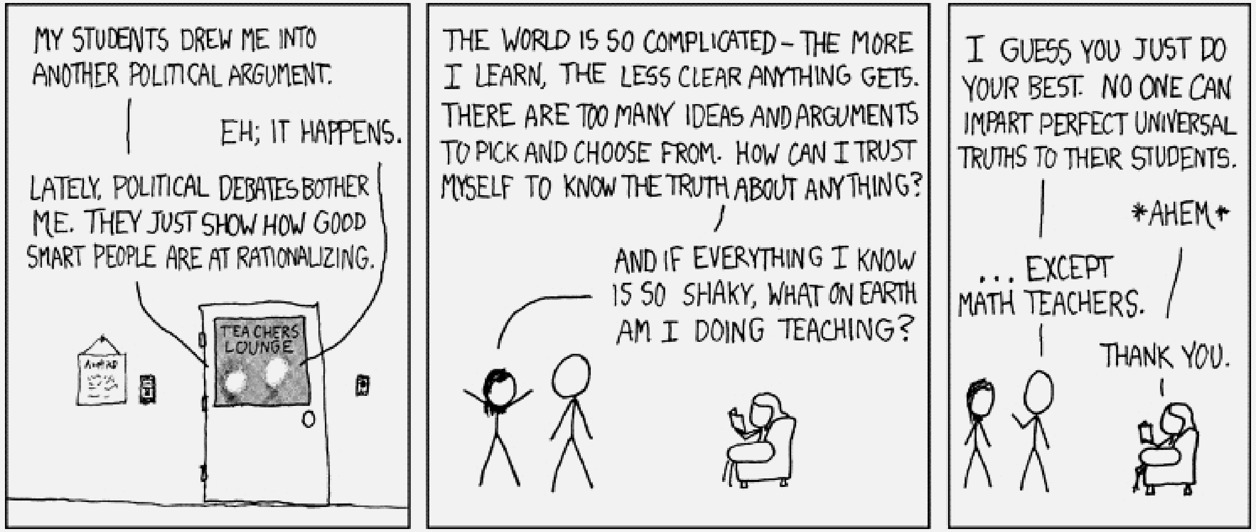
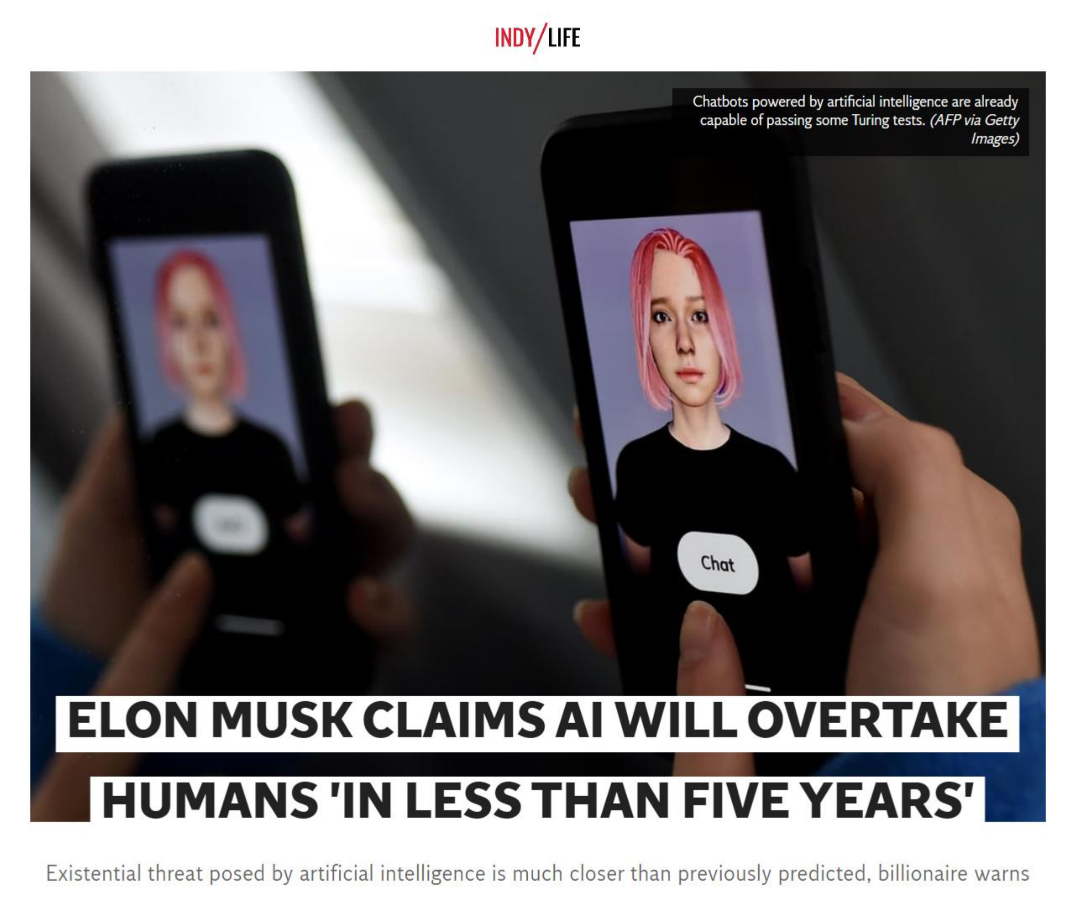
Undecidability

