CS2209A 2017

Applied Logic for Computer Science

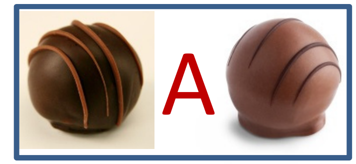
Lectures 14, 15

Set Theory and Related

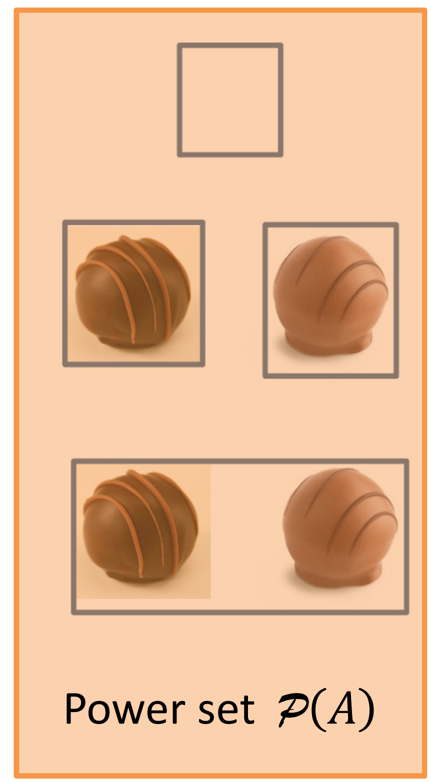
Instructor: Yu Zhen Xie

Power sets

- A **power set** of a set A, $\mathcal{P}(A)$, is a set of **all subsets** of A.
  - Think of sets as boxes of elements.
  - A subset of a set A is a box with elements of A (maybe all, maybe none, maybe some).
  - Then $\mathcal{P}(A)$ is a box containing boxes with elements of A.
  - When you open the box $\mathcal{P}(A)$, you don't see chocolates (elements of A), you see boxes.
  - A={1,2}, $\mathcal{P}(A) = \{\emptyset, \{1\}, \{2\}, \{1,2\}\}$
  - $A = \emptyset$, $\mathcal{P}(A) = \{\emptyset\}$.
    - They are not the same! There is nothing in A, and there is one element, an empty box, in $\mathcal{P}(A)$
- If A has n elements, then $\mathcal{P}(A)$ has 2^n elements.



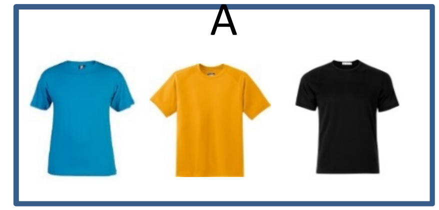
Subsets of A:



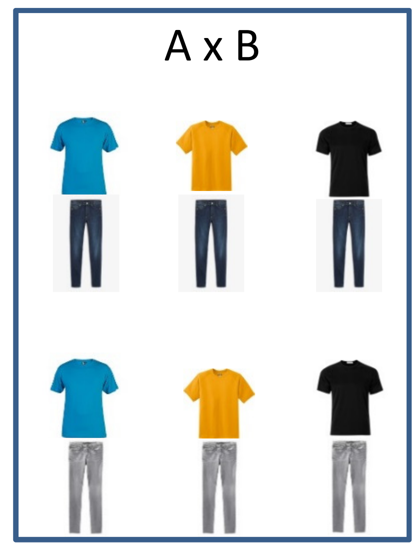
Cartesian products

- **Cartesian product** of A and B is a set of all pairs of elements with the first from A, and the second from B:
  - A x B = $\{(x, y) \mid x \in A,\ y \in B\}$
  - A={1,2,3}, B={a,b}
  - $A \times B = \{(1,a),(1,b),(2,a),(2,b),(3,a),(3,b)\}$
  - A={1,2}, $A \times A = \{(1,1),(1,2),(2,1),(2,2)\}$

	a	b
1	(1,a)	(1,b)
2	(2,a)	(2,b)
3	(3,a)	(3,b)

