CS70: Countability and Uncountability

Alex Psomas

June 30, 2016
Warning!

Warning:
Warning: I’m really loud!
Today.

One idea, from around 130 years ago.

At the heart of set theory.

Started a crisis in mathematics in the middle of the previous century!

The man who worked on this was described as:

- Genious?
- Renegade?
- Corrupter of youth?
- The King in the North?
Today.

One idea, from around 130 years ago.
Today.

One idea, from around 130 years ago.
At the heart of set theory.
One idea, from around 130 years ago.
At the heart of set theory.
Started a crisis in mathematics in the middle of the previous century!
One idea, from around 130 years ago.
At the heart of set theory.
Started a crisis in mathematics in the middle of the previous century!!!!!
Today.

One idea, from around 130 years ago.
At the heart of set theory.
Started a crisis in mathematics in the middle of the previous century!!!!!
The man who worked on this was described as:
One idea, from around 130 years ago.
At the heart of set theory.
Started a crisis in mathematics in the middle of the previous century!!!!!
The man who worked on this was described as:

- Genious?
Today.

One idea, from around 130 years ago.

At the heart of set theory.

Started a crisis in mathematics in the middle of the previous century!!!!!

The man who worked on this was described as:

- Genious?
- Renegade?
Today.

One idea, from around 130 years ago.
At the heart of set theory.
Started a crisis in mathematics in the middle of the previous century!!!!
The man who worked on this was described as:

- Genious?
- Renegade?
- Corrupter of youth?
Today.

One idea, from around 130 years ago.
At the heart of set theory.
Started a crisis in mathematics in the middle of the previous century!!!!!

The man who worked on this was described as:

- Genious?
- Renegade?
- Corrupter of youth?
- The King in the North?
One idea, from around 130 years ago.
At the heart of set theory.
Started a crisis in mathematics in the middle of the previous century!!!!!
The man who worked on this was described as:

- Genious?
- Renegade?
- Corrupter of youth?
- The King in the North?
The idea:

More than one infinities!!!!!!

The man:
Georg Cantor
The idea.

The idea: **More than one infinities!!!!!!**
The idea.

The idea: **More than one infinities!!!!!!**

The man:
The idea: *More than one infinities!!!!!!*

The man:

Georg Cantor
Life before Cantor

How many elements in \{1, 2, 4\}?

3

How many elements in \{1, 2, 4, 10, 13, 18\}?

6

How many primes?

Infinite!

How many elements in \mathbb{N}\{0\}?

Infinite!

How many elements in \mathbb{Z}?

Infinite!

How many elements in \mathbb{R}?

Infinite!

What is this infinity though?
The symbol you write after taking a limit....

Don't think about it....

Even Gauss: “I protest against the use of infinite magnitude as something completed, which is never permissible in mathematics. Infinity is merely a way of speaking, the true meaning being a limit which certain ratios approach indefinitely close, while others are permitted to increase without restriction.”
Life before Cantor

How many elements in \( \{1, 2, 4\} \)?
Life before Cantor

How many elements in \( \{1, 2, 4\} \)? 3
Life before Cantor

How many elements in \( \{1, 2, 4\} \)? 3
How many elements in \( \{1, 2, 4, 10, 13, 18\} \)?
Life before Cantor

How many elements in \(\{1, 2, 4\}\)? 3
How many elements in \(\{1, 2, 4, 10, 13, 18\}\)? 6
Life before Cantor

How many elements in \( \{1, 2, 4\} \)? 3

How many elements in \( \{1, 2, 4, 10, 13, 18\} \)? 6

How many primes?
Life before Cantor

How many elements in \{1, 2, 4\}? 3
How many elements in \{1, 2, 4, 10, 13, 18\}? 6
How many primes? Infinite!
Life before Cantor

How many elements in \( \{1, 2, 4\} \)? 3
How many elements in \( \{1, 2, 4, 10, 13, 18\} \)? 6
How many primes? Infinite!
How many elements in \( \mathbb{N} \)?
Life before Cantor

How many elements in \{1,2,4\}? 3
How many elements in \{1,2,4,10,13,18\}? 6
How many primes? Infinite!
How many elements in $\mathbb{N}$? Infinite!
Life before Cantor

How many elements in \{1, 2, 4\}? 3
How many elements in \{1, 2, 4, 10, 13, 18\}? 6
How many primes? Infinite!
How many elements in \(\mathbb{N}\)? Infinite!
How many elements in \(\mathbb{N} \setminus \{0\}\)?
Life before Cantor

How many elements in \(\{1, 2, 4\}\)? 3
How many elements in \(\{1, 2, 4, 10, 13, 18\}\)? 6
How many primes? Infinite!
How many elements in \(\mathbb{N}\)? Infinite!
How many elements in \(\mathbb{N} \setminus \{0\}\)? Infinite!
Life before Cantor

How many elements in \{1, 2, 4\}? 3
How many elements in \{1, 2, 4, 10, 13, 18\}? 6
How many primes? Infinite!
How many elements in \(\mathbb{N}\)? Infinite!
How many elements in \(\mathbb{N} \setminus \{0\}\)? Infinite!
How many elements in \(\mathbb{Z}\)?
Life before Cantor

How many elements in \( \{1, 2, 4\} \)? 3
How many elements in \( \{1, 2, 4, 10, 13, 18\} \)? 6
How many primes? Infinite!
How many elements in \( \mathbb{N} \)? Infinite!
How many elements in \( \mathbb{N} \setminus \{0\} \)? Infinite!
How many elements in \( \mathbb{Z} \)? Infinite!
Life before Cantor

How many elements in \( \{1, 2, 4\} \)?) 3
How many elements in \( \{1, 2, 4, 10, 13, 18\} \)?) 6
How many primes? Infinite!
How many elements in \( \mathbb{N} \)?) Infinite!
How many elements in \( \mathbb{N} \setminus \{0\} \)?) Infinite!
How many elements in \( \mathbb{Z} \)?) Infinite!
How many elements in \( \mathbb{R} \)?
Life before Cantor

How many elements in \{1, 2, 4\}? 3
How many elements in \{1, 2, 4, 10, 13, 18\}? 6
How many primes? Infinite!
How many elements in \(\mathbb{N}\)? Infinite!
How many elements in \(\mathbb{N} \setminus \{0\}\)? Infinite!
How many elements in \(\mathbb{Z}\)? Infinite!
How many elements in \(\mathbb{R}\)? Infinite!
Life before Cantor

How many elements in \( \{1, 2, 4\} \)? 3
How many elements in \( \{1, 2, 4, 10, 13, 18\} \)? 6
How many primes? Infinite!
How many elements in \( \mathbb{N} \)? Infinite!
How many elements in \( \mathbb{N} \setminus \{0\} \)? Infinite!
How many elements in \( \mathbb{Z} \)? Infinite!
How many elements in \( \mathbb{R} \)? Infinite!
What is this infinity though?
Life before Cantor

How many elements in \( \{1, 2, 4\} \)? 3
How many elements in \( \{1, 2, 4, 10, 13, 18\} \)? 6
How many primes? Infinite!
How many elements in \( \mathbb{N} \)? Infinite!
How many elements in \( \mathbb{N} \setminus \{0\} \)? Infinite!
How many elements in \( \mathbb{Z} \)? Infinite!
How many elements in \( \mathbb{R} \)? Infinite!
What is this infinity though?
The symbol you write after taking a limit....
Life before Cantor

How many elements in \( \{1, 2, 4\} \)? 3
How many elements in \( \{1, 2, 4, 10, 13, 18\} \)? 6
How many primes? Infinite!
How many elements in \( \mathbb{N} \)? Infinite!
How many elements in \( \mathbb{N} \setminus \{0\} \)? Infinite!
How many elements in \( \mathbb{Z} \)? Infinite!
How many elements in \( \mathbb{R} \)? Infinite!
What is this infinity though?
The symbol you write after taking a limit....
Don’t think about it....
Life before Cantor

How many elements in \{1,2,4\}? 3
How many elements in \{1,2,4,10,13,18\}? 6
How many primes? Infinite!
How many elements in \(\mathbb{N}\)? Infinite!
How many elements in \(\mathbb{N} \setminus \{0\}\)? Infinite!
How many elements in \(\mathbb{Z}\)? Infinite!
How many elements in \(\mathbb{R}\)? Infinite!

What is this infinity though?
The symbol you write after taking a limit....
Don’t think about it....

Even Gauss: “I protest against the use of infinite magnitude as something completed, which is never permissible in mathematics. Infinity is merely a way of speaking, the true meaning being a limit which certain ratios approach indefinitely close, while others are permitted to increase without restriction.”
Cantor’s questions

Is $\mathbb{N}$ smaller than $\mathbb{N}$?

Is $\mathbb{N}$ smaller than $\mathbb{Z}$?

What about $\mathbb{Z}^2$?

Is $\mathbb{N}$ smaller than $\mathbb{R}$?
Cantor’s questions

Is $\mathbb{N} \setminus \{0\}$ smaller than $\mathbb{N}$?
Cantor’s questions

Is $\mathbb{N} \setminus \{0\}$ smaller than $\mathbb{N}$?
Is $\mathbb{N}$ smaller than $\mathbb{Z}$?
Cantor’s questions

Is \( \mathbb{N} \setminus \{0\} \) smaller than \( \mathbb{N} \)?

Is \( \mathbb{N} \) smaller than \( \mathbb{Z} \)? What about \( \mathbb{Z}^2 \)?
Cantor’s questions

Is \( \mathbb{N} \setminus \{0\} \) smaller than \( \mathbb{N} \)?
Is \( \mathbb{N} \) smaller than \( \mathbb{Z} \)? What about \( \mathbb{Z}^2 \)?
Is \( \mathbb{N} \) smaller than \( \mathbb{R} \)?
Hilbert’s hotel

A hotel with infinite rooms.

Rooms are numbered from 1 to infinity.

Every room is occupied.

Room $i$ has guest $G_i$.

$G_0$ shows up. What do we do?

Move $G_1$ to room number 2.
Hilbert’s hotel

A hotel with infinite rooms.
A hotel with infinite rooms. Rooms are numbered from 1 to infinity.
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied.
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$. 

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$. 

\begin{center}
\begin{tabular}{cccc}
$G_1$ & $G_2$ & $G_3$ & $G_4$ \\
\end{tabular}
\end{center}
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room \( i \) has guest \( G_i \).

\[ \begin{array}{cccccc}
G_0 & G_1 & G_2 & G_3 & G_4 & \ldots \\
\end{array} \]

\( G_0 \) shows up. What do we do?
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$.

$G_0$ shows up. What do we do?
Move $G_1$ to room number 2.
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$. 

$G_0$ $\rightarrow$ $G_1$ $\rightarrow$ $G_3$ $\rightarrow$ $G_4$ $\rightarrow$ $\ldots$ 

$G_2$
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$.

$G_0 \rightarrow G_1 \rightarrow G_3 \rightarrow G_4 \rightarrow \ldots$

Move $G_2$ to room number 3.
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$.
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$.

Move $G_3$ to room number 4.
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$.

$G_0 \rightarrow G_1 \rightarrow G_2 \rightarrow G_3 \rightarrow \ldots$

And so on.
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room \(i\) has guest \(G_i\).

And so on.

Now \(G_0\) can go to room number 1!!
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$.

$G_0 \rightarrow G_1 \rightarrow G_2 \rightarrow G_3 \rightarrow \cdots$

And so on.

Now $G_0$ can go to room number 1!!
Hilbert’s hotel

A hotel with infinite rooms. Rooms are numbered from 1 to infinity. Every room is occupied. Room $i$ has guest $G_i$.

![Diagram of rooms and guests]

And so on.

Now $G_0$ can go to room number 1!!
Moral of the story

Number of rooms: $N \{0\}$

Number of guests: $N \{0\}$ is not smaller than $N \{0\}$.

$N \{0\}$ is not bigger than $N \{0\}$.

Why?

Because it's a subset.

Therefore, $N \{0\}$ must have the same number of elements as $N \{0\}$.

Is this a proof?

How would we show this formally???
Moral of the story

Number of rooms:
Moral of the story

Number of rooms: $\mathbb{N} \setminus \{0\}$
Moral of the story

Number of rooms: \( \mathbb{N} \setminus \{0\} \)
Number of guests:
Moral of the story

Number of rooms: \( \mathbb{N} \setminus \{0\} \)
Number of guests: \( \mathbb{N} \)
Moral of the story

Number of rooms: $\mathbb{N} \setminus \{0\}$
Number of guests: $\mathbb{N}$

$\mathbb{N} \setminus \{0\}$ is not smaller than $\mathbb{N}$. Why? Because it's a subset. Therefore, $\mathbb{N} \setminus \{0\}$ must have the same number of elements as $\mathbb{N}$. How would we show this formally???
Moral of the story

Number of rooms: $\mathbb{N} \setminus \{0\}$
Number of guests: $\mathbb{N}$

$\mathbb{N} \setminus \{0\}$ is not smaller than $\mathbb{N}$.