Mon, November 2, 2020

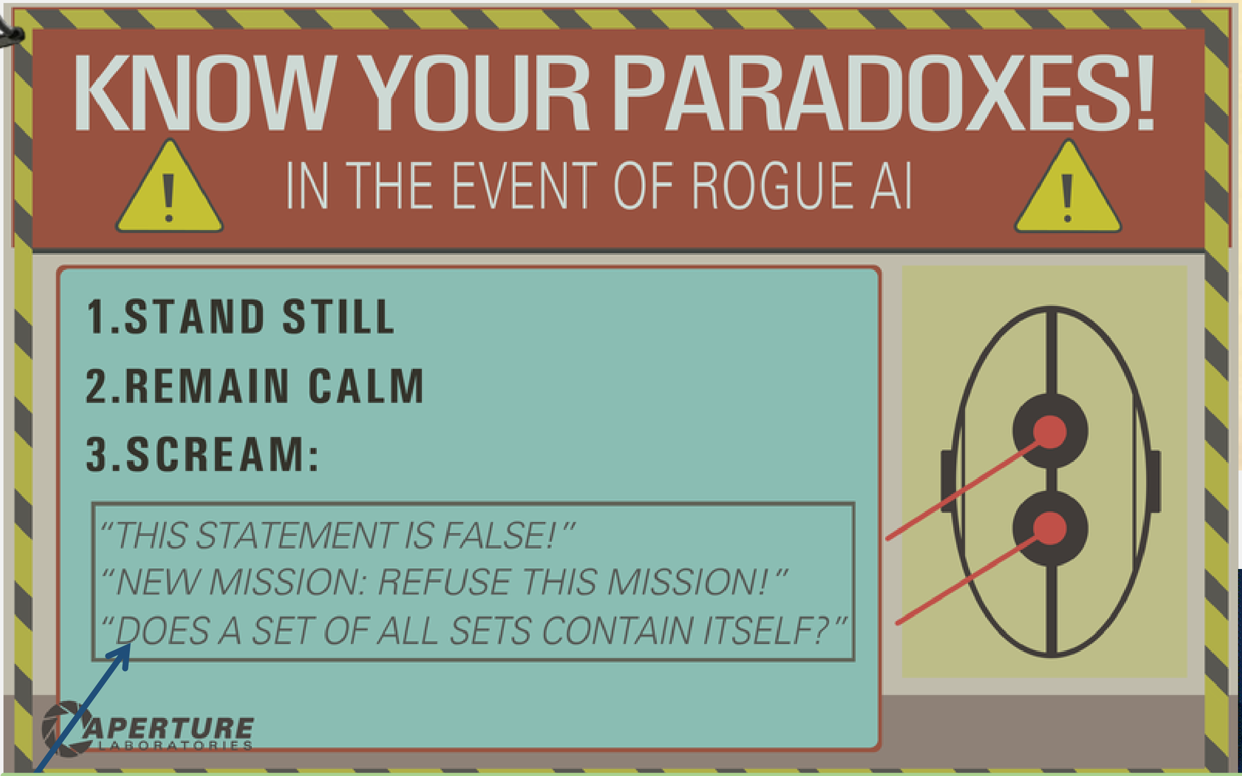


HW6 questions?

