Welcome to CS103!

- **Two Handouts**
  - Course Information
  - Syllabus
  - (Also available online if you'd like!)
- **Today:**
  - Course Overview
  - Introduction to Set Theory
  - The Limits of Computation

Key Questions in CS103

- Can computers solve all problems?

  - **Computability Theory**

- Why are some problems harder to solve than others?

  - **Complexity Theory**

- How can we be certain in our answers to these questions?

  - **Discrete Mathematics**

Instructor
Keith Schwarz (htiek@cs.stanford.edu)

Head TA
Maesen Churchill (maesenc@stanford.edu)

TAs
Lucille Benoit (lbenoit@stanford.edu)
Alex Cope (alexcope@stanford.edu)
Kevin Crain (craink@stanford.edu)
Edward Dai (ejdai@stanford.edu)
Kevin Gibbons (kgibbs@stanford.edu)
Devney Hamilton (devney@stanford.edu)
Shantanu Joshi (joshi4@stanford.edu)
Jimmy Lee (jdl39@stanford.edu)
Stephen Macke (smacke@cs.stanford.edu)
Colin Mayer (cmayer@stanford.edu)
Aditya Palnitkar (aditpal@stanford.edu)
Sanjana Rajan (srajan1@stanford.edu)
Devon Rifkin (drifkin@stanford.edu)
Logan Short (lshort@stanford.edu)
More to come!

Staff Email List: cs103-aut1415-staff@lists.stanford.edu
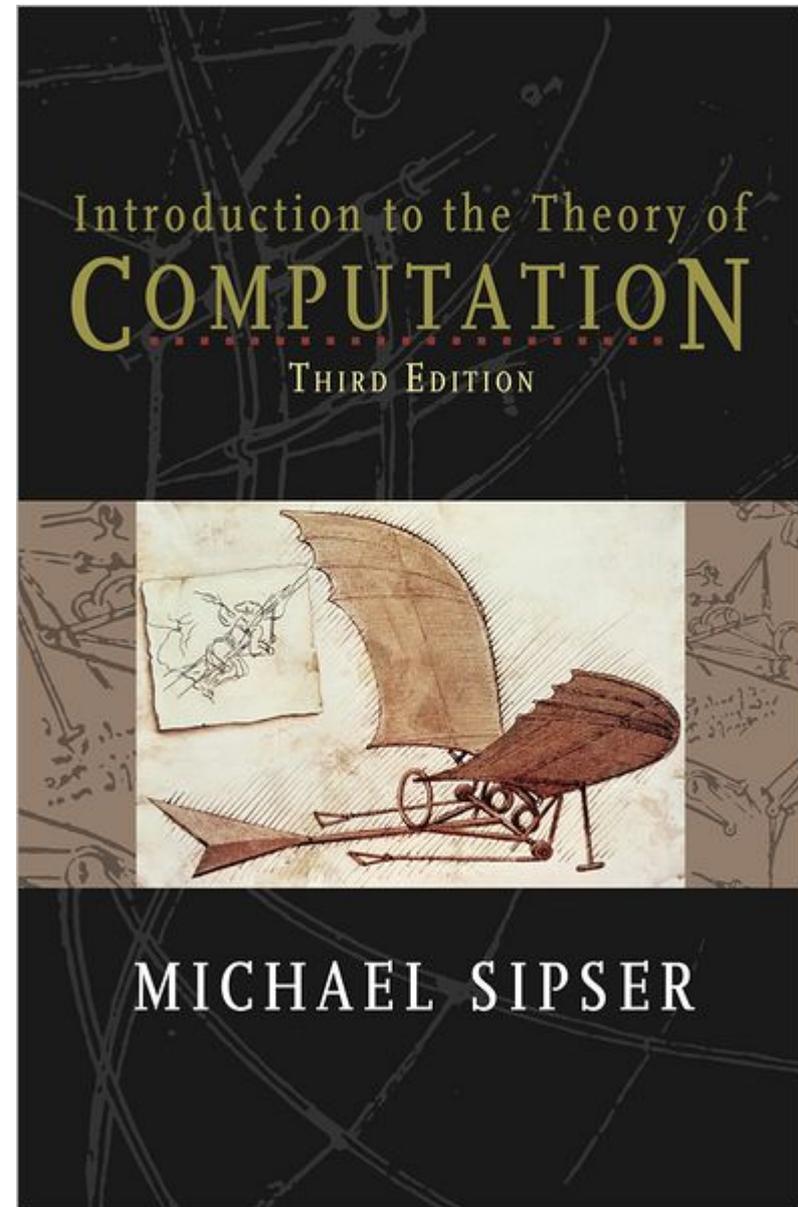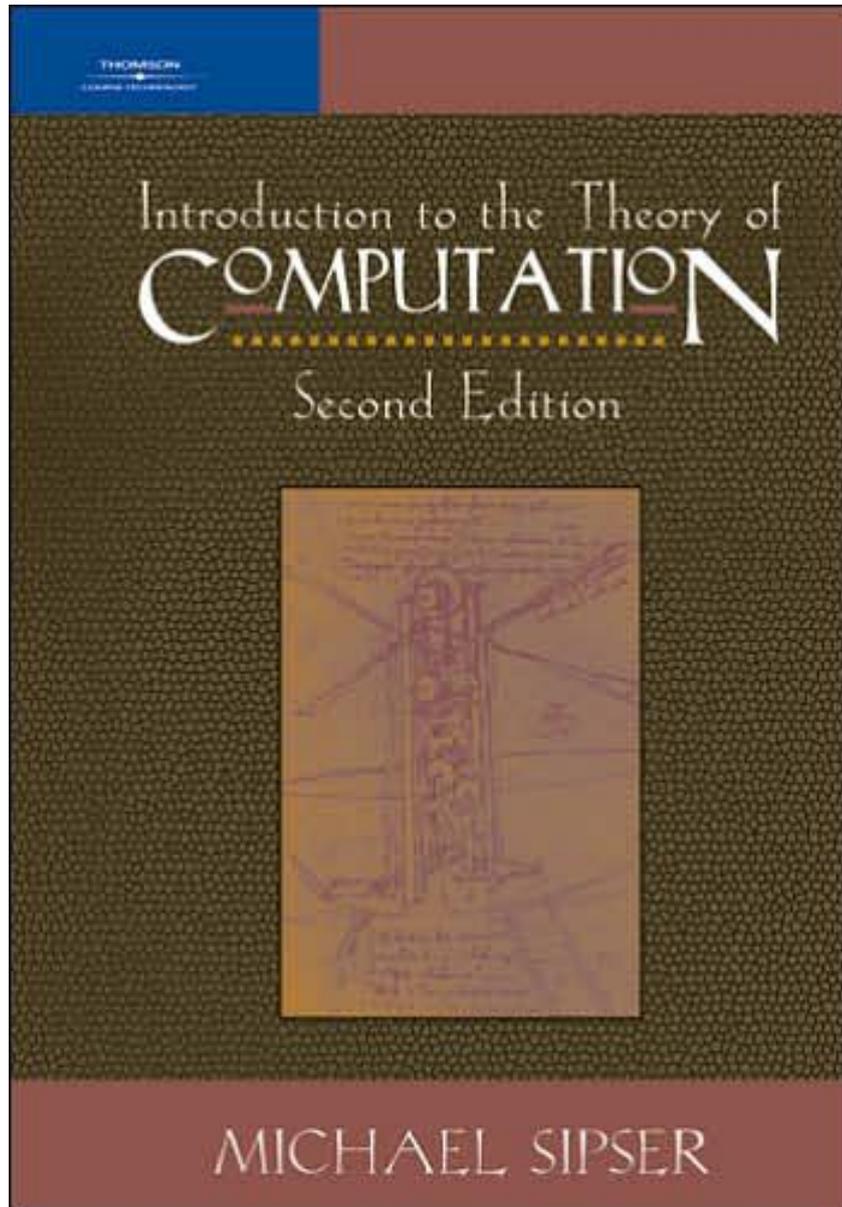
Course Website

http://cs103.stanford.edu

"Prerequisite"

CS106A

There aren't any math prerequisites for this course – high-school algebra should be enough!

Recommended Reading

Introduction to the Theory of COMPUTATION — Second Edition — MICHAEL SIPSER

Introduction to the Theory of COMPUTATION — THIRD EDITION — MICHAEL SIPSER

Online Course Notes

CS103

Mathematical Foundations of Computing

$L(M) = L((0+1)^*(00+11))$

start
q₁ → q₂ (0)
q₃ → q₄ (1)
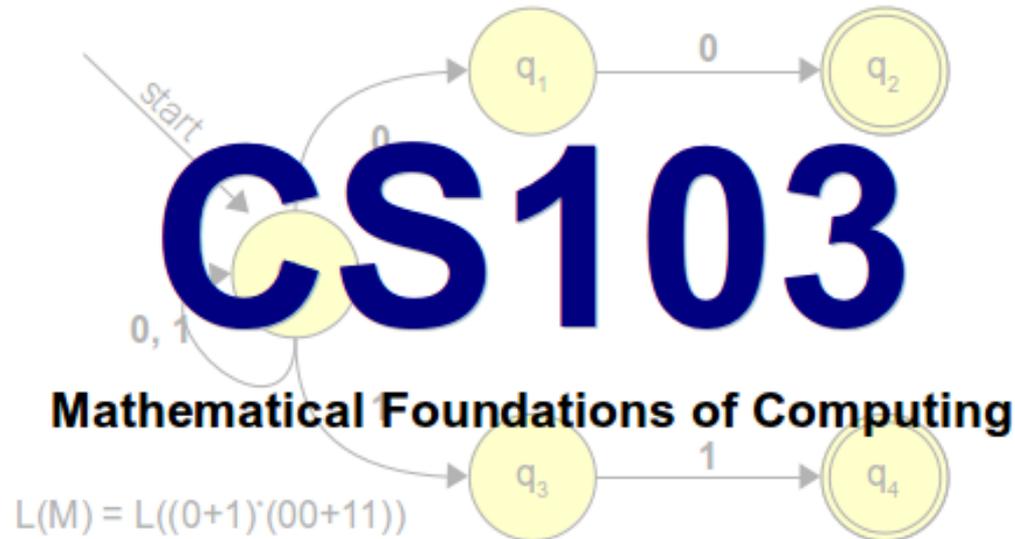
Handouts

00: Course Information
01: Syllabus

Discussion Problems

Coming soon!

