CSE 311 Lecture 27: Cardinality and Uncomputability

Emina Torlak and Sami Davies

Topics

Course evaluation

Is open; please tell us what you think!

Proving irregularity

A quick review of Lecture 26.

Languages and representations

How powerful are general-purpose programming languages?

Cardinality and countability

What does it mean for two sets to have the same size?

Uncomputability

Are there problems computers can't solve?

Course evaluation

Is open; please tell us what you think!

Proving irregularity

A quick review of Lecture 26.

A template for proving that a language L is not regular

① Suppose for contradiction that some DFA M recognizes L.

A template for proving that a language L is not regular

① Suppose for contradiction that some DFA M recognizes L.

② Consider the set $S = \{\ldots\}$.

S must be infinite, and for *every pair* of prefixes $s_a \neq s_b \in S$, there is a suffix t such that one of of $s_a t$, $s_b t$ is in L but not the other.

A template for proving that a language L is not regular

① Suppose for contradiction that some DFA M recognizes L.

② Consider the set $S = \{\ldots\}$.

③ Since S is infinite and M has finitely many states, there must be two strings $s_a, s_b \in S$ such that $s_a \neq s_b$ and both end in the same state of M.

S must be infinite, and for *every pair* of prefixes $s_a \neq s_b \in S$, there is a suffix t such that one of of $s_a t, s_b t$ is in L but not the other.

We don't get to choose s_a and s_b!

A template for proving that a language L is not regular

① Suppose for contradiction that some DFA M recognizes L.

② Consider the set $S = \{\ldots\}$.

③ Since S is infinite and M has finitely many states, there must be two strings $s_a, s_b \in S$ such that $s_a \neq s_b$ and both end in the same state of M.

④ Consider appending t to both s_a and s_b.

S must be infinite, and for *every pair* of prefixes $s_a \neq s_b \in S$, there is a suffix t such that one of of $s_a t$, $s_b t$ is in L but not the other.

We don't get to choose s_a and s_b!

t should be an accept suffix for s_a but not s_b.

A template for proving that a language L is not regular

① Suppose for contradiction that some DFA M recognizes L.

② Consider the set $S = \{\ldots\}$.

③ Since S is infinite and M has finitely many states, there must be two strings $s_a, s_b \in S$ such that $s_a \neq s_b$ and both end in the same state of M.

④ Consider appending t to both s_a and s_b.

⑤ Since s_a and s_b end in the same state of M, then $s_a t$ and $s_b t$ also end in the same state q of M. Since $s_a t \in L$ and $s_b t \notin L$, M does not recognize L.

S must be infinite, and for *every pair* of prefixes $s_a \neq s_b \in S$, there is a suffix t such that one of of $s_a t$, $s_b t$ is in L but not the other.

We don't get to choose s_a and s_b!

t should be an accept suffix for s_a but not s_b.

A template for proving that a language L is not regular

① Suppose for contradiction that some DFA M recognizes L.

② Consider the set $S = \{\ldots\}$.

③ Since S is infinite and M has finitely many states, there must be two strings $s_a, s_b \in S$ such that $s_a \neq s_b$ and both end in the same state of M.

④ Consider appending t to both s_a and s_b.

⑤ Since s_a and s_b end in the same state of M, then $s_a t$ and $s_b t$ also end in the same state q of M. Since $s_a t \in L$ and $s_b t \notin L$, M does not recognize L.

⑥ Since M was arbitrary, no DFA recognizes L.

S must be infinite, and for *every pair* of prefixes $s_a \neq s_b \in S$, there is a suffix t such that one of of $s_a t$, $s_b t$ is in L but not the other.

We don't get to choose s_a and s_b!

t should be an accept suffix for s_a but not s_b.

Example: prove that $L = \{0^n 1^n : n \geq 0\}$ is not regular

Suppose for contradiction that some DFA M recognizes L.

Consider the set $S =$

Example: prove that $L = \{0^n 1^n : n \geq 0\}$ is not regular

Suppose for contradiction that some DFA M recognizes L.

Consider the set $S = \{0^n : n \geq 0\}$.

Example: prove that $L = \{0^n 1^n : n \geq 0\}$ is not regular

Suppose for contradiction that some DFA M recognizes L.

Consider the set $S = \{0^n : n \geq 0\}$.

Since S is infinite and M has finitely many states, there must be two strings $0^a, 0^b \in S$ with $a \neq b$ that both end in the same state of M.

Example: prove that $L = \{0^n 1^n : n \geq 0\}$ is not regular

Suppose for contradiction that some DFA M recognizes L.

Consider the set $S = \{0^n : n \geq 0\}$.

Since S is infinite and M has finitely many states, there must be two strings $0^a, 0^b \in S$ with $a \neq b$ that both end in the same state of M.

Consider appending to both 0^a and 0^b.

