Welcome to CS103!
Are there “laws of physics” in computer science?
Key Questions in CS103

• What problems can you solve with a computer?
  • *Computability Theory*

• How can we be certain in our answers to these questions?
  • *Discrete Mathematics*
The Teaching Team

Amy Liu (Instructor)
Caroline Gao (TA)
Michel Dellepere (TA)
Ophir Horovitz (TA)
Lucas Sato (TA)

Amy’s Email: liuamyj@cs.stanford.edu

Staff Email List: cs103-sum2122-staff@lists.stanford.edu
Course Website

https://cs103.stanford.edu

All course content will be hosted here, except for lecture videos.
Some problem sets will have small coding components. We’ll also reference some concepts from CS106B/X, particularly recursion, throughout the quarter.

There aren’t any math prerequisites for this course – high-school algebra should be enough!
Problem Set 0

• Your first assignment, Problem Set 0, goes out today. It’s due Friday at 2:30PM Pacific.

• This assignment requires you to set up your development environment and to get set up on GradeScope.

• There’s no coding involved, but it’s good to start early anyway in case you encounter any technical issues setting up.
Recommended Reading

**How to Read and Do Proofs**
SIXTH EDITION
Daniel Solow
Wiley

**Introduction to the Theory of Computation**
Third Edition
Michael Sipser
Grading
Eight Problem Sets

Problem sets may be completed individually or in pairs.

Grading

50%

Problem Sets
Midterm Exam
48-hour take-home exam.
July 15th – 17th.
Grading

- Problem Sets: 50%
- Midterm: 20%
- Final Exam: 30%

**Final Exam**

Take-Home Exam. August 10th through August 12th
How to Succeed in CS103
“A little slope makes up for a lot of y-intercept.”
- John Ousterhout
We've got a big journey ahead of us.

*Let's get started!*
Introduction to Set Theory
“CS103 students”

“The chemical elements”

“US coins”

“Cool people”

“Cute animals”
A set is an unordered collection of distinct objects, which may be anything, including other sets.
A set is an unordered collection of distinct objects, which may be anything, including other sets.
A **set** is an unordered collection of distinct objects, which may be anything, including other sets.
Two sets are equal when they have the same contents, ignoring order.
Two sets are equal when they have the same contents, ignoring order.
Two sets are equal when they have the same contents, ignoring order.
Two sets are equal when they have the same contents, ignoring order.
Two sets are equal when they have the same contents, ignoring order.
Two sets are equal when they have the same contents, ignoring order.

These are two different descriptions of exactly the same set.
Sets cannot contain duplicate elements. Any repeated elements are ignored.
Sets cannot contain duplicate elements. Any repeated elements are ignored.
Sets cannot contain duplicate elements. Any repeated elements are ignored.

These are two different descriptions of exactly the same set.
The objects that make up a set are called the elements of that set.
The objects that make up a set are called the **elements** of that set.
The objects that make up a set are called the *elements* of that set.

This symbol means “is an element of.”
The objects that make up a set are called the **elements** of that set.
Sets can contain any number of elements.
Sets can contain any number of elements.

The **empty set** is the set with no elements.

We denote the empty set using this symbol: \( \emptyset \).

Sets can contain any number of elements.
Question: Are these objects equal?
Question: Are these objects equal?

This is a number.

This is a set.
It contains a number.
Question: Are these objects equal?
Question: Are these objects equal?
Question: Are these objects equal?

This is the empty set.

This is a set with the empty set in it.
Question: Are these objects equal?
No object \( x \) is equal to the set containing \( x \).

\[ x \not\in \{ x \} \]

This is a box that has \( x \) inside it.