Warning: AI is Taking Over Soon

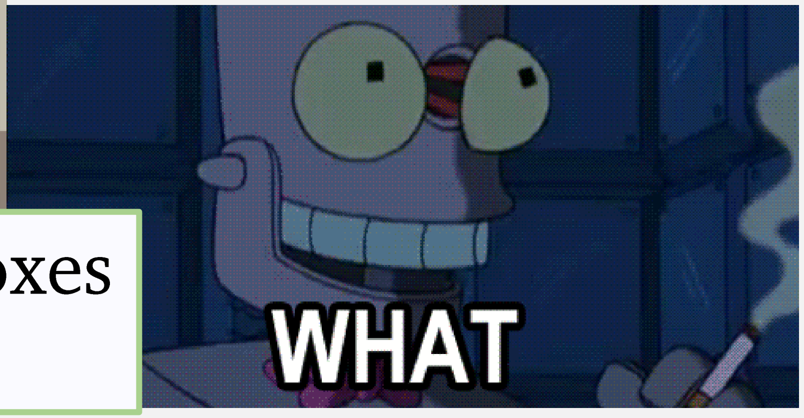


There's Hope (If You Pay Attention Today)



Bertrand Russell's Paradox (1901)

Today: A method for creating paradoxes (used by Russell and others)



Recap: Decidability of Regular and CFLs

- $A_{\mathsf{DFA}} = \{\langle B, w\rangle \mid B \text{ is a DFA that accepts input string } w\}$ Decidable
- $A_{\mathsf{NFA}} = \{\langle B, w\rangle \mid B \text{ is an NFA that accepts input string } w\}$ Decidable
- $A_{\mathsf{REX}} = \{\langle R, w\rangle \mid R \text{ is a regular expression that generates string } w\}$ Decidable
- $E_{\mathsf{DFA}} = \{\langle A\rangle \mid A \text{ is a DFA and } L(A) = \emptyset\}$ Decidable
- $EQ_{\mathsf{DFA}} = \{\langle A, B\rangle \mid A \text{ and } B \text{ are DFAs and } L(A) = L(B)\}$ Decidable
- $A_{\mathsf{CFG}} = \{\langle G, w\rangle \mid G \text{ is a CFG that generates string } w\}$ Decidable
- $E_{\mathsf{CFG}} = \{\langle G\rangle \mid G \text{ is a CFG and } L(G) = \emptyset\}$ Decidable
- $EQ_{\mathsf{CFG}} = \{\langle G, H\rangle \mid G \text{ and } H \text{ are CFGs and } L(G) = L(H)\}$ *Undecidable?*
- $A_{\mathsf{TM}} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$ *Undecidable?*

Thm: A_{TM} is Turing-recognizable

$A_{\mathsf{TM}} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$

U = "On input $\langle M, w\rangle$, where M is a TM and w is a string:

1. Simulate M on input w.

2. If M ever enters its accept state, *accept*; if M ever enters its reject state, *reject*."

- U = "run" function for TMs ("The Universal Turing Machine")
  - U Loops when M loops

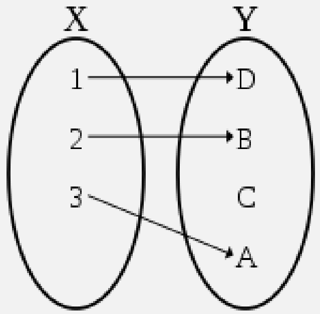
Thm: A_{TM} is undecidable

$$A_{\mathsf{TM}} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$$

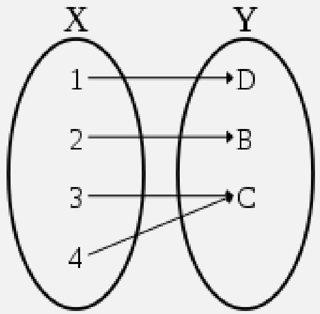
- ???