Example: prove that $L = \{0^n 1^n : n \geq 0\}$ is not regular

Suppose for contradiction that some DFA M recognizes L.

Consider the set $S = \{0^n : n \geq 0\}$.

Since S is infinite and M has finitely many states, there must be two strings $0^a, 0^b \in S$ with $a \neq b$ that both end in the same state of M.

Consider appending 1^a to both 0^a and 0^b.

Example: prove that $L = \{0^n 1^n : n \geq 0\}$ is not regular

Suppose for contradiction that some DFA M recognizes L.

Consider the set $S = \{0^n : n \geq 0\}$.

Since S is infinite and M has finitely many states, there must be two strings $0^a, 0^b \in S$ with $a \neq b$ that both end in the same state of M.

Consider appending 1^a to both 0^a and 0^b.

Since 0^a and 0^b end in the same state of M, then $0^a 1^a$ and $0^b 1^a$ also end in the same state q of M. Since $0^a 1^a \in L$ and $0^b 1^a \notin L$, M does not recognize L.

Example: prove that $L = \{0^n 1^n : n \geq 0\}$ is not regular

Suppose for contradiction that some DFA M recognizes L.

Consider the set $S = \{0^n : n \geq 0\}$.

Since S is infinite and M has finitely many states, there must be two strings $0^a, 0^b \in S$ with $a \neq b$ that both end in the same state of M.

Consider appending 1^a to both 0^a and 0^b.

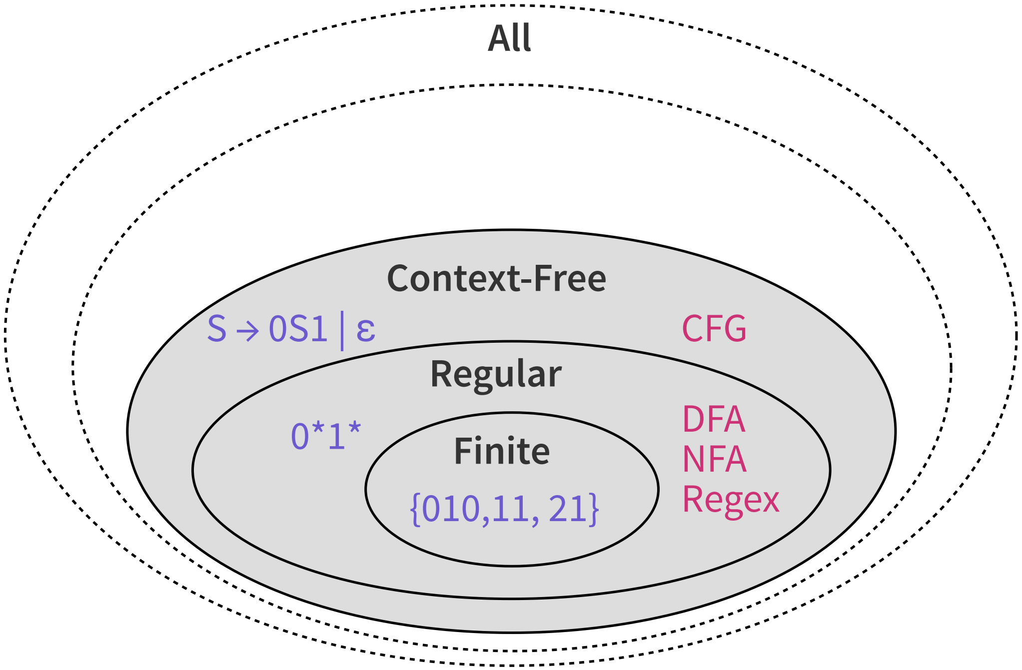
Since 0^a and 0^b end in the same state of M, then $0^a 1^a$ and $0^b 1^a$ also end in the same state q of M. Since $0^a 1^a \in L$ and $0^b 1^a \notin L$, M does not recognize L.

Since M was arbitrary, no DFA recognizes L.

Languages and representations

How powerful are general-purpose programming languages?