This is \( x \) itself.
Infinite Sets

- Some sets contain *infinitely many* elements!
- The set $\mathbb{N} = \{ 0, 1, 2, 3, \ldots \}$ is the set of all the *natural numbers*.
  - Some mathematicians don't include zero; in this class, assume that 0 is a natural number.
- The set $\mathbb{Z} = \{ \ldots, -2, -1, 0, 1, 2, \ldots \}$ is the set of all the *integers*.
  - $Z$ is from German “Zahlen.”
- The set $\mathbb{R}$ is the set of all *real numbers*.
  - $e \in \mathbb{R}$, $\pi \in \mathbb{R}$, $4 \in \mathbb{R}$, etc.
Describing Complex Sets

• Here are some English descriptions of infinite sets:
  “The set of all even natural numbers.”
  “The set of all real numbers less than 137.”
  “The set of all negative integers.”

• To describe complex sets like these mathematically, we'll use set-builder notation.
Even Natural Numbers

\{ \ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \ \}
Even Natural Numbers

\{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \}
Even Natural Numbers

\{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \}

The set of all $n$
Even Natural Numbers

\{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \}

The set of all $n$ where
Even Natural Numbers

\{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \}

The set of all \( n \) where \( n \) is a natural number.
Even Natural Numbers

\{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \}

The set of all \( n \)

where

\( n \) is a natural number

and \( n \) is even
Even Natural Numbers

\[ \{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \} \]

The set of all \( n \) where \( n \) is a natural number and \( n \) is even.

\{ 0, 2, 4, 6, 8, 10, 12, 14, 16, \ldots \}
Set Builder Notation

• A set may be specified in **set-builder notation**:

\[
\{ \, x \mid \text{some property } x \text{ satisfies} \, \}
\]

\[
\{ \, x \in S \mid \text{some property } x \text{ satisfies} \, \}
\]

• For example:

\[
\{ \, n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \, \}
\]

\[
\{ \, C \mid C \text{ is a set of US currency} \, \}
\]

\[
\{ \, r \in \mathbb{R} \mid r < 3 \, \}
\]

\[
\{ \, n \in \mathbb{N} \mid n < 3 \, \} \quad \text{(the set } \{0, 1, 2\})
\]
Combining Sets
Venn Diagrams

$A = \{ 1, 2, 3 \}$
$B = \{ 3, 4, 5 \}$
Venn Diagrams

\[ A = \{ 1, 2, 3 \} \]
\[ B = \{ 3, 4, 5 \} \]
\begin{align*}
A &= \{1, 2, 3\} \\
B &= \{3, 4, 5\}
\end{align*}
Venn Diagrams

\[ A = \{ 1, 2, 3 \} \]
\[ B = \{ 3, 4, 5 \} \]
A = \{ 1, 2, 3 \}
B = \{ 3, 4, 5 \}
Venn Diagrams

\[ A = \{1, 2, 3\} \]
\[ B = \{3, 4, 5\} \]
Venn Diagrams

\[ A \cup B \]

\[ A = \{1, 2, 3\} \]
\[ B = \{3, 4, 5\} \]

Union
\[ A \cup B \]
\[ \{1, 2, 3, 4, 5\} \]
Venn Diagrams

\[ A = \{ 1, 2, 3 \} \]
\[ B = \{ 3, 4, 5 \} \]
Venn Diagrams

$A = \{ 1, 2, 3 \}$

$B = \{ 3, 4, 5 \}$

Intersection

$A \cap B = \{ 3 \}$
A = \{ 1, 2, 3 \}
B = \{ 3, 4, 5 \}
Venn Diagrams

\[ A = \{ 1, 2, 3 \} \]
\[ B = \{ 3, 4, 5 \} \]

Difference
\[ A - B \]
\[ \{ 1, 2 \} \]
Venn Diagrams

\[ A = \{ 1, 2, 3 \} \]
\[ B = \{ 3, 4, 5 \} \]

