Topics:  
- Set theory remarks
- Resolution of Russell’s Paradox
- Some other paradoxes/theorems
- Start Ch.1 on Regular Languages

Section 1: Finite Automata (DFA’s)

- DFA: the idea
- DFA: definition \((Q, \Sigma, \delta, q_0, F)\)
- Regular Languages: recognized by some DFA
- \(A \cup B, A \cap B, A \cdot B, A^*\)
- Why NFAs? [Answer: for \(A \cdot B, A^*\)]

Section 2: Non-deterministic Finite Automata (NFA’s)

CANVAS SURVEY AT END
Breakout Room Questions:

① Come up with your own paradox (related to those of §5.6 of Handout)

② Give a DFA that recognizes
\[ \{0, 3, 6, 9, 12, 15, \ldots \} \subset \{0, \ldots, 9\}^* \]

③ Give a DFA that recognizes
\[ \{0, 3, 6, 9, 03, 06, 09, 12, 15, \ldots \} \]

④ Give a DFA that recognizes
\[ \{3, 0, 3, 6, 9, 03, 06, 09, 12, 15, \ldots \} \]

⑤ Is there a DFA that recognizes
\[ \{0, 7, 14, 21, 28, 35, 42, \ldots \} \]
⑥ Give a DFA that recognizes
\[ \{ 1^5, 1^7 \} \subseteq \{ 1 \}^* \]

⑦ Give a DFA that recognizes
\[ \{ 1^5, 1^7 \}^* \subseteq \{ 1 \}^* \]
Last Time:

4) Is the set of functions \( \mathbb{N} \to \mathbb{N} \) that you can describe by a finite string in English: countable or uncountable?

ASCII* \( \rightarrow \) \{ meanings as function \}, or some function

5) Yet, \( \{ \mathbb{N} \to \mathbb{N} \} \) is uncountable; proof:

Let's give a surjection \( \{ \mathbb{N} \to \mathbb{N} \} \rightarrow \{ \{ \mathbb{N} \to \{ \text{yes, no} \} \} \}

\( f : \mathbb{N} \to \mathbb{N} \rightarrow \hat{f}(n) = \{ \text{yes, if } f(n) = 1, \text{no, if } f(n) \neq 1 \}

\( \left( \text{functions } \mathbb{N} \to \{ \text{yes, no} \} \right) \overset{\text{bij}}{\longrightarrow} \text{powersets } (\mathbb{N}) \leftarrow \text{uncountable} \)

Fact: If \( \exists \text{ surjective } S \to T \) and \( T \) is uncountable, then \( S \) is uncountable

Breakup \( \rightarrow \) HW

\( \{ \text{function } A \to \mathbb{N} \} \overset{\text{surj}}{\longrightarrow} \{ \text{functions } A \to \{ \text{yes, no} \} \} \)
\[ f \mapsto \tilde{f} \text{ given by } \tilde{f}(a) = \begin{cases} \text{yes} & \text{if } f(a) = 1, \\ \text{no} & \text{if } f(a) > 2 \end{cases} \]

\[ \text{func } A \to \mathbb{N} \mapsto A \to \mathbb{N} \text{ surj } \{ \text{yes, no} \} \]

Cantor's Thm: Takes \( \mathcal{S} \to \text{Power}(\mathcal{S}) \).

Forms \( T = \{ s \in \mathcal{S} \mid s \notin f(s) \} \) is not in itself.

Russell's Paradox: If \( R = \{ S \text{ set } \mid S \notin S \} \)

you get a contradiction with either \( R \in R \) or \( R \notin R \).

Resolution (you have ask a set theorist or logician)

"the set of all sets" is too big to be set

so \( \{ S \text{ sets } \mid \text{blah} \} \) can be too large to be set

\( \{ S \mid S \notin S \} \) - "self-reference" + "negation"
§6 Handout gives a few more:

**Paradox (3):** What is the *smallest positive integer* not described by a phrase in English of at most 100 words? (less than 100 words)

Say it's the number

\[ n = 157324163 \doteq 279. \]

"Bland's Paradox" (probably due to Russell)

Self-reference + "not" = negation.