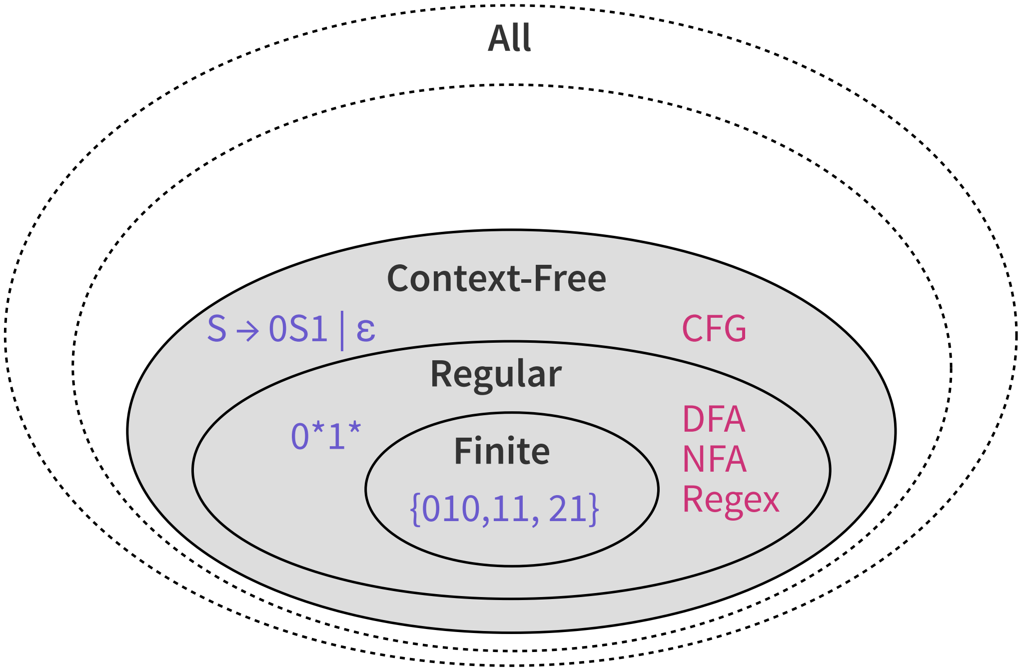
A hierarchy of languages and representations



A hierarchy of languages and representations

We can think of languages as functions from strings to booleans.

Such a function returns `true` iff a string is in the language.



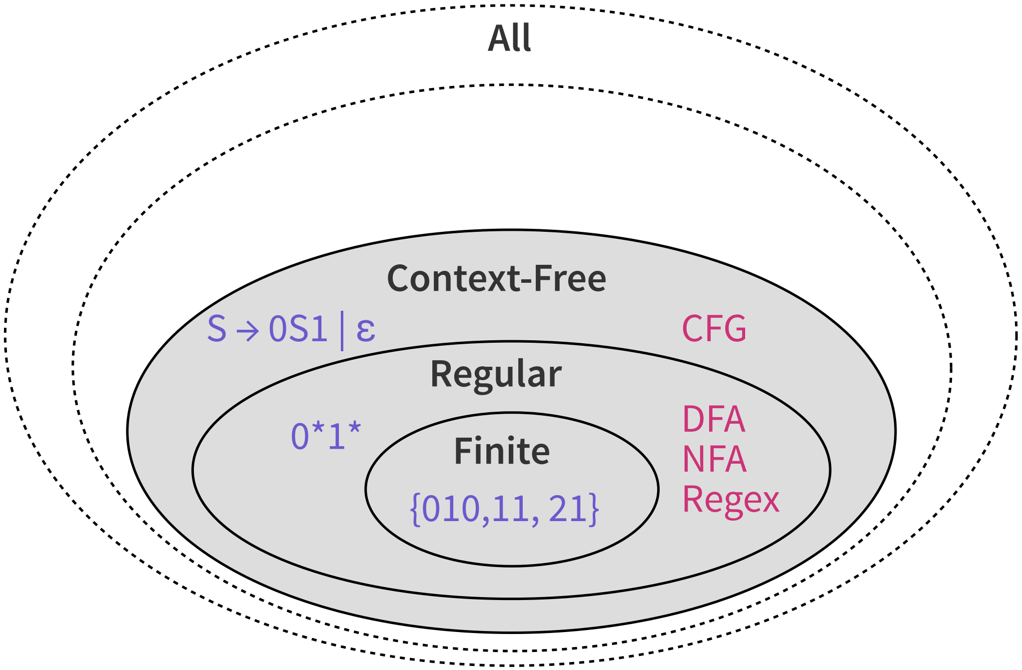
A hierarchy of languages and representations

We can think of languages as functions from strings to booleans.

Such a function returns `true` iff a string is in the language.

DFAs and CFGs can represent some functions but not others.

E.g., no DFA for $\{0^n 1^n : n \geq 0\}$.



A hierarchy of languages and representations

We can think of languages as functions from strings to booleans.