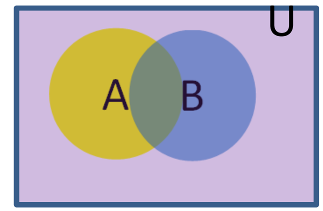


- Order of pairs does not matter, order within pairs does: $A \times B \neq B \times A$.
- Number of elements in $A \times B$ is $|A \times B| = |A| \cdot |B|$



- Can define the Cartesian product for any number of sets:
  - $A_1 \times A_2 \times \cdots \times A_k = \{(x_1, x_2, \dots x_k) \mid x_1 \in A_1 \dots x_k \in A_k\}$
  - $A = \{1,2,3\}$, B ={a,b}, C={3,4}
  - $A \times B \times C = \{(1,a,3),(1,a,4),\ (1,b,3),(1,b,4),$
    $(2,a,3),(2,a,4),\ (2,b,3),(2,b,4),$
    $(3,a,3),(3,a,4),\ (3,b,3),(3,b,4)\}$

Proofs with sets

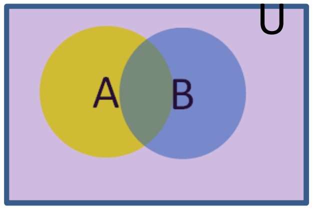


- Two ways to describe the purple area
- $\overline{A \cup B}$, $\overline{A} \cap \overline{B}$
  - $x \in \overline{A \cup B}$ when $x \notin A \cup B$
  - This happens when $x \notin A \land x \notin B$.
  - So $x \in \overline{A} \cap \overline{B}$.
    Since we picked an arbitrary x, then $\overline{A \cup B} \subseteq \overline{A} \cap \overline{B}$

  - Not quite done yet … Now let $x \in \overline{A} \cap \overline{B}$
  - Then $x \in \overline{A} \land x \in \overline{B}$. So $x \notin A \land x \notin B$.
  - $x \notin A \land x \notin B \equiv \neg(x \in A \lor x \in B)$. So $x \notin A \cup B$. Thus $x \in \overline{A \cup B}$.
  - Since x was an arbitrary element of $\overline{A} \cap \overline{B}$,
    then $\overline{A} \cap \overline{B} \subseteq \overline{A \cup B}$.
  - Therefore $\overline{A \cup B} = \overline{A} \cap \overline{B}$

Laws of set theory

- Two ways to describe the purple area
  - $\overline{A \cup B} = \overline{A} \cap \overline{B}$
- By similar reasoning,
  - $\overline{A \cap B} = \overline{A} \cup \overline{B}$



- Does this remind you of something?...
  - $\neg(p \vee q) \equiv \neg p \wedge \neg q$
  - **De Morgan's law** works in set theory!
  - What about other equivalences from logic?

More useful equivalences

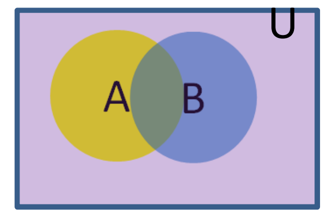
- For any formulas A, B, C:
  - A $\vee \neg A \equiv True$ $\quad$ $A \wedge \neg A \equiv False$
  - $True \vee A \equiv True$. $\quad$ $True \wedge A \equiv A$
  - $False \vee A \equiv A$. $\quad$ $False \wedge A \equiv False$
  - A$\vee$ $A \equiv A \wedge A \equiv A$

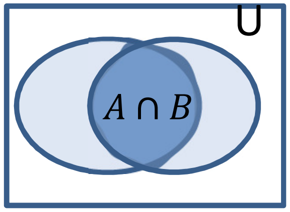
- Also, like in arithmetic (with ∨ as +, ∧ as *)
  - $A \vee B \equiv B \vee A \quad and \quad (A \vee B) \vee C \equiv A \vee (B \vee C)$
  - Same holds for ∧.
  - Also, $(A \vee B) \wedge C \equiv (A \wedge C) \vee (B \wedge C)$

- And unlike arithmetic
  - $(A \wedge B\ ) \vee C \equiv \ (A \vee C) \wedge (B \vee C)$