Kinds of Functions (a fn maps **Domain -> Range**)

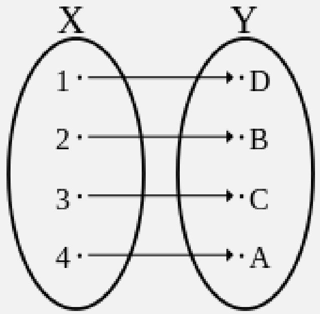
- **Injective**
  - A.k.a., “one-to-one”
  - Every element in Domain has a unique mapping
  - <u>How to remember</u>:
    - Domain is mapped “in” to the Range



- **Surjective**
  - A.k.a., “onto”
  - Every element in Range is mapped to
  - <u>How to remember</u>:
    - “Sur” = “over” (eg, survey); Domain is mapped “over” the Range



- **Bijective**
  - A.k.a., “correspondence” or “one-to-one correspondence”
  - Is both injective and surjective
  - Unique pairing of every element in Domain and Range



Countability

- A set is “countable” if it is:
  - Finite
  - Or there exists a bijection between the set and the natural numbers
    - This set then has the same size as the set of naturals numbers
    - This is called “countably infinite”

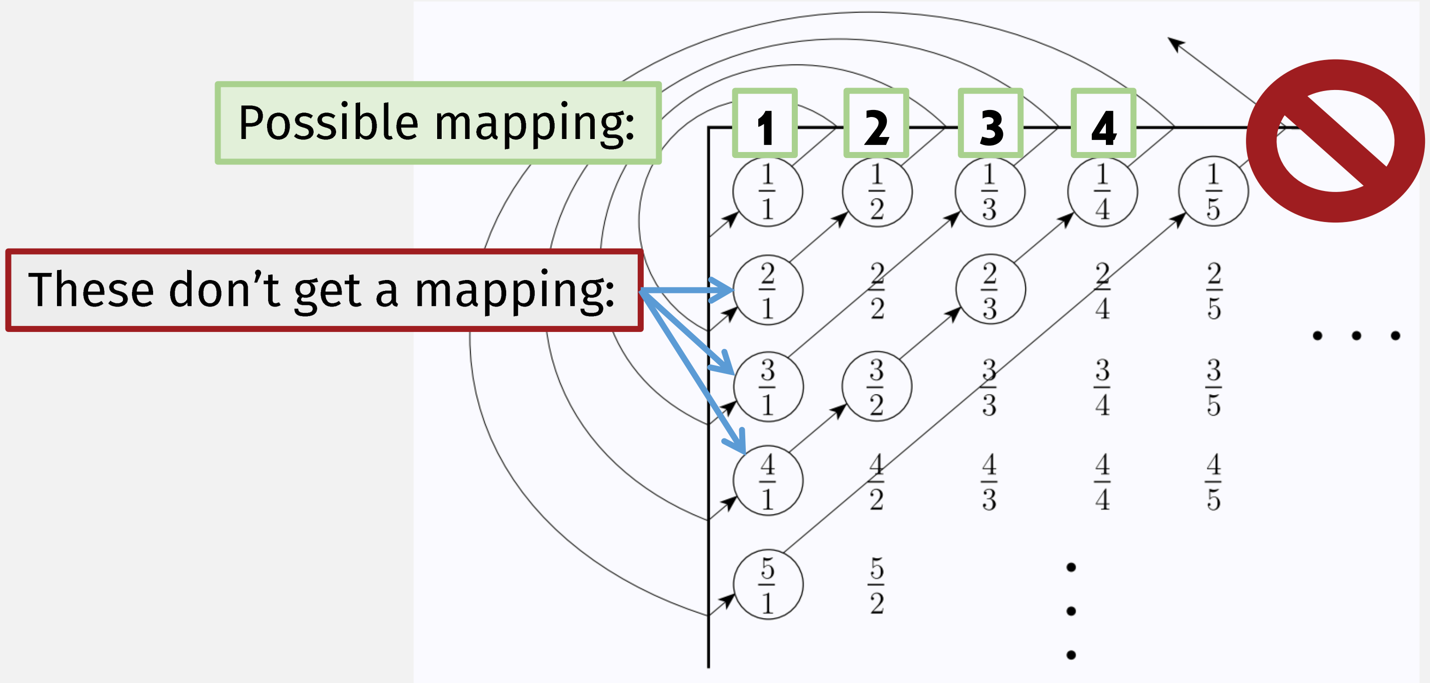
Which set is larger?

- The set of:
  - Natural numbers, or
  - Even numbers?
- They are the **same** size! Both are <u>countably infinite</u>

n	$f(n) = 2n$
1	2
2	4
3	6
$\vdots$	$\vdots$

Which set is larger?