**Paradox (4), §6:****

Leslie write about (and only about) those people who do not write about themselves.

Does Leslie write about themself? If yes - contradiction. If no - "

Also "barber paradox"
Later: "Halting problem" is undecidable

Proof: Assume it is decidable, and get a contradiction via "self-reference" + "negation"

Chapter 1 in textbook by Sipser:

Regular Languages ↔ languages described by "regular expressions"

Idea: \( \Sigma = \{a, b\} \),

Let \( L = \{ w \in \Sigma^* \mid w \text{ has at least 2 } a's \} \) (in its set of symbols)

\( = \{ aa, aab, bca, aba, bbabbab, \ldots \} \)

Our algorithm (informally):
- read each symbol (letter) of $w$, one by one

$$w = \sigma_1 \sigma_2 \cdots \sigma_k$$

$\sigma_i \in \Sigma = \{a, b\}$

- start here

\[ \begin{array}{c}
\text{b} \quad a \\
\text{a} \quad \text{b} \\
\text{a} \quad \text{b} \\
\text{b} \\
\end{array} \]

- you haven't
seen any a's

- "seen 1 a"

- seen 2 a's or more

Say $L = \{w \in \{1, 2\}^* | \text{end in 2}\}$

$\Sigma = \{1, 2\}$
Formally: A finite automaton is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$:

- $Q$: a set of states
- $\Sigma$: a set of symbols of alphabet of language
- $\delta: Q \times \Sigma \rightarrow Q$: "final states" "accepting states" "initial state"
- $q_0 \in Q$: "initial state"

Idea:

$(q, \sigma) \rightarrow$ what is the new state that you move to when in state $q$ you see a $\sigma$

"deterministic"

Each finite automaton (DFA), $M = (Q, \Sigma, \delta, q_0, F)$ "recognizes" $L = \{ \omega \in \Sigma^* | \text{ following the DFA, you finish at the last symbol of } \omega \text{ in a state in } F \}$

Def: $\Sigma$ alphabet, $L \subseteq \Sigma^*$, $L$ is regular iff $L$ is recognized by some finite automaton.
Otherwise we say \( L \) is non-regular.

\[ \text{Rem': DFA } M = (Q, \Sigma, \delta, q_0, F) \]

\[ \text{finite} \]

Example: Later \( L = \{ a^n b^n | m = n \} \)

is non-regular

\[ \begin{array}{c}
\text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \\
\text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \text{ } \\
a_m \rightarrow a \quad b_n \rightarrow b \\
m \quad n
\end{array} \]

Can specify a DFA by

\( 1 \) (\( Q, \Sigma, \delta, q_0, F \)) \quad \( 2 \) By graphs drawn

written out using notation in \([Sip]\)

Convention: \( \text{EVEN} = \left\{ \omega \in \{0, 1\}^* \mid \omega \text{ represents } \right\} \)

an even

number

\( = \{ 0, 2, 4, 6, 8, \ldots \} \) 

you have \( \leq ? \)
Simple for \textit{EVEN} = \{ \varepsilon, 0, 2, 4, 6, 8, 10, 12, \ldots \}.

\[ \begin{array}{c}
\text{Breakout rooms:} \\
\text{Problem 4, then problem 3} \\
of today breakout room problem}
\end{array} \]

Roughly 8-10 minutes
Problem 4:

\[ L_{\text{DN EASY}} \]

\[ = \{ \varepsilon, 0, 3, 6, 9, 00, 03, 06, 09, \ldots \} \]

Build DFA recognizing \( L_{\text{DN EASY}} \)

Q: How to modify for \( L = \{ 0, 3, 6, 9, 03, 06, \ldots \} \)
New: Survey under "Quiz"
for the canvas webpage
"Survey After Class on Sept 24"