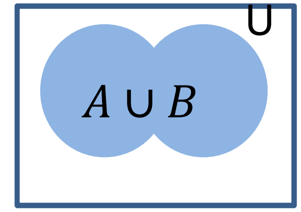
Propositions vs. sets

Propositional logic	Set theory
Negation $\neg p$	Complement $\overline{A}$
AND $p \wedge q$	Intersection $A \cap B$
OR $p \vee q$	Union $A \cup B$
FALSE	Empty set $\emptyset$
TRUE	Universe U

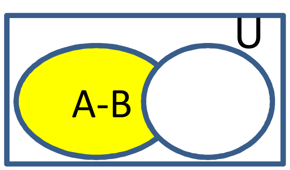


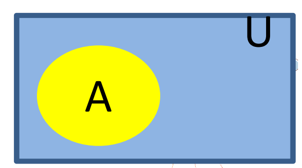


Laws of set theory



- For any sets A, B, C and universe U:
  - $A \cup \overline{A} = U$ $\quad$ $A \cap \overline{A} = \emptyset$
  - $U \cup A = U$. $\quad$ $U \cap A = A$
  - $\emptyset \cup A = A$. $\quad$ $\emptyset \cap A = \emptyset$
  - $A \cup A = A \cap A = A$
- Also, like in arithmetic (with ∨ as +, ∧ as *)
  - $A \cup B = B \cup A$ *and* $(A \cup B) \cup C = A \cup (B \cup C)$
  - Same holds for ∩.
  - Also, $(A \cup B) \cap C = (A \cap C) \cup (B \cap C)$
- And unlike arithmetic
  - $(A \cap B) \cup C \equiv (A \cup C) \cap (B \cup C)$





Boolean algebra

- The "algebra" of both propositional logic and set theory is called **Boolean algebra** (as opposed to algebra on numbers).

Propositional logic	Set theory	Boolean algebra
Negation $\neg p$	Complement $\overline{A}$	$\overline{a}$
$AND\ \ p \wedge q$	Intersection $A \cap B$	$a \cdot b$
OR $p \vee q$	Union $A \cup B$	$a + b$
FALSE	Empty set $\emptyset$	0
TRUE	Universe U	1

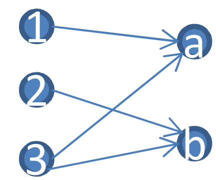
Axioms of Boolean algebra

1. $a + b = b + a, \qquad a \cdot b = b \cdot a$
2. (a+b)+c = a+(b+c) $\qquad (a \cdot b) \cdot c = a \cdot (b \cdot c)$
3. $a + (b \cdot c) = (a + b) \cdot (a + c)$
   $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$
4. There exist distinct elements 0 and 1 (among underlying set of elements B of the algebra) such that for all $a \in B$,
   $$a + 0 = a \qquad a \cdot 1 = a$$
5. For each $a \in B$ there exists an element $\overline{a} \in B$ such that
   $$a + \overline{a} = 1 \qquad a \cdot \overline{a} = 0$$

How about De Morgan, etc.? They can be derived from the axioms!

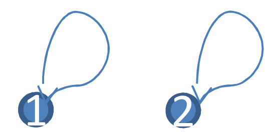
Relations

- **A relation** is a subset of a Cartesian product of sets.
  - If of two sets (set of pairs), call it a **binary** relation.
  - Of 3 sets (set of triples), **ternary**. Of k sets (set of tuples), **k-nary**
  - A={1,2,3}, B={a,b}
    - $A \times B = \{(1,a),(1,b),(2,a),(2,b),(3,a),(3,b)\}$
    - R = {(1,a), (2,b),(3,a), (3,b)} is a relation. So is R={(1,b)}.



Graph of R (bipartite)

  - A={1,2},
    - $A \times A = \{(1,1),(1,2),(2,1),(2,2)\}$
    - R={(1,1), (2,2)} (all pairs (x,y) where x=y)
    - R={(1,1),(1,2),(2,2)} (all pairs (x,y) where $x \leq y$).

