DECIDABLE LANGUAGES.

Languages represent computational problems

\text{eg: acceptance of DFA expressible as a language, } A_{\text{DFA}} \\
\text{(i.e. is } w \text{ accepted by DFA?)}

\[ A_{\text{DFA}} = \left\{ \langle D, w \rangle \mid D \text{ is a DFA that accepts } w \right\} \]

Showing that a language \((\text{say } A_{\text{DFA}})\) is decidable

\[ \iff \text{ showing corresponding computational problem is decidable} \]

Th1: \(A_{\text{DFA}}\) is decidable. \(\text{(i.e. given any string } x = \langle D', w' \rangle \text{ encoding, decide if Answer Yes/No using a TM, whether } D' \text{ (a DFA's encoding) accepts } w' \text{,)}\)

//All problems requiring non-tool answers rephrased as (combination of) problems requiring tool answers

\text{Proof: (construction)}

\[ M : \langle B, w \rangle \]

1. Simulate \(B\) on input \(w\)
2. If simulation ends in \text{accept}, then accept, else reject
\[ A_{\text{NFA}} = \{ \langle B, w \rangle \mid B \text{ is a NFA that accepts } w \} \]

**Theorem 2:** \( A_{\text{NFA}} \) is decidable

**Proof:** On input \( \langle B, w \rangle \):
1. Convert \( B \) into \( C \) (a DFA)
2. Run \( M \) on input \( w \)
   
   // procedure call
   
   if \( M \) accepts \( w \) then accept
   
   else reject

\[ A_{\text{REGEX}} = \{ \langle R, w \rangle \mid R \text{ is a regex that generates } w \} \]

**Theorem 3:** \( A_{\text{REGEX}} \) is decidable

**Proof:** \( P \) on input \( \langle R, w \rangle \):
1. Convert \( R \) into DFA \( A \)
2. Run \( M \) on \( w \)
   
   // procedure call
   
   if \( M \) accepts \( w \), then accept
   
   else reject
\[
E_{DFA} = \{ \langle A \rangle \mid A \text{ is a DFA and } L(A) = \emptyset \} \\
\rightarrow \text{i.e., set of encodings of DFAs whose language is } \emptyset
\]

**Th 4:** \( E_{DFA} \) is decidable

(i.e., given an encoding of a DFA, answer \( Y/N \) whether \( L(\text{DFA given}) \) is \( \emptyset \) by using a TM)

**Proof:**

[A] generate all strings over \( \Sigma \) of the DFA

Run DFA on each string \( s \)

If any \( s \) is accepted by DFA, then reject DFA

(from membership in \( E_{DFA} \))

Else keep looping

This TM is not a decider

[B] observe: if path from \( q_0 \) to \( q_{accept} \), then \( L(\text{DFA}) \neq \emptyset \); reject

the DFA from membership in \( E_{DFA} \)

**T:** On input \( \langle A \rangle \)

1) Mark \( q_0 \)

2) Repeat until no new state is marked
   For each unmarked \( q' \in Q \)
   If incoming edge from marked \( q \), then mark

3) If no \( q \in F \) is marked, then accept else reject
\[ \mathcal{EQ}_{\text{DFA}} = \{ \langle A, B \rangle \mid L(A) = L(B) \text{ and } A, B \text{ are DFAs} \} \]

**Theorem 5**: \( \mathcal{EQ}_{\text{DFA}} \) is decidable

**Proof**: observe \( L(A) = L(B) \) iff \( L(A) - L(B) = \emptyset \) and \( L(B) - L(A) = \emptyset \)

\[
\begin{align*}
\mathcal{L}(A) & \cap \mathcal{L}(B) \\
& = \mathcal{L}(C) = \emptyset
\end{align*}
\]

Using closure of R.L under \( \cup, \cap, \text{comp} \), conclude that \( \mathcal{L}(C) \) is a R.L.

**(Construction)**: On input \( \langle A, B \rangle \),

1) Construct DFA \( C \) such that \( L(C) = \) \( \text{//defined above.} \)

2) invoke \( E_{\text{DFA}} \) proof \( \text{//procedure} \)

3) Decide accordingly.
\[ A_{CFG} = \{ \langle G, w \rangle \mid G \text{ is a CFG that generates } w \} \]

**Th 6**: \( A_{CFG} \) is decidable. // Caution: only many derivations

**Proof (construction)**: 
1) Convert \( G \) in \( G' \) (Chomsky N.F.)
   // any derivation of \( w \) occurs in \( 2|w|-1 \) steps, guaranteed.
2) List all derivations using \( G' \), having \( 2|w|-1 \) steps.
3) If any of these derivations yield \( w \), then accept else reject.

\[ A_{PDA} = \{ \langle B, w \rangle \mid B \text{ is a PDA that accepts string } w \} \]

**Th 7**: \( A_{PDA} \) is decidable

\[ E_{CFG} = \{ \langle G \rangle \mid L(G) = \emptyset \text{ and } G \text{ is CFG} \} \]