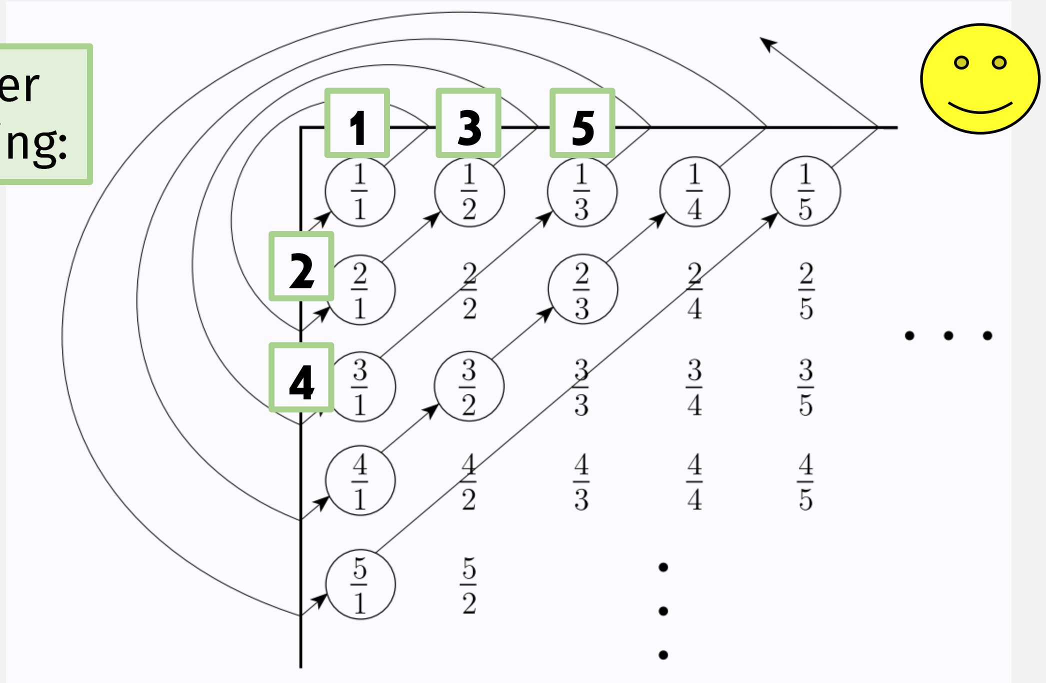
- The set of:
  - Natural numbers $\mathcal{N}$, or
  - Positive rational numbers? $\mathcal{Q} = \{\frac{m}{n} | m, n \in \mathcal{N}\}$
- They are the **same** size! Both are countably infinite



Which set is larger?

- The set of:
  - Natural numbers $\mathcal{N}$, or
  - Positive rational numbers? $\mathcal{Q} = \{\frac{m}{n} | m, n \in \mathcal{N}\}$
- They are the **same** size! Both are <u>countably infinite</u>

Another mapping:



Which set is larger?

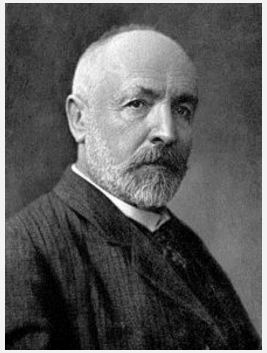
- The set of:
  - Natural numbers, or $\mathcal{N}$
  - Real numbers? $\mathcal{R}$
- There are **more** real numbers. It is <u>uncountably infinite</u>.
- Proof by contradiction:
  - Assume a bijection between natural and real numbers exists.
  - We show in any bijection, some real number is not mapped to:
  - Choose number different at each position

n	$f(n)$
1	$3.\underline{1}4159\ldots$
2	$55.5\underline{5}555\ldots$
3	$0.12\underline{3}45\ldots$
4	$0.500\underline{0}0\ldots$
$\vdots$	$\vdots$

$$x = 0.4641\ \ldots$$

  - This number is <u>not</u> included in mapping
  - Contradiction!