Difference
\[ A \setminus B \]
\[ \{ 1, 2 \} \]
Venn Diagrams

A = \{ 1, 2, 3 \}
B = \{ 3, 4, 5 \}
Venn Diagrams

A = \{ 1, 2, 3 \}
B = \{ 3, 4, 5 \}

Symmetric Difference
\[ A \Delta B = \{ 1, 2, 4, 5 \} \]
Venn Diagrams

\[ A \cap B \]

\[ A \Delta B \]
Venn Diagrams
Venn Diagrams for Four Sets

Question to ponder: why don’t we just draw four circles?
Venn Diagrams for Five Sets

[Diagram with various sets and intersections labeled with words like "Bang Theory," "Enchilada," "Board," "Foot," "League," etc.]
Venn Diagrams for Seven Sets

http://moebio.com/research/sevensets/
Subsets and Power Sets
Subsets

• A set $S$ is called a \textit{subset} of a set $T$ (denoted $S \subseteq T$) if all elements of $S$ are also elements of $T$.

• Examples:
  
  • $\{ 1, 2, 3 \} \subseteq \{ 1, 2, 3, 4 \}$
  • $\{ b, c \} \subseteq \{ a, b, c, d \}$
  • $\{ H, \text{He}, \text{Li} \} \subseteq \{ H, \text{He}, \text{Li} \}$
  • $\mathbb{N} \subseteq \mathbb{Z}$ \textit{(every natural number is an integer)}
  • $\mathbb{Z} \subseteq \mathbb{R}$ \textit{(every integer is a real number)}
Subsets and Elements

Set $S$
Subsets and Elements

Set $S$ contains the elements $\{2\}$, with $2 \in S$. There are also other elements, such as a smiley face and a coin, indicating the diversity within $S$. The star symbol also represents an element of $S$. 
Subsets and Elements

Set $S$ includes the element 2.
Subsets and Elements

Set $S$ contains the set $\{2\}$, and $2 \in S$. The element 2 is represented by the smiling face and the coin.
Subsets and Elements

Set $S$

$\{2\} \in S$
Subsets and Elements

General intuition: $x \in S$ means you can **point at $x$ inside of $S$**.
Subsets and Elements

Set $S$

$\{2\}$
Subsets and Elements

Set $S$

$\{2\} \subseteq S$
Subsets and Elements

Set $S$

$\{2\} \subseteq S$

$\{1, 2\} \subseteq S$
Subsets and Elements

Set $S$

$\{2\}$

$2$

$\{\star, 2\} \notin S$
Subsets and Elements

Set $S$ is not contained in $S$.

$\{2\} \notin S$
Subsets and Elements

Set $S$

$\{2\} \in S$
Subsets and Elements

Set $S$

$\{2\} \in S$
Subsets and Elements

Set $S$

$\{2\}$

$\{2\} \subseteq S$
Subsets and Elements

Set $S$

$\{2\} \subseteq S$
Subsets and Elements

Set $S$

$\{2\} \subseteq S$
Subsets and Elements

General intuition: $A \subseteq B$ if you can form $A$ by circling elements of $B$. 

$\{2\} \subseteq S$
Set $S$ contains the subset $\{2\}$. The element 2 does not belong to $S$, denoted as $2 \notin S$. 
Subsets and Elements

Set $S$

\{2\}

\(2 \notin S\)  
(Since 2 isn't a set.)
Subsets and Elements

Set $S$

$\{2\}$

2

\star

\smiley
Subsets and Elements

Set $S$

$\emptyset \subseteq S$

$\{2\}$

2

$\emptyset \subseteq S$
Subsets and Elements

Set $S$

$\{2\}$

$\emptyset \subseteq S$
Subsets and Elements

Set $S$

$\{2\}$

$\emptyset \notin S$
Subsets and Elements

Set $S$

$\{2\}$

$\emptyset \notin S$
Subsets and Elements

• We say that $S \in T$ if, among the elements of $T$, one of them is exactly the object $S$.

• We say that $S \subseteq T$ if $S$ is a set and every element of $S$ is also an element of $T$. ($S$ has to be a set for the statement $S \subseteq T$ to be true.)