**Th 8**: \( E_{CFG} \) is decidable. // Caution: only many \( w \)’s

**TM R**: On input \( \langle G \rangle \):
1) Mark all terminals
2) Repeat until no new variable get marked.
   Mark any var \( A \), where \( A \rightarrow v_1v_2 \ldots \ldots v_k \)
   and each symbol \( v_1 \ldots v_k \) is marked
3) If \( S \) is not marked then accept else reject
\[ \text{EQ}_{\text{CFG}} = \{ \langle G, H \rangle \mid L(G) = L(H) \text{ and } G, H \text{ are NPDA}s \} \]

**Th 9**: \( \text{EQ}_{\text{CFG}} \) is NOT decidable

**Th 10** (Corollary of Th. 6):

Every CFL is decidable

---

**NB**: R.L. = R.Exp. = R. Grammar = PDA
CFL \( \equiv \) PDA

CFL \( \equiv \) \begin{align*}
\text{CFG} & \equiv \text{PDA} \\
\text{CSL} & \equiv \text{Context-Sensitive G.} = \text{L.B.A. (NEW)} \end{align*}

Restricted G. \( \equiv \) TM

CHOMSKY HIERARCHY [read link on web]
HALTING PROBLEM

\[ A_{TM} = \{ \langle M, w \rangle \mid TM M accepts w \} \]

Th: \( A_{TM} \) is recognizable.

Proof: by construction.

U: On input \( \langle M, w \rangle \)

1) Simulate operation \( M \) on \( w \)
2) if \( M \) accepts \( w \), then \( U \) accepts \( \langle M, w \rangle \)
   else if \( M \) rejects \( w \), then \( U \) rejects \( \langle M, w \rangle \)
   (else if) -----

/// Digression: MATH!

which set is larger?

1) \( N \) \((1,2,3,\ldots)\)
2) \( \Sigma^* \) where \( \Sigma = \{0,1\} \)
3) \( \mathbb{E} \) = \{2,4,6,8,\ldots\}
4) \( \mathbb{Q} \) (rational #s)
   \[ = \{ \frac{a}{b} \mid a, b \in N \} \]
5) \( \mathbb{R} \) (real #s)
   \[ = \{ \text{numbers expressible in decimals} \} \]
Defn: Function: \( f : \mathbb{N} \to B \)

'into' fn: \( \exists b \in B \)

\( \forall n \in \mathbb{N} \) where
\( f(n) = b \).

'onto' fn: \( \forall b \in B, \exists n \in \mathbb{N} \) \( f(n) = b \)

Correspondence: is a 1-1 onto function.

Defn: Set \( B \) is countable if \( B \) is finite or \( B \) had same size as \( \mathbb{N} \)
(ie, \( f : \mathbb{N} \to B \) is a correspondence)

\( \forall N \in \mathbb{N} \) \( \forall b \in E, f(\frac{b}{2}) = b \)

\( \forall N \in \mathbb{N} \)
\( \forall (a, b) \in \mathbb{N} \)

preimage of \( \frac{a}{b} \)?

Hint: \( \sum_{i=1}^{\infty} i + j \)
Theorem: \( \mathbb{R} \) is uncountable.

Proof: by contradiction. (nested, use construction).

Assume \( \mathbb{R} \) is countable.

<table>
<thead>
<tr>
<th>( n )</th>
<th>( f(n) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3145</td>
</tr>
<tr>
<td>2</td>
<td>8212</td>
</tr>
<tr>
<td>3</td>
<td>235</td>
</tr>
<tr>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

Now, we generate a real \( x \) such that:

\[ f(n) = x, \quad (\text{ie } x \neq f(n) \text{ for any } n) \]

Let \( j \)th fractional position value of \( f(j) \) be \( \neq j \)th fractional position value.

\[ x = 354 \cdots \]

Claim: \( x \) not in \( f(n) \) for any \( n \).

(because) \( x \) differs from \( f(n) \) in its \( n \)th decimal position.

END DIGRESSION

Observe:

1) Countably many TMs
2) Uncountably many languages.

\( \Rightarrow \) more languages than TMs \( \Rightarrow \) some languages will not be recognizable.
Th: There are Turing-recognizable languages that are not countable.

Proof: \( \{TM_S\} \) is countable.

1. \( \{TM_S\} \subseteq \Sigma^* \) \( \forall TM \in \Sigma^* \)

\( \Sigma^* = \{ \varepsilon, 0, 1, 00, 01, 10, 11, 000, \ldots \} \)

Show \( f: N \to \Sigma^* \) is a correspondence.

Hint: \( s \in \Sigma^* \mid |s| = k \) has a pre-image \( n \) in \( N \), where

\[
\sum_{i=0}^{k-1} 2^i
\]

2. \( L = \{ \text{all languages} \} \)

Show \( L \) is uncountable.

Observe: \( \{ \text{all infinite binary sequences} \} \) is uncountable.