Georg Cantor



- Invented set theory
- Came up with countable infinity in 1873
- To show a set is uncountable: “diagonalization” technique

Diagonalization with Turing Machines

Result of Giving a TM its own Encoding as Input

All TM Encodings

	$\langle M_1 \rangle$	$\langle M_2 \rangle$	$\langle M_3 \rangle$	$\langle M_4 \rangle$	$\cdots$	$\langle D \rangle$	$\cdots$
M_1	*accept*	*reject*	*accept*	*reject*		*accept*	
M_2	*accept*	*accept*	*accept*	*accept*	$\cdots$	*accept*	$\cdots$
M_3	*reject*	*reject*	*reject*	*reject*		*reject*	
M_4	*accept*	*accept*	*reject*	*reject*		*accept*	
$\vdots$		$\vdots$			$\ddots$		
D	*reject*	*reject*	*accept*	*accept*		?	
$\vdots$		$\vdots$					$\ddots$

All TMs

Try to construct "opposite" TM

TM D can't exist!

What should happen here?

Thm: A_{TM} is undecidable

$$A_{\mathsf{TM}} = \{\langle M, w\rangle \mid M \text{ is a TM and } M \text{ accepts } w\}$$

- Proof by contradiction.
- Assume A_{TM} is decidable. Then there exists a decider:

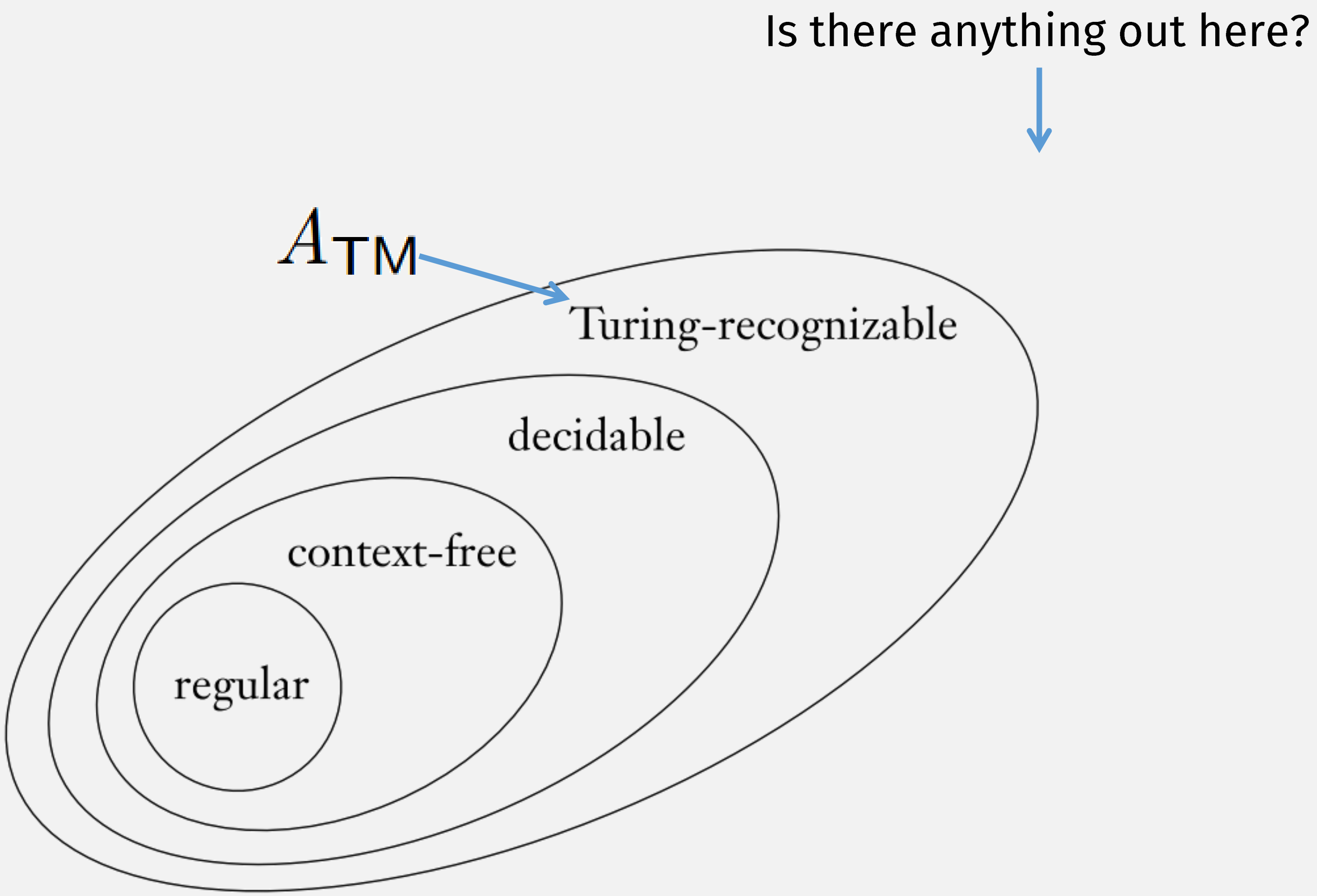
$$H(\langle M, w\rangle) = \begin{cases} \textit{accept} & \text{if } M \text{ accepts } w \\ \textit{reject} & \text{if } M \text{ does not accept } w \end{cases}$$