$\mathbb{N} \setminus \{0\}$ is not bigger than $\mathbb{N}$. 
Number of rooms: $\mathbb{N} \setminus \{0\}$
Number of guests: $\mathbb{N}$

$\mathbb{N} \setminus \{0\}$ is **not** smaller than $\mathbb{N}$.

$\mathbb{N} \setminus \{0\}$ is not bigger than $\mathbb{N}$. Why?
Moral of the story

Number of rooms: $\mathbb{N} \setminus \{0\}$
Number of guests: $\mathbb{N}$

$\mathbb{N} \setminus \{0\}$ is not smaller than $\mathbb{N}$.

$\mathbb{N} \setminus \{0\}$ is not bigger than $\mathbb{N}$. Why? Because it’s a subset.
Moral of the story

Number of rooms: \( \mathbb{N} \setminus \{0\} \)
Number of guests: \( \mathbb{N} \)

\( \mathbb{N} \setminus \{0\} \) is **not** smaller than \( \mathbb{N} \).

\( \mathbb{N} \setminus \{0\} \) is not bigger than \( \mathbb{N} \). Why? Because it’s a subset.

Therefore, \( \mathbb{N} \setminus \{0\} \) must have the same number of elements as \( \mathbb{N} \).
Moral of the story

Number of rooms: $\mathbb{N} \setminus \{0\}$
Number of guests: $\mathbb{N}$

$\mathbb{N} \setminus \{0\}$ is **not** smaller than $\mathbb{N}$.

$\mathbb{N} \setminus \{0\}$ is not bigger than $\mathbb{N}$. Why? Because it's a subset.

Therefore, $\mathbb{N} \setminus \{0\}$ must have the same number of elements as $\mathbb{N}$.

Is this a proof?
Number of rooms: $\mathbb{N}\setminus\{0\}$
Number of guests: $\mathbb{N}$

$\mathbb{N}\setminus\{0\}$ is **not** smaller than $\mathbb{N}$.

$\mathbb{N}\setminus\{0\}$ is not bigger than $\mathbb{N}$. Why? Because it’s a subset.

Therefore, $\mathbb{N}\setminus\{0\}$ must have the same number of elements as $\mathbb{N}$.

Is this a proof? How would we show this formally???
Countable.

Definition:
$S$ is countable if there is a bijection between $S$ and some subset of $\mathbb{N}$.
If the subset of $\mathbb{N}$ is finite, $S$ has finite cardinality.
If the subset of $\mathbb{N}$ is infinite, $S$ is countably infinite.
Definition: S is **countable** if there is a bijection between S and some subset of \( \mathbb{N} \).
Definition: $S$ is **countable** if there is a bijection between $S$ and some subset of $\mathbb{N}$.

If the subset of $\mathbb{N}$ is finite, $S$ has finite **cardinality**.
Countable.

Definition: S is countable if there is a bijection between S and some subset of \( \mathbb{N} \).

If the subset of \( \mathbb{N} \) is finite, S has finite cardinality.

If the subset of \( \mathbb{N} \) is infinite, S is countably infinite.
Bijections?

- **Bijection:** one to one and onto.
- **Onto:** not a function.
Bijections?

One to one.

- **X** to **Y**:
  - 1 → D
  - 2 → B
  - 3 → C

- **X** to **Y**:
  - 1 \cdot → D
  - 2 \cdot → B
  - 3 \cdot → C
  - 4 \cdot → A

- **X** to **Y**:
  - 1 → D
  - 2 → B
  - 3 → C
  - 4 → A

- **X** to **Y**:
  - 1 → D
  - 2 → B
  - 3 → C
  - 4 → A
Bijections?

One to one.

Bijection: one to one and onto.

\[ X \rightarrow Y \]
\[ 1 \rightarrow D \]
\[ 2 \rightarrow B \]
\[ 3 \rightarrow C \]
\[ 4 \rightarrow A \]

\[ X \rightarrow Y \]
\[ 1 \rightarrow D \]
\[ 2 \rightarrow B \]
\[ 3 \rightarrow C \]
\[ 4 \rightarrow A \]
Bijections?

One to one.

Bijection: one to one and onto.

Onto.
Bijections?

One to one.  Bijection: one to one and onto.

Onto.  Not a function.
Countable.

- Enumerable means countable.
Countable.

- Enumerable means countable.
- Subsets of countable sets are countable.
Countable.

- Enumerable means countable.
- Subsets of countable sets are countable.
  For example the set \( \{14, 54, 5332, 10^{12} + 4\} \) is countable.
Countable.

- Enumerable means countable.

- Subsets of countable sets are countable. For example the set \( \{14, 54, 5332, 10^{12} + 4\} \) is countable. (It has 4 elements)
Countable.

- Enumerable means countable.
- Subsets of countable sets are countable.
  For example the set \{14, 54, 5332, 10^{12} + 4\} is countable. (It has 4 elements) Even numbers are countable.
Enumerable means countable.

Subsets of countable sets are countable. For example, the set \(\{14, 54, 5332, 10^{12} + 4\}\) is countable. (It has 4 elements) Even numbers are countable. Prime numbers are countable.
Countable.

- Enumerable means countable.
- Subsets of countable sets are countable.
  For example, the set \( \{14, 54, 5332, 10^{12} + 4\} \) is countable. (It has 4 elements) Even numbers are countable. Prime numbers are countable. Multiples of 3 are countable.
Countable.

- Enumerable means countable.
- Subsets of countable sets are countable. For example the set \( \{14, 54, 5332, 10^{12} + 4\} \) is countable. (It has 4 elements) Even numbers are countable. Prime numbers are countable. Multiples of 3 are countable.
- All countably infinite sets have the same cardinality as each other.
Back to Hilbert’s hotel

Where's the function?

We want a bijection from:

\[
N \{0\} \to N
\]

\[
f(x) = x - 1.
\]

Maps every number from \(N \{0\}\) to a number in \(N\), and every number in \(x \in N\) has exactly one number \(y \in N \{0\}\) such that \(f(y) = x\).

What if we had a bijection from \(N\) to \(N \{0\}\)?

Same thing! Bijection means that the sets have the same size.

Invert it and you'll get a bijection from \(N \{0\}\) to \(N\).
Back to Hilbert’s hotel

Where’s the function?

G₀ → G₁ → G₂ → G₃ → ...
Back to Hilbert’s hotel

Where’s the function?
We want a bijection from:
Back to Hilbert’s hotel

Where’s the function?
We want a bijection from: \( \mathbb{N} \setminus \{0\} \)
Where’s the function?
We want a bijection from: \( \mathbb{N} \setminus \{0\} \) to
Back to Hilbert’s hotel

Where’s the function?

We want a bijection from: $\mathbb{N} \setminus \{0\}$ to $\mathbb{N}$. 

\[
f(x) = x - 1.
\]

Maps every number from $\mathbb{N} \setminus \{0\}$ to a number in $\mathbb{N}$, and every number in $x \in \mathbb{N}$ has exactly one number $y \in \mathbb{N} \setminus \{0\}$ such that $f(y) = x$.

What if we had a bijection from $\mathbb{N}$ to $\mathbb{N} \setminus \{0\}$?

Same thing! Bijection means that the sets have the same size. Invert it and you’ll get a bijection from $\mathbb{N} \setminus \{0\}$ to $\mathbb{N}$. 

Diagram:

- $G_0 \rightarrow G_1 \rightarrow G_2 \rightarrow G_3 \rightarrow \ldots$
Back to Hilbert’s hotel

Where’s the function?
We want a bijection from: \( \mathbb{N} \setminus \{0\} \) to \( \mathbb{N} \).
\[
f(x) = x - 1.
\]
Back to Hilbert's hotel

Where's the function?

We want a bijection from: $\mathbb{N} \setminus \{0\}$ to $\mathbb{N}$.

$f(x) = x - 1$. Maps every number from $\mathbb{N} \setminus \{0\}$ to a number in $\mathbb{N}$, and
Where’s the function?

We want a bijection from: $\N \setminus \{0\}$ to $\N$.

$f(x) = x - 1$. Maps every number from $\N \setminus \{0\}$ to a number in $\N$, and every number in $x \in \N$ has exactly one number $y \in \N \setminus \{0\}$ such that $f(y) = x$. 

Back to Hilbert’s hotel

$G_0 \rightarrow G_1 \rightarrow G_2 \rightarrow G_3 \rightarrow \ldots$
Back to Hilbert’s hotel

Where’s the function?

We want a bijection from: $\mathbb{N} \setminus \{0\}$ to $\mathbb{N}$.

$f(x) = x - 1$. Maps every number from $\mathbb{N} \setminus \{0\}$ to a number in $\mathbb{N}$, and every number in $x \in \mathbb{N}$ has exactly one number $y \in \mathbb{N} \setminus \{0\}$ such that $f(y) = x$.

What if we had a bijection from $\mathbb{N}$ to $\mathbb{N} \setminus \{0\}$?
Where’s the function?

We want a bijection from: \( \mathbb{N} \setminus \{0\} \) to \( \mathbb{N} \).

\[ f(x) = x - 1. \]
Maps every number from \( \mathbb{N} \setminus \{0\} \) to a number in \( \mathbb{N} \), and every number in \( x \in \mathbb{N} \) has exactly one number \( y \in \mathbb{N} \setminus \{0\} \) such that \( f(y) = x \).

What if we had a bijection from \( \mathbb{N} \) to \( \mathbb{N} \setminus \{0\} \)?
Same thing! Bijection means that the sets have the same size.
Back to Hilbert’s hotel

Where’s the function?

We want a bijection from: \( \mathbb{N} \setminus \{0\} \) to \( \mathbb{N} \).

\( f(x) = x - 1 \). Maps every number from \( \mathbb{N} \setminus \{0\} \) to a number in \( \mathbb{N} \), and every number in \( x \in \mathbb{N} \) has exactly one number \( y \in \mathbb{N} \setminus \{0\} \) such that \( f(y) = x \).

What if we had a bijection from \( \mathbb{N} \) to \( \mathbb{N} \setminus \{0\} \)?

Same thing! Bijection means that the sets have the same size. Invert it and you’ll get a bijection from \( \mathbb{N} \setminus \{0\} \) to \( \mathbb{N} \).
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds?
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate:
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0,
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2,
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0,
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1,
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...

- $Z$ all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3, ...
  When will we get to $-1$???
  New Enumeration: 0, −1, 1, −2, 2, ...
  Bijection: $f(z) = 2|z| - \text{sign}(z)$.
  Where \( \text{sign}(z) = 1 \) if \( z > 0 \) and \( \text{sign}(z) = 0 \) otherwise.
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.

- $Z$ all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3, ...
  When will we get to $-1$???
  New Enumeration: 0, $-1$, 1, $-2$, 2, ...
  Bijection: $f(z) = 2|z| - \text{sign}(z)$.
  Where $\text{sign}(z) = 1$ if $z > 0$ and $\text{sign}(z) = 0$ otherwise.
Examples

Countably infinite (same cardinality as naturals)

- \( E \) even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.

\[ f(e) = e / 2. \]

\(-\text{Z} - \) all integers.
Twice as big?
Enumerate: 0, 1, 2, 3, ...
When will we get to \(-1\)???
New Enumeration: 0, \(-1\), 1, \(-2\), 2, ...
Bijection:
\[ f(z) = 2 \cdot |z| - \text{sign}(z). \]
Where \( \text{sign}(z) = 1 \) if \( z > 0 \) and \( \text{sign}(z) = 0 \) otherwise.
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: $0, 2, 4, ...$
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$. 

$\mathbb{Z}$ - all integers.
Twice as big?
Enumerate: $0, 1, 2, 3, ...$
When will we get to $-1$???
New Enumeration: $0, -1, 1, -2, 2, ...$
Bijection: $f(z) = 2|z| - \text{sign}(z)$.
Where $\text{sign}(z) = 1$ if $z > 0$ and $\text{sign}(z) = 0$ otherwise.
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$.

- $Z$ - all integers.
Examples

Countably infinite (same cardinality as naturals)

- \( E \) even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: \( f(e) = e/2 \).

- \( Z \) - all integers.
  Twice as big?
Examples

Countably infinite (same cardinality as naturals)

- **E** even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: \( f(e) = e/2 \).

- **Z** - all integers.
  Twice as big?
  Enumerate: 0,
Examples

Countably infinite (same cardinality as naturals)

- \( E \) even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1 , 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: \( f(e) = e/2 \).

- \( Z \) - all integers.
  Twice as big?
  Enumerate: 0, 1,
Examples

Countably infinite (same cardinality as naturals)

- \( E \) even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1 , 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: \( f(e) = e/2 \).

- \( Z \) - all integers.
  Twice as big?
  Enumerate: 0, 1, 2,
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$.

- $Z$- all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3,
Examples

Countably infinite (same cardinality as naturals)

- \(E\) even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: \(f(e) = e/2\).

- \(Z\) - all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3,...
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$.

- $Z$ - all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3, ...
  When will we get to $-1$???
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$.

- $Z$- all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3, ...
  When will we get to $-1$???
  New Enumeration: 0,
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$.

- $Z$: all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3, ...
  When will we get to $-1$???
  New Enumeration: 0, $-1$, ...
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$.