Resources

Course Reader PDF
Lecture Videos

Lectures

Coming soon!

uction to discrete
eory, and complexity
ter ahead of us filled
sults in the power and
ope that you're able to

Grading Policies

- 40% Assignments
- 15% Midterm I
- 15% Midterm II
- 30% Final Exam

Final Exam

Friday, December 12
12:15PM – 3:15PM
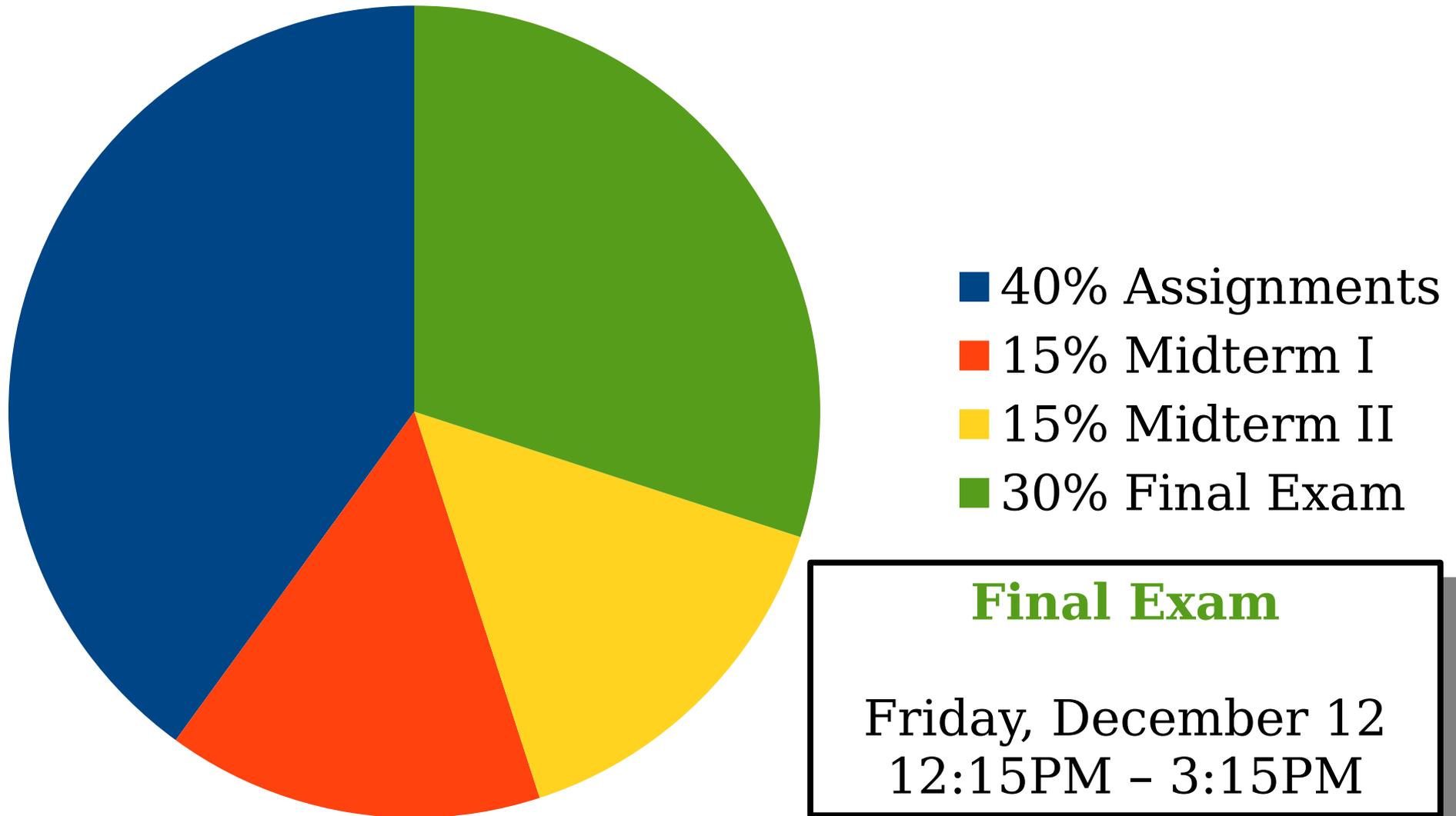
Let's Get Started!

Introduction to Set Theory

"CS103 students"

"All the computers on the
Stanford network"

"Cool people"

"The chemical elements"

"Cute animals"

"US coins"

A **set** is an unordered collection of distinct objects, which may be anything (including other sets).

Set notation: Curly braces with commas separating out the elements

A **set** is an unordered collection of distinct objects, which may be anything (including other sets).

These are the same set!
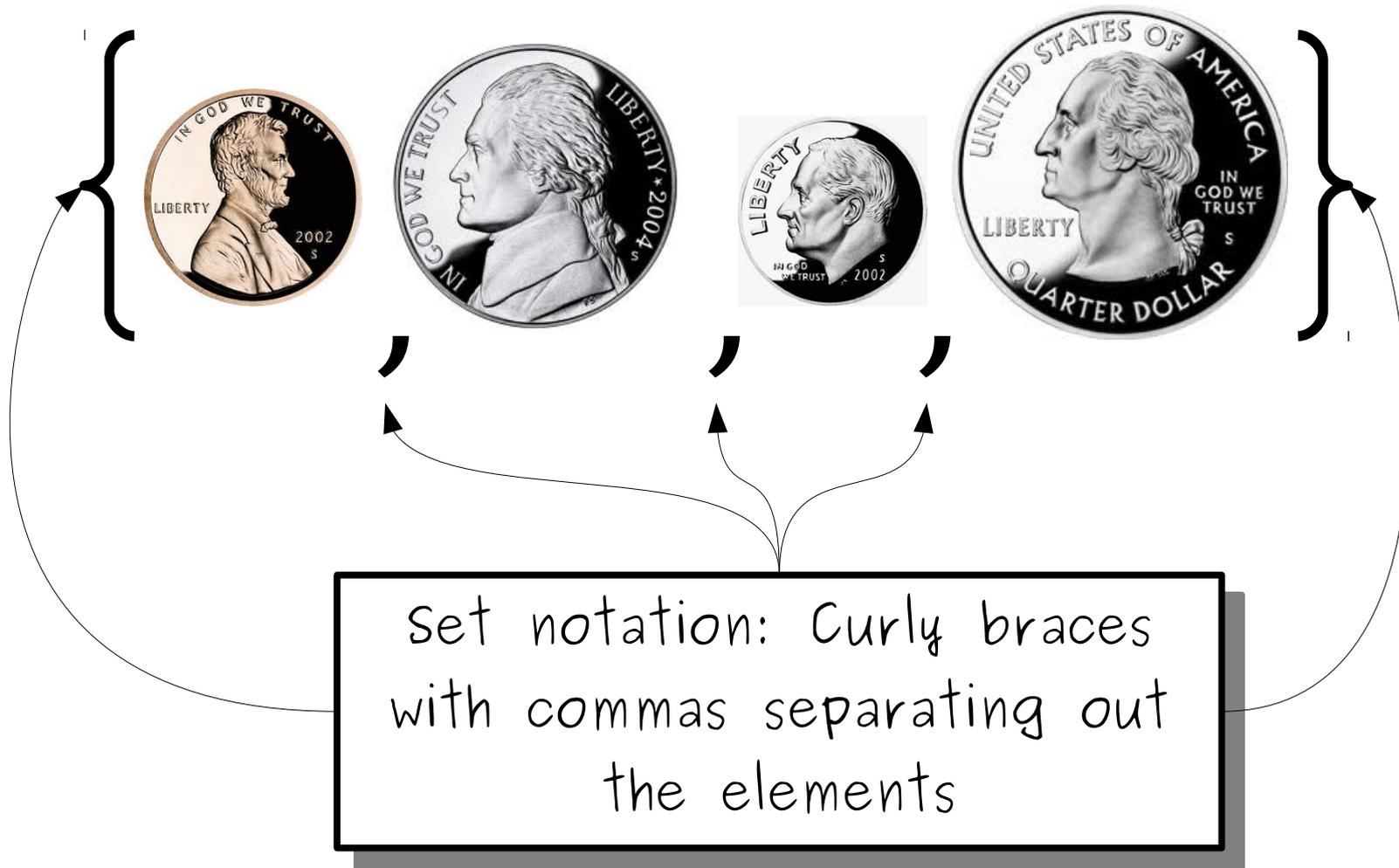
A **set** is an *unordered* collection of distinct objects, which may be anything (including other sets).

A **set** is an unordered collection of *distinct* objects, which may be anything (including other sets).

$$\{\ \} \ = \ \varnothing$$

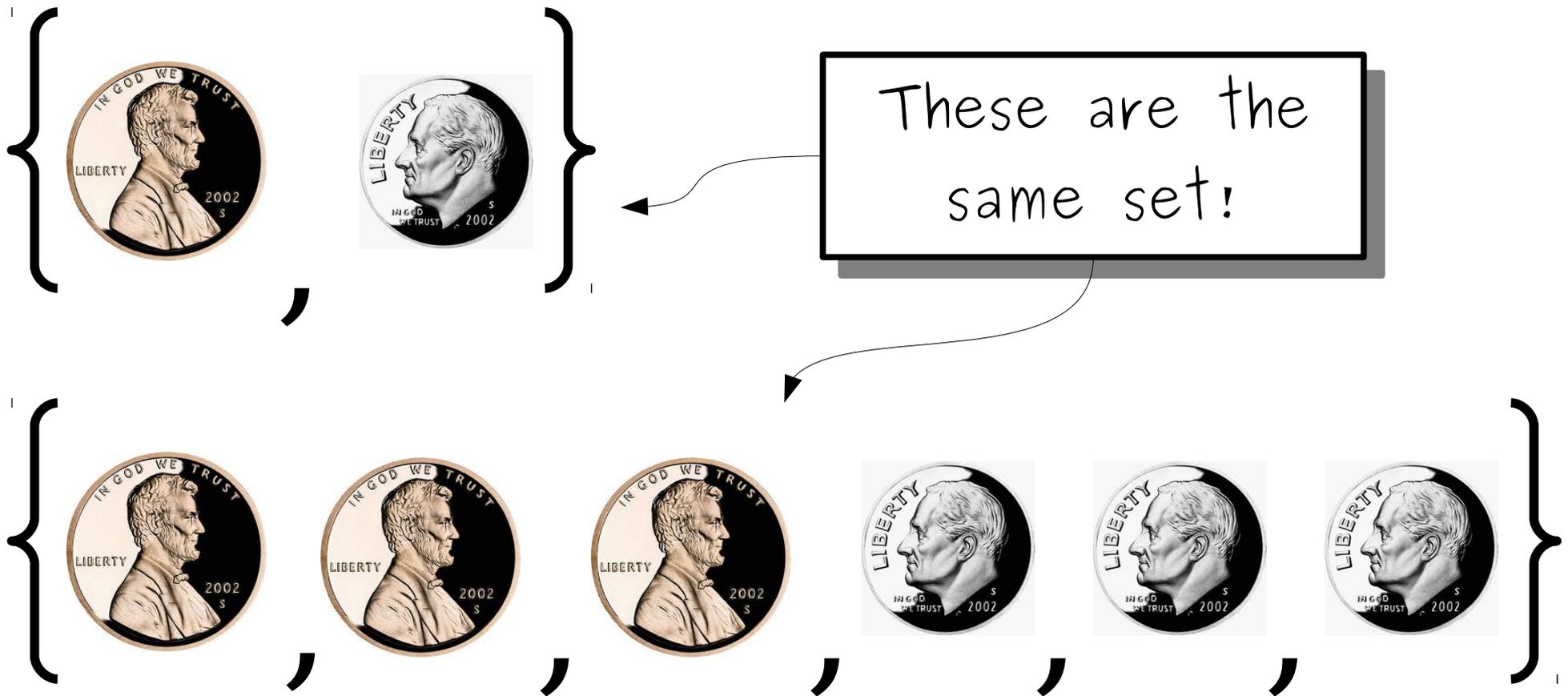The **empty set** contains no elements.

We use this symbol to denote the empty set.

A **set** is an unordered collection of distinct objects, which may be anything (including other sets).

This is a
number.

This is a set.
It contains a
number.
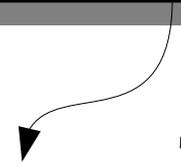
$$1 \neq \{1\}$$

Are these equal to one another?

Membership

Is in this set?

Membership

 $\notin$ $\left\{ \vphantom{\rule{0pt}{3cm}} \right.$, , , $\left. \vphantom{\rule{0pt}{3cm}} \right\}$

Is in this set?

Set Membership

- Given a set S and an object x, we write

$$x \in S$$

if x is contained in S, and

$$x \notin S$$

otherwise.

- If $x \in S$, we say that x is an **element** of S.
- Given any object x and any set S, either $x \in S$ or $x \notin S$.