• Although these concepts are similar, they are not the same! Not all elements of a set are subsets of that set and vice-versa.

• We have a resource on the course website, the Guide to Elements and Subsets, that explores this in more depth.
This is the **power set** of $S$, the set of all subsets of $S$. We write the power set of $S$ as $\mathcal{P}(S)$.

Formally, $\mathcal{P}(S) = \{ T \mid T \subseteq S \}$.

*(Do you see why?)*
What is $\wp(\emptyset)$?

**Answer:** \{\emptyset\}

*Remember that $\emptyset \neq \{\emptyset\}$!*
Cardinality
Cardinality

- The **cardinality** of a set is the number of elements it contains.
- If $S$ is a set, we denote its cardinality as $|S|$.
- Examples:
  - $|\{\text{whimsy, mirth}\}| = 2$
  - $|\{\{a, b\}, \{c, d, e, f, g\}, \{h\}\}| = 3$
  - $|\{1, 2, 3, 3, 3, 3, 3\}| = 3$
  - $|\{n \in \mathbb{N} | n < 4\}| = |\{0, 1, 2, 3\}| = 4$
  - $|\emptyset| = 0$
  - $|\{\emptyset\}| = 1$
The Cardinality of $\mathbb{N}$

• What is $|\mathbb{N}|$?
  • There are infinitely many natural numbers.
  • $|\mathbb{N}|$ can't be a natural number, since it's infinitely large.
The Cardinality of $\mathbb{N}$

• What is $|\mathbb{N}|$?
  • There are infinitely many natural numbers.
  • $|\mathbb{N}|$ can't be a natural number, since it's infinitely large.

• We need to introduce a new term.

• Let's define $\aleph_0 = |\mathbb{N}|$.
  • $\aleph_0$ is pronounced “aleph-zero,” “aleph-nought,” or “aleph-null.”
Consider the set

\[ S = \{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \}. \]

What is \(|S|\)?
How Big Are These Sets?
How Big Are These Sets?
Comparing Cardinalities

• *By definition*, two sets have the same size if there is a way to pair their elements off without leaving any elements uncovered.

• The intuition:
Comparing Cardinalities

- By definition, two sets have the same size if there is a way to pair their elements off without leaving any elements uncovered.

- The intuition:

  Everything has been paired up, and this one is all alone.
Infinite Cardinalities

\[ S = \{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \} \]

Two sets have the same size if there is a way to pair their elements off without leaving any elements uncovered.
Infinite Cardinalities

\[ S = \{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \} \]

Two sets have the same size if there is a way to pair their elements off without leaving any elements uncovered.
Infinite Cardinalities

$$\mathbb{N} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad \ldots$$

$$S \quad 0 \quad 2 \quad 4 \quad 6 \quad 8 \quad \ldots$$

$$S = \{ \ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \ \}$$
Infinite Cardinalities

\[
\begin{align*}
\mathbb{N} & \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad \ldots \\
S & \quad 0 \quad 2 \quad 4 \quad 6 \quad 8 \quad 10 \quad 12 \quad 14 \quad 16 \quad \ldots
\end{align*}
\]

\[n \leftrightarrow 2n\]

\[S = \{ n \mid n \in \mathbb{N} \text{ and } n \text{ is even} \}\]

\[|S| = |\mathbb{N}| = \aleph_0\]
Infinite Cardinalities

\[ \mathbb{N} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad \ldots \]

\[ \mathbb{Z} \quad \ldots \quad -3 \quad -2 \quad -1 \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad \ldots \]
Infinite Cardinalities

\[ \mathbb{N} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad \ldots \]

\[ \mathbb{Z} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad \ldots \]

\[ \ldots \quad -3 \quad -2 \quad -1 \]
Infinite Cardinalities

\[ \mathbb{N} : 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ \ldots \]

\[ \mathbb{Z} : 0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ \ldots \]