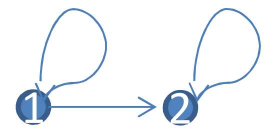


Graph of {(1,1),(2,2)}

  - A=PEOPLE
    - COUPLES ={(x,y) | Loves(x,y)}
    - PARENTS ={(x,y) | Parent(x,y)}
  - A=PEOPLE, B=DOGS, C=PLACES
    - WALKS = {(x,y,z) | x walks y in z}
      - Jane walks Buddy in spring bank park.

Types of binary relations

- A binary relation $R \subseteq A \times A$ is
  - **Reflexive** if $\forall x \in A, R(x,x)$
    - Every x is related to itself.
    - E.g. A={1,2}, R_1 = { (1,1), (2,2), (1,2)}
    - On A = $\mathbb{Z}$, $R_2 = \{(x,y) | x = y\}$ is reflexive
    - But not $R_3 = \{(x,y) |\ x < y\}$



R_1

  - **Symmetric** if $\forall x, y \in A,\ (x,y) \in R \leftrightarrow\ (y,x) \in R$
    - R_1 and R_3 above are not symmetric. R_2 is.
    - A = $\mathbb{Z}$, $R_4 = \{(x,y) | x \equiv y \ mod\ 3\ \}$ is symmetric.
  - **Transitive** if $\forall x, y, z \in A,\ (x,y) \in R \wedge (y,z) \in R \rightarrow\ (x,z) \in R$
    - R_1, R_2, R_3, R_4 are all transitive.
    - $R_5 = \{(x,y) | x, y \in \mathbb{Z} \wedge x + 1 = y\}$ is not transitive.
    - PARENT = $\{(x,y) | x, y \in PEOPLE \wedge x\ \ is\ a\ parent\ of\ \ y\}$ is not.
    - A **transitive closure** of a relation R is a relation $R^* = \{(x,z) \mid \exists k \in \mathbb{N}\ \ \exists y_0, \dots, y_k \in A\ \ (x = y_0 \wedge z = y_k \wedge \forall i \in \{0, \dots, k-1\}\ R(y_i, y_{i+1})\}$
      - That is, can get from x to z following R arrows.

Types of binary relations

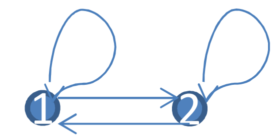
- A binary relation $R \subseteq A \times A$ is
  - **Anti-reflexive** if $\forall x \in A, \neg R(x, x)$
    - R can be neither reflexive nor anti-reflexive.
    - E.g. A={1,2}, $R_6 = \{(1,2)\}$
      - but not $R_1 = \{$ (1,1), (2,2), (1,2)$\}$ (reflexive)
      - nor $R_7 = \{$(1,1), (1,2)$\}$ (neither)
    - For $A = \mathbb{Z}$, not $R_2 = \{(x, y) | x = y\}$
      - Nor $R_4 = \{(x, y) | x \equiv y \; mod \; 3 \}$
    - But $R_3 = \{(x, y) | \; x < y\}$ is anti-reflexive.
      - So are $R_5 = \{(x, y) \in \mathbb{Z} \times \mathbb{Z} \mid x + 1 = y\}$
      - And PARENT = $\{(x, y) \in PEOPLE \times PEOPLE \mid x \;\; is \; a \; parent \; of \;\; y\}$

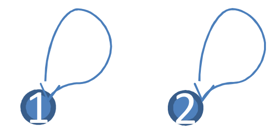
Graph of {(1,2)}

  - **Anti-symmetric**
    if $\forall x, y \in A, (x, y) \in R \wedge (y, x) \in R \rightarrow x = y$
    - $R_1, R_3, R_5, R_6, R_7, PARENT$ are anti-symmetric. R_4 is not.
    - R_2 is both symmetric and anti-symmetric.
    - $R_8 = \{(1,2), (2,1), (1,3)\}$ is neither symmetric nor anti-symmetric.