(Proof technique:) like "\( R \) is uncountable"

Let \( B = \{ \text{infinite binary sequences} \} \)

\( f: L \to B \) is a correspondence.

\( \Sigma^* = \{ \varepsilon, 0, 1, 00, 01, 10, 11, 000, \ldots \} \)

Example lang.

\( A = \{ 0, 01, 00, 000, \ldots \} \)

\( x = 01010011 \ldots \)
\( f(A) \Rightarrow X_A \) is a 1-1 onto, i.e., correspondence. 
\( X_A \) is an \( \infty \) seq, called the "characteristic seq."

ie, \( f : L \rightarrow B \), is a correspondence.

Here \( f(A) = X_A \), where \( A \in L \) and \( X_A \in B \) (characteristic seq).

As \( B \) is uncountable, \( L \) is also uncountable.

\[ \Sigma^* : \text{set of finite strings, but } \infty \text{ members in it,} \]
\[ \text{can be listed} \]
\[ \text{1-1 correspond with } N \text{ is countable} \]

\( B \) : cannot be listed.
\[ \text{set of } \infty \text{ binary strings} \]

\( B \not\subseteq \Sigma^* \)

\[ A_{TM} = \{ \langle M, w \rangle \mid M \text{ accepts } w \} \]

Th: \( A_{TM} \) is undecidable.

Proof: (by contradiction):
Assume \( H(\langle M, w \rangle) = \begin{cases} \text{accept} & \text{if } M \text{ accepts } w \\ \text{reject} & \text{if } [M \text{ does not accept }] w \end{cases} \)

\[ D(\langle M \rangle) = \begin{cases} \text{accept} & \text{if } M \text{ does not accept } \langle M \rangle \\ \text{reject} & \text{if } M \text{ accepts } \langle M \rangle \end{cases} \]
D: On input \( \langle M \rangle \)

1. Runs \( H(\langle M, \langle M' \rangle \rangle) \)
   
   i.e., Run \( H \) on input \( \langle M, \langle M' \rangle \rangle \)

   /// \( \langle M \rangle \) i representation/encoding of \( M \) using \( \Sigma \)

2. \( H \) is a decider
   
   \[ \text{if } H \text{ accepts, then } D \text{ rejects } \langle M \rangle \]
   
   \[ \text{else } H \text{ rejects, then } D \text{ accepts } \langle M \rangle \]

\[ L(D) = \{ \langle M \rangle \mid M \text{ on input } \langle M \rangle \text{ is not accepted} \} \]

   /// decidable using call to \( H \)

Question: does \( \langle D \rangle \in L(D) \)?

\[ D(\langle D \rangle) = \begin{cases} 
\text{accept} & \text{if } D \text{ does not accept } \langle D \rangle \\
\text{reject} & \text{if } D \text{ accepts } \langle D \rangle 
\end{cases} \]

\[ \rightarrow \text{ absurdity! } \because H \text{ cannot exist.} \]

| \( \langle M_1 \rangle \) | \( \langle M_2 \rangle \) | \( \langle M_3 \rangle \) | \cdots | \( \langle D \rangle \) |
|---|---|---|---|
| \( M_1 \) | A | rej | A | \cdots | 
| \( M_2 \) | A | A | A | \cdots | 
| \( M_3 \) | rej | A | rej | \cdots | 

Red color: behavior of \( H \).
A_{TM} not decidable. \( L = \{ <M> \} \) is ∞

uncountable

Use REDUCIBILITY to prove languages are not decidable.

What about not Turing-recognizable languages?

NOT Turing-recognizable LANGUAGES

Defn: \( L \) is co-Turing recognizable if \( L \) is the complement of a Turing-recognizable language.

Th: \( L \) is decidable \( \iff \) \( L \) is \( T \)-recognizable and \( L \) is co-\( T \)-recognizable.

\( \Rightarrow \) \( L \) is decidable, \( \therefore \) \( \overline{L} \) is \( T \)-recognizable

\( \iff \) \( \overline{L} \) is \( T \)-recognizable

\( \iff \) \( \overline{L} \) is co-\( T \)-recognizable

\( \iff \) Let \( M_1 \) recognize \( L \)

\( \iff \) Let \( M_2 \) recognize \( \overline{L} \)
M: On input w:
1) Run $M_1$ on w and $M_2$ on w alternating steps
2) If $M_1$ accepts w then accept w;
   If $M_2$ accepts w then reject w.

Any $w \in L$ or $w \in \overline{L}$

\[ \therefore \ M_1 \text{ accepts } w \text{ or } M_2 \text{ accepts } w \]
\[ \therefore \ M \text{ is a decider} \]
\[ - \text{ accepts } w \in L \]
\[ - \text{ rejects } w \notin L \]

\[ \therefore \ L \text{ and } \overline{L} \text{ must reside in the set } T\text{-decidable} \]
(Inner circle). \[ \square \]

Cor: $A^\text{T}_\text{TM}$ is not $T$-recognizable.

Proof: (by contradiction). $A^\text{T}_\text{TM}$ is not decidable.