\[ \ldots \ -3 \ -2 \ -1 \]
Infinite Cardinalities

Two sets have the same size if there is a way to pair their elements off without leaving any elements uncovered.
Infinite Cardinalities

\[ \mathbb{N} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad \ldots \]

\[ \mathbb{Z} \quad \ldots \quad -3 \quad -2 \quad -1 \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad \ldots \]
Infinite Cardinalities

\[ \mathbb{N} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad \ldots \]

\[ \mathbb{Z} \quad \ldots \quad -3 \quad -2 \quad -1 \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad \ldots \]
Infinite Cardinalities

\[ \mathbb{N} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad \ldots \]

\[ \mathbb{Z} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad \ldots \]

\[ \ldots \quad -3 \quad -2 \quad -1 \]
Infinite Cardinalities

\[ \mathbb{N} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad \ldots \]

\[ \mathbb{Z} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad \ldots \]

\[ \ldots \quad -3 \quad -2 \quad -1 \]
Infinite Cardinalities

\[ \mathbb{N} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad 7 \quad 8 \quad \ldots \]

\[ \mathbb{Z} \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad \ldots \]

...  -3  -2  -1

Pair nonnegative integers with even natural numbers.
Pair nonnegative integers with even natural numbers.
Pair nonnegative integers with even natural numbers.
Infinite Cardinalities

Pair nonnegative integers with even natural numbers.
Pair negative integers with odd natural numbers.
Infinite Cardinalities

Pair nonnegative integers with even natural numbers.
Pair negative integers with odd natural numbers.

\[ |\mathbb{N}| = |\mathbb{Z}| = \aleph_0 \]
Important Question:

Do all infinite sets have the same cardinality?
\[ S = \{ \} \]

\[ \wp(S) = \{ \emptyset, \{ \} \} \]

\[ |S| < |\wp(S)| \]
\[ S = \{ \emptyset, , \} \]

\[ \varphi(S) = \{ \emptyset, , \} \]

\[ |S| < |\varphi(S)| \]
\[ S = \{ a, b, c, d \} \]

\[ \mathcal{P}(S) = \{ \emptyset, \{ a \}, \{ b \}, \{ c \}, \{ d \}, \{ a, b \}, \{ a, c \}, \{ a, d \}, \{ b, c \}, \{ b, d \}, \{ c, d \}, \{ a, b, c \}, \{ a, b, d \}, \{ a, c, d \}, \{ b, c, d \}, \{ a, b, c, d \} \} \]

\[ |S| < |\mathcal{P}(S)| \]
If $|S|$ is infinite, what is the relation between $|S|$ and $|\wp(S)|$?

Does $|S| = |\wp(S)|$?
If $|S| = |\wp(S)|$, we can pair up the elements of $S$ and the elements of $\wp(S)$ without leaving anything out.
If $|S| = |\wp(S)|$, we can pair up the elements of $S$ and the elements of $\wp(S)$ without leaving anything out.
If $|S| = |\wp(S)|$, we can pair up the elements of $S$ and the subsets of $S$ without leaving anything out.
If $|S| = |\wp(S)|$, we can pair up the elements of $S$ and the subsets of $S$ without leaving anything out.
If $|S| = |\wp(S)|$, we can pair up the elements of $S$ and the subsets of $S$ without leaving anything out.

What would that look like?
$X_0 \leftrightarrow \left\{ \begin{array}{c} X_0, X_2, X_4, \ldots \end{array} \right\}$

$X_1 \leftrightarrow \left\{ \begin{array}{c} X_3, X_5, \ldots \end{array} \right\}$

$X_2 \leftrightarrow \left\{ \begin{array}{c} X_0, X_1, X_2, X_5, \ldots \end{array} \right\}$

$X_3 \leftrightarrow \left\{ \begin{array}{c} X_1, X_4, \ldots \end{array} \right\}$

