Lecture 15

Last time
- Decidable languages
- Designing deciders

Today
- Designing deciders
- Undecidable languages
- Diagonalization

Sofya Raskhodnikova
Recall

$$A_{DFA} = \{ \langle D, w \rangle \mid D \text{ is a DFA that accepts string } w \}$$

$$E_{DFA} = \{ \langle D \rangle \mid D \text{ is a DFA and } L(D) = \emptyset \}$$

$$EQ_{DFA} = \{ \langle D_1, D_2 \rangle \mid D_1, D_2 \text{ DFAs and } L(D_1) = L(D_2) \}$$

$$A_{DFA}, E_{DFA}, EQ_{DFA}, A_{CFG}, E_{CFG} \text{ are decidable.}$$
Classes of languages

- Recognizable
- Decidable
- CFL
- Regular
Theorem. Every CFL is decidable.

Proof: Let G be a CFG for L. Design a TM $M_G$ that decides L.

- Is it a good idea to convert G to an equivalent PDA P and have $M_G$ simulate P?
I-clicker problem (frequency: AC)

G is a CFG for L. Design a TM $M_G$ that decides L.

Is it a good idea to convert G to an equivalent PDA P and have $M_G$ simulate P?

A. Yes. Why not?
B. No, we can’t always convert G to an equivalent PDA.
C. No, P might loop on some inputs.
D. No, because we don’t have any input to run P on.
E. None of the above.
I-clicker problem (frequency: AC)

G is a CFG for L. Design a TM \( M_G \) that decides L.

A decider for which language is useful as a subroutine?

A. for \( A_{DFA} \)
B. for \( E_{DFA} \)
C. for \( EQ_{DFA} \)
D. for \( A_{CFG} \)
E. for \( E_{CFG} \)
Theorem. Every CFL is decidable.

Proof: Let G be a CFG for L. Design a TM $M_G$ that decides L.

• Is it a good idea to convert G to an equivalent PDA P and have $M_G$ simulate P?

$M = \text{``On input } w:\text{''}

1. Run the decider for $A_{CFG}$ on input $<G,w>$.
2. Accept if yes. O.w. reject.”
Classes of languages

recognizable

decidable

CFL

regular
Theorem. \( \text{INFINITE}_{\text{DFA}} \) is decidable.

\[
\text{INFINITE}_{\text{DFA}} = \{ \langle D \rangle \mid D \text{ is a DFA and } L(D) \text{ is infinite} \}
\]

Idea: Let \( n \) be the number of states in \( D \). 
\( L(D) \) is infinite iff \( D \) accepts a string of length \( \geq n \).

Proof: The following TM \( M \) decides \( \text{INFINITE}_{\text{DFA}} \).

\[
\text{M = `On input } \langle D \rangle, \text{ where } D \text{ is a DFA:}
\]

1. Let \( n \) be the number of states in \( D \).
2. Let \( C \) be a DFA for \( \{ w \mid |w| \geq n \} \).
3. Build a DFA \( B \) for \( L(C) \cap L(D) \).
4. Run a decider for \( E_{\text{DFA}} \) on \( \langle B \rangle \).
5. Accept if it rejects. O.w. reject.”
### Problems in language theory

<table>
<thead>
<tr>
<th>$A_{DFA}$ decidable</th>
<th>$A_{CFG}$ decidable</th>
<th>$A_{TM}$ ?</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_{DFA}$ decidable</td>
<td>$E_{CFG}$ decidable</td>
<td>$E_{TM}$ ?</td>
</tr>
<tr>
<td>$EQ_{DFA}$ decidable</td>
<td>$EQ_{CFG}$ ?</td>
<td>$EQ_{TM}$ ?</td>
</tr>
</tbody>
</table>
We will prove that there are some undecidable languages:

• i.e., problems a computer cannot solve no matter how long it computes

The proof idea is “simple:”

There are more languages than there are Turing Machines.
A language $L$ is **undecidable** if there is no TM that decides $L$.

If $L$ is undecidable, then every TM must either:

1. Accept (infinitely many) strings $s \notin L$.
2. Reject (infinitely many) strings $s \in L$.
3. Loop forever on (infinitely many) strings.
Let $\mathbb{N} = \{1, 2, \ldots\}$ be the natural numbers.
Let $E = \{2, 4, 6, \ldots\}$ be the even natural numbers.
Let $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$ be the integers.