Infinite Sets

- Some sets contain *infinitely many* elements!
- The set $\mathbb{N}$ = { 0, 1, 2, 3, ...} 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 $\mathbb{Z}$ = { ..., -2, -1, 0, 1, 2, ... } 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 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

$$\{\ n\ |\ n \in \mathbb{N} \text{ and } n \text{ is even }\}$$

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***:

$$\{ \, x \mid some \ property \ x \ satisfies \, \}$$

- For example:

$$\{ \, r \mid r \in \mathbb{R} \text{ and } r < 137 \, \}$$

$$\{ \, n \mid n \text{ is an even natural number} \, \}$$

$$\{ \, S \mid S \text{ is a set of US currency} \, \}$$

$$\{ \, a \mid a \text{ is cute animal} \, \}$$

Combining Sets

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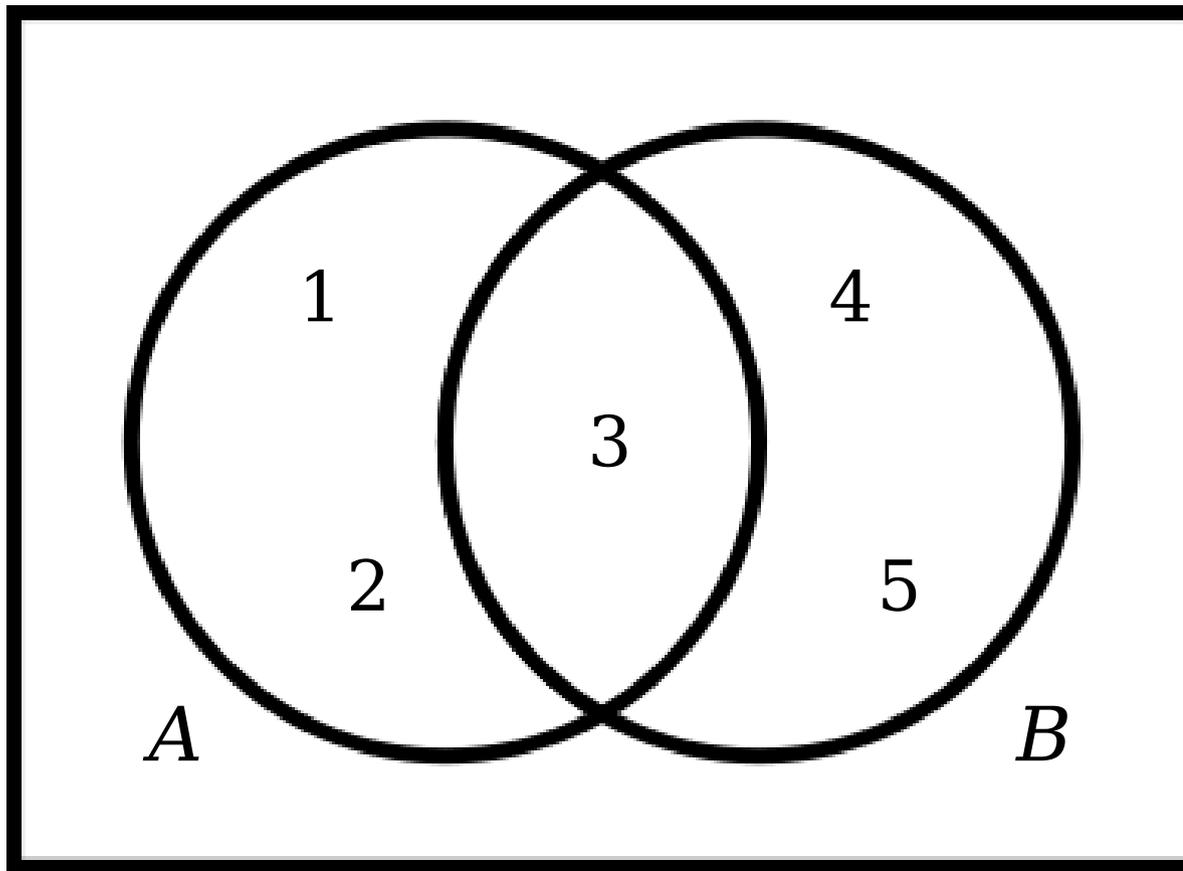
$A = \{\ 1, 2, 3\ \}$
$B = \{\ 3, 4, 5\ \}$

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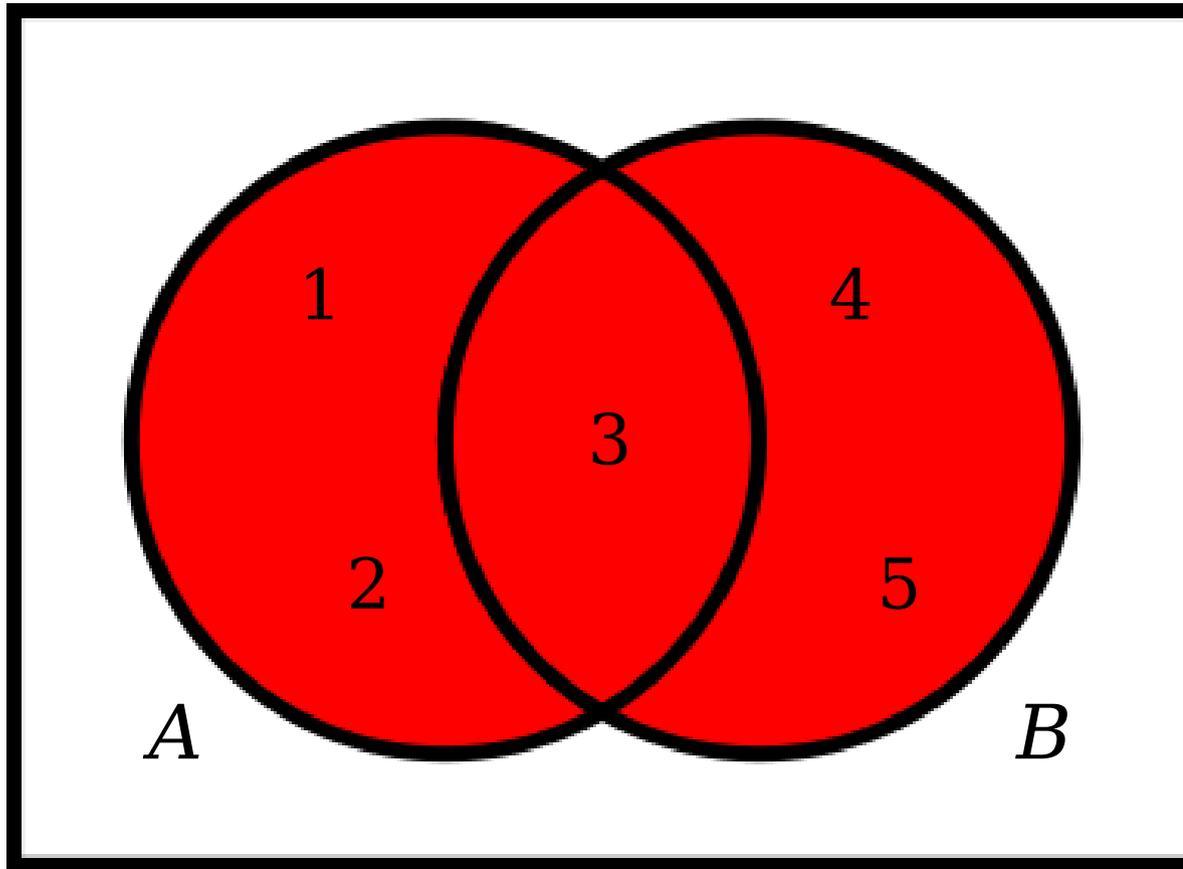
Union
$A \cup B$
{ 1, 2, 3, 4, 5 }

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

Venn Diagrams
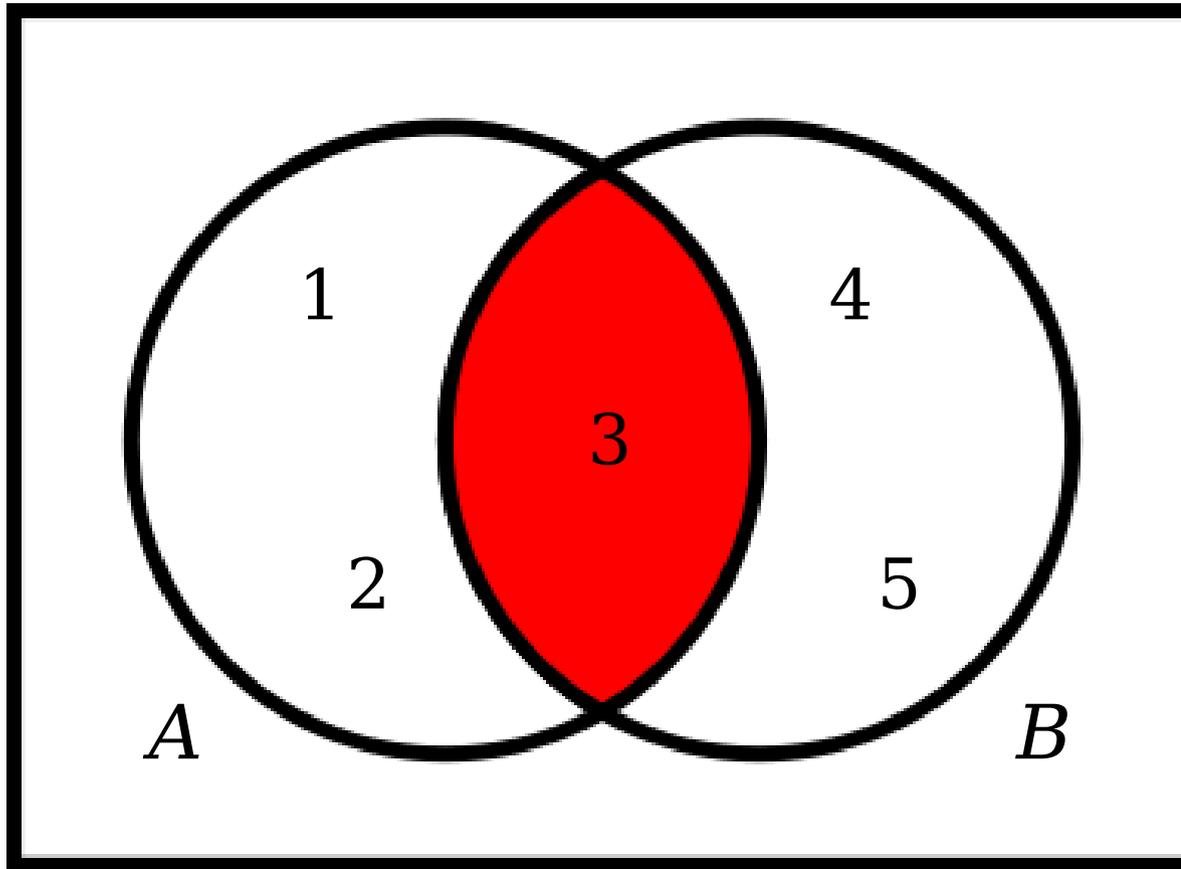
Intersection
$A \cap B$
{ 3 }