$X_4 \leftrightarrow \left\{ \begin{array}{c} X_2, \ldots \end{array} \right\}$

$X_5 \leftrightarrow \left\{ \begin{array}{c} X_0, X_4, X_5, \ldots \end{array} \right\}$

$\ldots \leftrightarrow \left\{ \begin{array}{c} \ldots \end{array} \right\}$
<table>
<thead>
<tr>
<th>$X_0$</th>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$X_3$</th>
<th>$X_4$</th>
<th>$X_5$</th>
<th>$\ldots$</th>
</tr>
</thead>
</table>

$X_0 \leftrightarrow \{ X_0, X_2, X_4, \ldots \}$

$X_1 \leftrightarrow \{ X_3, X_5, \ldots \}$

$X_2 \leftrightarrow \{ X_0, X_1, X_2, X_5, \ldots \}$

$X_3 \leftrightarrow \{ X_1, X_4, \ldots \}$

$X_4 \leftrightarrow \{ X_2, \ldots \}$

$X_5 \leftrightarrow \{ X_0, X_4, X_5, \ldots \}$

$\ldots \leftrightarrow \{ \ldots \}$
<table>
<thead>
<tr>
<th></th>
<th>X0</th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
<th>X5</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>X0</td>
<td>X0,</td>
<td>X2,</td>
<td>X4,</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td></td>
<td>X3,</td>
<td>X5,</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X2</td>
<td>X0,</td>
<td>X2,</td>
<td>X5,</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X3</td>
<td>X1,</td>
<td></td>
<td>X4,</td>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X4</td>
<td></td>
<td>X2,</td>
<td></td>
<td>...</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X5</td>
<td>X0,</td>
<td></td>
<td>X4,</td>
<td>X5,</td>
<td>...</td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>
Which element is paired with this set?

$$\begin{bmatrix}
\{X_0, X_2, X_4, \ldots\} \\
\{X_0, X_2, \ldots\} \\
\{X_0, X_2, \ldots\} \\
\{X_1, X_3, X_5, \ldots\} \\
\{X_1, X_3, X_5, \ldots\} \\
\{X_1, X_3, X_5, \ldots\} \\
\{X_0, X_2, \ldots\} \\
\{X_0, X_2, \ldots\} \\
\{X_0, X_2, \ldots\}
\end{bmatrix}$$
Flip this set. Swap what’s included and what’s excluded.
Which element is paired with this set?
Which element is paired with this set?

\begin{array}{cccccc}
X_0 & X_1 & X_2 & X_3 & X_4 & X_5 \\
\hline
X_0 & X_0, & X_2, & X_4, & \ldots \\
X_1 & \text{green} & \text{red} & X_3, & X_5, & \ldots \\
X_2 & X_0, & X_2, & X_5, & \ldots \\
X_3 & X_1, & \text{yellow} & X_4, & \ldots \\
X_4 & \text{yellow} & X_2, & \ldots \\
X_5 & X_0, & X_4, & X_5, & \ldots \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
\end{array}
Which element is paired with this set?
<table>
<thead>
<tr>
<th></th>
<th>X0</th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
<th>X5</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>X0</td>
<td>X0,</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>X1</td>
<td></td>
<td>X3,</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>X2</td>
<td>X0,</td>
<td>X2,</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>X3</td>
<td>X1,</td>
<td></td>
<td>X4,</td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>X4</td>
<td></td>
<td>X2,</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>X5</td>
<td>X0,</td>
<td></td>
<td></td>
<td>X4,</td>
<td></td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>...</td>
</tr>
</tbody>
</table>

Which element is paired with this set?
Which element is paired with this set?
Which element is paired with this set?