Equivalence

- A binary relation $R \subseteq A \times A$ is an **equivalence** if R is reflexive, symmetric and transitive.
    - E.g. **A={1,2}**, $R = \{(1,1),(2,2)\}$ or $R = A \times A$
    - Not R_1 = { (1,1), (2,2), (1,2)} nor $R_3 = \{(x,y) | \ x < y\}$
    - On A = $\mathbb{Z}$, $R_2 = \{(x,y)|x = y\}$ is an equivalence
    - So is $R_4 = \{(x,y)|x \equiv y \ mod \ 3 \ \}$
        - Reflexive: $\forall x \in \mathbb{Z}, \ x \ \equiv x \ mod \ 3$
        - Symmetric: $\forall x,y \in \mathbb{Z}, \ x \equiv y \bmod 3 \ \rightarrow y \equiv x \ \ mod \ 3$
        - Transitive: $\forall x,y,z \ \in \mathbb{Z}, \ x \equiv y \bmod 3 \wedge \ y \equiv z \ \ mod \ 3 \rightarrow x \equiv z \ \ mod \ 3$

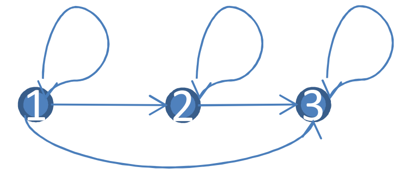


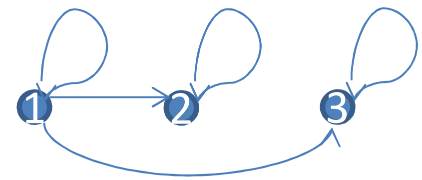


- An equivalence relation partitions A into **equivalence classes**:
    - Intersection of any two equivalence classes is ∅
    - Union of all equivalence classes is A.
    - R_4: $\mathbb{Z} = \{x \ |x \equiv 0 \ mod \ 3\} \cup \{\mathrm{x} \ | \ \mathrm{x} \equiv 1 \ mod \ 3\} \cup \{x \ |x \equiv 2 \ mod \ 3\}$
    - $R = A \times A$ gives rise to a single equivalence class. $R = \{(1,1),(2,2)\}$ to two.

Partial and total orders

- A binary relation $R \subseteq A \times A$ is an **order** if R is **reflexive**, **anti-symmetric** and **transitive**.
  - R is a **total order** if $\forall x, y \in A \;\; R(x,y) \vee R(y,x)$
    - That is, every two elements of A are related.
    - E.g. $R_1 = \{(x,y) | x,y \in \mathbb{Z} \wedge x \leq y\}$ is a total order.
    - So is alphabetical order of English words.
    - But not $R_2 = \{(x,y) | x,y \in \mathbb{Z} \wedge x < y\}$
      - not reflexive, so not an order.



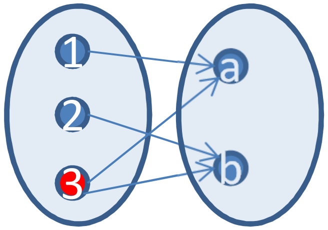


  - Otherwise, R is a **partial order**.
    - $SUBSETS = \{(A,B) \mid A, B \; are \; sets \wedge \; A \subseteq B \; \}$ is a partial order.
      - Reflexive: $\forall A, \;\; A \subseteq A$
      - Anti-symmetric: $\forall \, A, B \;\; A \subseteq B \wedge B \subseteq A \rightarrow A = B$
      - Transitive: $\forall A, B, C \;\; A \subseteq B \wedge B \subseteq C \rightarrow A \subseteq C$
      - Not total: if A ={1,2} and B ={1,3}, then neither $A \subseteq B$ nor $B \subseteq A$
    - $DIVISORS = \{(x,y) | \; x, y \in \mathbb{N} \wedge x, y \geq 2 \;\; \wedge \exists z \in \mathbb{N} \;\; y = z \cdot x\}$ is a partial order.
    - PARENT is not an order. But ANCESTOR would be, if defined so that each person is an ancestor of themselves. It is a partial order.