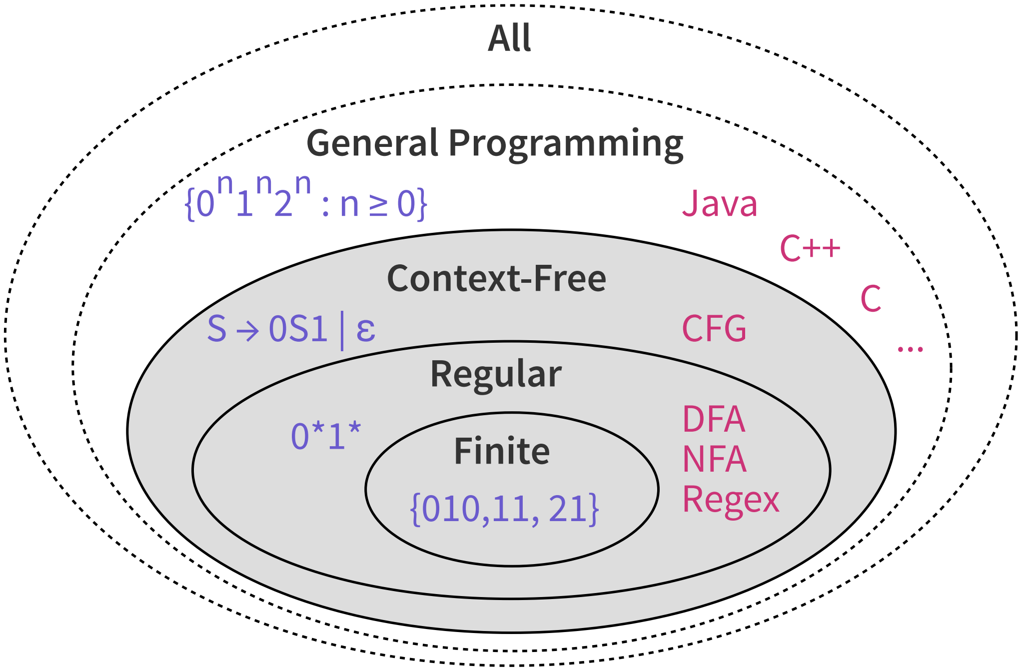
Such a function returns `true` iff a string is in the language.

DFAs and CFGs can represent some functions but not others.

E.g., no DFA for $\{0^n 1^n : n \geq 0\}$.

General-purpose programs can represent even more functions.

E.g., no CFG for $\{0^n 1^n 2^n : n \geq 0\}$.



A hierarchy of languages and representations

We can think of languages as functions from strings to booleans.

Such a function returns `true` iff a string is in the language.

DFAs and CFGs can represent some functions but not others.

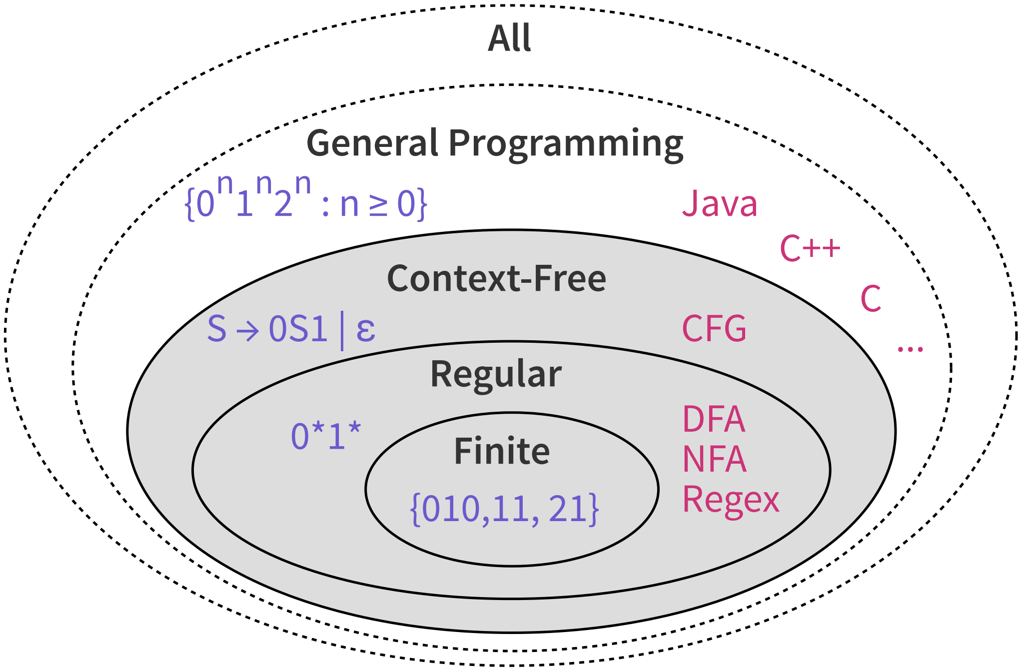
E.g., no DFA for $\{0^n 1^n : n \geq 0\}$.

General-purpose programs can represent even more functions.

E.g., no CFG for $\{0^n 1^n 2^n : n \geq 0\}$.

Are there some functions no program can represent?