<table>
<thead>
<tr>
<th></th>
<th>X0</th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
<th>X5</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>X0</td>
<td>X0,</td>
<td></td>
<td>X2,</td>
<td></td>
<td>X4,</td>
<td></td>
<td>...</td>
</tr>
<tr>
<td>X1</td>
<td></td>
<td>X3,</td>
<td></td>
<td>X5,</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X2</td>
<td>X0,</td>
<td>X2,</td>
<td></td>
<td></td>
<td>X5,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X3</td>
<td>X1,</td>
<td></td>
<td>X4,</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X4</td>
<td></td>
<td>X2,</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X5</td>
<td>X0,</td>
<td></td>
<td>X4,</td>
<td></td>
<td>X5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

...
Which element is paired with this set?

<table>
<thead>
<tr>
<th></th>
<th>X0</th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
<th>X5</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>X0</td>
<td>X0,</td>
<td>X2,</td>
<td>X4,</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X1</td>
<td></td>
<td></td>
<td></td>
<td>X3,</td>
<td>X5,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X2</td>
<td>X0,</td>
<td></td>
<td></td>
<td></td>
<td>X5,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X3</td>
<td>X1,</td>
<td></td>
<td></td>
<td></td>
<td>X4,</td>
<td></td>
<td></td>
</tr>
<tr>
<td>X4</td>
<td></td>
<td>X2,</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X5</td>
<td>X0,</td>
<td></td>
<td>X4,</td>
<td>X5,</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

...
The Diagonalization Proof

- No matter how we pair up elements of $S$ and subsets of $S$, the complemented diagonal won't appear in the table.
  - In row $n$, the $n$th element must be wrong.
- No matter how we pair up elements of $S$ and subsets of $S$, there is always at least one subset left over.
- This result is **Cantor's theorem**: Every set is strictly smaller than its power set:
  \[
  \text{If } S \text{ is a set, then } |S| < |\mathcal{P}(S)|.\]
Two Infinities...

• By Cantor's Theorem:

\[ |\mathbb{N}| < |\mathcal{P}(\mathbb{N})| \]
...And Beyond!

- By Cantor's Theorem:
  \[ |\mathbb{N}| < |\wp(\mathbb{N})| \]
  \[ |\wp(\mathbb{N})| < |\wp(\wp(\mathbb{N}))| \]
  \[ |\wp(\wp(\mathbb{N}))| < |\wp(\wp(\wp(\mathbb{N})))| \]
  \[ |\wp(\wp(\wp(\mathbb{N})))| < |\wp(\wp(\wp(\wp(\mathbb{N}))))| \]
  ...

- *Not all infinite sets have the same size!*
- *There is no biggest infinity!*
- *There are infinitely many infinities!*
What does this have to do with computation?
“The set of all computer programs”

“The set of all problems to solve”
Where We're Going

• A **string** is a sequence of characters.

• We're going to prove the following results:
  • There are *at most* as many programs as there are strings.
  • There are *at least* as many problems as there are sets of strings.

• This leads to some *incredible* results – we'll see why in a minute!
Where We're Going

A *string* is a sequence of characters. We're going to prove the following results:

- There are *at most* as many programs as there are strings.
- There are *at least* as many problems as there are sets of strings.

This leads to some *incredible* results – we'll see why in a minute!
The source code of a computer program is just a (long, structured, well-commented) string of text.

All programs are strings, but not all strings are necessarily programs.

\[ |\text{Programs}| \leq |\text{Strings}| \]
Where We're Going

• A *string* is a sequence of characters.

• We're going to prove the following results:
  • There are *at most* as many programs as there are strings.
  • There are *at least* as many problems as there are sets of strings.

• This leads to some *incredible* results – we'll see why in a minute!
Where We're Going

• A *string* is a sequence of characters.

• We're going to prove the following results:
  • There are *at most* as many programs as there are strings. ✓
  • There are *at least* as many problems as there are sets of strings.

• This leads to some *incredible* results – we'll see why in a minute!
Where We're Going

A *string* is a sequence of characters.

We're going to prove the following results:

- There are **at most** as many programs as there are strings. ✓
- There are **at least** as many problems as there are sets of strings.