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

Venn Diagrams

Difference

$A - B$

$\{\ 1,\ 2\ \}$

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

Venn Diagrams

Difference

$A \setminus B$

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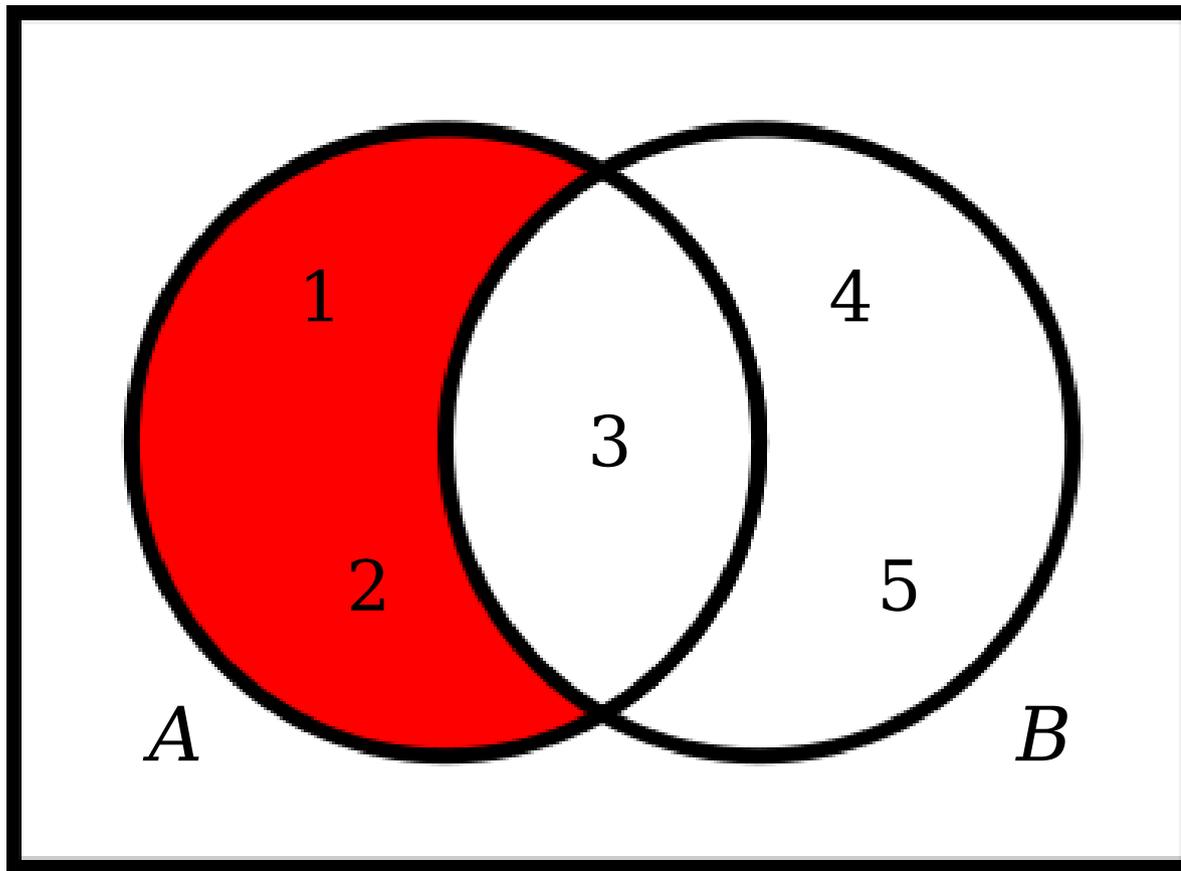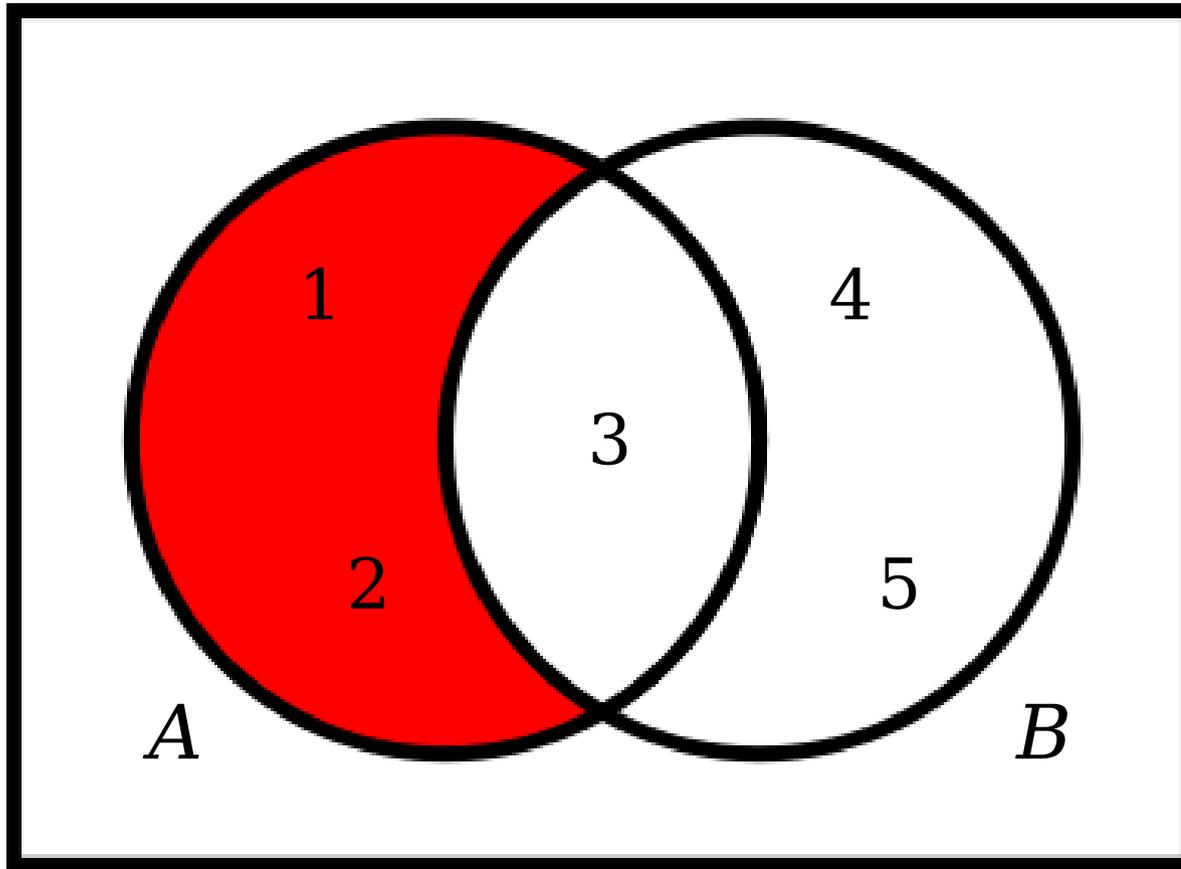
A = { 1, 2, 3 }
B = { 3, 4, 5 }
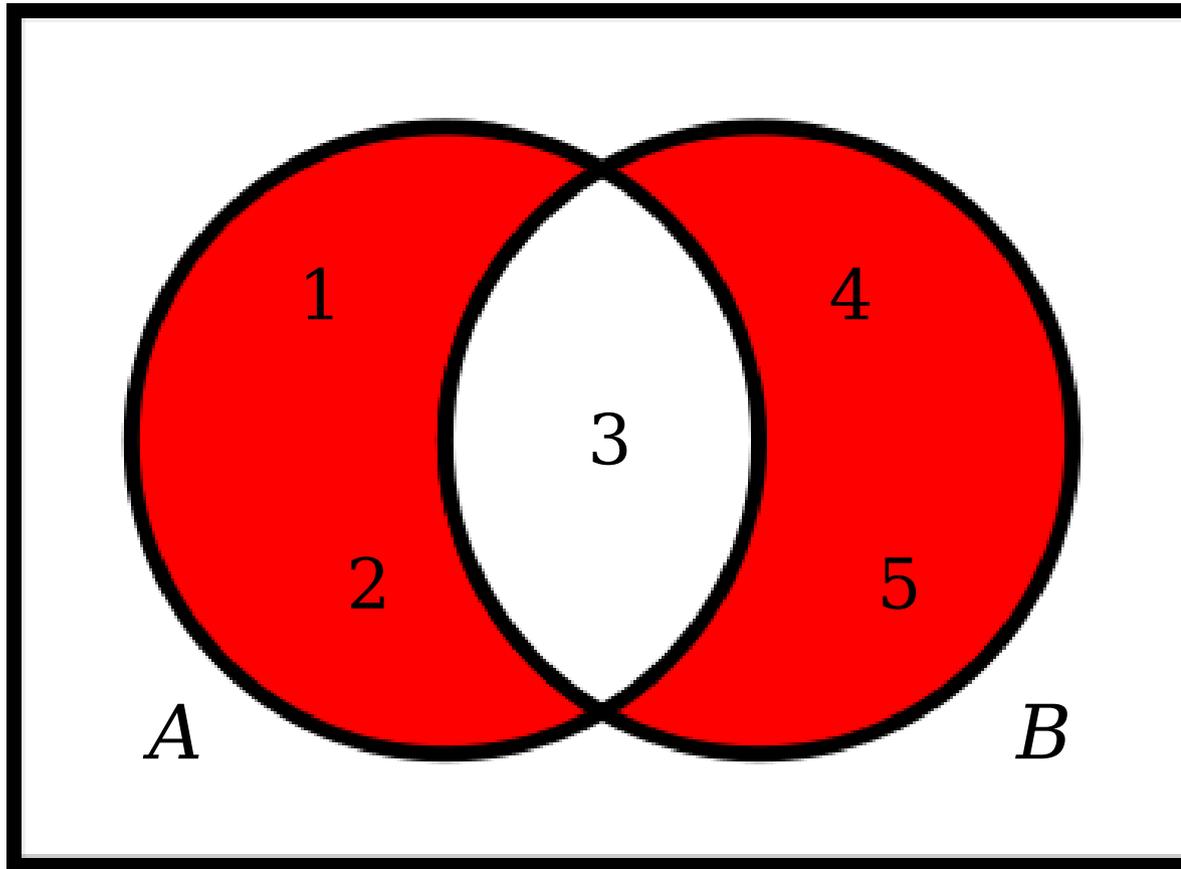
Venn Diagrams