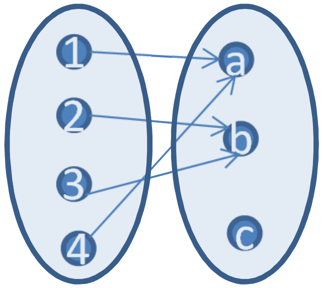
- An order may have **minimal** and **maximal** elements (maybe multiple)
  - $x \in A$ is minimal in R if $\forall y \in A \;\; y \neq x \rightarrow \;\neg R(y,x)$
    - and maximal if $\forall y \in A \; y \neq x \rightarrow \neg R(x,y)$
  - $\emptyset$ is minimal in SUBSETS (its unique minimum); universe is maximal (its unique maximum).
  - All primes are minimal in DIVISORS, and there are no maximal elements.

Functions

- **A function** $f: X \to Y$ is a **relation** on $\mathrm{X} \times Y$ such that for every $x \in \mathrm{X}$ there is **at most one** $y \in Y$ for which (x, y) is in the relation.
  - Usual notation: $f(x) = y$
    - y is an **image** of x under f.
  - X is the **domain** of f
  - Y is the **codomain** of f
  - **Range** of f (**image** of X under f):
    - $\{\mathrm{y} \in Y \mid \exists x \in X, f(x) = y\}$
  - **Preimage** of a given $y \in Y$:
    - $\{x \in X \mid f(x) = y\}$
      - Preimage of b is {2,3}.



This R is not a function

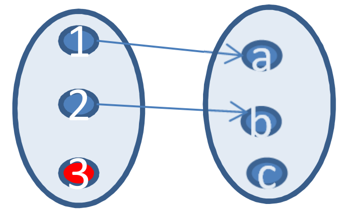


This R is a function with domain {1,2,3,4}, codomain {a,b,c} and range {a,b}

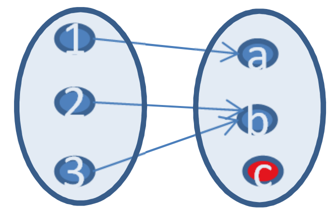
Functions

- **A function $f: X \rightarrow Y$ is**
  - **Total**: $\forall x \in X\ \exists y \in Y\ f(x) = y$
    - f: $\mathbb{Z} \rightarrow \mathbb{Z}$
    - $f(x) = x + 1$ is total.
    - $f(x) = \frac{100}{x}$ is not total. Why?
  - **Onto**: $\forall y \in Y\ \exists x \in X\ f(x) = y$
    - $f(x) = x + 1$ is onto over $\mathbb{Z}$, but not over $\mathbb{N}$
  - **One-to-one:** $\forall x_1, x_2 \in X\ (f(x_1) = \mathrm{f}(\mathrm{x}_1) \rightarrow x_1 = x_2)$
    - $f(x) = x + 1$ is one-to-one.
    - $f(x) = x^2$ is not one-to-one
  - **Bijection**: both one-to-one and onto.
    - $f(x) = x + 1$ is a bijection over $\mathbb{Z}$.