- $Z$ - all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3, ...
  When will we get to $-1$???
  New Enumeration: 0, $-1$, 1,
Examples

Countably infinite (same cardinality as naturals)

- \(E\) even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: \(f(e) = e/2\).

- \(Z\) - all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3, ...
  When will we get to \(-1\)??
  New Enumeration: 0, \(-1\), 1, \(-2\),
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: $0, 2, 4, ...$
  $0$ maps to $0$, $2$ maps to $1$, $4$ maps to $2$, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$.

- $Z$ - all integers.
  Twice as big?
  Enumerate: $0, 1, 2, 3, ...$
  When will we get to $-1$???
  New Enumeration: $0, -1, 1, -2, 2...$
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$.

- $Z$- all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3, ...
  When will we get to $-1$???
  New Enumeration: 0, $-1, 1, -2, 2$...
  Bijection: $f(z) = 2|z| - sign(z)$. 

$sign(z)$ = 1 if $z > 0$ and $0$ otherwise.
Examples

Countably infinite (same cardinality as naturals)

- $E$ even numbers.
  Where are the odds? Half as big?
  Enumerate: 0, 2, 4, ...
  0 maps to 0, 2 maps to 1, 4 maps to 2, ...
  Enumeration naturally corresponds to function.
  No two evens map to the same natural.
  For every natural, there is a corresponding even.
  Bijection: $f(e) = e/2$.

- $\mathbb{Z}$ - all integers.
  Twice as big?
  Enumerate: 0, 1, 2, 3, ...
  When will we get to $-1$???
  New Enumeration: 0, $-1$, $1$, $-2$, $2$...
  Bijection: $f(z) = 2|z| - \text{sign}(z)$.
  Where $\text{sign}(z) = 1$ if $z > 0$ and $\text{sign}(z) = 0$ otherwise.
Examples: Countable by enumeration

- $\mathbb{N} \times \mathbb{N}$ - Pairs of integers.
Examples: Countable by enumeration

- \( \mathbb{N} \times \mathbb{N} \) - Pairs of integers. Square of countably infinite?
Examples: Countable by enumeration

- $\mathbb{N} \times \mathbb{N}$ - Pairs of integers.
  Square of countably infinite?
  Enumerate: $(0, 0), (0, 1), (0, 2), \ldots$
Examples: Countable by enumeration

- \( \mathbb{N} \times \mathbb{N} \) - Pairs of integers.
  Square of countably infinite?
  Enumerate: \((0,0),(0,1),(0,2),\ldots \) ???

(dovetailing) \((a,b)\) at position \((a+b+1)(a+b)/2 + b\) in this order.
Examples: Countable by enumeration

- $\mathbb{N} \times \mathbb{N}$ - Pairs of integers.
  Square of countably infinite?
  Enumerate: $(0,0), (0,1), (0,2), \ldots$ ???
  Never get to $(1,1)$!
Examples: Countable by enumeration

- $\mathbb{N} \times \mathbb{N}$ - Pairs of integers.
  Square of countably infinite?
  Enumerate: $(0,0), (0,1), (0,2), \ldots$ ???
  Never get to $(1,1)$!
  Enumerate: $(0,0)$,
Examples: Countable by enumeration

- $\mathbb{N} \times \mathbb{N}$ - Pairs of integers.  
  Square of countably infinite?  
  Enumerate: $(0, 0), (0, 1), (0, 2), \ldots$ ???  
  Never get to $(1, 1)$!  
  Enumerate: $(0, 0), (1, 0)$,
Examples: Countable by enumeration

- \( \mathbb{N} \times \mathbb{N} \) - Pairs of integers.
  Square of countably infinite?
  Enumerate: \((0,0),(0,1),(0,2),\ldots\) ???
  Never get to \((1,1)\)!
  Enumerate: \((0,0),(1,0),(0,1),\ldots\)
Examples: Countable by enumeration

- \( \mathbb{N} \times \mathbb{N} \) - Pairs of integers.
  Square of countably infinite?
  Enumerate: \((0,0), (0,1), (0,2), \ldots \) ???
  Never get to \((1,1)\)!
  Enumerate: \((0,0), (1,0), (0,1), (2,0), \ldots \)
Examples: Countable by enumeration

- \( \mathbb{N} \times \mathbb{N} \) - Pairs of integers.
  Square of countably infinite?
  Enumerate: \((0, 0), (0, 1), (0, 2), \ldots \) ???
  Never get to \((1, 1)\)!
  Enumerate: \((0, 0), (1, 0), (0, 1), (2, 0), (1, 1), \ldots \)
Examples: Countable by enumeration

- $\mathbb{N} \times \mathbb{N}$ - Pairs of integers.
  
  Square of countably infinite?
  
  Enumerate: $(0, 0), (0, 1), (0, 2), \ldots$ ???
  
  Never get to $(1, 1)$!
  
  Enumerate: $(0, 0), (1, 0), (0, 1), (2, 0), (1, 1), (0, 2)\ldots$
Examples: Countable by enumeration

- $\mathbb{N} \times \mathbb{N}$ - Pairs of integers.
  
  Square of countably infinite?
  
  Enumerate: $(0, 0), (0, 1), (0, 2), \ldots$ ???
  
  Never get to $(1, 1)$!
  
  Enumerate: $(0, 0), (1, 0), (0, 1), (2, 0), (1, 1), (0, 2) \ldots$ (dovetailing)
Examples: Countable by enumeration

- $\mathbb{N} \times \mathbb{N}$ - Pairs of integers.
  Square of countably infinite?
  Enumerate: $(0, 0), (0, 1), (0, 2), \ldots$ ???
  Never get to $(1, 1)$!
  Enumerate: $(0, 0), (1, 0), (0, 1), (2, 0), (1, 1), (0, 2) \ldots$ (dovetailing)
Examples: Countable by enumeration

- \( \mathbb{N} \times \mathbb{N} \) - Pairs of integers.

  Square of countably infinite?
  Enumerate: \((0, 0), (0, 1), (0, 2), \ldots \) ???
  Never get to \((1, 1)\)!
  Enumerate: \((0, 0), (1, 0), (0, 1), (2, 0), (1, 1), (0, 2) \ldots \) (dovetailing)

\[
\begin{align*}
(0, 0) & \rightarrow (0, 1) \rightarrow (0, 2) \rightarrow (0, 3) \\
(1, 0) & \rightarrow (1, 1) \rightarrow (1, 2) \ldots \\
(2, 0) & \rightarrow (2, 1) \ldots \\
(3, 0) & \ldots \\
\vdots
\end{align*}
\]

\((a, b)\) at position \((a + b + 1)(a + b)/2 + b\) in this order.
Rationals

All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$. 
All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$. Enumerate: list 0, positive and negative.
All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$.
Enumerate: list 0, positive and negative. How?
Rationals

All rational numbers \( \mathbb{Q} \): \( \frac{a}{b} \), such that \( a, b \in \mathbb{Z} \), and \( b \neq 0 \).
Enumerate: list 0, positive and negative. How?
Same as \( \mathbb{Z}^2 \)!!!!
Rationals

All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$.
Enumerate: list 0, positive and negative. How?
Same as $\mathbb{Z}^2$!!!! In fact, $\mathbb{Z}^2$ is ”bigger” than $\mathbb{Q}$.
Rationals

All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$. Enumerate: list 0, positive and negative. How?
Same as $\mathbb{Z}^2$!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$. So let’s show $\mathbb{Z}^2$ is countable.
Rationals

All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$.
Enumerate: list 0, positive and negative. How?
Same as $\mathbb{Z}^2$!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$.
So let’s show $\mathbb{Z}^2$ is countable.
All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$. Enumerate: list 0, positive and negative. How?

Same as $\mathbb{Z}^2$!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$.

So let’s show $\mathbb{Z}^2$ is countable.

Enumerate: (0,0),...
Rationals

All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$. Enumerate: list 0, positive and negative. How?

Same as $\mathbb{Z}^2$!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$. So let's show $\mathbb{Z}^2$ is countable.

Enumerate: $(0,0), (1,0), \ldots$
Rationals

All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$.
Enumerate: list 0, positive and negative. How?
Same as $\mathbb{Z}^2$!!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$.
So let’s show $\mathbb{Z}^2$ is countable.
Enumerate: (0,0), (1,0), (1,1),

![Diagram of enumeration of rationals](image)
Rationals

All rational numbers \( \mathbb{Q} \): \( \frac{a}{b} \), such that \( a, b \in \mathbb{Z} \), and \( b \neq 0 \).

Enumerate: list 0, positive and negative. How?
Same as \( \mathbb{Z}^2 \)!!!! In fact, \( \mathbb{Z}^2 \) is "bigger" than \( \mathbb{Q} \).
So let’s show \( \mathbb{Z}^2 \) is countable.

Enumerate: (0,0), (1,0), (1,1), (0,1),...
Rationals

All rational numbers \( Q: \frac{a}{b} \), such that \( a, b \in \mathbb{Z} \), and \( b \neq 0 \).

Enumerate: list 0, positive and negative. How?

Same as \( \mathbb{Z}^2 \)!!!! In fact, \( \mathbb{Z}^2 \) is ”bigger” than \( Q \).

So let’s show \( \mathbb{Z}^2 \) is countable.

Enumerate: (0,0), (1,0), (1,1), (0,1), (-1,1)...

[Diagram of spiral enumeration]
All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$. 
Enumerate: list 0, positive and negative. How? 
Same as $\mathbb{Z}^2$!!! In fact, $\mathbb{Z}^2$ is ”bigger” than $\mathbb{Q}$. 
So let’s show $\mathbb{Z}^2$ is countable. 
Enumerate: $(0,0), (1,0), (1,1), (0,1), (-1,1)$... 
Will eventually get to any pair.
Rationals

All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$.
Enumerate: list 0, positive and negative. How?
Same as $\mathbb{Z}^2$!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$.
So let’s show $\mathbb{Z}^2$ is countable.

Enumerate: $(0,0), (1,0), (1,1), (0,1), (-1,1)\ldots$
Will eventually get to any pair.
Two different pairs cannot map to the same natural number/same position in the spiral.
All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$. 
Enumerate: list 0, positive and negative. How?
Same as $\mathbb{Z}^2$!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$.
So let’s show $\mathbb{Z}^2$ is countable.

Enumerate: (0,0), (1,0), (1,1), (0,1), (-1,1)...
Will eventually get to any pair.
Two different pairs cannot map to the same natural number/same position in the spiral.
Every natural has a "corresponding" pair.
All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$. 
Enumerate: list 0, positive and negative. How? 
Same as $\mathbb{Z}^2$!!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$. 
So let’s show $\mathbb{Z}^2$ is countable.

Enumerate: (0,0), (1,0), (1,1), (0,1), (-1,1)...
Will eventually get to any pair.
Two different pairs cannot map to the same natural number/same position in the spiral.
Every natural has a "corresponding" pair.
Where’s my bijection???
All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$. 
Enumerate: list 0, positive and negative. How?
Same as $\mathbb{Z}^2$!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$.
So let's show $\mathbb{Z}^2$ is countable.

Enumerate: (0,0), (1,0), (1,1), (0,1), (-1,1)...
Will eventually get to any pair.
Two different pairs cannot map to the same natural number/same position in the spiral.
Every natural has a "corresponding" pair.
Where's my bijection?? Too complicated!
All rational numbers \( \mathbb{Q} \): \( \frac{a}{b} \), such that \( a, b \in \mathbb{Z} \), and \( b \neq 0 \). 
Enumerate: list 0, positive and negative. How?
Same as \( \mathbb{Z}^2 \)!!! In fact, \( \mathbb{Z}^2 \) is "bigger" than \( \mathbb{Q} \). 
So let’s show \( \mathbb{Z}^2 \) is countable.

Enumerate: (0,0), (1,0), (1,1), (0,1), (-1,1)...
Will eventually get to any pair.
Two different pairs cannot map to the same natural number/same position in the spiral.
Every natural has a "corresponding" pair.
Where’s my bijection?? Too complicated! Enumeration is good enough:
A set \( S \) is countable if it can be enumerated in a sequence, i.e., if all of its elements can be listed as a sequence \( a_1, a_2, \ldots \).
All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$.

Enumerate: list 0, positive and negative. How?

Same as $\mathbb{Z}^2$!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$.

So let's show $\mathbb{Z}^2$ is countable.

Enumerate: $(0,0), (1,0), (1,1), (0,1), (-1,1)\ldots$

Will eventually get to any pair.

Two different pairs cannot map to the same natural number/same position in the spiral.

Every natural has a "corresponding" pair.

Where's my bijection??? Too complicated! Enumeration is good enough:

A set $S$ is countable if it can be enumerated in a sequence, i.e., if all of its elements can be listed as a sequence $a_1, a_2, \ldots$. Make sure that (1) different elements map to different naturals.
All rational numbers $\mathbb{Q}$: $\frac{a}{b}$, such that $a, b \in \mathbb{Z}$, and $b \neq 0$.

Enumerate: list 0, positive and negative. How?

Same as $\mathbb{Z}^2$!!! In fact, $\mathbb{Z}^2$ is "bigger" than $\mathbb{Q}$.