Symmetric
Difference
$A \,\Delta\, B$

{ 1, 2, 4, 5 }

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

Venn Diagrams
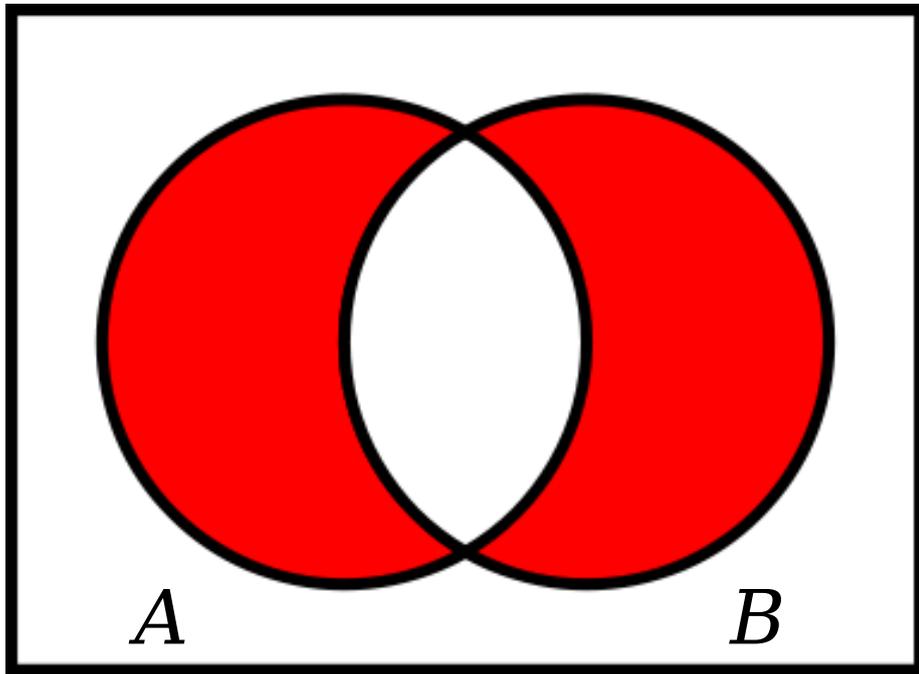
$A \Delta B$
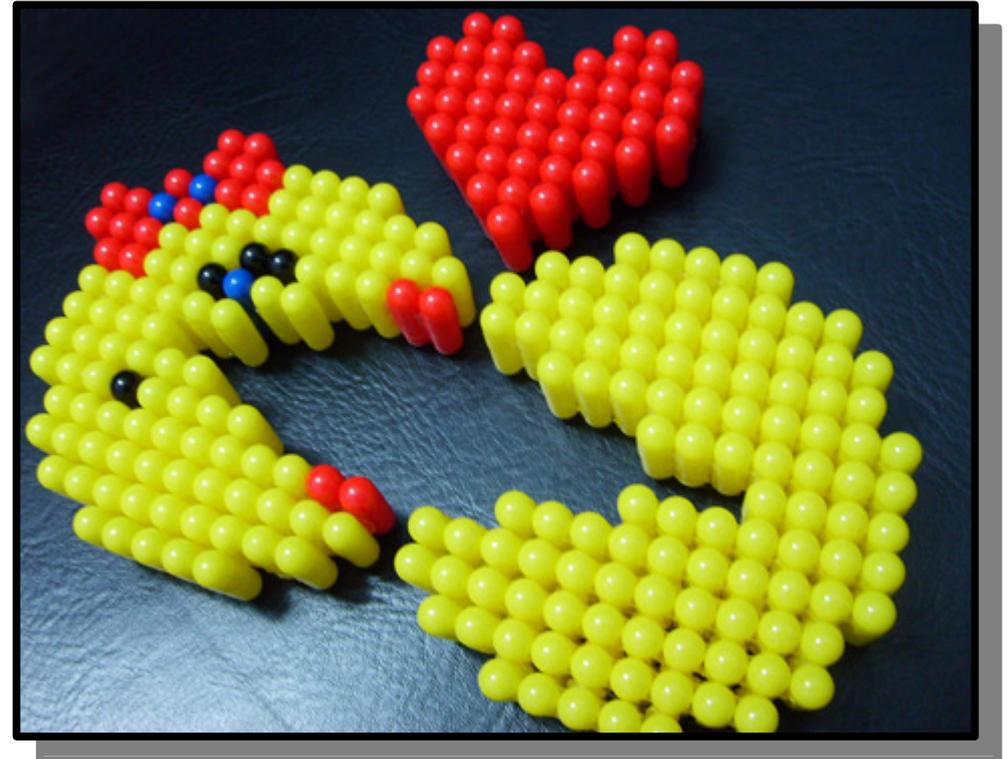
Venn Diagrams
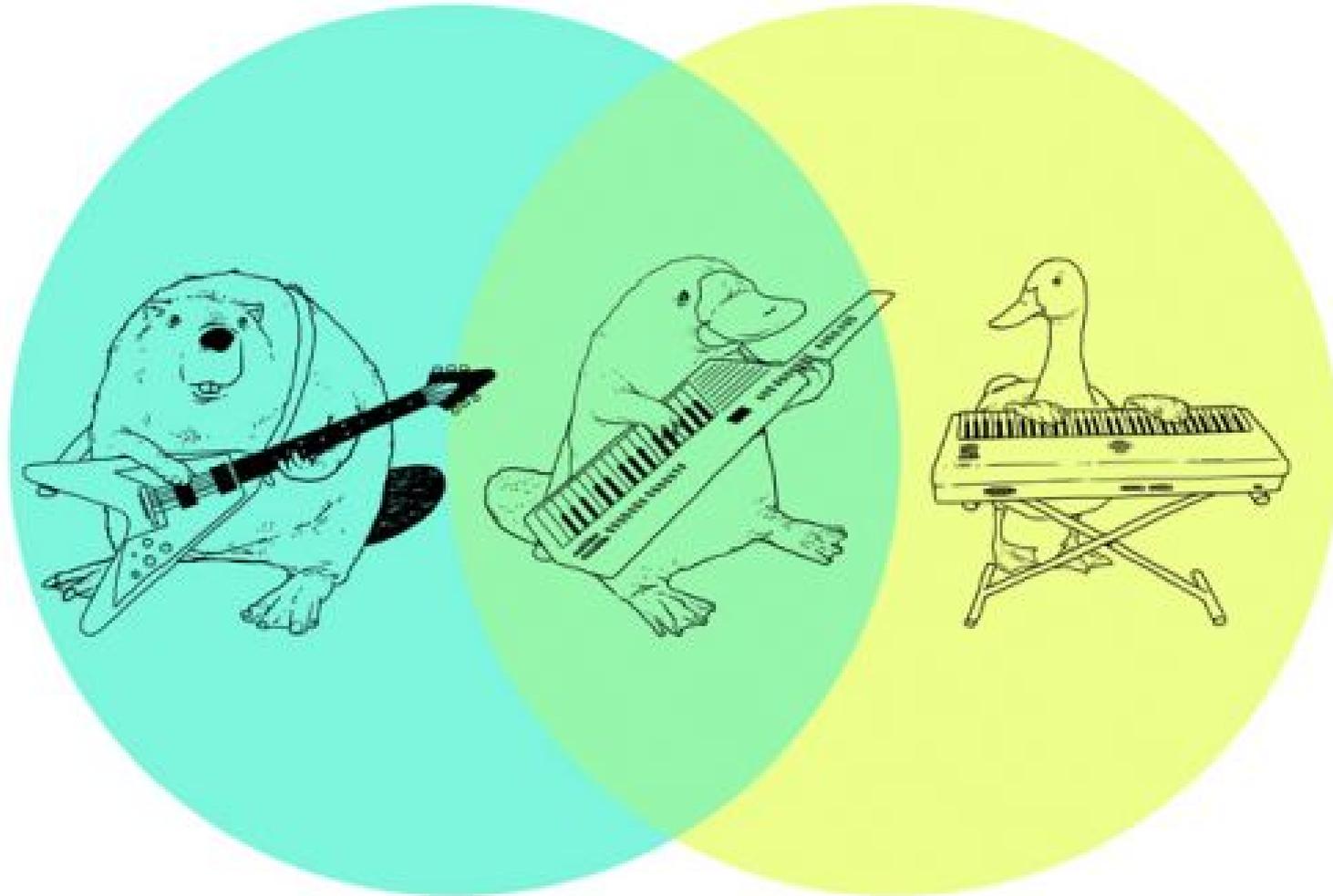
Venn Diagrams for Three Sets
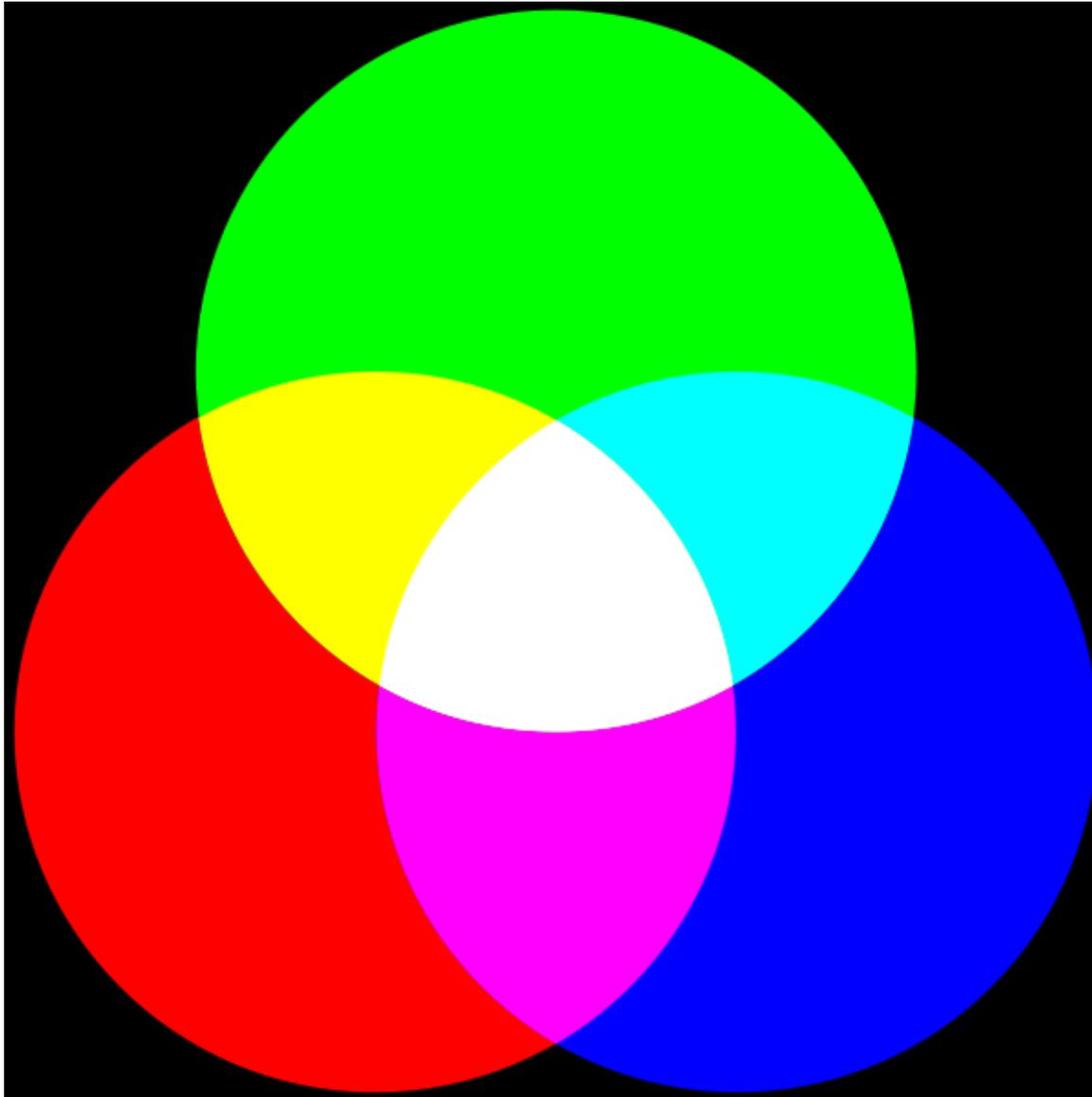
Venn Diagrams for Four Sets
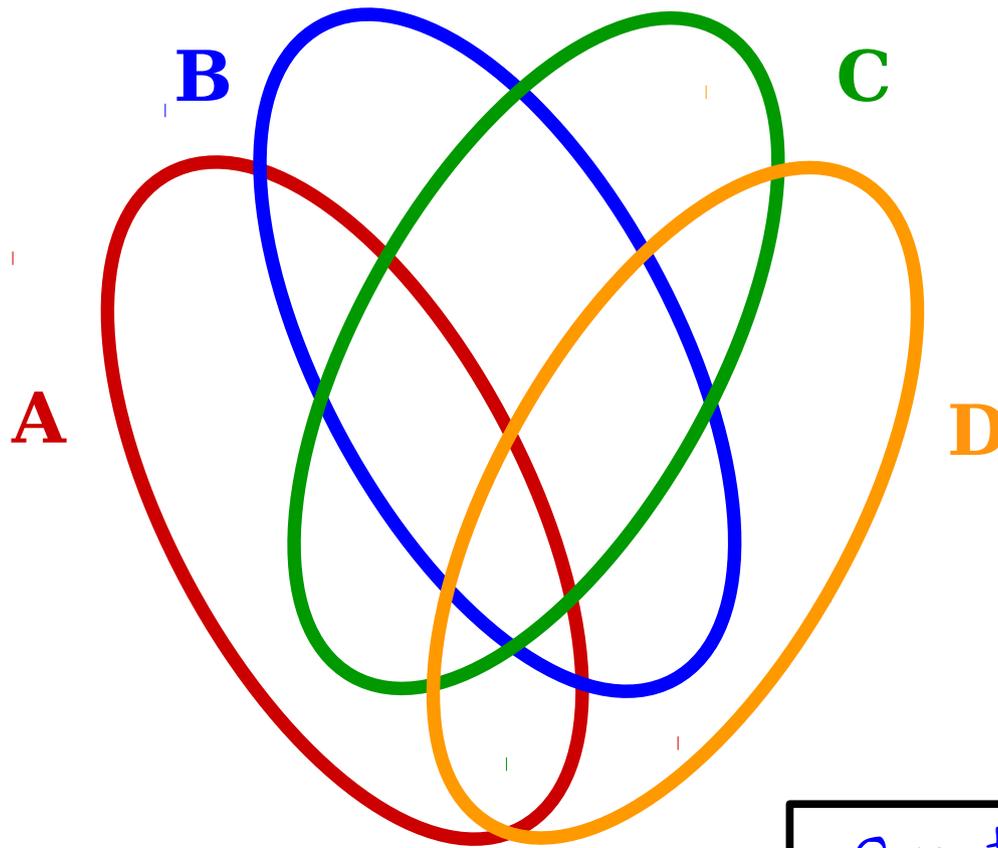
Question to ponder: why can't we just draw four circles?

