Will it ever stop?

\( A_{TM} = \{ <M, w> | M \text{ is a TM and } M \text{ accepts } w \} \)

–undecidable

–remember, **decidable** means that the TM will eventually reach an accept or reject state;

–it will halt

–U is a Universal TM

–TM U recognizes \( A_{TM} \):

•1. Simulate M on input w with U

•2. If M accepts then U accepts; if M rejects then U rejects; *if M never halts then U never halts*

•If we could get U to halt, then we could get M to halt
Counting

• Diagonalization
  – how can we determine if two infinite sets are the same size? (Georg Cantor)
  – cannot just count them up
  – the two sets are the same size if the elements of one set can be paired with the elements from the other set (no counting!)
  – define a function as a correspondence

Correspondence

• A and B are sets, F is a function from A to B; F: A → B
  – F is one-to-one if it never maps two different elements to the same place, if F(a) ≠ F(c) whenever a ≠ c
  – F is onto if it hits every element of B, for each b ∈ B this is an a ∈ A such that F(a) = b
  – A and B are the same size if there is a one-to-one, onto function F
  – F is a correspondence
Application

• Let N be the set of natural numbers, let E be the set of even natural numbers.

• If we can find a correspondence function between these two infinite sets, they are the same size \( F: N \to E \)
  \[ f(n) = ? \]

• Definition: a set is countable if it is finite or in correspondence with the set of natural numbers

Diagonalization

• Is \( Q = \{ \frac{m}{n} \mid m,n \in N \} \) countable?
  – What is Q?
  – We can make a list of all the elements in Q, and match them with the elements in N

<table>
<thead>
<tr>
<th>1/1</th>
<th>1/2</th>
<th>1/3</th>
<th>1/4</th>
<th>1/5</th>
</tr>
</thead>
<tbody>
<tr>
<td>2/1</td>
<td>2/2</td>
<td>2/3</td>
<td>2/4</td>
<td>2/5</td>
</tr>
<tr>
<td>3/1</td>
<td>3/2</td>
<td>3/3</td>
<td>3/4</td>
<td>3/5</td>
</tr>
<tr>
<td>4/1</td>
<td>4/2</td>
<td>4/3</td>
<td>4/4</td>
<td>4/5</td>
</tr>
</tbody>
</table>
What could ever be uncountable?

- The set of Real Numbers, \( \mathbb{R} \)
- Proof by contradiction

- Assume \( \mathbb{R} \) is countable
- There must exist a correspondence function \( f \) with the set \( \mathbb{N} \)
- Find some number \( x \in \mathbb{R} \) that is not paired with a number \( p \in \mathbb{N} \)
- We will construct this number \( x \)

Real Numbers are uncountable

- Assume \( F \) exists
- Construct \( x \) such \( x \neq f(p) \) for any \( p \)
- \( x \) is between 0 and 1
- Ensure \( x \neq f(1) \), set the 10\(^{th}\)s’ place to 4
- Ensure \( x \neq f(2) \), set the 100\(^{th}\)s’ place to 6
- Forever….
- Never select 0 or 9 since \( .1999… = .2000 \)
- We know \( x \neq f(p) \) for any \( p \) since \( x \) differs from \( f(p) \) in the \( p^{th} \) decimal place
Why do we care?

- Some languages are not TM recognizable
  - show that the set of all TMs is countable
- Each TM recognizes exactly one language
  - show that the set of all languages is not countable
  - some language must not match to a TM
  - for a finite alphabet, $\Sigma, \Sigma^*$ is countable
- A finite set of strings of each length

Some languages are not TM recog.

- show that the set of all TMs, $T$, is countable
- show that the set of all languages, $L$, is not
The Halting Problem, Proof

• \( A_{\text{TM}} = \{ <M, w> | M \text{ is a TM and } M \text{ accepts } w \} \)
  – undecidable, may never halt

  – assume \( A_{\text{TM}} \) is decidable and that \( H \) is a TM decider (always halts) for \( A_{\text{TM}} \)
  – on input \( <M,w> \):

    \[
    H(<M,w>) \begin{cases} 
    \text{accept if } M \text{ accepts } w \\
    \text{rejects if } M \text{ does not accept } w 
    \end{cases}
    \]

The Halting Problem, Proof, cont.

• Construct a TM, \( D \), with \( H \) as subroutine.
• \( D \) calls \( H \) to determine what \( M \) does when input is \( M \)'s encoding. \( D \) does the opposite.
  – \( D = \) On input \( <M>, \text{ where } M \text{ is a TM} \)
  1) Run \( H \) on \( <M, <M>> \)
  2) If \( H \) accepts, reject. If \( H \) rejects, accept.

  \[
  D(<D>) \begin{cases} 
  \text{accept if } D \text{ does not accept } <D> \\
  \text{reject if } D \text{ accepts } <D> 
  \end{cases}
  \]

Contradiction! We can use diagonalization to explore this further
Encode TM as string

• Assume $\Sigma = \{0, 1\}; \Gamma = \{0, 1, \nabla\}$
• Encode elements of $\delta$ using 1s

$\delta(q_i, x) = (q_j, y, M)$ is

• $\text{en}(q_0)0\text{en}(x)0\text{en}(q_j)0\text{en}(y)0\text{en}(M)$

– two 0s separate transitions,
  beginning and end marked with 000
  $q_0$ is start
  $q_1$ is accept
  $q_{n-1}$ is reject

• We could build a TM to check to see if a string is a legal encoding of a deterministic TM
  – what does that language look like?