So let’s show $\mathbb{Z}^2$ is countable.

Enumerate: (0,0), (1,0), (1,1), (0,1), (-1,1)...

Will eventually get to any pair.

Two different pairs cannot map to the same natural number/same position in the spiral.

Every natural has a "corresponding" pair.

Where’s my bijection??? Too complicated! Enumeration is good enough:

A set $S$ is countable if it can be enumerated in a sequence, i.e., if all of its elements can be listed as a sequence $a_1, a_2, \ldots$. Make sure that (1) different elements map to different naturals. (2) every natural gets an element.
Let’s get real

Is the set of Reals countable?
Let’s get real

Is the set of Reals countable?
Lets consider the reals $[0, 1]$. 
Let’s get real

Is the set of Reals countable?

Let’s consider the reals $[0, 1]$.

Each real has a decimal representation.
Let’s get real

Is the set of Reals countable?

Let’s consider the reals [0, 1].

Each real has a decimal representation.

.500000000...
Is the set of Reals countable?

Let's consider the reals [0, 1].

Each real has a decimal representation.
.500000000... (1/2)
Is the set of Reals countable?

Let's consider the reals \([0, 1]\).

Each real has a decimal representation.
\(0.500000000\ldots\) (1/2)
\(0.785398162\ldots\)
Let's get real

Is the set of Reals countable?

Let's consider the reals $[0, 1]$.

Each real has a decimal representation.
.500000000... (1/2)
.785398162... $\pi/4$
Let’s get real

Is the set of Reals countable?

Let’s consider the reals $[0, 1]$.

Each real has a decimal representation.

.500000000... (1/2)
.785398162... $\pi/4$
.367879441...
Is the set of Reals countable?

Let's consider the reals $[0, 1]$.

Each real has a decimal representation.

- $0.5000000000...$ (1/2)
- $0.785398162...$ $\pi/4$
- $0.367879441...$ $1/e$
Let’s get real

Is the set of Reals countable?

Let’s consider the reals $[0, 1]$.

Each real has a decimal representation.

- $.500000000...$ (1/2)
- $.785398162...$ $\pi/4$
- $.367879441...$ $1/e$
- $.632120558...$
Let’s get real

Is the set of Reals countable?

Let’s consider the reals $[0, 1]$.

Each real has a decimal representation.

- $0.500000000...$ (1/2)
- $0.785398162...$ $\pi/4$
- $0.367879441...$ $1/e$
- $0.632120558...$ $1−1/e$
Let’s get real

Is the set of Reals countable?

Let's consider the reals $[0,1]$.

Each real has a decimal representation.

- .500000000... (1/2)
- .785398162... $\pi/4$
- .367879441... $1/e$
- .632120558... $1 - 1/e$
- .345212312...
Let’s get real

Is the set of Reals countable?

Let's consider the reals $[0, 1]$.

Each real has a decimal representation.

$0.500000000...$ (1/2)

$0.785398162...$ $\pi/4$

$0.367879441...$ $1/e$

$0.632120558...$ $1 - 1/e$

$0.345212312...$ Some real number
Let’s get real

Is the set of Reals countable?

Let's consider the reals $[0, 1]$.

Each real has a decimal representation.

- $.500000000...$ $(1/2)$
- $.785398162...$ $\pi/4$
- $.367879441...$ $1/e$
- $.632120558...$ $1 - 1/e$
- $.345212312...$ Some real number
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0,1]$. 

Construct "diagonal" number: 

$$0: \ldots500000000\ldots$$

$$1: \ldots785398162\ldots$$

$$2: \ldots367879441\ldots$$

$$3: \ldots632120558\ldots$$

$$4: \ldots345212312\ldots$$

... 

Diagonal Number:

Digit $i$ is 7 if number $i$'s $i$th digit is not 7 and 6 otherwise.

Diagonal number for a list differs from every number in list! 

Diagonal number is real.

Contradiction!

Subset $[0,1]$ is not countable!!
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

Construct "diagonal" number:
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: .500000000...
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: .500000000...
1: .785398162...

...
Diagonalization.

If countable, there exists a listing (enumeration), \( L \) contains all reals in \([0, 1]\). For example

0: \(.500000000\ldots\)
1: \(.785398162\ldots\)
2: \(.367879441\ldots\)

Construct "diagonal" number: \( .77677 \ldots \)

Diagonal Number:

Digit

\( i \) is 7 if number

\( i \)'s

\( i \)th digit is not 7

and 6 otherwise.

Diagonal number for a list differs from every number in list!

Diagonal number not in list.

Diagonal number is real.

Contradiction!

Subset \([0, 1]\) is not countable!!
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: \( .500000000... \)
1: \( .785398162... \)
2: \( .367879441... \)
3: \( .632120558... \)

Construct "diagonal" number:

\[ .77677... \]

Diagonal Number:

Digit \( i \) is 7 if number \( i \)'s \( i \)th digit is not 7 and 6 otherwise.

Diagonal number for a list differs from every number in list!

Diagonal number not in list.

Diagonal number is real.

Contradiction!

Subset $[0, 1]$ is not countable!!
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: .500000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: .500000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...

\[ \vdots \]
Diagonalization.

If countable, there exists a listing (enumeration), \( L \) contains all reals in \([0, 1]\). For example

0: .500000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...

\vdots

Construct “diagonal” number:
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0,1]$. For example

0: .500000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...

...,

Construct “diagonal” number: .7
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: .500000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...

Construct “diagonal” number: .77
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: 0.500000000...
1: 0.785398162...
2: 0.367879441...
3: 0.632120558...
4: 0.345212312...

Construct “diagonal” number: 0.776
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: .500000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...

::

Construct "diagonal" number: .7767
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: .500000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...

Construct “diagonal” number: .77677

Diagonal number for a list differs from every number in list!

Diagonal number not in list.

Diagonal number is real.

Contradiction!

Subset $[0, 1]$ is not countable!!
Diagonalization.

If countable, there exists a listing (enumeration), \( L \) contains all reals in \([0, 1]\). For example

0: .5000000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...
...

Construct “diagonal” number: .77677…
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: \(0.500000000\ldots\)
1: \(0.785398162\ldots\)
2: \(0.367879441\ldots\)
3: \(0.632120558\ldots\)
4: \(0.345212312\ldots\)

::

Construct “diagonal” number: \(0.77677\ldots\)

Diagonal Number:
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0,1]$. For example

0: .500000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...

\vdots

Construct “diagonal” number: .77677\ldots

Diagonal Number: Digit $i$ is 7 if number $i$’s $i$th digit is not 7
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0,1]$. For example

0: \[.500000000... \]
1: \[.785398162... \]
2: \[.367879441... \]
3: \[.632120558... \]
4: \[.345212312... \]

\[\vdots\]

Construct “diagonal” number: \[.77677... \]

Diagonal Number: Digit $i$ is 7 if number $i$’s $i$th digit is not 7 and 6 otherwise.
Diagonalization.

If countable, there exists a listing (enumeration), \( L \) contains all reals in \([0, 1]\). For example

0: \( .5000000000... \)
1: \( .785398162... \)
2: \( .367879441... \)
3: \( .632120558... \)
4: \( .345212312... \)

: 

Construct “diagonal” number: \( .77677... \)

Diagonal Number: Digit \( i \) is 7 if number \( i \)’s \( i \)th digit is not 7 and 6 otherwise.

Diagonal number for a list differs from every number in list!
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0,1]$. For example

$0: \, .500000000...$
$1: \, .785398162...$
$2: \, .367879441...$
$3: \, .632120558...$
$4: \, .345212312...$

... 

Construct “diagonal” number: $0.77677...$

**Diagonal Number**: Digit $i$ is 7 if number $i$’s $i$th digit is not 7 and 6 otherwise.

Diagonal number for a list differs from every number in list!
Diagonal number not in list.
Diagonalization.

If countable, there exists a listing (enumeration), \( L \) contains all reals in \([0, 1]\). For example

0: \( .500000000... \)
1: \( .785398162... \)
2: \( .367879441... \)
3: \( .632120558... \)
4: \( .345212312... \)

Construct “diagonal” number: \( .77677... \)

Diagonal Number: Digit \( i \) is 7 if number \( i \)'s \( i \)th digit is not 7 and 6 otherwise.

Diagonal number for a list differs from every number in list!
Diagonal number not in list.
Diagonal number is real.
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: \[0.500000000...\]
1: \[0.785398162...\]
2: \[0.367879441...\]
3: \[0.632120558...\]
4: \[0.345212312...\]

Construct “diagonal” number: \[0.77677...\]

Diagonal Number: Digit $i$ is 7 if number $i$’s $i$th digit is not 7 and 6 otherwise.

Diagonal number for a list differs from every number in list! Diagonal number not in list.

Diagonal number is real.

Contradiction!
Diagonalization.

If countable, there exists a listing (enumeration), $L$ contains all reals in $[0, 1]$. For example

0: .500000000...
1: .785398162...
2: .367879441...
3: .632120558...
4: .345212312...

Construct “diagonal” number: .77677...

Diagonal Number: Digit $i$ is 7 if number $i$’s $i$th digit is not 7 and 6 otherwise.

Diagonal number for a list differs from every number in list!

Diagonal number not in list.

Diagonal number is real.

Contradiction!

Subset $[0, 1]$ is not countable!!
All reals?

Subset [0, 1] is not countable!!
All reals?

Subset [0, 1] is not countable!!
What about all reals?
All reals?

Subset [0, 1] is not countable!!

What about all reals?
Uncountable.
All reals?

Subset $[0, 1]$ is not countable!!

What about all reals?
Uncountable.

Any subset of a countable set is countable.
All reals?

Subset \([0,1]\) is not countable!!

What about all reals?
Uncountable.

Any subset of a countable set is countable.

If reals are countable then so is \([0,1]\).
Diagonalization.

1. Assume that a set $S$ can be enumerated.
1. Assume that a set $S$ can be enumerated.
2. Consider an arbitrary list of all the elements of $S$. 
Diagonalization.

1. Assume that a set $S$ can be enumerated.
2. Consider an arbitrary list of all the elements of $S$.
3. Use the diagonal from the list to construct a new element $t$.
Diagonalization.

1. Assume that a set $S$ can be enumerated.
2. Consider an arbitrary list of all the elements of $S$.
3. Use the diagonal from the list to construct a new element $t$.
4. Show that $t$ is different from all elements in the list.
5. Show that $t$ is in $S$.
6. Contradiction.
Diagonalization.

1. Assume that a set $S$ can be enumerated.
2. Consider an arbitrary list of all the elements of $S$.
3. Use the diagonal from the list to construct a new element $t$.
4. Show that $t$ is different from all elements in the list $\implies t$ is not in the list.
1. Assume that a set $S$ can be enumerated.
2. Consider an arbitrary list of all the elements of $S$.
3. Use the diagonal from the list to construct a new element $t$.
4. Show that $t$ is different from all elements in the list $\Rightarrow t$ is not in the list.
5. Show that $t$ is in $S$.

Diagonalization.
Diagonalization.

1. Assume that a set $S$ can be enumerated.
2. Consider an arbitrary list of all the elements of $S$.
3. Use the diagonal from the list to construct a new element $t$.
4. Show that $t$ is different from all elements in the list $\implies t$ is not in the list.
5. Show that $t$ is in $S$.
6. Contradiction.
Another diagonalization.

The set of all subsets of $N$. 

$\{0\}$, $\{0, \ldots, 7\}$, evens, odds, primes, multiples of 10

Assume is countable.

There is a listing, $L$, that contains all subsets of $N$.

Define a diagonal set, $D$: If $i$th set in $L$ does not contain $i$, $i \in D$. Otherwise $i \notin D$.

$D$ is different from $i$th set in $L$ for every $i$. 

$\Rightarrow D$ is not in the listing.

$L$ does not contain all subsets of $N$.

Contradiction.

Theorem: The set of all subsets of $N$ is not countable.

(The set of all subsets of $S$, is the powerset of $N$. )
Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$: $\{0\}$,
Another diagonalization.

The set of all subsets of \( N \).

Example subsets of \( N \): \( \{0\} \), \( \{0,\ldots,7\} \),
Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$: $\{0\}$, $\{0, \ldots, 7\}$, evens,
Another diagonalization.

The set of all subsets of $\mathbb{N}$.

Example subsets of $\mathbb{N}$: \{0\}, \{0,\ldots,7\},
evens, odds,
Another diagonalization.

The set of all subsets of $\mathbb{N}$.

Example subsets of $\mathbb{N}$: $\{0\}$, $\{0, \ldots, 7\}$, evens, odds, primes,
Another diagonalization.

The set of all subsets of $\mathbb{N}$.

Example subsets of $\mathbb{N}$:  
- $\{0\}$, $\{0,\ldots,7\}$,
- evens, odds, primes, multiples of 10

Assume is countable.
Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$: $\{0\}$, $\{0,\ldots,7\}$, evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, $L$, that contains all subsets of $N$. 

Theorem: The set of all subsets of $N$ is not countable.

(The set of all subsets of $S$, is the powerset of $N$.)
Another diagonalization.