Venn Diagrams for Five Sets
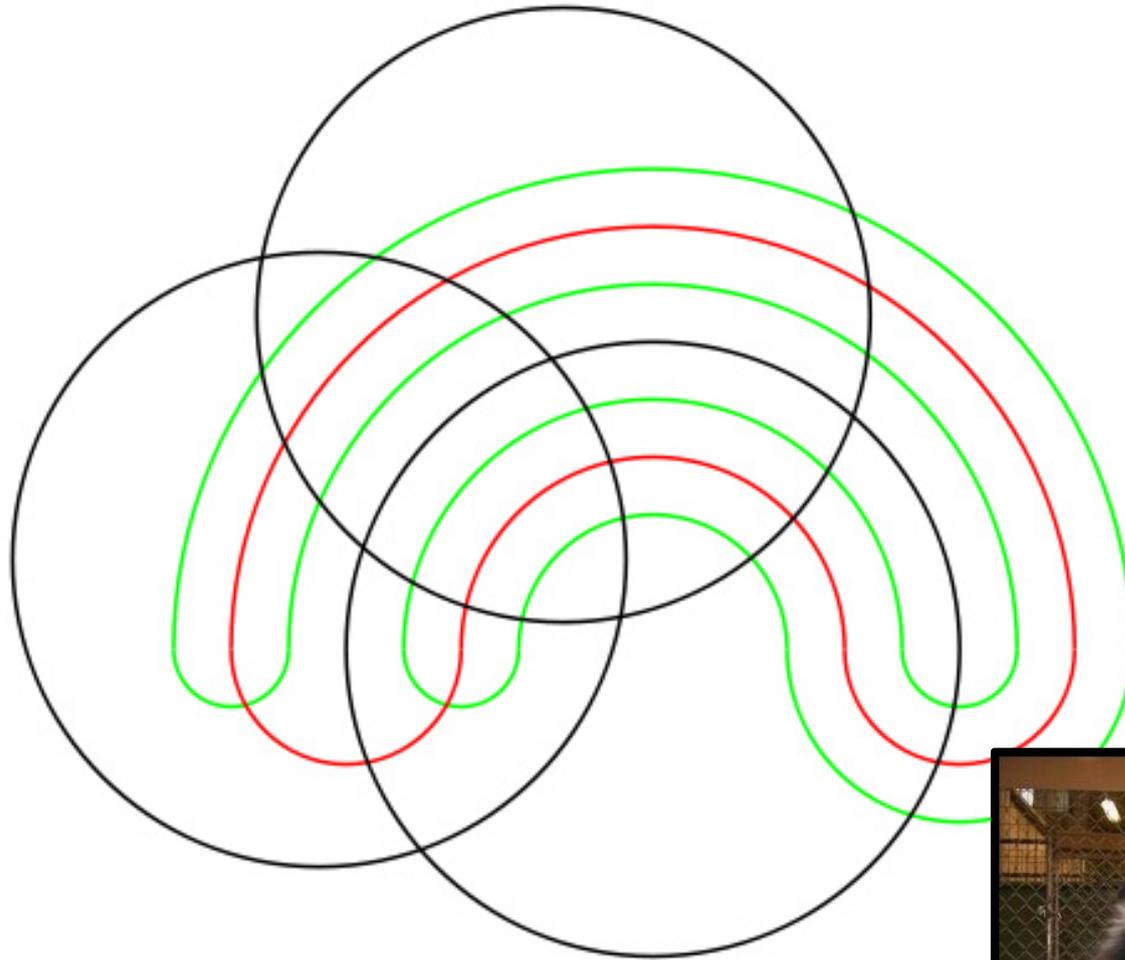
Venn Diagrams for Seven Sets

http://moebio.com/research/sevensets/

Subsets and Power Sets

Subsets

- A set S is a **subset** of a set T (denoted $S \subseteq T$) if all elements of S are also elements of T.

- Examples:
  - $\{ 1, 2, 3 \} \subseteq \{ 1, 2, 3, 4 \}$
  - $\mathbb{N} \subseteq \mathbb{Z}$ (*every natural number is an integer*)
  - $\mathbb{Z} \subseteq \mathbb{R}$ (*every integer is a real number*)

What About the Empty Set?

- A set S is a **subset** of a set T (denoted **$S \subseteq T$**) if all elements of S are also elements of T.

- Are there any sets S where $\emptyset \subseteq S$?

- Equivalently, is there a set S where the following statement is true?

  "All elements of $\emptyset$ are also elements of S"

- **Yes!** In fact, this statement is true for every choice of S!

Vacuous Truth

- A statement of the form

  **"All objects of type *P*
  are also of type *Q*"**

  is called ***vacuously true*** if there are no objects of type *P*.

- Vacuously true statements are true *by definition*. This is a convention used throughout mathematics.

- Some examples:

  - All unicorns are pink.

  - All unicorns are blue.

  - Every element of Ø is also an element of *S*.

$$S = \left\{ \text{🪙}, \text{🪙} \right\}$$

$$\wp(S) = \left\{ \varnothing, \left\{ \text{🪙} \right\}, \left\{ \text{🪙} \right\}, \left\{ \text{🪙}, \text{🪙} \right\} \right\}$$

$\wp(S)$ is the **_power set_** of S (the set of all subsets of S)

Formally, $\wp(S) = \{\ T \mid T \subseteq S\ \}$

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 by writing $|S|$.

- Examples:

  - $|\{a, b, c, d, e\}| = 5$

  - $|\{\{a, b\}, \{c, d, e, f, g\}, \{h\}\}| = 3$

  - $|\{1, 2, 3, 3, 3, 3, 3\}| = 3$

  - $|\{\, n \in \mathbb{N} \mid n < 137\}| = 137$

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?
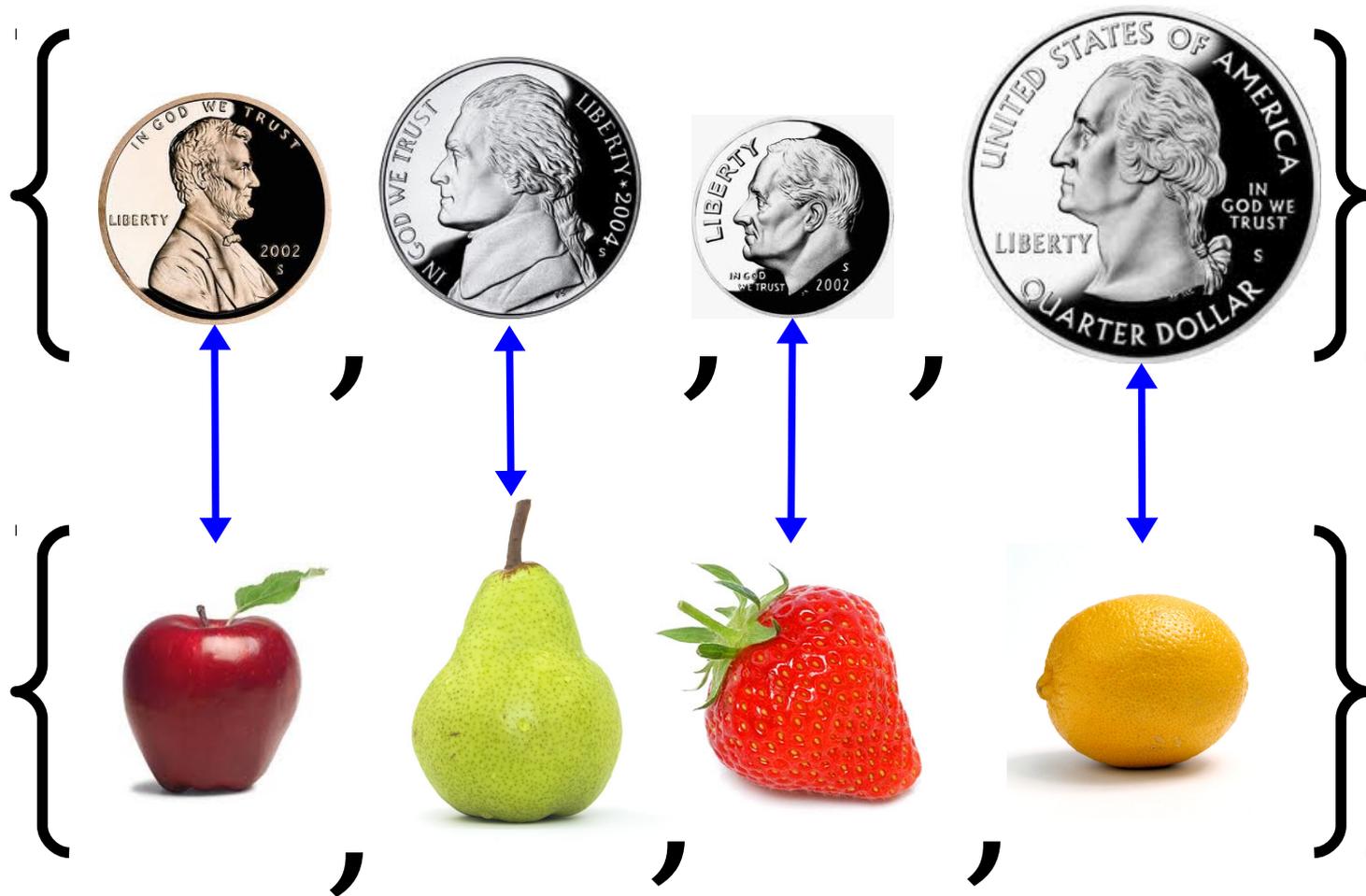
Comparing Cardinalities

- *By definition,* two sets have the same size if their elements can be paired off with no elements remaining.

- The intuition:

Comparing Cardinalities

- *By definition,* two sets have the same size if their elements can be paired off with no elements remaining.

