems on CFG

- \( \text{EQ}_{\text{CFG}} = \{ <G_1, G_2> \mid G_1, G_2 \text{ CFG s.t. } L(G_1) = L(G_2) \} \)
- \( \text{SUB}_{\text{CFG}} = \{ <G_1, G_2> \mid G_1, G_2 \text{ CFG s.t. } L(G_1) \subseteq L(G_2) \} \)
- \( \text{SUP}_{\text{CFG}} = \{ <G_1, G_2> \mid G_1, G_2 \text{ CFG s.t. } L(G_1) \supseteq L(G_2) \} \)

- Can you give reductions between any two of these problems? In what direction?
  - \( \text{EQ}_{\text{CFG}} < \text{SUB}_{\text{CFG}} ? \)
  - \( \text{SUB}_{\text{CFG}} < \text{SUP}_{\text{CFG}} ? \)
  - \( \text{SUP}_{\text{CFG}} < \text{EQ}_{\text{CFG}} ? \)
Reduction: $\text{SUB}_{\text{CFG}} < \text{SUP}_{\text{CFG}}$

- $\text{SUB}_{\text{CFG}} = \{<G_1,G_2> \mid G_1, G_2 \text{ CFG s.t. } L(G_1) \subseteq L(G_2) \}$
- $\text{SUP}_{\text{CFG}} = \{<G_1,G_2> \mid G_1, G_2 \text{ CFG s.t. } L(G_1) \supseteq L(G_2) \}$
- $F(<G_1,G_2>) = <G_2,G_1>$

Which statement is true?

A) $F$ is a computable function
B) $F(\text{SUB}_{\text{CFG}}) \subseteq \text{SUP}_{\text{CFG}}$
C) $F(\text{SUB}_{\text{CFG}}) \subseteq \text{SUP}_{\text{CFG}}$
D) all of the above
Reduction: $\text{SUP}_{\text{CFG}} < \text{EQ}_{\text{CFG}}$

- $\text{SUP}_{\text{CFG}} = \{ <G_1, G_2> | G_1, G_2 \text{ CFG s.t. } L(G_2) \subseteq L(G_1) \}$
- $\text{EQ}_{\text{CFG}} = \{ <G_1, G_2> | G_1, G_2 \text{ CFG s.t. } L(G_1) = L(G_2) \}$
- Observation: $A \subseteq B \leftrightarrow A \cup B = B$
- $F(<G_1, G_2>) = ???
Reduction: $\text{EQ}_{\text{CFG}} < \text{SUB}_{\text{CFG}}$

- $\text{SUB}_{\text{CFG}} = \{ <G_1, G_2> | G_1, G_2 \text{ CFG s.t. } L(G_1) \subseteq L(G_2) \}$
- $\text{EQ}_{\text{CFG}} = \{ <G_1, G_2> | G_1, G_2 \text{ CFG s.t. } L(G_1) = L(G_2) \}$
- $F(<G_1, G_2>) = ???$
Undecidable Problems

• $\text{ALL}_{\text{CFG}} = \{ <G> \mid G \text{ is a CFG and } L(G) = \Sigma^* \}$

• Sipser Theorem 5.13: $\text{ALL}_{\text{CFG}}$ is undecidable

• What can you say about $\text{EQ}_{\text{CFG}}$?

• $\text{ALL}_{\text{CFG}} < \text{EQ}_{\text{CFG}}$

• $F(G) =$
  
  – Let $G' = "S \rightarrow aS \mid bS \mid \ldots \mid \epsilon"$
  
  – Output $<G,G'>$

• $\text{EQ}_{\text{CFG}}$ is undecidable
Undecidable Problems

- \( \mathbf{ALL}_{\text{CFG}} = \{ <G> \mid G \text{ is a CFG and } L(G) = \Sigma^* \} \)
- Sipser Theorem 5.13: \( \mathbf{ALL}_{\text{CFG}} \) is undecidable
- What can you say about \( \mathbf{EQ}_{\text{CFG}} \)?
- \( \mathbf{ALL}_{\text{CFG}} < \mathbf{EQ}_{\text{CFG}} \)
- \( \mathbf{F}(G) = \)
  - Let \( G' = "S \rightarrow aS | bS | \ldots | \varepsilon" \)
  - Output \( <G,G'> \)
- \( \mathbf{EQ}_{\text{CFG}} \) is undecidable

\( \mathbf{SUB}_{\text{CFG}} \) is also undecidable. Which of the following is a valid justification?