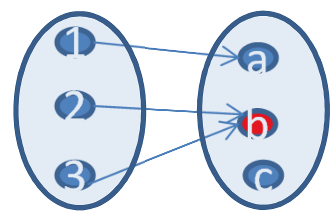
Not total



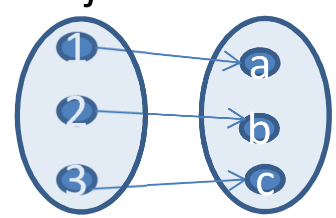
Not onto



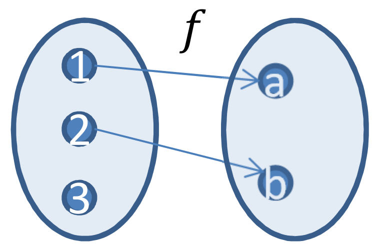
Not one-to-one



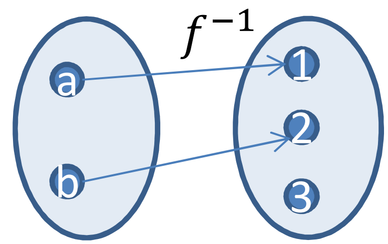
Bijection



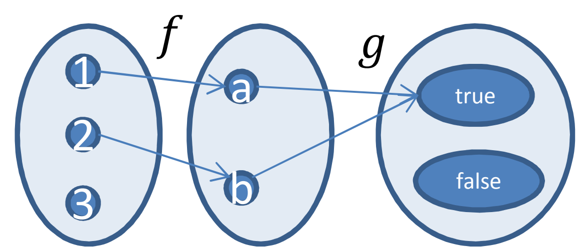
Functions



- An **inverse** of f is $f^{-1}: Y \to X$, such that $f^{-1}(y) = x$ iff $f(x) = y$
  - $f(x) = x + 1, f^{-1}(y) = y - 1$
  - *Only one-to-one functions have an inverse*



- **Composition** of $f: X \to Y$ and $g: Y \to Z$ is $g \circ f: X \to Z$ such that $(g \circ f)(x) = g(f(x))$
  - $f(x) = \frac{x}{5},\ g(x) = \lceil x \rceil$, over $\mathbb{R}$
    - $\lceil x \rceil$ is ceiling: x rounded up to nearest integer.
  - $(g \circ f)(x) = g(f(x)) = \left\lceil \frac{x}{5} \right\rceil$
  - $(f \circ g)(x) = f(g(x)) = \frac{\lceil x \rceil}{5}$
  - $(g \circ f)(12.5) = \lceil 2.5 \rceil = 3$
  - $(f \circ g)(12.5) = 13/5 = 2.6$
    - Order matters!



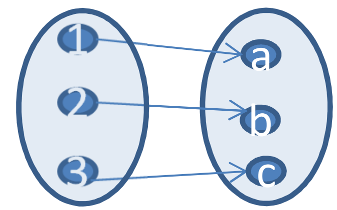
Puzzle: the barber

- In a certain village, there is a (male) barber who shaves all and only those men of the village who do not shave themselves.

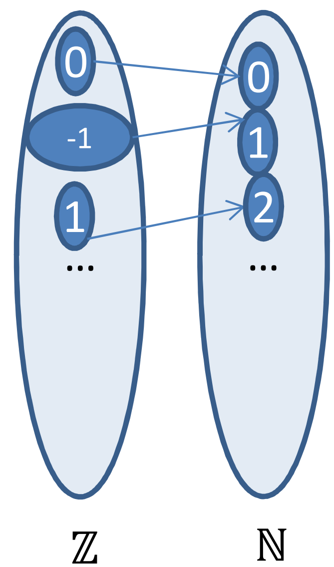
- *Question: who shaves the barber?*

Cardinalities of infinite sets

- Two finite sets A and B have the **same cardinality** if they have the same number of elements
  - That is, for each element of A there is exactly one matching element of B.



- For infinite sets A and B, define |A|=|B| iff there exists a **bijection** between A and B.
  - If there is both a one-to-one function from A to B, and an onto function from A to B.
- A set A is **countable** iff |A| = |ℕ|.
  - $\mathbb{Z}$ is countable: take $f: \mathbb{Z} \to \mathbb{N}$, $f(x) = 2x$ if $x \geq 0$, else $f(x) = -(1 + 2x)$
  - Set of all finite strings over {0,1}, denoted $\{0,1\}^*$, is countable.
    - Empty string, 0, 1, 00, 01, 10, 11, 000, 001, …
  - An infinite subset of a countable language is countable. A Cartesian product of countable languages is countable:
    - $\mathbb{N} \times \mathbb{N}$: (0,0), (0,1), (1,0), (2,0), (1,1), (0,2), (3,0), (2,1), (1,2),…
  - $\mathbb{Q}$ is countable: $\mathbb{Q} \subset \mathbb{Z} \times \mathbb{Z}$



Diagonalization: $\mathbb{R}$

- Is there a bigger infinity?
  - Yes! In particular, $\mathbb{R}$ is uncountable. Even [0,1) interval of the real line is uncountable!
    - Reals may have infinite strings of digits after the decimal point.
    - Imagine if there were a numbered list of all reals in [0,1)
      - $a_0, a_1, a_2, a_3, \ldots$
    - For example:
      - a_1 = 0.23145…
      - a_2 = 0.30000…
      - …
  - Let number *d* be:
    - *d*[i]=$(a_i[i] + 1) \ mod \ 10$
    - Here, $[i]$ is i^{th} digit.
    - This *d* is a valid real number!