- The intuition:

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

Infinite Cardinalities

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

$$S \quad 0 \quad 2 \quad 4 \quad 6 \quad 8 \quad 10 \quad 12 \quad 14 \quad 16 \quad \ldots$$
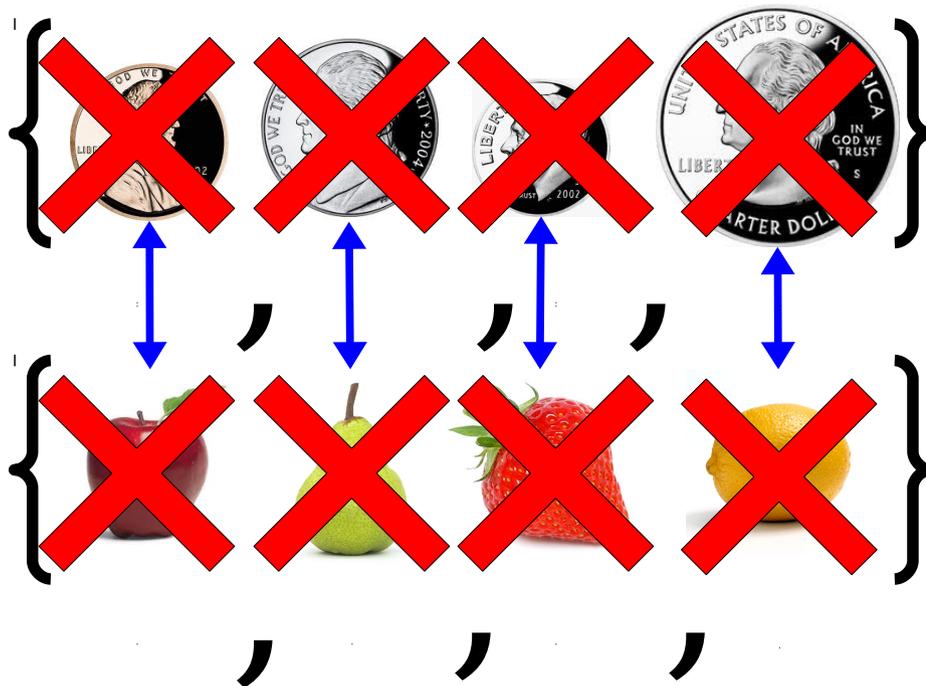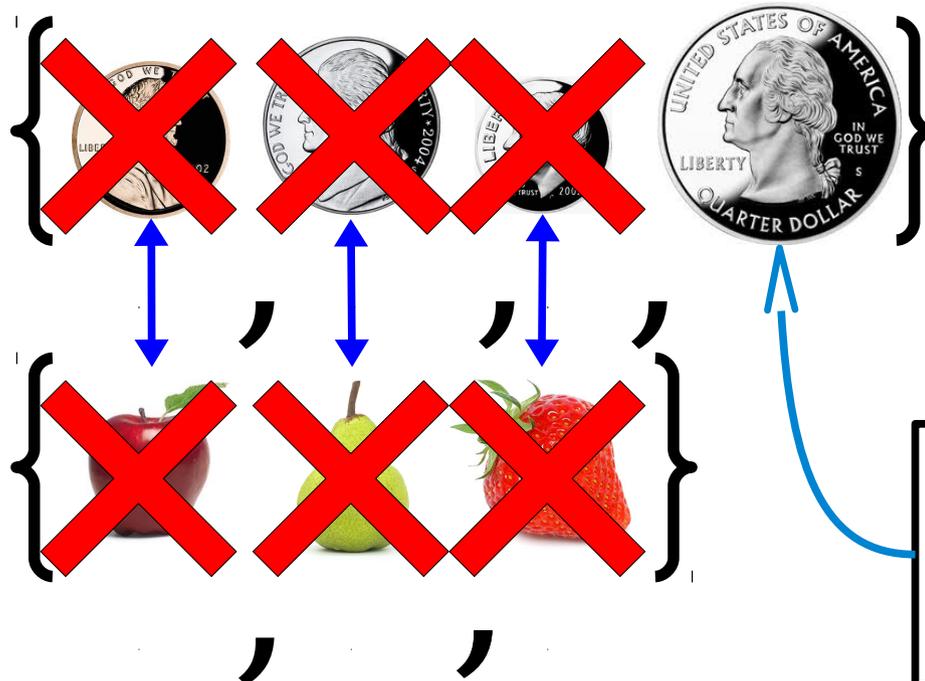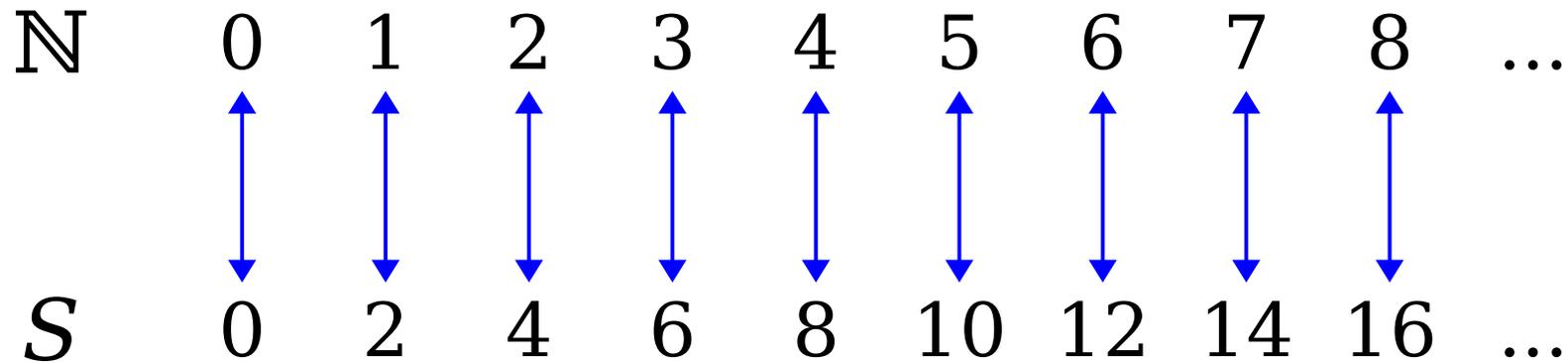
$$n \leftrightarrow 2n$$

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

$$|S| = |\mathbb{N}| = \aleph_0$$

Infinite Cardinalities

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

$$\mathbb{Z} \quad 0 \quad \text{-}1 \quad 1 \quad \text{-}2 \quad 2 \quad \text{-}3 \quad 3 \quad \text{-}4 \quad 4 \quad \ldots$$
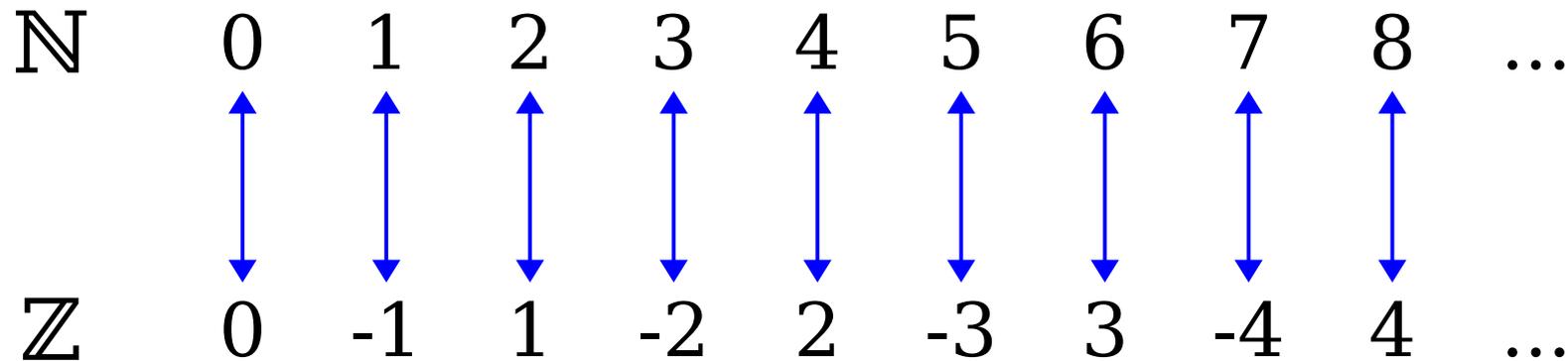
$$|\mathbb{N}| = |\mathbb{Z}| = \aleph_0$$

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

Important Question

Do all infinite sets have
the same cardinality?

$$S = \left\{ \text{(penny)}, \text{(dime)} \right\}$$

$$\wp(S) = \left\{ \varnothing, \left\{ \text{(dime)} \right\}, \left\{ \text{(penny)} \right\}, \left\{ \text{(penny)}, \text{(dime)} \right\} \right\}$$

$$|S| < |\wp(S)|$$

$$S = \{ \text{🪙}, \text{🪙}, \text{⬤} \}$$

$$\wp(S) = \left\{ \begin{array}{c} \emptyset, \{\text{🪙}\}, \{\text{🪙}\}, \{\text{⬤}\}, \\[2mm] \{\text{🪙}\,\text{🪙}\}, \{\text{🪙}\,\text{⬤}\}, \{\text{🪙}\,\text{⬤}\}, \\[2mm] \{\text{🪙}\,\text{🪙}\,\text{⬤}\} \end{array} \right\}$$

$$|S| < |\wp(S)|$$

$$S = \{a, b, c, d\}$$

$$\wp(S) = \{$$
$$\varnothing,$$
$$\{a\}, \{b\}, \{c\}, \{d\},$$
$$\{a, b\}, \{a, c\}, \{a, d\}, \{b, c\}, \{b, d\}, \{c, d\}$$
$$\{a, b, c\}, \{a, b, d\}, \{a, c, d\}, \{b, c, d\},$$
$$\{a, b, c, d\}$$
$$\}$$

$$|S| < |\wp(S)|$$

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 subsets of S without leaving anything out.

What would that look like?

x_0

x_1

x_2

x_3

x_4

x_5

$\ldots$

$x_0 \longleftrightarrow \{\ x_0,\ x_2,\ x_4,\ ...\ \}$

$x_1 \longleftrightarrow \{\ x_0,\ x_3,\ x_4,\ ...\ \}$

$x_2 \longleftrightarrow \{\ x_4,\ ...\ \}$

$x_3 \longleftrightarrow \{\ x_1,\ x_4,\ ...\ \}$

$x_4 \longleftrightarrow \{\ x_0,\ x_5,\ ...\ \}$

$x_5 \longleftrightarrow \{\ x_0,\ x_1,\ x_2,\ x_3,\ x_4,\ x_5,\ ...\ \}$

...

x_0	x_1	x_2	x_3	x_4	x_5	...

$x_0 \longleftrightarrow \{ x_0, x_2, x_4, ... \}$

$x_1 \longleftrightarrow \{ x_0, x_3, x_4, ... \}$

$x_2 \longleftrightarrow \{ x_4, ... \}$

$x_3 \longleftrightarrow \{ x_1, x_4, ... \}$

$x_4 \longleftrightarrow \{ x_0, x_5, ... \}$

$x_5 \longleftrightarrow \{ x_0, x_1, x_2, x_3, x_4, x_5, ... \}$

...

	x_0	x_1	x_2	x_3	x_4	x_5	...
$x_0 \longleftrightarrow$	**Y**	**N**	**Y**	**N**	**Y**	**N**	**...**

$x_1 \longleftrightarrow \{ x_0, x_3, x_4, ... \}$

$x_2 \longleftrightarrow \{ x_4, ... \}$

$x_3 \longleftrightarrow \{ x_1, x_4, ... \}$

$x_4 \longleftrightarrow \{ x_0, x_5, ... \}$

$x_5 \longleftrightarrow \{ x_0, x_1, x_2, x_3, x_4, x_5, ... \}$

...

	x_0	x_1	x_2	x_3	x_4	x_5	...
$x_0 \leftrightarrow$	**Y**	**N**	**Y**	**N**	**Y**	**N**	**...**
$x_1 \leftrightarrow$	**Y**	**N**	**N**	**Y**	**Y**	**N**	**...**

$x_2 \leftrightarrow \{\ x_4,\ ...\ \}$

$x_3 \leftrightarrow \{\ x_1,\ x_4,\ ...\ \}$

$x_4 \leftrightarrow \{\ x_0,\ x_5,\ ...\ \}$

$x_5 \leftrightarrow \{\ x_0,\ x_1,\ x_2,\ x_3,\ x_4,\ x_5,\ ...\ \}$

...

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	Y	N	Y	N	Y	N	...
x_1	Y	N	N	Y	Y	N	...
x_2	N	N	N	N	Y	N	...
x_3	N	Y	N	N	Y	N	...
x_4	Y	N	N	N	N	Y	...
x_5	Y	Y	Y	Y	Y	Y	...
...	...	...	...	...	...	...	...

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**Y**	N	Y	N	Y	N	...
x_1	Y	**N**	N	Y	Y	N	...
x_2	N	N	**N**	N	Y	N	...
x_3	N	Y	N	**N**	Y	N	...
x_4	Y	N	N	N	**N**	Y	...
x_5	Y	Y	Y	Y	Y	**Y**	...
...	...	...	...	...	...	...	**...**

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**Y**	N	Y	N	Y	N	...
x_1	Y	**N**	N	Y	Y	N	...
x_2	N	N	**N**	N	Y	N	...
x_3	N	Y	N	**N**	Y	N	...
x_4	Y	N	N	N	**N**	Y	...
x_5	Y	Y	Y	Y	Y	**Y**	...
...	...	...	...	...	...	...	**...**

Y	N	N	N	N	Y	...