The set of all subsets of \( N \).

Example subsets of \( N \): \( \{0\} \), \( \{0, \ldots, 7\} \), evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, \( L \), that contains all subsets of \( N \).
- Define a diagonal set, \( D \):

\[
D = \begin{cases} 
\text{if } i \text{th set in } L \text{ does not contain } i, \\
\text{otherwise } i \not\in D.
\end{cases}
\]

\( D \) is different from the \( i \)th set in \( L \) for every \( i \).

\( D \) is not in the listing.

\( D \) is a subset of \( N \).

\( L \) does not contain all subsets of \( N \).

Contradiction.

Theorem: The set of all subsets of \( N \) is not countable.

(The set of all subsets of \( S \), is the powerset of \( N \).)
Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$: $\{0\}, \{0, \ldots, 7\}$, evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, $L$, that contains all subsets of $N$.
- Define a diagonal set, $D$:
  If $i$th set in $L$ does not contain $i$, $i \in D$.

Theorem: The set of all subsets of $N$ is not countable.

(The set of all subsets of $S$, is the powerset of $N$.)

Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$: $\{0\}$, $\{0,\ldots,7\}$, evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, $L$, that contains all subsets of $N$.
- Define a diagonal set, $D$:
  If $i$th set in $L$ does not contain $i$, $i \in D$.
  otherwise $i \notin D$.

Theorem: The set of all subsets of $N$ is not countable.
Another diagonalization.

The set of all subsets of \( \mathbb{N} \).

Example subsets of \( \mathbb{N} \): \{0\}, \{0, \ldots, 7\}, evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, \( L \), that contains all subsets of \( \mathbb{N} \).
- Define a diagonal set, \( D \):
  - If \( i \)th set in \( L \) does not contain \( i \), \( i \in D \).
  - otherwise \( i \notin D \).
- \( D \) is different from \( i \)th set in \( L \) for every \( i \).
Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$: $\{0\}$, $\{0, \ldots, 7\}$, evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, $L$, that contains all subsets of $N$.
- Define a diagonal set, $D$:
  If $i$th set in $L$ does not contain $i$, $i \in D$.
  otherwise $i \notin D$.
- $D$ is different from $i$th set in $L$ for every $i$.
  $\implies$ $D$ is not in the listing.
Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$: $\{0\}$, $\{0, \ldots, 7\}$, evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, $L$, that contains all subsets of $N$.
- Define a diagonal set, $D$:
  If $i$th set in $L$ does not contain $i$, $i \in D$.
  otherwise $i \notin D$.
- $D$ is different from $i$th set in $L$ for every $i$.
  $\implies D$ is not in the listing.
- $D$ is a subset of $N$. 

(Theorem: The set of all subsets of $N$ is not countable.

(The set of all subsets of $S$, is the powerset of $N$.))
Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$: $\{0\}$, $\{0,\ldots,7\}$, evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, $L$, that contains all subsets of $N$.
- Define a diagonal set, $D$:
  If $i$th set in $L$ does not contain $i$, $i \in D$.
  otherwise $i \notin D$.
- $D$ is different from $i$th set in $L$ for every $i$.
  $\implies$ $D$ is not in the listing.
- $D$ is a subset of $N$.
- $L$ does not contain all subsets of $N$.

Theorem: The set of all subsets of $N$ is not countable.
(The set of all subsets of $S$, is the powerset of $N$.)
Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$: $\{0\}$, $\{0,\ldots,7\}$, evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, $L$, that contains all subsets of $N$.
- Define a diagonal set, $D$:
  If $i$th set in $L$ does not contain $i$, $i \in D$.
  otherwise $i \notin D$.

- $D$ is different from $i$th set in $L$ for every $i$.
  $D$ is not in the listing.

- $D$ is a subset of $N$.
- $L$ does not contain all subsets of $N$.
  Contradiction.
Another diagonalization.

The set of all subsets of \( N \).

Example subsets of \( N \): \( \{0\} \), \( \{0, \ldots, 7\} \), evens, odds, primes, multiples of 10

- Assume is countable.
- There is a listing, \( L \), that contains all subsets of \( N \).
- Define a diagonal set, \( D \):
  - If \( i \)th set in \( L \) does not contain \( i \), \( i \in D \).
  - otherwise \( i \notin D \).
- \( D \) is different from \( i \)th set in \( L \) for every \( i \).
  \( \implies \) \( D \) is not in the listing.
- \( D \) is a subset of \( N \).
- \( L \) does not contain all subsets of \( N \).
  Contradiction.

Theorem: The set of all subsets of \( N \) is not countable.
Another diagonalization.

The set of all subsets of $N$.

Example subsets of $N$:  
- $\{0\}$,  
- $\{0, \ldots, 7\}$,  
- evens, odds, primes, multiples of 10

► Assume is countable.

► There is a listing, $L$, that contains all subsets of $N$.

► Define a diagonal set, $D$:  
  If $i$th set in $L$ does not contain $i$, $i \in D$.  
  otherwise $i \notin D$.

► $D$ is different from $i$th set in $L$ for every $i$.  
  $\Rightarrow$ $D$ is not in the listing.

► $D$ is a subset of $N$.

► $L$ does not contain all subsets of $N$.  
  Contradiction.

**Theorem:** The set of all subsets of $N$ is not countable.  
(The set of all subsets of $S$, is the powerset of $N$.)
Another diagonalization.

\[
\begin{align*}
s_1 &= 0 0 0 0 0 0 0 0 0 0 0 0 
& \quad \vdots \\
s_2 &= 1 1 1 1 1 1 1 1 1 1 1 1 
& \quad \vdots \\
s_3 &= 0 1 0 1 0 1 0 1 0 1 0 1 
& \quad \vdots \\
s_4 &= 1 0 1 0 1 0 1 0 1 0 1 0 
& \quad \vdots \\
s_5 &= 1 1 0 1 0 1 1 0 1 0 1 1 
& \quad \vdots \\
s_6 &= 0 0 1 1 0 1 1 0 1 1 0 1 
& \quad \vdots \\
s_7 &= 1 0 0 0 1 0 0 1 0 0 1 0 
& \quad \vdots \\
s_8 &= 0 0 1 1 0 0 1 1 0 0 1 1 
& \quad \vdots \\
s_9 &= 1 1 0 0 1 1 0 0 1 1 0 1 
& \quad \vdots \\
s_{10} &= 1 1 0 1 1 1 0 0 1 0 1 0 
& \quad \vdots \\
s_{11} &= 1 1 0 1 0 1 0 0 1 0 0 0 
& \quad \vdots \\
\vdots &= \vdots \vdots \vdots \vdots \vdots \vdots \vdots \\
\vdots &= \vdots \vdots \vdots \vdots \vdots \vdots \vdots \\
\end{align*}
\]

\[
s = 1 0 1 1 1 0 1 0 0 1 1 \ldots
\]
Countable or uncountable??

- Binary strings?
Countable or uncountable??

- Binary strings?
- Trees?
Countable or uncountable??

- Binary strings?
- Trees?
- Weighted trees?
Countable or uncountable??

- Binary strings?
- Trees?
- Weighted trees?
- Inputs to the stable marriage algorithm?
Countable or uncountable??

- Binary strings?
- Trees?
- Weighted trees?
- Inputs to the stable marriage algorithm?
- Mathematical proofs?
Countable or uncountable??

- Binary strings?
- Trees?
- Weighted trees?
- Inputs to the stable marriage algorithm?
- Mathematical proofs?
- Programs in Java?
Countable or uncountable??

- Binary strings?
- Trees?
- Weighted trees?
- Inputs to the stable marriage algorithm?
- Mathematical proofs?
- Programs in Java?
- All possible endings to Game of Thrones?
Countable or uncountable??

- Binary strings?
- Trees?
- Weighted trees?
- Inputs to the stable marriage algorithm?
- Mathematical proofs?
- Programs in Java?
- All possible endings to Game of Thrones?
- All subsets of Reals?
Countable or uncountable??

- Binary strings?
- Trees?
- Weighted trees?
- Inputs to the stable marriage algorithm?
- Mathematical proofs?
- Programs in Java?
- All possible endings to Game of Thrones?
- All subsets of Reals?
- Functions from $\mathbb{N}$ to $\mathbb{N}$?
Countable or uncountable??

- Binary strings?
- Trees?
- Weighted trees?
- Inputs to the stable marriage algorithm?
- Mathematical proofs?
- Programs in Java?
- All possible endings to Game of Thrones?
- All subsets of Reals?
- Functions from $\mathbb{N}$ to $\mathbb{N}$?

You already know some of these.....
Countable or uncountable??

- Binary strings?
- Trees?
- Weighted trees?
- Inputs to the stable marriage algorithm?
- Mathematical proofs?
- Programs in Java?
- All possible endings to Game of Thrones?
- All subsets of Reals?
- Functions from $\mathbb{N}$ to $\mathbb{N}$?

You already know some of these..... Think about induction!
What happened with Cantor?

Cantor's work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite

After Cantor: Countable

For example \{1, 2, 3\} ▶ Infinite and countable. For example \mathbb{N}, \mathbb{Z}, ...

Uncountable. For example \left[0, 1\right], \mathbb{R}...

▶ Bigger than uncountable!

(Math 135, Math 136, Math 227A...

Everyone was upset! Many puzzled... Many openly hostile to Cantor... Cantor was clinically depressed. In and out of hospitals until the end of his life. Died in poverty...
Cantor’s work between 1874 and 1884 is the origin of set theory.
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content.
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite

▶ Countable

For example \{1, 2, 3\} ▶ Infinite and countable. For example \(N, Z, \ldots\)

▶ Uncountable.

For example \([0, 1], \mathbb{R}\) ▶ Bigger than uncountable!

Everyone was upset! Many puzzled... Many openly hostile to Cantor... Cantor was clinically depressed. In and out of hospitals until the end of his life. Died in poverty...
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite

After Cantor:

- Countable

Everyone was upset! Many puzzled... Many openly hostile to Cantor... Cantor was clinically depressed. In and out of hospitals until the end of his life. Died in poverty...
Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite
After Cantor:

- Countable
  - Finite and countable.
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content.

Before Cantor: Finite, Infinite
After Cantor:

- Countable
  - Finite and countable. For example \{1, 2, 3\}
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite

After Cantor:

- Countable
  - Finite and countable. For example \( \{1, 2, 3\} \)
  - Infinite and countable.
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite

After Cantor:

▶ Countable

▶ Finite and countable. For example \( \{1, 2, 3\} \)

▶ Infinite and countable. For example \( \mathbb{N}, \mathbb{Z}, \ldots \)
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite
After Cantor:

- Countable
  - Finite and countable. For example \{1, 2, 3\}
  - Infinite and countable. For example \(\mathbb{N}, \mathbb{Z}, \ldots\)

- Uncountable.
Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite
After Cantor:

- Countable
  - Finite and countable. For example \{1, 2, 3\}
  - Infinite and countable. For example \(\mathbb{N}, \mathbb{Z}, \ldots\)
- Uncountable. For example \([0, 1], \mathbb{R}\ldots\)
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite

After Cantor:

- Countable
  - Finite and countable. For example \{1, 2, 3\}
  - Infinite and countable. For example \(\mathbb{N}, \mathbb{Z}, \ldots\)
- Uncountable. For example \([0, 1], \mathbb{R}\ldots\)
- Bigger than uncountable!
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite

After Cantor:

- Countable
  - Finite and countable. For example \{1, 2, 3\}
  - Infinite and countable. For example \(\mathbb{N}, \mathbb{Z}, \ldots\)

- Uncountable. For example \([0, 1], \mathbb{R}\ldots\)

- Bigger than uncountable! (Math 135, Math 136, Math 227A \ldots \)
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite
After Cantor:

- Countable
  - Finite and countable. For example \{1, 2, 3\}
  - Infinite and countable. For example \(\mathbb{N}, \mathbb{Z}, \ldots\)

- Uncountable. For example \([0, 1], \mathbb{R} \ldots\)

- Bigger than uncountable! (Math 135, Math 136, Math 227A ...)

Everyone was upset!
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite
After Cantor:

- Countable
  - Finite and countable. For example \{1, 2, 3\}
  - Infinite and countable. For example \(\mathbb{N}, \mathbb{Z}, \ldots\)

- Uncountable. For example \([0, 1], \mathbb{R}\ldots\)

- Bigger than uncountable! (Math 135, Math 136, Math 227A \ldots\)

Everyone was upset! Many puzzled...
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite

After Cantor:

- **Countable**
  - Finite and countable. For example \{1, 2, 3\}
  - Infinite and countable. For example \(\mathbb{N}, \mathbb{Z}, \ldots\)

- **Uncountable.** For example [0, 1], \(\mathbb{R}\)...
- Bigger than uncountable! (Math 135, Math 136, Math 227A ... )

Everyone was upset! Many puzzled... Many openly hostile to Cantor...
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite

After Cantor:

- Countable
  - Finite and countable. For example \{1, 2, 3\}
  - Infinite and countable. For example \(\mathbb{N}, \mathbb{Z}, \ldots\)

- Uncountable. For example \([0, 1], \mathbb{R}\ldots\)
- Bigger than uncountable! (Math 135, Math 136, Math 227A \ldots\)

Everyone was upset! Many puzzled... Many openly hostile to Cantor... Cantor was clinically depressed.
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite
After Cantor:

- Countable
  - Finite and countable. For example \{1, 2, 3\}
  - Infinite and countable. For example \mathbb{N}, \mathbb{Z}, ...