A) \( \mathbf{SUB}_{\text{CFG}} <_m \mathbf{EQ}_{\text{CFG}} \)
B) \( \mathbf{SUB}_{\text{CFG}} <_m \mathbf{SUP}_{\text{CFG}} \)
C) \( \mathbf{EQ}_{\text{CFG}} <_m \mathbf{SUB}_{\text{CFG}} \)
D) \( \mathbf{SUB}_{\text{CFG}} <_m \mathbf{ALL}_{\text{CFG}} \)
$E_{TM}$ is undecidable

- $A_{TM} = \{ <M,w> \mid M \text{ is a TM and } M(w) \text{ accepts} \}$
- $E_{TM} = \{<M> \mid M \text{ is a TM and } L(M) \text{ is empty} \}$
- We already proved that $A_{TM}$ is RE, but not coRE
- How can we prove that $E_{TM}$ is coRE, not RE
  - $E_{TM}$ is not RE: Reduce $A_{TM} <_m E_{TM}$
  - $E_{TM}$ is coRE: Reduce $E_{TM} <_m A_{TM}$
$E_{TM}$ is not RE

- $A_{TM} = \{ <M, w> | M \text{ is a TM and } M(w) \text{ accepts} \}$
- $E_{TM} = \{ <M> | M \text{ is a TM and } L(M) \text{ is empty} \}$

- $A_{TM} \prec_m E_{TM}$

- $F(<M, w>) =$
  1. If $M$ not a valid TM, let $M'(x) = \text{reject}$
  2. Otherwise, build $M'(x) = \text{"if } (x == w) \text{ then } M(x) \text{ else reject"}$
  3. Output $<M'>$
$E_{TM}$ is not RE

- $A_{TM} = \{<M,w> | M \text{ is a TM and } M(w) \text{ accepts}\}$
- $E_{TM} = \{<M> | M \text{ is a TM and } L(M) \text{ is empty}\}$

- $A_{TM} \not\preceq_m E_{TM}$

- $F(<M,w>) =$
  1. If $M$ not a valid TM, let $M'(x) = \text{reject}$
  2. Otherwise, build $M'(x) = \text{"if } (x == w) \text{ then } M(x) \text{ else reject"}$
  3. Output $<M'>$

What can you say about $L(M')$?

A) $L(M') = L(M)$
B) $L(M') = \{x | x == w\}$
C) $L(M') \subseteq \{w\}$
D) $w \in L(M')$
E) None of the above
$E_{TM}$ is not RE (alternative reduction)

- $A_{TM} = \{ <M,w> | M \text{ is a TM and } M(w) \text{ accepts} \}$
- $E_{TM} = \{ <M> | M \text{ is a TM and } L(M) \text{ is empty} \}$

- $A_{TM} \leq_m E_{TM}$

- $F(<M,w>) =$
  1. If $M$ not a valid TM, let $M'(x) = \text{reject}$
  2. Otherwise, build $M'(x) = M(w)$
  3. Output $<M'>$
$E_{TM}$ is in coRE

- $A_{TM} = \{ <M,w> \mid M$ is a TM and $M(w)$ accepts $\}$
- $E_{TM} = \{<M> \mid M$ is a TM and $L(M)$ is empty $\}$

- Method 1: give a TM $M$ such that $L(M) = E_{TM}$
- Method 2: $E_{TM} \prec_m A_{TM}$

- Hint: given $M$ can you build $M'$ such that
  - if $L(M)$ is not empty, then $M'$ always accepts
  - if $L(M)$ is empty, then $M'$ always loops
For next Time

- Happy Thanksgiving
- HW7 out, due next week
- Reading: Sipser *Chapters 5*