That's what we'll study in these last two lectures :)

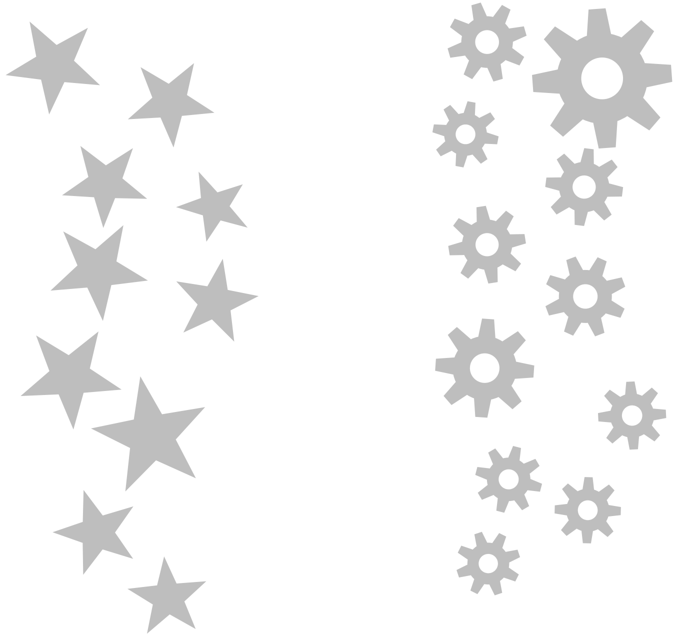


Cardinality and countability

What does it mean for two sets to have the same size?

Understanding cardinality

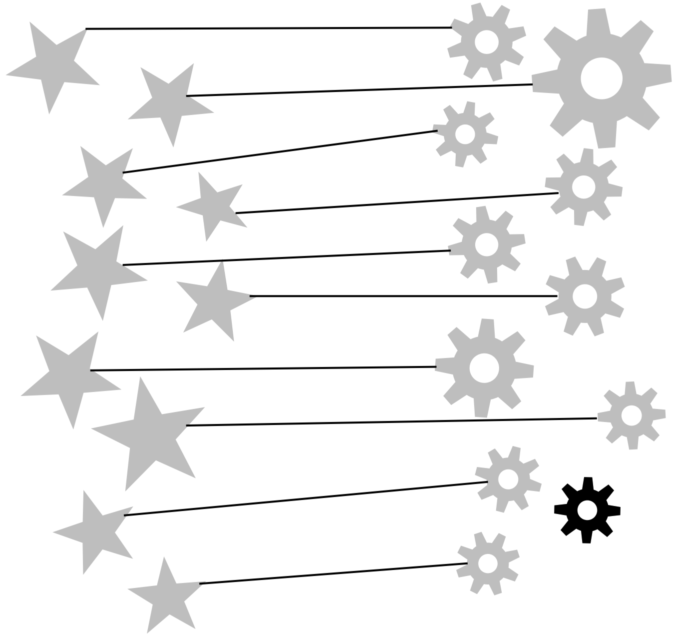
What does it mean for two sets to have the same size?



Understanding cardinality

What does it mean for two sets to have the same size?

We can establish a *one-to-one correspondence* between their elements.

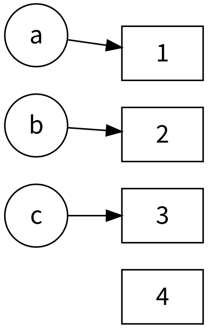


Defining one-to-one correspondences

One-to-one (injective) functions

A function $f\colon A \longrightarrow B$ is *one-to-one* (1-1) if every output corresponds to at most one input:

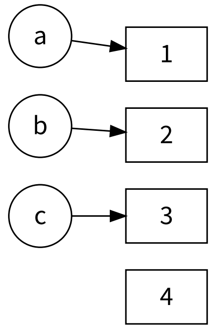
$f(x) = f(x') \Rightarrow x = x'$ for all $x, x' \in A$.



Defining one-to-one correspondences

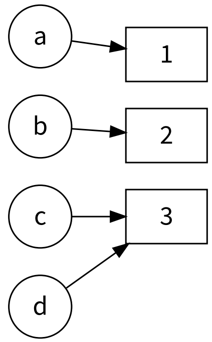
One-to-one (injective) functions

A function $f: A \longrightarrow B$ is *one-to-one* (1-1) if every output corresponds to at most one input:
$f(x) = f(x') \Rightarrow x = x'$ for all $x, x' \in A$.



Onto (surjective) functions

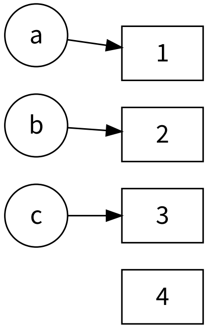
A function $f: A \longrightarrow B$ is *onto* if there is at least one input for every output: for each $y \in B$, there is an $x \in A$ such that $f(x) = y$.



Defining one-to-one correspondences

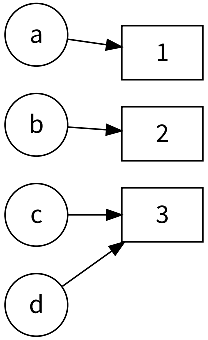
One-to-one (injective) functions

A function $f: A \to B$ is *one-to-one* (1-1) if every output corresponds to at most one input: $f(x) = f(x') \Rightarrow x = x'$ for all $x, x' \in A$.