- Uncountable. For example [0, 1], \mathbb{R}...

- Bigger than uncountable! (Math 135, Math 136, Math 227A ...)

Everyone was upset! Many puzzled... Many openly hostile to Cantor... Cantor was clinically depressed. In and out of hospitals until the end of his life.
What happened with Cantor?

Cantor’s work between 1874 and 1884 is the origin of set theory. No one had realized that set theory had any nontrivial content. Before Cantor: Finite, Infinite
After Cantor:

- Countable
  - Finite and countable. For example \( \{1, 2, 3\} \)
  - Infinite and countable. For example \( \mathbb{N}, \mathbb{Z}, ... \)

- Uncountable. For example \( [0, 1], \mathbb{R}... \)

- Bigger than uncountable! (Math 135, Math 136, Math 227A ... )

Everyone was upset! Many puzzled... Many openly hostile to Cantor... Cantor was clinically depressed. In and out of hospitals until the end of his life. Died in poverty...
Let's look at the foundations!

Clear ambition:

Become the new Euclid.

Make up a bunch of axioms for number theory.

(In the case of geometry "A straight line segment can be drawn joining any two points" etc)

Everything that is true in number theory can be inferred from the axioms.

Writes Basic Laws of Arithmetic vol. 1.

680 pages (Amazon).

About to publish vol. 2.

And then......

Disaster!!
Cantor’s legacy

Gottlob Frege: Let’s look at the foundations!
Cantor’s legacy

Gottlob Frege: Let’s look at the foundations!
Clear ambition:
Gottlob Frege: Let’s look at the foundations!
Clear ambition: Become the new Euclid.
Gottlob Frege: Let’s look at the foundations!
Clear ambition: Become the new Euclid.
Make up a bunch of axioms for number theory.
Gottlob Frege: Let’s look at the foundations!
Clear ambition: Become the new Euclid.
Make up a bunch of axioms for number theory. (In the case of geometry “A straight line segment can be drawn joining any two points” etc)
Gottlob Frege: Let’s look at the foundations!
Clear ambition: Become the new Euclid.
Make up a bunch of axioms for number theory. (In the case of geometry ”A straight line segment can be drawn joining any two points” etc)
Everything that is true in number theory can be inferred from the axioms.
Gottlob Frege: Let’s look at the foundations!
Clear ambition: Become the new Euclid.
Make up a bunch of axioms for number theory. (In the case of geometry “A straight line segment can be drawn joining any two points” etc)
Everything that is true in number theory can be inferred from the axioms.
Writes Basic Laws of Arithmetic vol. 1.
Gottlob Frege: Let’s look at the foundations! Clear ambition: Become the new Euclid. Make up a bunch of axioms for number theory. (In the case of geometry "A straight line segment can be drawn joining any two points" etc) Everything that is true in number theory can be inferred from the axioms. Writes Basic Laws of Arithmetic vol. 1. 680 pages (Amazon).
Gottlob Frege: Let’s look at the foundations!
Clear ambition: Become the new Euclid.
Make up a bunch of axioms for number theory. (In the case of geometry "A straight line segment can be drawn joining any two points" etc)
Everything that is true in number theory can be inferred from the axioms.
Writs Basic Laws of Arithmetic vol. 1. 680 pages (Amazon).
About to publish vol. 2.
Cantor’s legacy

Gottlob Frege: Let’s look at the foundations!
Clear ambition: Become the new Euclid.
Make up a bunch of axioms for number theory. (In the case of geometry “A straight line segment can be drawn joining any two points” etc)
Everything that is true in number theory can be inferred from the axioms.
Writes Basic Laws of Arithmetic vol. 1. 680 pages (Amazon).
About to publish vol. 2. And then......
Cantor’s legacy

Gottlob Frege: Let’s look at the foundations!
Clear ambition: Become the new Euclid.
Make up a bunch of axioms for number theory. (In the case of
gometry "A straight line segment can be drawn joining any two
points" etc)
Everything that is true in number theory can be inferred from the
axioms.
Wites Basic Laws of Arithmetic vol. 1. 680 pages (Amazon).
About to publish vol. 2. And then......
Disaster!!
A bug

Bertrand Russell finds a bug!
A bug

Bertrand Russell finds a bug!
A bug

Bertrand Russell finds a bug!

Frege’s reaction:
Bertrand Russell finds a bug!

Frege’s reaction: "Hardly anything more unfortunate can befall a scientific writer than to have one of the foundations of his edifice shaken after the work is finished. This was the position I was placed in by a letter of Mr. Bertrand Russell, just when the printing of this volume was nearing its completion."
Zisimos Lorentzatos.
”Beware of systems grandiose, of mathematically strict causalities as you’re trying, stone by stone, to found the goldenwoven tower of the logical, castle and fort immune to contradiction. Designed in two volumes, the foundational laws of arithmetic, or Grundgesetze of der arithmetic in 1893, the first, 1903 the second. A life’s work. Hammer on chisel blows for years and years. So far, so good. But as Frege Gottlob was correcting, content, the printer’s proofs already of the second volume, one cursed logic paradox, one not admitting refutation, question by Russell Bertrand, forced, without delay, the great thinker of Mecklemburg to add a last paragraph to his system, show me a great thinker who would resist the truth, accepting the reversible disaster. His foundations in ruin, his logic flawed, his work wasted, and his two volumes imagine the colossal set back, odd load and ballast for the refuge cart.”
Russell’s Paradox.

- "This statement is false"
Russell’s Paradox.

- "This statement is false"
  Is the statement above true?

- A barber says "I shave all and only those men who do not shave themselves."
Russell’s Paradox.

- "This statement is false"
  Is the statement above true?

- A barber says "I shave all and only those men who do not shave themselves."
  Who shaves the barber??
Russell’s Paradox.

- "This statement is false"
  Is the statement above true?

- A barber says "I shave all and only those men who do not shave themselves."
  Who shaves the barber??

- Self reference........
Russell’s Paradox.

Naive Set Theory: Any definable collection is a set.
Russell’s Paradox.

Naive Set Theory: Any definable collection is a set.
Let’s think about the set of all sets that don’t contain themselves.
Russell’s Paradox.

Naive Set Theory: Any definable collection is a set.
Let’s think about the set of all sets that don’t contain themselves. Call it $A$. 
Russell’s Paradox.

Naive Set Theory: Any definable collection is a set.
Let’s think about the set of all sets that don’t contain themselves. Call it $A$.

Does $A$ contain itself?
Russell’s Paradox.

Naive Set Theory: Any definable collection is a set.
Let’s think about the set of all sets that don’t contain themselves. Call it $A$.
Does $A$ contain itself?
Oops!
Russell’s Paradox.

Naive Set Theory: Any definable collection is a set.
Let’s think about the set of all sets that don’t contain themselves. Call it $A$.

Does $A$ contain itself?

Oops!

What type of object is a set that contain sets?
Russell’s Paradox.

Naive Set Theory: Any definable collection is a set.
Let’s think about the set of all sets that don’t contain themselves. Call it $A$.

Does $A$ contain itself?
Oops!

What type of object is a set that contain sets?
Change Axioms!
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis...
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- Consistent: You can't prove false statements
- Complete: Everything true can be proven.

Other people in this story: Russell, Whitehead, Wittgenstein, Hilbert (We must know. We will know.)... Until 1931.
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- Consistent:
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- Consistent: You can’t prove false statements
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- Consistent: You can’t prove false statements
- Complete:
They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- Consistent: You can’t prove false statements
- Complete: Everything true can be proven.
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- Consistent: You can’t prove false statements
- Complete: Everything true can be proven.

Other people in this story:
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- Consistent: You can’t prove false statements
- Complete: Everything true can be proven.

Other people in this story: Russell
They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- **Consistent**: You can’t prove false statements
- **Complete**: Everything true can be proven.

Other people in this story: Russell, Whitehead
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- Consistent: You can’t prove false statements
- Complete: Everything true can be proven.

Other people in this story: Russell, Whitehead, Wittgenstein
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- **Consistent:** You can’t prove false statements
- **Complete:** Everything true can be proven.

Other people in this story: Russell, Whitehead, Wittgenstein, Hilbert (We must know. We will know.)...
Changing Axioms?

They did keep trying to put all of mathematics on a firm basis... Trying to find a set of axioms such that is

- **Consistent**: You can’t prove false statements
- **Complete**: Everything true can be proven.

Other people in this story: Russell, Whitehead, Wittgenstein, Hilbert
(We must know. We will know.) ... Until 1931.
Changing Axioms?

Kurt Gödel: Any set of axioms is either inconsistent (can prove false statements) or incomplete (true statements cannot be proven.) Concrete example: Continuum hypothesis (see official notes if interested)
Kurt Gödel:

Any set of axioms is either inconsistent (can prove false statements) or incomplete (true statements cannot be proven.) Concrete example: Continuum hypothesis (see official notes if interested)
Changing Axioms?

Kurt Gödel:
Any set of axioms is either
Changing Axioms?

Kurt Gödel:
Any set of axioms is either inconsistent (can prove false statements) or
Changing Axioms?

Kurt Gödel:
Any set of axioms is either inconsistent (can prove false statements) or incomplete (true statements cannot be proven.)
Changing Axioms?

Kurt Gödel:
Any set of axioms is either inconsistent (can prove false statements) or incomplete (true statements cannot be proven.)

Concrete example:
Continuum hypothesis (see official notes if interested)
Gödel

Russell was fine... but for two schizophrenic children.

Wittgenstein... multiple tragedies in his family.

Dangerous work?

See Logicomix by Doxiadis, Papadimitriou (my advisor!), Papadatos, Di Donna.
Gödel ..starved himself out of fear of being poisoned..
Gödel starved himself out of fear of being poisoned.

Russell
Gödel ..starved himself out of fear of being poisoned..

Russell .. was fine...
Gödel ..starved himself out of fear of being poisoned..
Russell .. was fine.....but for two schizophrenic children..
▶ Gödel ..starved himself out of fear of being poisoned..
▶ Russell .. was fine.....but for two schizophrenic children..
▶ Wittgenstein
Gödel ..starved himself out of fear of being poisoned..
Russell .. was fine.....but for two schizophrenic children..
Wittgenstein ... multiple tragedies in his family.
Gödel ..starved himself out of fear of being poisoned..
Russell .. was fine.....but for two schizophrenic children..
Wittgenstein ... multiple tragedies in his family.
Dangerous work?
- Gödel ..starved himself out of fear of being poisoned..
- Russell .. was fine.....but for two schizophrenic children..
- Wittgenstein ... multiple tragedies in his family.
- Dangerous work?
- See Logicomix by Doxiadis, Papadimitriou (my advisor!), Papadatos, Di Donna.
Next Topic: Undecidability.

- Undecidability. A happy ending?

Thus, for any nondeterministic Turing machine $M$ that runs in some polynomial time $p(n)$, we can devise an algorithm that takes an input $w$ of length $n$ and produces $E_{M,w}$. The running time is $O(p^2(n))$ on a multitape deterministic Turing machine and...

Man, I just wanted to learn how to program video games.

SIPSE CH 7

$N_1 = \{A \vdash V B_1 \} A \{A \vdash V B_1 \} A \cdots A$

$N = N_1$
Turing
Is it actually useful?

Turing: Write me a program checker!

A program that checks that the compiler works!

How about... Check that the compiler terminates on a certain input.

\[\text{HALT}(P, I)\]

\[P\] - program
\[I\] - input.

Determines if \(P(I)\) (run on \(I\)) halts or loops forever.

Notice: Need a computer... with the notion of a stored program!!!!

Program is a text string.
Text string can be an input to a program.
Program can be an input to a program.
Is it actually useful?

Turing: Write me a program checker!
A program that checks that the compiler works!
Is it actually useful?

Turing: Write me a program checker!
A program that checks that the compiler works!
How about.. Check that the compiler terminates on a certain input.
Is it actually useful?

Turing: Write me a program checker!
A program that checks that the compiler works!
How about.. Check that the compiler terminates on a certain input.
\[ HALT(P, I) \]
Is it actually useful?

Turing: Write me a program checker!

A program that checks that the compiler works!

How about.. Check that the compiler terminates on a certain input.

\textit{HALT}(P, I)

\begin{itemize}
  \item \textit{P} - program
\end{itemize}
Is it actually useful?

Turing: Write me a program checker!

A program that checks that the compiler works!

How about.. Check that the compiler terminates on a certain input.

\[\text{HALT}(P, I)\]
\begin{itemize}
  \item \textit{P} - program
  \item \textit{I} - input.
\end{itemize}
Is it actually useful?

Turing: Write me a program checker!

A program that checks that the compiler works!

How about.. Check that the compiler terminates on a certain input.