Which one is largest?
A. $\mathbb{N}$
B. $E$
C. $\mathbb{Z}$
D. the same size.
Are there more blue or yellow dots?
A function $f : A \rightarrow B$ is

- **1-to-1** (or *injective*) if
  
  \[ f(a) \neq f(b) \text{ for } a \neq b. \]

- **onto** (or *surjective*) if for all $b \in B$, some $a \in A$ maps to $b$: $f(a) = b$.

- **correspondence** (or *bijective*) if
  
  it is 1-to-1 and onto, i.e.,
  each $a \in A$ maps to a unique $b \in B$, and each $b \in B$ has a unique $a \in A$ mapping to it.
How to compare sizes of infinite sets?

- Two sets are **the same size** if there is a bijection between them.
- A set is **countable** if it is
  - finite or
  - it has the same size as \( \mathbb{N} \), the set of natural numbers.
Examples of countable sets

\[ \emptyset, \{0\}, \{0,1\}, \{0,1, \ldots, 255\} \]

\[ E = \{2,4,6,\ldots\} \]

\[ O = \{1,3,5,7,\ldots\} \]

\[ \text{SQUARES} = \{1,4,9,16,25\ldots\} \]

\[ \text{POWERS} = \{1,2,4,8,16,32\ldots\} \]

\[ |\text{POWERS}| = |\text{SQUARES}| = |E| = |O| = |N| \]
There is a bijection between $\mathbb{N}$ and $\mathbb{N} \times \mathbb{N}$.

\[(0,0) \quad (0,1) \quad (0,2) \quad (0,3) \quad (0,4) \ldots \]
\[(1,0) \quad (1,1) \quad (1,2) \quad (1,3) \quad (1,4) \ldots \]
\[(2,0) \quad (2,1) \quad (2,2) \quad (2,3) \quad (2,4) \ldots \]
\[(3,0) \quad (3,1) \quad (3,2) \quad (3,3) \quad (3,4) \ldots \]
\[(4,0) \quad (4,1) \quad (4,2) \quad (4,3) \quad (4,4) \ldots \]
\{0,1\}^* \text{ is countable}

\{ \langle M \rangle \mid M \text{ is a TM} \} \text{ is countable}

\mathbb{Q}^{+} = \{ \frac{p}{q} \mid p,q \in \mathbb{Z}^{+} \} \text{ is countable!}

Is any set \textit{uncountable}?
Theorem. There is no bijection from the positive integers to the real interval (0,1).

Proof: Suppose $f$ is such a function:

<table>
<thead>
<tr>
<th>$n$</th>
<th>$f(n)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.28347279…</td>
</tr>
<tr>
<td>2</td>
<td>0.88388384…</td>
</tr>
<tr>
<td>3</td>
<td>0.77635284…</td>
</tr>
<tr>
<td>4</td>
<td>0.11111111…</td>
</tr>
<tr>
<td>5</td>
<td>0.12345678…</td>
</tr>
<tr>
<td>…</td>
<td>…</td>
</tr>
</tbody>
</table>

Construct $b \in (0, 1)$ that does not appear in the table:

$b = 0.d_1d_2d_3…$, where $d_i \neq \text{digit } i \text{ of } f(i)$. 

Sofya Raskhodnikova; based on slides by Nick Hopper
The process of constructing a counterexample by “contradicting the diagonal” is called **DIAGONALIZATION**
Let $L$ be any set and $P(L)$ be the power set of $L$.

**Theorem:** There is no onto map from $L$ to $P(L)$

**Proof:** Assume, for a contradiction, that there is an onto map $f : L \rightarrow P(L)$

We construct a set $S$ that cannot be the output, $f(y)$, for any $y \in L$.

Let $S = \{ x \in L \mid x \notin f(x) \}$

If $S = f(y)$ then $y \in S$ if and only if $y \notin S$.
How is that diagonalization?

<table>
<thead>
<tr>
<th>x</th>
<th>$y_1 \in f(x)$?</th>
<th>$y_2 \in f(x)$?</th>
<th>$y_3 \in f(x)$?</th>
<th>$y_4 \in f(x)$?</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y_1$</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>$y_2$</td>
<td>N</td>
<td>Y</td>
<td>N</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>$y_3$</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>$y_4$</td>
<td>Y</td>
<td>N</td>
<td>N</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

$$(y_i \in S) = Y \text{ iff } (y_i \in f(y_i)) = N$$
For all sets L, P(L) has more elements than L