The Halting Problem

\[ A_{TM} = \{ <M,w> \mid M \text{ is a TM that accepts string } w \} \]

- Turing-recognizable? \text{YES}
- Turing-decidable?
  NO \text{we'll see}

write a Python function that
- input:
  a description of a TM M
  and a string w
- output True if w accepted by M
  (enough for T-recognizable)
- output True if w accepted by M
  False o/w
  \rightarrow for T-decidable

intuition:
don't know when to say False

T-decidable
T-recognizable

TM
A bit about infinite sets and their sizes (diagonalization):

Def 4.12: Let $A, B$ be sets and let $f: A \to B$. We say that $f$ is
- **one-to-one** if $f(a) \neq f(b)$ for every $a \neq b$
- **onto** if for every $b \in B$ there exists $a \in A$ such that $f(a) = b$

If $f$ is one-to-one and onto, then $A, B$ are the **same size** and $f$ is called **correspondence**.

Example: $\mathbb{N} = \{1,2,3,4,5,\ldots\}$ and $\{2,4,6,8,\ldots\}$

$f: \mathbb{N} \to \{2,4,6,8,\ldots\}$ $\quad f(x) = 2x$
The Halting Problem

A bit about infinite sets and their sizes (diagonalization):

**Def 4.12:** Let \( A, B \) be sets and let \( f: A \rightarrow B \). We say that \( f \) is
- **one-to-one** if \( f(a) \neq f(b) \) for every \( a \neq b \)
- **onto** if for every \( b \in B \) there exists \( a \in A \) such that \( f(a) = b \)

If \( f \) is one-to-one and onto, then \( A, B \) are the **same size** and \( f \) is called **correspondence**.

**Example:** \( \mathbb{N} = \{1, 2, 3, 4, 5, \ldots\} \) and \( \{2, 4, 6, 8, \ldots\} \)

**Def 4.14:** A set is **countable** if it is finite or has the same size as \( \mathbb{N} \).
Are \( \mathbb{Q} \) (rational numbers) and \( \mathbb{R} \) (real numbers) countable?

- **YES**

Consider \( \mathbb{Q}_{\geq 0} \):

- integers
- \( \frac{1}{2} \) that is not an integer
- \( \frac{2}{3} \) that has not been used above
- \( \frac{3}{4} \) — 11

We need to describe \( f: N \to \mathbb{Q}_{\geq 0} \) one-on-one onto

\[ Q_{\geq 0} \text{ same size as } N \]

Now, \( f \) is defined by the yellow line.
The Halting Problem

[Section 4.2]

Are \( \mathbb{Q} \) (rational numbers) and \( \mathbb{R} \) (real numbers) countable?

No

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

\( \rightarrow \) we have one-on-one onto \( f : \mathbb{N} \rightarrow \mathbb{R} \)

\[
\begin{align*}
f(0) &= 3.01415 \\
f(1) &= 2.58556 \\
f(2) &= 11.400 \\
f(3) &= \ldots
\end{align*}
\]

we will construct a number that is not being mapped to by \( f \):

0.261 etc.

\( \uparrow \)

differ from the first decimal digit in \( f(0) \)
diff. from the second dec. digit in \( f(1) \)
the i-th dec. digit diff than i-th dec. digit in \( f(i-1) \)

\( \Rightarrow \) number not mapped to \( \Rightarrow \) \( f \) not onto.
Cor 4.18: There is a language that is not Turing-recognizable.

Idea:
- The set of Turing-recognizable languages is countable.
  - Every Turing-recognizable language can be described by a TM.
  - Every TM can be described by a string of 0's and 1's.

- The set of all strings is countable.

Just list the strings in lexicographical order:
- \( \emptyset, 0, 1, 00, 10, 11, 000, 001, 010, 011, \ldots \)

Set of all languages is uncountable.
- Suppose not.
- We have a function \( f \) onto one-on-one:
  \[ N \rightarrow \text{all languages} \]
- \( f(0) = \{ \emptyset, 0, 1, 111 \} \)
- \( f(1) = \{ 0, 00 \} \)
- \( f(2) = \{ 0, 111, 1111, \ldots \} \)

We construct a language that is not being mapped by \( f \).
- \( f(0) \) includes \( \emptyset \), my lang will not.
- \( f(0) \) includes 0, my lang will not.
The Halting Problem

Thm 4.11: $A_{TM}$ is not decidable.

Recall: $A_{TM} = \{ <M,w> \mid M$ is a TM that accepts string $w \}$

Recall that $A_{TM}$ is $T$-recognizable.

Here we say NOT $T$-decidable.

Pf: by contradiction, suppose decidable, i.e. we have a func (TM that always finishes) that decides whether a given $M$ accepts a given $w$.

Let's create a func weird ($M$):

- accept if $M$ does not accept $M$ (return True)
- reject if $M$ does accept $M$ (return False)

What happens if: weird (weird)

- accepts if weird does not accept weird
- rejects if weird does accept weird

Let's call it decideAccept ($M,w$)

Weird does not run $M$, it simply calls decideAccept ($M,M$)

Def weird ($M$):

- return not decideAccept ($M,M$)

// contradiction - decideAccept cannot exist!
Thm 4.22: A language $L$ is decidable iff $L$ is Turing-recognizable and $L^c$ is Turing-recognizable (we say that $L$ is co-Turing-recognizable).

If: $\Rightarrow$ want: if $L$ is T-decidable then $L$ is T-recognizable and $L^c$ is T-recognizable.

\[ \Rightarrow \text{immediately follows: T-recognizable} \]

\[ \text{To show } L \text{ is T-recognizable: take } T \text{ and switch } q_{accept} \text{ and } q_{reject}. \]

$\Leftrightarrow$ if I have a TM $T_1$ for $L$ and $T_2$ for $L^c$ then we want to construct $T$ for $L$ s.t. $T$ always gets to either $q_{accept}$ or $q_{reject}$.

Let's suppose that $T$ has 2 tapes (equivalent to regular TM)

- Tape 1: simulate $T_1$
- Tape 2: simulate $T_2$

}\simultaneously$

$\Rightarrow$ if $T_1$ accepts, accept; if $T_2$ accepts, reject $\Leftrightarrow$ one of them has $q_{accept}$, i.e. we always either accept or reject.