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**Y**	N	Y	N	Y	N	...
x_1	Y	**N**	N	Y	Y	N	...
x_2	N	N	**N**	N	Y	N	...
x_3	N	Y	N	**N**	Y	N	...
x_4	Y	N	N	N	**N**	Y	...
x_5	Y	Y	Y	Y	Y	**Y**	...
...	...	...	...	...	...	...	**...**

Which row in the table is paired with this set?

| N | Y | Y | Y | Y | N | ... |

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	Y	N	Y	N	Y	N	...
x_1	Y	N	N	Y	Y	N	...
x_2	N	N	N	N	Y	N	...
x_3	N	Y	N	N	Y	N	...
x_4	Y	N	N	N	N	Y	...
x_5	Y	Y	Y	Y	Y	Y	...
...	...	...	...	...	...	...	...

N	Y	Y	Y	Y	N	...

Which row in the table is paired with this set?

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**Y**	N	Y	N	Y	N	...
x_1	Y	**N**	N	Y	Y	N	...
x_2	N	N	**N**	N	Y	N	...
x_3	N	Y	N	**N**	Y	N	...
x_4	Y	N	N	N	**N**	Y	...
x_5	Y	Y	Y	Y	Y	**Y**	...
...	...	...	...	...	...	...	**...**

N	Y	Y	Y	Y	N	...

Which row in the table is paired with this set?

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	Y	N	Y	N	Y	N	...
x_1	Y	N	N	Y	Y	N	...
x_2	N	N	N	N	Y	N	...
x_3	N	Y	N	N	Y	N	...
x_4	Y	N	N	N	N	Y	...
x_5	Y	Y	Y	Y	Y	Y	...
...	...	...	...	...	...	...	...

Which row in the table is paired with this set?

| N | Y | Y | Y | Y | N | ... |

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**Y**	N	Y	N	Y	N	...
x_1	Y	**N**	N	Y	Y	N	...
x_2	N	N	**N**	N	Y	N	...
x_3	N	Y	N	**N**	Y	N	...
x_4	Y	N	N	N	**N**	Y	...
x_5	Y	Y	Y	Y	Y	**Y**	...
...	...	...	...	...	...	...	**...**

| N | Y | Y | **Y** | Y | N | ... |

Which row in the table is paired with this set?

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	Y	N	Y	N	Y	N	...
x_1	Y	N	N	Y	Y	N	...
x_2	N	N	N	N	Y	N	...
x_3	N	Y	N	N	Y	N	...
x_4	Y	N	N	N	N	Y	...
x_5	Y	Y	Y	Y	Y	Y	...
...	...	...	...	...	...	...	...

N	Y	Y	Y	Y	N	...

Which row in the table is paired with this set?

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	Y	N	Y	N	Y	N	...
x_1	Y	N	N	Y	Y	N	...
x_2	N	N	N	N	Y	N	...
x_3	N	Y	N	N	Y	N	...
x_4	Y	N	N	N	N	Y	...
x_5	Y	Y	Y	Y	Y	Y	...
...	...	...	...	...	...	...	...

N	Y	Y	Y	Y	N	...

Which row in the table is paired with this set?

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**Y**	N	Y	N	Y	N	...
x_1	Y	**N**	N	Y	Y	N	...
x_2	N	N	**N**	N	Y	N	...
x_3	N	Y	N	**N**	Y	N	...
x_4	Y	N	N	N	**N**	Y	...
x_5	Y	Y	Y	Y	Y	**Y**	...
...	...	...	...	...	...	...	...

N	Y	Y	Y	Y	N	...

Which row in the table is paired with this set?

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	N	N	N	N	N	Y	...
x_1	Y	Y	Y	Y	Y	Y	...
x_2	N	Y	N	Y	Y	N	...
x_3	Y	N	Y	N	Y	N	...
x_4	N	N	Y	Y	N	Y	...
x_5	Y	N	N	N	N	Y	...
...	...	...	...	...	...	...	...

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	N	N	N	N	N	Y	...
x_1	Y	Y	Y	Y	Y	Y	...
x_2	N	Y	N	Y	Y	N	...
x_3	Y	N	Y	N	Y	N	...
x_4	N	N	Y	Y	N	Y	...
x_5	Y	N	N	N	N	Y	...
...	...	...	...	...	...	...	...

N	Y	N	N	N	Y	...

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**N**	N	N	N	N	Y	...
x_1	Y	**Y**	Y	Y	Y	Y	...
x_2	N	Y	**N**	Y	Y	N	...
x_3	Y	N	Y	**N**	Y	N	...
x_4	N	N	Y	Y	**N**	Y	...
x_5	Y	N	N	N	N	**Y**	...
...	...	...	...	...	...	...	...

| Y | N | Y | Y | Y | N | ... |

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	N	N	N	N	N	Y	...
x_1	Y	Y	Y	Y	Y	Y	...
x_2	N	Y	N	Y	Y	N	...
x_3	Y	N	Y	N	Y	N	...
x_4	N	N	Y	Y	N	Y	...
x_5	Y	N	N	N	N	Y	...
...	...	...	...	...	...	...	...

Y	N	Y	Y	Y	N	...

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**N**	N	N	N	N	Y	...
x_1	Y	**Y**	Y	Y	Y	Y	...
x_2	N	Y	**N**	Y	Y	N	...
x_3	Y	N	Y	**N**	Y	N	...
x_4	N	N	Y	Y	**N**	Y	...
x_5	Y	N	N	N	N	**Y**	...
...	...	...	...	...	...	...	...

Y	**N**	**Y**	**Y**	**Y**	**N**	**...**

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	N	N	N	N	N	Y	...
x_1	Y	Y	Y	Y	Y	Y	...
x_2	N	Y	N	Y	Y	N	...
x_3	Y	N	Y	N	Y	N	...
x_4	N	N	Y	Y	N	Y	...
x_5	Y	N	N	N	N	Y	...
...	...	...	...	...	...	...	...

Y	N	Y	Y	Y	N	...

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**N**	N	N	N	N	Y	...
x_1	Y	**Y**	Y	Y	Y	Y	...
x_2	N	Y	**N**	Y	Y	N	...
x_3	Y	N	Y	**N**	Y	N	...
x_4	N	N	Y	Y	**N**	Y	...
x_5	Y	N	N	N	N	**Y**	...
...	...	...	...	...	...	...	**...**

Y	N	Y	**Y**	Y	N	...

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	N	N	N	N	N	Y	...
x_1	Y	Y	Y	Y	Y	Y	...
x_2	N	Y	N	Y	Y	N	...
x_3	Y	N	Y	N	Y	N	...
x_4	N	N	Y	Y	N	Y	...
x_5	Y	N	N	N	N	Y	...
...	...	...	...	...	...	...	...

| Y | N | Y | Y | Y | N | ... |

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	N	N	N	N	N	Y	...
x_1	Y	Y	Y	Y	Y	Y	...
x_2	N	Y	N	Y	Y	N	...
x_3	Y	N	Y	N	Y	N	...
x_4	N	N	Y	Y	N	Y	...
x_5	Y	N	N	N	N	Y	...
...	...	...	...	...	...	...	...

Y	N	Y	Y	Y	N	...

	x_0	x_1	x_2	x_3	x_4	x_5	...
x_0	**N**	N	N	N	N	Y	...
x_1	Y	**Y**	Y	Y	Y	Y	...
x_2	N	Y	**N**	Y	Y	N	...
x_3	Y	N	Y	**N**	Y	N	...
x_4	N	N	Y	Y	**N**	Y	...
x_5	Y	N	N	N	N	**Y**	...
...	...	...	...	...	...	...	...

Y	N	Y	Y	Y	N	...

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 nth 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:

  If S is a set, then $|S| < |\wp(S)|$.

Infinite Cardinalities

- By Cantor's Theorem:

$$|\mathbb{N}| < |\wp(\mathbb{N})|$$

$$|\wp(\mathbb{N})| < |\wp(\wp(\mathbb{N}))|$$

$$|\wp(\wp(\mathbb{N}))| < |\wp(\wp(\wp(\mathbb{N})))|$$

$$|\wp(\wp(\wp(\mathbb{N})))| < |\wp(\wp(\wp(\wp(\mathbb{N}))))|$$

$$\ldots$$

- ***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!

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 $w \in S$.

Strings and Problems

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

- Suppose that S is the set

$$S = \{ \text{"}a\text{"}, \text{"}b\text{"}, \text{"}c\text{"}, \ldots \text{"}z\text{"} \}$$

- 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_ ∈ _S_.

- Suppose that S is the set

  $S = \{$ "0", "1", "2", … "9", "10", "11", … $\}$

- 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* ∈ *S*.

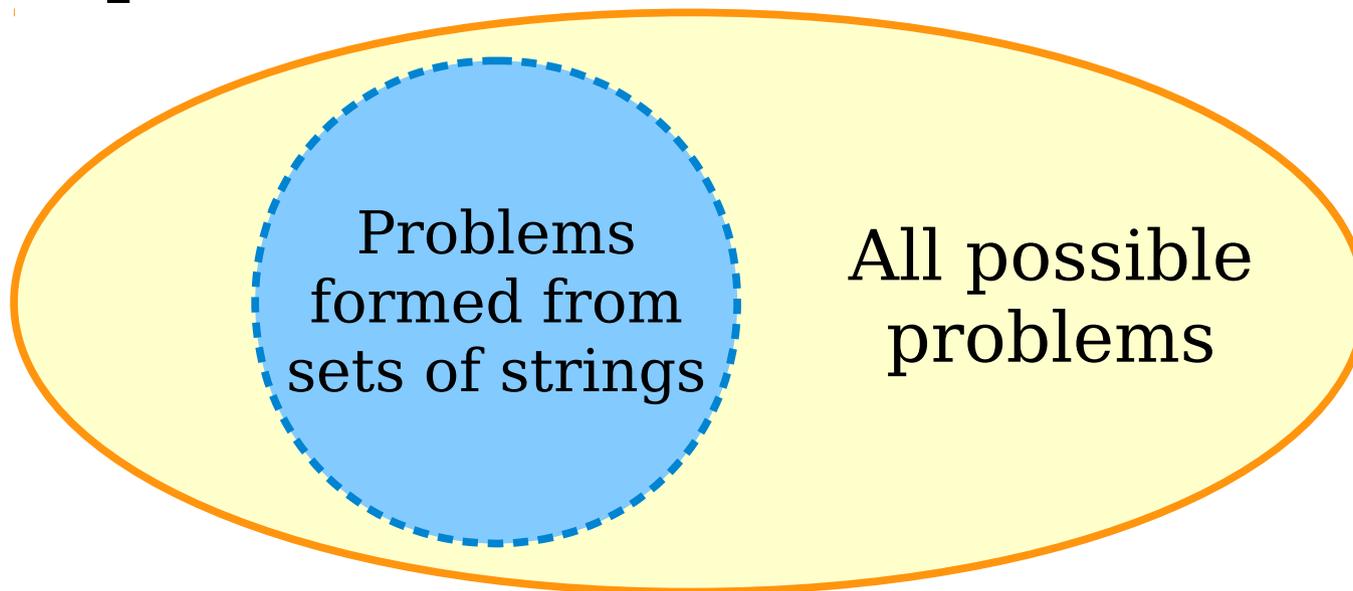- Suppose that S is the set

$$S = \{\, p \mid p \text{ is a legal Java program} \,\}$$

- From this set S, we get this problem:

Given a string *w*, determine whether *w* is a legal Java program.

Strings and Problems

- Every set of strings gives rise to a unique problem to solve.

- Other problems exist as well.

|Sets of Strings| ≤ |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!

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.

|Programs| ≤ |Strings| < |Sets of Strings| ≤ |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.

|Programs| < |Problems|

There are more problems to solve than there are programs to solve them.

|Programs| < |Problems|

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, but we don't know *which* problems are impossible!

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?