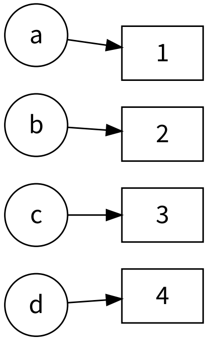
Onto (surjective) functions

A function $f: A \to B$ is *onto* if there is at least one input for every output: for each $y \in B$, there is an $x \in A$ such that $f(x) = y$.



One-to-one correspondences (bijections)

A function $f: A \to B$ is a *one-to-one correspondence* if it is both one-to-one and onto.



Defining cardinality

Cardinality of two sets

Sets A and B have the same *cardinality* if there is a one-to-one correspondence between them, i.e., there is a bijection $f: A \to B$.

Defining cardinality

Cardinality of two sets

Sets A and B have the same *cardinality* if there is a one-to-one correspondence between them, i.e., there is a bijection $f\colon A \to B$.

This definition also works for infinite sets!

Defining cardinality

Cardinality of two sets

Sets A and B have the same *cardinality* if there is a one-to-one correspondence between them, i.e., there is a bijection $f: A \to B$.

This definition also works for infinite sets!

Example: do $\mathbf{N}$ and even natural numbers have the same cardinality?

Defining cardinality

Cardinality of two sets

Sets A and B have the same *cardinality* if there is a one-to-one correspondence between them, i.e., there is a bijection $f\colon A \to B$.

This definition also works for infinite sets!

Example: do $\mathbb{N}$ and even natural numbers have the same cardinality?

Yes! The 1-1 correspondence is $f(n) = 2n$.

0	1	2	3	4	5	6	7	8	9	10	…
0	2	4	6	8	10	12	14	16	18	20	…

Countable sets

Countable set

A set is *countable* iff it has the same cardinality as some subset of $\mathbb{N}$.

Countable sets

Countable set

A set is *countable* iff it has the same cardinality as some subset of N.

Equivalently, we can say that …

A set S is countable iff there is an *onto* function $g : \mathrm{N} \to S$.

Countable sets

Countable set

A set is *countable* iff it has the same cardinality as some subset of N.

Equivalently, we can say that …

A set S is countable iff there is an *onto* function $g : \mathrm{N} \to S$.

And we can also say that …

A set S is countable iff we can order its elements: $S = \{x_0, x_1, x_2, \ldots\}$.

Countable sets

Countable set

A set is *countable* iff it has the same cardinality as some subset of $\mathbb{N}$.

Equivalently, we can say that …

A set S is countable iff there is an *onto* function $g : \mathbb{N} \to S$.

And we can also say that …

A set S is countable iff we can order its elements: $S = \{x_0, x_1, x_2, \ldots\}$.

Example: is the set $\mathbb{Z}$ of all integers countable?

Countable sets

Countable set

A set is *countable* iff it has the same cardinality as some subset of $\mathbb{N}$.

Equivalently, we can say that …

A set S is countable iff there is an *onto* function $g : \mathbb{N} \to S$.

And we can also say that …

A set S is countable iff we can order its elements: $S = \{x_0, x_1, x_2, \dots\}$.

Example: is the set $\mathbb{Z}$ of all integers countable?

$\mathbb{N}$	0	1	2	3	4	5	6	7	8	9	10	…
$\mathbb{Z}$	0	1	-1	2	-2	3	-3	4	-4	5	-5	…

Is the set $\mathbf{Q}^+$ of positive rationals countable?

There are infinitely many rationals between any two rational numbers.

Is the set $\mathbf{Q}^+$ of positive rationals countable?

There are infinitely many rationals between any two rational numbers.

1 / 1	1 / 2	1 / 3	1 / 4	1 / 5	1 / 6	1 / 7	1 / 8	…
2 / 1	2 / 2	2 / 3	2 / 4	2 / 5	2 / 6	2 / 7	2 / 8	…
3 / 1	3 / 2	3 / 3	3 / 4	3 / 5	3 / 6	3 / 7	3 / 8	…
4 / 1	4 / 2	4 / 3	4 / 4	4 / 5	4 / 6	4 / 7	4 / 8	…
5 / 1	5 / 2	5 / 3	5 / 4	5 / 5	5 / 6	5 / 7	5 / 8	…
6 / 1	6 / 2	6 / 3	6 / 4	6 / 5	6 / 6	6 / 7	6 / 8	…
7 / 1	7 / 2	7 / 3	7 / 4	7 / 5	7 / 6	7 / 7	7 / 8	…
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	…

Is the set $\mathbb{Q}^+$ of positive rationals countable?

There are infinitely many rationals between any two rational numbers.

