Today’s Topics:

1. Functions and set sizes
2. Infinite set sizes

1. Functions and set sizes

Set sizes

- Let $X,Y$ be finite sets, $f: X \rightarrow Y$ a function
- Theorem: If $f$ is injective then $|X| \leq |Y|$.
- Try and prove yourself first
Let $X, Y$ be finite sets, $f: X \to Y$ a function.

Theorem: If $f$ is injective then $|X| \leq |Y|$.

Proof by picture (not really a proof, just for intuition):

Actual proof:
Consider the set $S=\{(x, f(x)) : x \in X\}$. This is called the "graph of $f$".
Since $f$ is a function, each $x \in X$ appears exactly once, hence $|S| = |X|$.
Since $f$ is injective, the values of $f(x)$ are all distinct, i.e. each value $y \in Y$ appears in at most one pair. Hence $|S| \leq |Y|$.
So, $|X| \leq |Y|$.
QED.

Set sizes

Let $X, Y$ be finite sets, $f: X \to Y$ a function.

Theorem: If $f$ is surjective then $|X| \geq |Y|$.

Try and prove yourself first
Set sizes

Let \( X,Y \) be finite sets, \( f : X \to Y \) a function

Theorem: If \( f \) is surjective then \( |X| \geq |Y| \).

Proof:
Consider again the set \( S = \{(x,f(x)) : x \in X\} \).
Since \( f \) is a function, each \( x \in X \) appears exactly once hence \( |S| = |X| \).
Since \( f \) is surjective, all values \( y \in Y \) appear in at least one pair. So \( |S| \geq |Y| \).
So, \( |X| \geq |Y| \).
QED.

Set sizes

Let \( X,Y \) be finite sets, \( f : X \to Y \) a function

Theorem: If \( f \) is bijective then \( |X| = |Y| \).

Try and prove yourself first

Proof by picture (not really a proof, just for intuition):

Set sizes

Let \( X,Y \) be finite sets, \( f : X \to Y \) a function

Theorem: If \( f \) is bijective then \( |X| = |Y| \).

Proof: \( f \) is bijective so it is both injective (hence \( |X| \leq |Y| \)) and surjective (hence \( |X| \geq |Y| \)). So \( |X| = |Y| \). QED.

Set sizes
3. Infinite set sizes

Let $X, Y$ be infinite sets (e.g. natural numbers, prime numbers, reals)

We want to define a notion of set size (called cardinality) which will make sense.

Turns out, it is easier to just define which sets are smaller than other ones.

Infinite set sizes

Let $X, Y$ be infinite sets (e.g. natural numbers, prime numbers, reals).

We define $|X| \leq |Y|$ if there is an injective function $f: X \rightarrow Y$.

If $Z =$ integers and $E =$ even integers, is

A. $|Z| \leq |E|$
B. $|E| \leq |Z|$
C. Both
D. Neither

|Z| = |E|

How do we prove this?

$|E| \leq |Z|$ because we can define $f: E \rightarrow Z$ by $f(x) = x$. It is injective.

$|Z| \leq |E|$ because we can define $f: Z \rightarrow E$ by $f(x) = 2x$. It is injective.
Infinite set sizes

- Interesting facts:
  - $|Z| = |E| = |\text{prime numbers}| = |N|
  - This “size”, or cardinality, is called countable and denoted as $\aleph_0$
  - It is the smallest infinite size
  - These are sets which you can list in a sequence

- BUT!
  - $|\text{reals}| > |Z|$
  - There are “really” more reals than integers
  - We will prove that!

Countable sets

- Theorem: The rationals are countable, $|Q| = \aleph_0$

Proof:
- It’s enough to list positive rationals, because if $Q_+ = \{x_1, x_2, x_3, \ldots\}$ then we have
  $Q = \{0, x_1, -x_1, x_2, -x_2, x_3, -x_3, \ldots\}$
- Solution: list $x/y$ by first enumerating $s=x+y$, then
  $x=1, 2, \ldots, s$
  $1/1, 1/2, 2/1, 1/3, 2/3/1, 1/4, 2/3, 3/2, 4/1, \ldots$

Enumerating the rationals
Reals are uncountable

- **Theorem**: $|\mathbb{R}| > \aleph_0$
- **Proof**: by contradiction
- Say we can list all the real numbers in $[0,1]$ in a sequence $x_1, x_2, x_3, ...$
- For each number, write its representation in base 10, eg 0.12893453...
- Let $x_i = 0.b_{i,1} b_{i,2} b_{i,3} b_{i,4} ...$

Define a number $x = 0.c_1 c_2 c_3 c_4 ...$ such that $c_i \neq b_{i,i}$ for all $i$
- $c$ cannot be equal to any $x_i$ since their $i$-th digits differ
- Hence, $c$ is a real number which is not in our list
- A contradiction. The reals cannot be listed.

---

**Diagonilaztion**

- This proof technique is called *diagonilaztion*
- It is used to create an object different from all the element in our list
- It is very powerful
- For example, it can be used to show that it is generally *impossible* to know whether a computer code is correct, or even simpler questions, like if it loops forever or not