- If H exists, then we can create:

$D =$ "On input $\langle M\rangle$, where M is a TM:
1. Run H on input $\langle M, \langle M\rangle\rangle$.
2. Output the opposite of what H outputs. That is, if H accepts, *reject*; and if H rejects, *accept*."

- But D does not exist! Contradiction!

Turing Unrecognizable?



Thm: Some langs are not Turing-recognizable

- Lemma 1: For any alphabet Σ, the **set of all strings** in Σ^* is *countable*
  - Count strings of length 0, then
  - Count strings of length 1, …
- Lemma 2: The **set of all TMs** is *countable*
  - Because every TM M can be encoded as a string $<M>$
  - And set of all strings is countable (Lemma 1)
- Lemma 3: The **set of all infinite binary sequences** $\mathcal{B}$ is *uncountable*
  - Diagonalization proof
- Lemma 4: The **set of all languages** is *uncountable*
  - There is a mapping to $\mathcal{B}$

Mapping a Lang to a Binary Sequence

All Possible Strings (countable)	$\Sigma^* = \{\ \varepsilon,\ 0,\ 1,\ 00,\ 01,\ 10,\ 11,\ 000,\ 001,\ \cdots\ \}$
Some Language	$A = \{\ 0,\ 00,\ 01,\ 000,\ 001,\ \cdots\ \}$
Its Binary Sequence	$\chi_A = 0\ 1\ 0\ 1\ 1\ 0\ 0\ 1\ 1\ \cdots$

1 if lang has this string, 0 otherwise

Thm: Some langs are not Turing-recognizable

- Lemma 1: For any alphabet Σ, the **set of all strings** in Σ^* is *countable*
  - Count strings of length 0, then
  - Count strings of length 1, …
- Lemma 2: The **set of all TMs** is *countable*
  - Because every TM M can be encoded as a string $<M>$
  - And set of all strings is countable (Lemma 1)
- Lemma 3: The **set of all infinite binary sequences** $\mathcal{B}$ is *uncountable*
  - Diagonalization proof
- Lemma 4: The **set of all languages** is *uncountable*
  - There is a mapping to $\mathcal{B}$

Thm: Some langs are not Turing-recognizable

- Lemma 1: For any alphabet Σ, the **set of all strings** in Σ^* is *countable*
  - Count strings of length 0, then
  - Count strings of length 1, …
- Lemma 2: The **set of all TMs** is *countable*
  - Because every TM M can be encoded as a string $<M>$
  - And set of all strings is countable (Lemma 1)
- Lemma 3: The **set of all infinite binary sequences** $\mathcal{B}$ is *uncountable*
  - Diagonalization proof
- Lemma 4: The **set of all languages** is *uncountable*
  - There is a mapping to $\mathcal{B}$
- Corollary 5:
  - TMs countable, langs uncountable => some lang not recognized by a TM

Co-Turing-Recognizability

- A language is **co-Turing-recognizable** if …
- … it is the complement of a Turing-recognizable language.

Thm: Decidable ⇔ Turing & co-Turing-recognizable

- => If a language is decidable,
  then it is Turing-recognizable and co-Turing-recognizable.
  - Decidable langs are subset of recognizable langs
  - Complement is closed for decidable langs
- <= If a language is Turing- and co-Turing recognizable,
  then it is decidable.