1 / 1	1 / 2	1 / 3	1 / 4	1 / 5	1 / 6	1 / 7	1 / 8	…
2 / 1	2 / 2	2 / 3	2 / 4	2 / 5	2 / 6	2 / 7	2 / 8	…
3 / 1	3 / 2	3 / 3	3 / 4	3 / 5	3 / 6	3 / 7	3 / 8	…
4 / 1	4 / 2	4 / 3	4 / 4	4 / 5	4 / 6	4 / 7	4 / 8	…
5 / 1	5 / 2	5 / 3	5 / 4	5 / 5	5 / 6	5 / 7	5 / 8	…
6 / 1	6 / 2	6 / 3	6 / 4	6 / 5	6 / 6	6 / 7	6 / 8	…
7 / 1	7 / 2	7 / 3	7 / 4	7 / 5	7 / 6	7 / 7	7 / 8	…
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	…

Counting $\mathbb{Q}^+$ with dovetailing

The set of all positive rational numbers is countable.

$$\mathbb{Q}^+ = \{1/1, 2/1, 1/2, 3/1, 2/2, 1/3, 4/1, 2/3, 3/2, \ldots\}$$

List elements in the order of the sum of the numerator and denominator, breaking ties according to the denominator.

Only k pairs of positive numbers add up to $k+1$, so every positive rational number comes up some point.

This technique is called *dovetailing*.

Σ^* is countable for every finite Σ

How would we show this?

Alphabetical / lexicographic order doesn't work (infinitely many A's):

A, AA, AAA, AAAA, AAAAA, …

Σ^* is countable for every finite Σ

How would we show this?

Alphabetical / lexicographic order doesn't work (infinitely many A's):

A, AA, AAA, AAAA, AAAAA, …

Use dovetailing again!

List strings in the order of length, breaking ties lexicographically.

There are only $|\Sigma|^k$ strings on length k.

Σ^* is countable for every finite Σ

How would we show this?

Alphabetical / lexicographic order doesn't work (infinitely many A's):

A, AA, AAA, AAAA, AAAAA, …

Use dovetailing again!

List strings in the order of length, breaking ties lexicographically.

There are only $|\Sigma|^k$ strings on length k.

For example, $\{0, 1\}^*$ is countable:

$\{\varepsilon, 0, 1, 00, 01, 10, 11, 000, 001, 010, 011, 100, 101, 110, 111, \ldots\}$

The set of all Java programs is countable

The set of all Java programs is countable

Java programs are just strings in Σ^* where Σ is the alphabet of ASCII characters. Since Σ^* is countable, so is the set of all Java programs.

The set of all Java programs is countable

Java programs are just strings in Σ^* where Σ is the alphabet of ASCII characters. Since Σ^* is countable, so is the set of all Java programs.

This is true for other programming languages too: C, C++, Python, JavaScript, etc.

The set of all Java programs is countable

Java programs are just strings in Σ^* where Σ is the alphabet of ASCII characters. Since Σ^* is countable, so is the set of all Java programs.

This is true for other programming languages too: C, C++, Python, JavaScript, etc.

So, is everything countable?

Real numbers are not countable

Theorem [due to Cantor]

The set of real numbers between 0 and 1, $[0, 1)$, is not countable.

Real numbers are not countable

Theorem [due to Cantor]

The set of real numbers between 0 and 1, $[0, 1)$, is not countable.

Proof will be by contradiction. Using a method called *diagonalization*.

Proof that $[0, 1)$ is uncountable: preliminaries

First, note that every number in $[0, 1)$ has an infinite decimal expansion:

$$
\begin{aligned}
1/2 &= 0.5000000000000000000000\ldots \\
1/3 &= 0.3333333333333333333333\ldots \\
1/7 &= 0.1428571428571428571428 5\ldots \\
\pi - 3 &= 0.1415926535897932384626 4\ldots \\
1/5 &= 0.1999999999999999999999\ldots \\
&= 0.2000000000000000000000\ldots
\end{aligned}
$$

This representation is unique except for the cases where the decimal expansion ends in all 0's or all 9's. We will use the all 0's representation.

Proof that $[0, 1)$ is uncountable: diagonalization

Suppose for contradiction that there is a list $\{r_0, r_1, r_2, \ldots\}$ of all real numbers in $[0, 1)$.

r_0	$0.500000000000\ldots$
r_1	$0.333333333333\ldots$
r_2	$0.142857142857\ldots$
r_3	$0.141592653589\ldots$
r_4	$0.200000000000\ldots$
$\vdots$	

Proof that $[0, 1)$ is uncountable: diagonalization

Suppose for contradiction that there is a list $\{r_0, r_1, r_2, \ldots\}$ of all real numbers in $[0, 1)$.

Consider the digits $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., the n-th digit of r_n for $n \in \mathbb{N}$.

r_0	0.500000000000…
r_1	0.333333333333…
r_2	0.142857142857…
r_3	0.141592653589…
r_4	0.200000000000…
⋮	

Proof that $[0, 1)$ is uncountable: diagonalization

Suppose for contradiction that there is a list $\{r_0, r_1, r_2, \ldots\}$ of all real numbers in $[0, 1)$.