0.	r[1]	r[2]	r[3]	r[4]	r[5]	…	r[k]		
a_0	2	3	1	4	5	…			
1	3	0	0	0	0	…			
2	9	9	9	9	9	…			
…									
k	2	1	3	4	3	…	5	…	
…									
d	3	1	0	…	…	…	6	…	

  - But if number *d* were in the list, e.g. k^{th}, a contradiction
    - It would have to differ from itself in k^{th} place.

Diagonalization: languages

- An **alphabet** is a finite set of symbols.
  - For example, {0,1} is the binary alphabet.
- A **language** is a set of finite strings over a given alphabet.
  - For example, $\{0,1\}^*$ is the set of all finite binary strings.
  - PRIMES ⊂ $\{0,1\}^*$ is all strings coding prime numbers in binary.
  - PYTHON ⊂ $\{0,1\}^*$ is all strings coding valid Python programs in binary.
- Every language is countable.
  - $\{0,1\}^*$, PRIMES, PYTHON are countable
- Set of all languages is uncountable.
  - Put "yes" if $s \in L$, "no" if $s \notin L$
  - Let language D be:
    - s ∈ D iff $s \notin L_s$
  - If D were in the list, e.g. as L_k, a contradiction
    - It would have to differ from itself in k^{th} place.

		0	1	00	01	...	s_k		
L_0	yes	yes	no	yes	yes	...			
L_1	yes	no	yes	no	yes	...			
	no	no	no	no	no	...			
...									
L_k	no	yes	yes	no	yes	...	yes	...	
...									
D	no	yes	yes	...	...	...	no	...	

- So there is a language for which there is no Python program which would correctly print "yes" on strings in the language, and "no" otherwise.

Puzzle: the barber club

- In a certain barber's club,
  - Every member has shaved at least one other member
  - No member shaved himself
  - No member has been shaved by more than one member
  - There is a member who has never been shaved.

- *Question: how many barbers are in this club?*

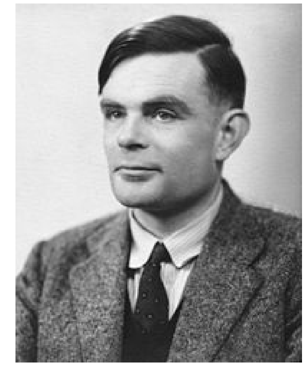
…

Infinitely many!

Barber 0 grows a beard.

For all n$\in \mathbb{N}$, barber n shaves barber n+1

The Halting Problem



- A specific example of a problem not solvable by any program: the **Halting problem,** invented by **Alan Turing**:
  - Input:
    - Prog: A program as piece of code (e.g., in Python):
    - x: Input to that program.
  - Output:
    - “yes” if this Prog(x) stops (that is, program Prog stops on input x).
    - “no” if Prog goes into an infinite loop on input x.
  - Suppose there is a program Halt(Prog, x) which always stops and prints “yes” or “no” correctly.
    - Nothing wrong with giving a piece of code as an input to another program.
  - Then there is a program HaltOnItself(Prog) = Halt(Prog,Prog)
  - And a program Diag(Prog):
    - if Halt(Prog, Prog) says “yes”, go into infinite loop (e.g. add “while 0 <1: “ to Halt’s code).
    - if Halt(Prog, Prog) says “no”, stop.
  - Now, what should Diag(Diag) do?...
    - Paradox! It is like a barber who shaves everybody who does not shave himself.
    - So the program Diag does not exist… Thus the program Halt does not exist!

- So there is no program that would always stop and give the right answer for the Halting problem.