$HALT(P, I)$

$P$ - program

$I$ - input.

Determines if $P(I)$ ($P$ run on $I$) halts or loops forever.

Notice:
...with the notion of a stored program!!!!

Program is a text string.

Text string can be an input to a program.

Program can be an input to a program.
Is it actually useful?

Turing: Write me a program checker!
A program that checks that the compiler works!
How about.. Check that the compiler terminates on a certain input.

$\text{HALT}(P, I)$

$P$ - program

$I$ - input.

Determines if $P(I)$ ($P$ run on $I$) halts or loops forever.

Notice:
Is it actually useful?

Turing: Write me a program checker!
A program that checks that the compiler works!
How about.. Check that the compiler terminates on a certain input.

\textit{HALT}(P, I)

\begin{itemize}
  \item \textit{P} - program
  \item \textit{I} - input.
\end{itemize}

Determines if \textit{P}(\textit{I}) (\textit{P} run on \textit{I}) halts or loops forever.

Notice:
Need a computer
Is it actually useful?

Turing: Write me a program checker!
A program that checks that the compiler works!
How about.. Check that the compiler terminates on a certain input.

\[ \text{HALT}(P, I) \]

- \( P \) - program
- \( I \) - input.

Determines if \( P(I) \) (\( P \) run on \( I \)) halts or loops forever.

Notice:
Need a computer
...with the notion of a stored program!!!!
Is it actually useful?

Turing: Write me a program checker!
A program that checks that the compiler works!
How about.. Check that the compiler terminates on a certain input.

$HALT(P, I)$
- $P$ - program
- $I$ - input.

Determines if $P(I)$ ($P$ run on $I$) halts or loops forever.

Notice:
Need a computer
...with the notion of a stored program!!!!
Program is a text string.
Is it actually useful?

Turing: Write me a program checker!
A program that checks that the compiler works!
How about.. Check that the compiler terminates on a certain input.

\[ \text{HALT}(P, I) \]

- \( P \) - program
- \( I \) - input.

Determines if \( P(I) \) (\( P \) run on \( I \)) halts or loops forever.

Notice:
Need a computer
...with the notion of a stored program!!!!
Program is a text string.
Text string can be an input to a program.
Is it actually useful?

Turing: Write me a program checker!
A program that checks that the compiler works!
How about.. Check that the compiler terminates on a certain input.

\[ \text{HALT}(P, I) \]
\[ P - \text{program} \]
\[ I - \text{input}. \]

Determines if \( P(I) \) (\( P \) run on \( I \)) halts or loops forever.

Notice:
Need a computer
...with the notion of a stored program!!!!
Program is a text string.
Text string can be an input to a program.
Program can be an input to a program.
Implementing HALT.

HALT \((P, I)\)

- **P** - program
- **I** - input.

Determines if \(P(I)\) (\(P\) run on \(I\)) halts or loops forever.

Run \(P\) on \(I\) and check!

How long do you wait?

Something about infinity here, maybe?

Theorem:

There is no program HALT.
Implementing HALT.

\[ HALT(P, I) \]
Implementing HALT.

\[ \text{HALT}(P, I) \]
\[ P \] - program
Implementing HALT.

\[ \text{HALT}(P, I) \]
\[ P \text{ - program} \]
\[ I \text{ - input.} \]
Implementing HALT.

\[ \text{HALT}(P, I) \]

- \( P \) - program
- \( I \) - input.

Determines if \( P(I) \) (\( P \) run on \( I \)) halts or loops forever.
Implementing HALT.

\[ \text{HALT}(P, I) \]

- \( P \) - program
- \( I \) - input.

Determines if \( P(I) \) (\( P \) run on \( I \)) halts or loops forever.

Run \( P \) on \( I \) and check!
Implementing HALT.

\[ \text{HALT}(P, I) \]
\[ P - \text{program} \]
\[ I - \text{input.} \]

Determines if \( P(I) \) (\( P \) run on \( I \)) halts or loops forever.

Run \( P \) on \( I \) and check!

How long do you wait?
Implementing HALT.

\[ \text{HALT}(P, I) \]

- \( P \) - program
- \( I \) - input.

Determines if \( P(I) \) (\( P \) run on \( I \)) halts or loops forever.

Run \( P \) on \( I \) and check!

How long do you wait?

Something about infinity here, maybe?
Implementing HALT.

\[ HALT(P, I) \]

\[ P \text{ - program} \]
\[ I \text{ - input.} \]

Determines if \( P(I) \) (\( P \) run on \( I \)) halts or loops forever.

Run \( P \) on \( I \) and check!

How long do you wait?

Something about infinity here, maybe?

**Theorem**: There is no program HALT.
Halt does not exist.

**Proof:**

```python
import HALT;
function Turing( Program P )
{
    if ( HALT( P , P .toString() ) == "halts" )
       while(true);
    else
       system.exit();
}
Run Turing(Turing).

Does Turing(Turing) halt?

Turing(Turing) halts ⇒ HALT(Turing, Turing.toString() ) = halts ⇒ Turing(Turing) loops forever.

Turing(Turing) loops forever ⇒ HALT(Turing, Turing.toString() ) ≠ halts ⇒ Turing(Turing) halts. ( goes to system.exit() )

Contradiction.

Program HALT does not exist!
Halt does not exist.

Proof: Assume there is a program \textit{HALT}(\cdot,\cdot).
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot,\cdot)$.

Code:
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot,\cdot)$.

Code:
import HALT;

Run Turing(Turing).

Does Turing(Turing) halt?

Turing(Turing) halts $\Rightarrow$ then $HALT(Turing, Turing.toString())$ halts $\Rightarrow$ Turing(Turing) loops forever.

Turing(Turing) loops forever $\Rightarrow$ then $HALT(Turing, Turing.toString()) \neq$ halts $\Rightarrow$ Turing(Turing) halts. (goes to system.exit())

Contradiction.

Program HALT does not exist!
Halt does not exist.

Proof: Assume there is a program $HALT(·, ·)$.

Code:
import HALT;
function Turing( Program P ) {

Run Turing(Turing).
Does Turing(Turing) halt?

Turing(Turing) halts $⇒$ then HALT(Turing, Turing.toString() ) $⇒$ Turing(Turing) loops forever.

Turing(Turing) loops forever $⇒$ then HALT(Turing, Turing.toString() ) $̸⇒$ Turing(Turing) halts. ( goes to system.exit() )
Contradiction.

Program HALT does not exist!
Halt does not exist.

Proof: Assume there is a program $HALT(\cdot, \cdot)$.

Code:

```python
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true);
    else:
        system.exit();
}
Run Turing(Turing).

Does Turing(Turing) halt?

Turing(Turing) halts $\Rightarrow$ then $HALT(Turing, Turing.toString() ) = halts$ $\Rightarrow$ Turing(Turing) loops forever.

Turing(Turing) loops forever $\Rightarrow$ then $HALT(Turing, Turing.toString() ) \neq halts$ $\Rightarrow$ Turing(Turing) halts. ( goes to system.exit() )

Contradiction.

Program $HALT$ does not exist!
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot, \cdot)$.

Code:

```python
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts"):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
Run Turing(Turing).
Does Turing(Turing) halt?
Turing(Turing) halts
=⇒ then HALT(Turing, Turing.toString() )
=⇒ Turing(Turing) loops forever.
Turing(Turing) loops forever
=⇒ then HALT(Turing, Turing.toString() ) \neq halts
=⇒ Turing(Turing) halts. (goes to system.exit())
Contradiction.
Program HALT does not exist!
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot,\cdot)$.

Code:

```python
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
```

Run Turing(Turing).
Does Turing(Turing) halt?

Turing(Turing) halts $\Rightarrow$ then $HALT(Turing, Turing.toString()) = halts$ $\Rightarrow$ Turing(Turing) loops forever.

Turing(Turing) loops forever $\Rightarrow$ then $HALT(Turing, Turing.toString()) \neq halts$ $\Rightarrow$ Turing(Turing) halts. (goes to system.exit())

Contradiction.

Program HALT does not exist!
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot,\cdot)$.

Code:
```
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
Run Turing(Turing).
```
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot, \cdot)$.

Code:

```java
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
Run Turing(Turing).
Does Turing(Turing) halt?
```
**Halt does not exist.**

**Proof:** Assume there is a program $HALT(\cdot,\cdot)$.

Code:

```python
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
Run Turing(Turing).
Does Turing(Turing) halt?
Turing(Turing) halts
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot,\cdot)$.

Code:

```python
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}

Run Turing(Turing).

Does $Turing(Turing)$ halt?

$Turing(Turing)$ halts

$\implies$ then $HALT(Turing, Turing.toString()) = halts$
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot,\cdot)$.

Code:
```plaintext
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
Run Turing(Turing).
Does Turing(Turing) halt?

Turing(Turing) halts
$\implies$ then $HALT(Turing, Turing.toString() ) = halts$
$\implies$ Turing(Turing) loops forever.
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot, \cdot)$.

Code:

```python
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
Run Turing(Turing).
Does Turing(Turing) halt?

Turing(Turing) halts
⇒ then HALT(Turing, Turing.toString() ) = halts
⇒ Turing(Turing) loops forever.

Turing(Turing) loops forever
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot, \cdot)$.

Code:

```plaintext
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}

Run Turing(Turing).
Does Turing(Turing) halt?

Turing(Turing) halts
$\implies$ then HALT(Turing, Turing.toString() ) = halts
$\implies$ Turing(Turing) loops forever.

Turing(Turing) loops forever
$\implies$ then HALT(Turing, Turing.toString() ) $\neq$ halts
Halt does not exist.

Proof: Assume there is a program $HALT(\cdot,\cdot)$.

Code:

```python
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
Run Turing(Turing).
Does Turing(Turing) halt?

Turing(Turing) halts
$\implies$ then HALT(Turing, Turing.toString() ) = halts
$\implies$ Turing(Turing) loops forever.

Turing(Turing) loops forever
$\implies$ then HALT(Turing, Turing.toString() ) $\neq$ halts
$\implies$ Turing(Turing) halts. ( goes to system.exit() )
Halt does not exist.

**Proof:** Assume there is a program \( HALT(\cdot,\cdot) \).

Code:

```plaintext
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}

Run Turing(Turing).
Does \texttt{Turing(Turing)} halt?

\texttt{Turing(Turing)} halts
\implies then HALT(Turing, Turing.toString() ) = halts
\implies Turing(Turing) loops forever.

\texttt{Turing(Turing)} loops forever
\implies then HALT(Turing, Turing.toString() ) \neq halts
\implies Turing(Turing) halts. ( goes to system.exit() )

Contradiction.
Halt does not exist.

**Proof:** Assume there is a program $HALT(·, ·)$.

Code:

```python
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
Run Turing(Turing).
Does Turing(Turing) halt?

Turing(Turing) halts
$\implies$ then HALT(Turing, Turing.toString() ) $=$ halts
$\implies$ Turing(Turing) loops forever.

Turing(Turing) loops forever
$\implies$ then HALT(Turing, Turing.toString() ) $\neq$ halts
$\implies$ Turing(Turing) halts. ( goes to system.exit() )

Contradiction. Program HALT does not exist!
Halt does not exist.

**Proof:** Assume there is a program $HALT(\cdot,\cdot)$.

Code:
```java
import HALT;
function Turing( Program P ) {
    if ( HALT( P, P.toString() ) == "halts" ):
        while(true); (go in an infinite loop)
    else:
        system.exit();
}
Run Turing(Turing).
Does Turing(Turing) halt?

Turing(Turing) halts
$\implies$ then $HALT(Turing, Turing.toString() ) = \text{halts}$
$\implies$ Turing(Turing) loops forever.

Turing(Turing) loops forever
$\implies$ then $HALT(Turing, Turing.toString() ) \neq \text{halts}$
$\implies$ Turing(Turing) halts. (goes to system.exit() )

Contradiction. Program HALT does not exist!
Another view of proof: diagonalization.

Any program is a fixed length string.
Another view of proof: diagonalization.

Any program is a fixed length string. Fixed length strings are enumerable.
Another view of proof: diagonalization.

Any program is a fixed length string.  
Fixed length strings are enumerable.  
If HALT existed, we could use it to make the following table:
Another view of proof: diagonalization.

Any program is a fixed length string. Fixed length strings are enumerable. If HALT existed, we could use it to make the following table:

<table>
<thead>
<tr>
<th></th>
<th>&quot;P_1&quot;</th>
<th>&quot;P_2&quot;</th>
<th>&quot;P_3&quot;</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_1</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>...</td>
</tr>
<tr>
<td>P_2</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>P_3</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>...</td>
</tr>
</tbody>
</table>

Program \( P_1 \) halts on input "\( P_1 \)" and "\( P_2 \)", doesn’t halt on input "\( P_3 \)", and so on...
Another view of proof: diagonalization.