This leads to some *incredible* results – we'll see why in a minute!
Strings and Problems

• There is a connection between the number of sets of strings and the number of problems to solve.

• Let \( S \) be any set of strings. This set \( S \) gives rise to a problem to solve:

  **Given a string \( w \), determine whether \( w \in S \).**
Given a string $w$, determine whether $w \in S$.

- Suppose that $S$ is the set
  \[ S = \{ "a", "b", "c", \ldots, "z" \} \]

- From this set $S$, we get this problem:
  
  **Given a string $w$, determine whether $w$ is a single lower-case English letter.**
Strings and Problems

Given a string \( w \), determine whether \( w \in S \).

- Suppose that \( S \) is the set
  \[
  S = \{ "0", "1", "2", \ldots, "9", "10", "11", \ldots \}
  \]
- From this set \( S \), we get this problem:
  
  Given a string \( w \), determine whether \( w \) represents a natural number.
Strings and Problems

Given a string $w$, determine whether $w \in S$.

- Suppose that $S$ is the set
  \[
  S = \{ \ p \mid p \text{ is a legal C++ program} \ \}
  \]
- From this set $S$, we get this problem:
  
  Given a string $w$, determine whether $w$ is a legal C++ program.
Strings and Problems

• Every set of strings gives rise to a unique problem to solve.
• Other problems exist as well.

\[ |\text{Sets of Strings}| \leq |\text{Problems}| \]
Where We're Going

- A *string* is a sequence of characters.
- We're going to prove the following results:
  - There are *at most* as many programs as there are strings. ✓
  - There are *at least* as many problems as there are sets of strings.
- This leads to some *incredible* results – we'll see why in a minute!
Where We're Going

• A *string* is a sequence of characters.

• We're going to prove the following results:
  • There are *at most* as many programs as there are strings. ✔
  • There are *at least* as many problems as there are sets of strings. ✔

• This leads to some *incredible* results – we'll see why in a minute!
Where We're Going

A *string* is a sequence of characters.

We're going to prove the following results:

- There are *at most* as many programs as there are strings. ✓
- There are *at least* as many problems as there are sets of strings. ✓

• **This leads to some *incredible* results – we'll see why in a minute!**
Where We're Going

A *string* is a sequence of characters.

We're going to prove the following results:

There are *at most* as many programs as there are strings. ✓

There are *at least* as many problems as there are sets of strings. ✓

• This leads to some *incredible* results – we'll see why *in a minute!* *right now!*
Every computer program is a string.

So, the number of programs is at most the number of strings.

From Cantor's Theorem, we know that there are more sets of strings than strings.

There are at least as many problems as there are sets of strings.

$$|\text{Programs}| \leq |\text{Strings}| < |\wp(\text{Strings})| \leq |\text{Problems}|$$
Every computer program is a string.

So, the number of programs is at most the number of strings.

From Cantor's Theorem, we know that there are more sets of strings than strings.

There are at least as many problems as there are sets of strings.

\[ |\text{Programs}| < |\text{Problems}| \]
There are more problems to solve than there are programs to solve them.
It Gets Worse

• Using more advanced set theory, we can show that there are infinitely more problems than solutions.

• In fact, if you pick a totally random problem, the probability that you can solve it is zero.

• **More troubling fact:** We've just shown that some problems are impossible to solve with computers, but we don't know which problems those are!
We need to develop a more nuanced understanding of computation.
Where We're Going

• **What makes a problem impossible to solve with computers?**
  • Is there a deep reason why certain problems can't be solved with computers, or is it completely arbitrary?
  • How do you know when you're looking at an impossible problem?
  • Are these real-world problems, or are they highly contrived?

• **How do we know that we're right?**
  • How can we back up our pictures with rigorous proofs?
  • How do we build a mathematical framework for studying computation?
Next Time

- **Mathematical Proof**
  - What is a mathematical proof?
  - How can we prove things with certainty?