Thm: Decidable ⇔ Turing & co-Turing-recognizable

- => If a language is decidable, then it is Turing-recognizable and co-Turing-recognizable.
  - Decidable langs are subset of recognizable langs
  - Complement is closed for decidable langs
- <= If a language is Turing- and co-Turing recognizable, then it is decidable.
  - Let $M1$ = recognizer for the language,
  - And $M2$ = recognizer for its complement
  - Decider M:
    - Run 1 step on $M1$,
    - Run 1 step on $M2$,
    - Repeat, until one machine accepts. If it's $M1$, accept. If it's $M2$, reject
  - One of $M1$ or $M2$ must accept and halt, so M halts and is a decider

A Turing-unrecognizable language

- We've proved:

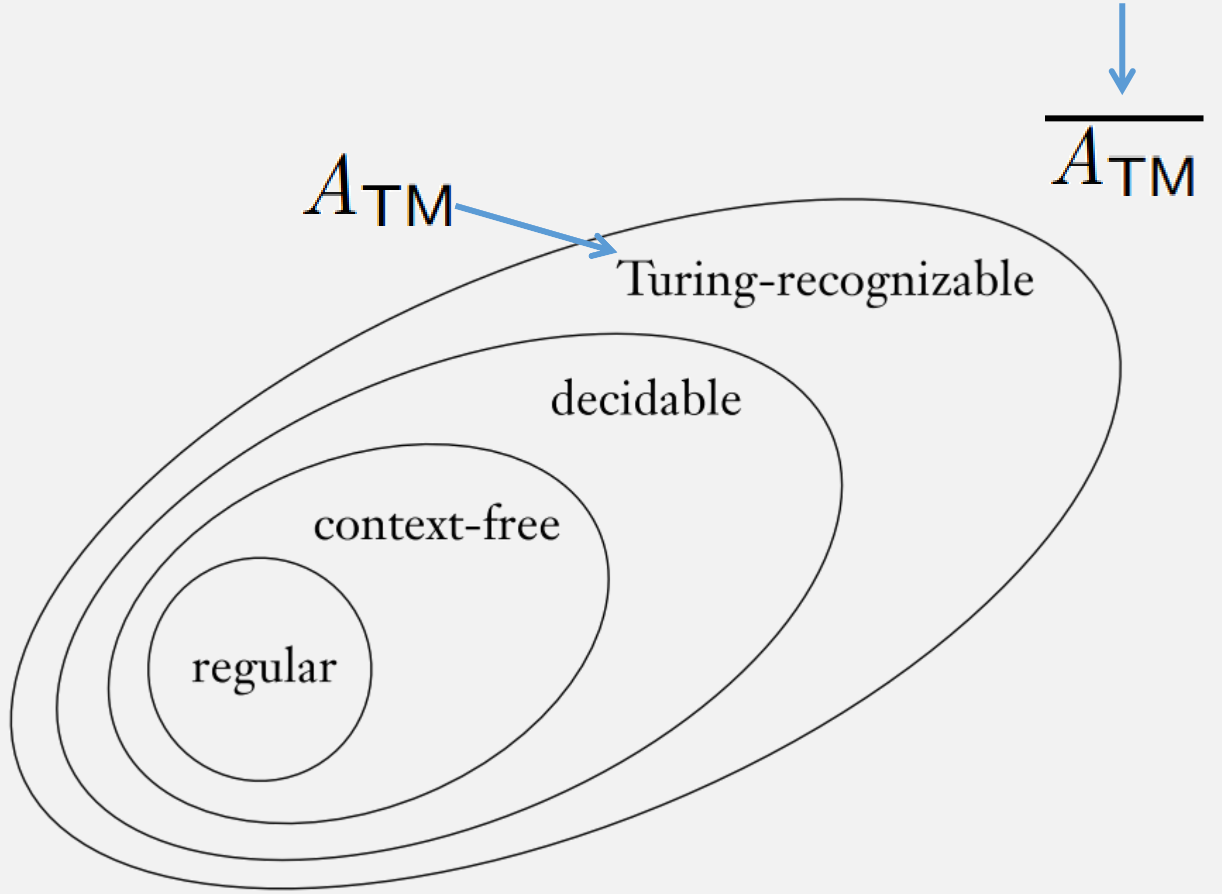
  A_{TM} is Turing-recognizable

  A_{TM} is undecidable

- So:

  $\overline{A_{\mathsf{TM}}}$ is not Turing-recognizable

Is there anything out here?



Check-in Quiz 11/2

On gradescope

End of Class Survey 11/2

See course website