Any program is a fixed length string.
Fixed length strings are enumerable.
If HALT existed, we could use it to make the following table:

<table>
<thead>
<tr>
<th></th>
<th>&quot;P_1&quot;</th>
<th>&quot;P_2&quot;</th>
<th>&quot;P_3&quot;</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_1</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>...</td>
</tr>
<tr>
<td>P_2</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>P_3</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>...</td>
</tr>
</tbody>
</table>

Program P_1 halts on input "P_1" and "P_2", doesn’t halt on input "P_3", and so on...
Turing is different from every P_i on the diagonal.
Another view of proof: diagonalization.

Any program is a fixed length string.
Fixed length strings are enumerable.
If HALT existed, we could use it to make the following table:

<table>
<thead>
<tr>
<th></th>
<th>&quot;P1&quot;</th>
<th>&quot;P2&quot;</th>
<th>&quot;P3&quot;</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>P₁</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>...</td>
</tr>
<tr>
<td>P₂</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>P₃</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Program \( P₁ \) halts on input "\( P₁ \)" and "\( P₂ \)", doesn’t halt on input "\( P₃ \)", and so on...
Turing is different from every \( P_i \) on the diagonal.
Turing is not on list.
Another view of proof: diagonalization.

Any program is a fixed length string. Fixed length strings are enumerable. If HALT existed, we could use it to make the following table:

<table>
<thead>
<tr>
<th></th>
<th>( P_1 )</th>
<th>( P_2 )</th>
<th>( P_3 )</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P_1 )</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>...</td>
</tr>
<tr>
<td>( P_2 )</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>( P_3 )</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Program \( P_1 \) halts on input \( P_1 \) and \( P_2 \), doesn’t halt on input \( P_3 \), and so on...

Turing is different from every \( P_i \) on the diagonal. Turing is not on list. But, Turing is a program.
Another view of proof: diagonalization.

Any program is a fixed length string.  
Fixed length strings are enumerable.  
If HALT existed, we could use it to make the following table:

<table>
<thead>
<tr>
<th></th>
<th>&quot;P_1&quot;</th>
<th>&quot;P_2&quot;</th>
<th>&quot;P_3&quot;</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_1</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>...</td>
</tr>
<tr>
<td>P_2</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>P_3</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>...</td>
</tr>
</tbody>
</table>

Program P_1 halts on input "P_1" and "P_2", doesn’t halt on input "P_3", and so on...
Turing is different from every P_i on the diagonal.
Turing is not on list. But, Turing is a program.
Turing can be constructed from Halt.
Another view of proof: diagonalization.

Any program is a fixed length string. Fixed length strings are enumerable. If HALT existed, we could use it to make the following table:

<table>
<thead>
<tr>
<th></th>
<th>&quot;P1&quot;</th>
<th>&quot;P2&quot;</th>
<th>&quot;P3&quot;</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>...</td>
</tr>
<tr>
<td>P2</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>P3</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>:</td>
<td>:</td>
<td>:</td>
<td>:</td>
</tr>
</tbody>
</table>

Program \( P_1 \) halts on input "\( P_1 \)" and "\( P_2 \)", doesn’t halt on input "\( P_3 \)", and so on...

Turing is different from every \( P_i \) on the diagonal. Turing is not on list. But, Turing is a program. Turing can be constructed from Halt.

Halt does not exist!
Another view of proof: diagonalization.

Any program is a fixed length string. Fixed length strings are enumerable. If HALT existed, we could use it to make the following table:

<table>
<thead>
<tr>
<th></th>
<th>$P_1$</th>
<th>$P_2$</th>
<th>$P_3$</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_1$</td>
<td>H</td>
<td>H</td>
<td>L</td>
<td>...</td>
</tr>
<tr>
<td>$P_2$</td>
<td>L</td>
<td>L</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>$P_3$</td>
<td>L</td>
<td>H</td>
<td>H</td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

Program $P_1$ halts on input "$P_1" and "$P_2"., doesn’t halt on input "$P_3", and so on...

Turing is different from every $P_i$ on the diagonal. Turing is not on list. But, Turing is a program. Turing can be constructed from Halt.

Halt does not exist!
Wow, that was easy!
Wow, that was easy!
We should be famous!
No computers for Turing!

In Turing’s time.
No computers for Turing!

In Turing’s time.
No computers.
No computers for Turing!

In Turing’s time.
No computers.
Concept of program as data wasn’t really there.
Undecidable problems.

Does a program ever print “Hello World”?
Undecidable problems.

Does a program ever print “Hello World”?
Find exit points and add statement: **Print** “Hello World.”
Undecidable problems.

Does a program ever print “Hello World”? Find exit points and add statement: **Print** “Hello World.”

Is there program that makes other programs faster?
Undecidable problems.

Does a program ever print “Hello World”? Find exit points and add statement: **Print** “Hello World.”

Is there a program that makes other programs faster?

Is there a program that decides if two other programs are equivalent?
Undecidable problems.

Does a program ever print “Hello World”? Find exit points and add statement: Print “Hello World.”

Is there program that makes other programs faster?

Is there program that decides if two other programs are equivalent?

Does this computer program have any security vulnerabilities?
More about Alan Turing.

- Brilliant codebreaker during WWII, helped break German Enigma Code (which probably shortened war by 1 year).
More about Alan Turing.

- Brilliant codebreaker during WWII, helped break German Enigma Code (which probably shortened war by 1 year).
- Seminal paper in numerical analysis:
More about Alan Turing.

- Brilliant codebreaker during WWII, helped break German Enigma Code (which probably shortened war by 1 year).
- Seminal paper in numerical analysis: Condition number.
More about Alan Turing.

- Brilliant codebreaker during WWII, helped break German Enigma Code (which probably shortened war by 1 year).
- Seminal paper in numerical analysis: Condition number.
- Seminal paper in mathematical biology.
More about Alan Turing.

- Brilliant codebreaker during WWII, helped break German Enigma Code (which probably shortened war by 1 year).
- Seminal paper in numerical analysis: Condition number.
- Seminal paper in mathematical biology.
- Movie:
Turing: personal.

Tragic ending...
Turing: personal.

Tragic ending...

- Arrested as a homosexual
Turing: personal.

Tragic ending...

- Arrested as a homosexual
- Given choice of prison or (quackish) injections to eliminate sex drive;

(A bite from the apple....) accident?
Turing: personal.

Tragic ending...

- Arrested as a homosexual
- Given choice of prison or (quackish) injections to eliminate sex drive;
- Took injections.
Turing: personal.

Tragic ending...

- Arrested as a homosexual
- Given choice of prison or (quackish) injections to eliminate sex drive;
- Took injections.
- Lost security clearance...

(A bite from the apple....) accident?
Turing: personal.

Tragic ending...

- Arrested as a homosexual
- Given choice of prison or (quackish) injections to eliminate sex drive;
- Took injections.
- Lost security clearance...
- Denied entry into the United States...
Turing: personal.

Tragic ending...

- Arrested as a homosexual
- Given choice of prison or (quackish) injections to eliminate sex drive;
- Took injections.
- Lost security clearance...
- Denied entry into the United States...
- Suffered from depression;
Turing: personal.

Tragic ending...

- Arrested as a homosexual
- Given choice of prison or (quackish) injections to eliminate sex drive;
- Took injections.
- Lost security clearance...
- Denied entry into the United States...
- Suffered from depression;
- Suicided with cyanide at age 42.
Turing: personal.

Tragic ending...

- Arrested as a homosexual
- Given choice of prison or (quackish) injections to eliminate sex drive;
- Took injections.
- Lost security clearance...
- Denied entry into the United States...
- Suffered from depression;
- Suicided with cyanide at age 42.
  (A bite from the apple....)
Turing: personal.

Tragic ending...

- Arrested as a homosexual
- given choice of prison or (quackish) injections to eliminate sex drive;
- took injections.
- lost security clearance...
- denied entry into the United States...
- suffered from depression;
- suicided with cyanide at age 42.
  (A bite from the apple....) accident?
British Apology.

British Apology.

Gordon Brown. 2009. “Thousands of people have come together to demand justice for Alan Turing and recognition of the appalling way he was treated.
Gordon Brown. 2009. “Thousands of people have come together to demand justice for Alan Turing and recognition of the appalling way he was treated. While Turing was dealt with under the law of the time and we can’t put the clock back,
Gordon Brown. 2009. “Thousands of people have come together to demand justice for Alan Turing and recognition of the appalling way he was treated. While Turing was dealt with under the law of the time and we can’t put the clock back, his treatment was of course utterly unfair and I am pleased to have the chance to say how deeply sorry I and we all are for what happened to him.”

2013. Granted Royal pardon.
Gordon Brown. 2009. “Thousands of people have come together to demand justice for Alan Turing and recognition of the appalling way he was treated. While Turing was dealt with under the law of the time and we can’t put the clock back, his treatment was of course utterly unfair and I am pleased to have the chance to say how deeply sorry I and we all are for what happened to him. Alan and the many thousands of other gay men who were convicted as he was convicted under homophobic laws were treated terribly.”
Gordon Brown. 2009. “Thousands of people have come together to demand justice for Alan Turing and recognition of the appalling way he was treated. While Turing was dealt with under the law of the time and we can’t put the clock back, his treatment was of course utterly unfair and I am pleased to have the chance to say how deeply sorry I and we all are for what happened to him. Alan and the many thousands of other gay men who were convicted as he was convicted under homophobic laws were treated terribly. Over the years millions more lived in fear of conviction.
Gordon Brown. 2009. “Thousands of people have come together to demand justice for Alan Turing and recognition of the appalling way he was treated. While Turing was dealt with under the law of the time and we can’t put the clock back, his treatment was of course utterly unfair and I am pleased to have the chance to say how deeply sorry I and we all are for what happened to him. Alan and the many thousands of other gay men who were convicted as he was convicted under homophobic laws were treated terribly. Over the years millions more lived in fear of conviction. ..........
Gordon Brown. 2009. “Thousands of people have come together to demand justice for Alan Turing and recognition of the appalling way he was treated. While Turing was dealt with under the law of the time and we can’t put the clock back, his treatment was of course utterly unfair and I am pleased to have the chance to say how deeply sorry I and we all are for what happened to him. Alan and the many thousands of other gay men who were convicted as he was convicted under homophobic laws were treated terribly. Over the years millions more lived in fear of conviction. ..........

So on behalf of the British government, and all those who live freely thanks to Alan’s work I am very proud to say: we’re sorry, you deserved so much better.”
Gordon Brown. 2009. “Thousands of people have come together to demand justice for Alan Turing and recognition of the appalling way he was treated. While Turing was dealt with under the law of the time and we can’t put the clock back, his treatment was of course utterly unfair and I am pleased to have the chance to say how deeply sorry I and we all are for what happened to him. Alan and the many thousands of other gay men who were convicted as he was convicted under homophobic laws were treated terribly. Over the years millions more lived in fear of conviction. ........
So on behalf of the British government, and all those who live freely thanks to Alan’s work I am very proud to say: we’re sorry, you deserved so much better.”
2013. Granted Royal pardon.
Infinity is interesting!
Summary

Infinity is interesting!
And mind boggling
Infinity is interesting!
   And mind boggling
Computer Programs are an interesting thing.
Infinity is interesting!
And mind boggling
Computer Programs are an interesting thing.
Like Math.
Infinity is interesting!
   And mind boggling
Computer Programs are an interesting thing.
   Like Math.
Deep connection between mathematical proofs and computer programs.
Infinity is interesting!
And mind boggling
Computer Programs are an interesting thing.
Like Math.
Deep connection between mathematical proofs and computer programs.
Computer Programs cannot completely “understand” computer programs.
Infinity is interesting!
   And mind boggling
Computer Programs are an interesting thing.
   Like Math.
Deep connection between mathematical proofs and computer programs.
Computer Programs cannot completely “understand” computer programs.
Example: no computer program can tell if any other computer program HALTS.
Infinity is interesting!
   And mind boggling
Computer Programs are an interesting thing.
   Like Math.
Deep connection between mathematical proofs and computer programs.
Computer Programs cannot completely “understand” computer programs.
Example: no computer program can tell if any other computer program HALTS.
Programming is a super power.
HOW MATH WORKS:

STEP 1: INSIGHT
MY GOD. I WONDER IF THIS IS TRUE.

STEP 2: RESISTANCE
IMPOSSIBLE! INSANE! IT'S NOT JUST INCORRECT; IT'S AN ENTIRELY NEW CATEGORY OF STUPID!

STEP 3: DEBATE
IT LOOKS RIGHT, BUT IT CAN'T BE RIGHT. PERHAPS WE COULD RESTRUCTURE ALL OF MATHEMATICS IN A WAY THAT MAKES IT WRONG.

STEP 4: ADDITIONAL DECADES OF DEBATE.
YOU SAY, Ex falso quodlibet.
I SEE YOUR MOTHERS WITH THEIR THING.
THE FACULTY OF MADNESS.

STEP 5: CHANGING OF THE GUARD.
I WILL NEVER UNDERSTAND IT. I WILL NEVER BELIEVE IT. AS I GO INTO DEATH, WITH MY FINAL BREATHE I SPIT ON YOUR THEOREM.

STEP 6: TRANSMISSION TO STUDENTS.
HOW DO YOU NOT GET THIS CONCEPT? WE SPENT AN HOUR ON IT YESTERDAY.