Consider the digits $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., the n-th digit of r_n for $n \in \mathbb{N}$.

For each such digit x_i, construct the digit $\hat{x}_i$ as follows:

- If $x_i = 1$ then $\hat{x}_i = 0$.
- If $x_i \neq 1$ then $\hat{x}_i = 1$.

r_0	$0.500000000000\ldots$
r_1	$0.333333333333\ldots$
r_2	$0.142857142857\ldots$
r_3	$0.141592653589\ldots$
r_4	$0.200000000000\ldots$
$\vdots$	

Proof that $[0, 1)$ is uncountable: diagonalization

Suppose for contradiction that there is a list $\{r_0, r_1, r_2, \ldots\}$ of all real numbers in $[0, 1)$.

Consider the digits $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., the n-th digit of r_n for $n \in \mathbb{N}$.

For each such digit x_i, construct the digit $\hat{x}_i$ as follows:

- If $x_i = 1$ then $\hat{x}_i = 0$.
- If $x_i \neq 1$ then $\hat{x}_i = 1$.

Now, consider the number $\hat{r} = 0.\hat{x}_0\hat{x}_1\hat{x}_2\hat{x}_3\ldots$

r_0	$0.\mathbf{5}00000000000\ldots$
r_1	$0.3\mathbf{3}3333333333\ldots$
r_2	$0.14\mathbf{2}857142857\ldots$
r_3	$0.141\mathbf{5}92653589\ldots$
r_4	$0.2000\mathbf{0}0000000\ldots$
$\vdots$	

Proof that $[0, 1)$ is uncountable: diagonalization

Suppose for contradiction that there is a list $\{r_0, r_1, r_2, \ldots\}$ of all real numbers in $[0, 1)$.

Consider the digits $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., the n-th digit of r_n for $n \in \mathrm{N}$.

For each such digit x_i, construct the digit $\hat{x}_i$ as follows:

- If $x_i = 1$ then $\hat{x}_i = 0$.
- If $x_i \neq 1$ then $\hat{x}_i = 1$.

Now, consider the number $\hat{r} = 0.\hat{x}_0\hat{x}_1\hat{x}_2\hat{x}_3\ldots$

Note that $r_n \neq \hat{r}$ for any $n \in \mathrm{N}$ because they differ on the n-th digit.

r_0	$0.\mathbf{5}00000000000\ldots$
r_1	$0.3\mathbf{3}3333333333\ldots$
r_2	$0.14\mathbf{2}857142857\ldots$
r_3	$0.141\mathbf{5}92653589\ldots$
r_4	$0.2000\mathbf{0}0000000\ldots$
$\vdots$	

Proof that $[0, 1)$ is uncountable: diagonalization

Suppose for contradiction that there is a list $\{r_0, r_1, r_2, \ldots\}$ of all real numbers in $[0, 1)$.

Consider the digits $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., the n-th digit of r_n for $n \in \mathbb{N}$.

For each such digit x_i, construct the digit $\hat{x}_i$ as follows:

- If $x_i = 1$ then $\hat{x}_i = 0$.
- If $x_i \neq 1$ then $\hat{x}_i = 1$.

Now, consider the number $\hat{r} = 0.\hat{x}_0\hat{x}_1\hat{x}_2\hat{x}_3\ldots$

Note that $r_n \neq \hat{r}$ for any $n \in \mathbb{N}$ because they differ on the n-th digit.

So the list doesn't include $\hat{r}$, which is a contradiction. Thus the set $[0, 1)$ is uncountable.

r_0	$0.500000000000\ldots$
r_1	$0.333333333333\ldots$
r_2	$0.142857142857\ldots$
r_3	$0.141592653589\ldots$
r_4	$0.200000000000\ldots$
$\vdots$	

The set of all functions $f\colon \mathbf{N} \to \{0, 1\}$ is uncountable

The set of all functions $f: \mathbb{N} \to \{0, 1\}$ is uncountable

Suppose for contradiction that there is a list $\{f_1, f_2, f_3, \ldots\}$ of functions from $\mathbb{N}$ to $\{0, 1\}$.

f_0	0 0 0 0 0 0 0 0 0…
f_1	1 1 1 1 1 1 1 1 1…
f_2	0 1 0 1 0 1 0 1 0…
f_3	0 1 1 1 0 1 1 1 0…
f_4	1 1 0 0 0 1 1 0 0…
$\vdots$	

The set of all functions $f: \mathbb{N} \to \{0, 1\}$ is uncountable

Suppose for contradiction that there is a list $\{f_1, f_2, f_3, \ldots\}$ of functions from $\mathbb{N}$ to $\{0, 1\}$.

Consider the outputs $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., $f_n(n)$ for $n \in \mathbb{N}$.

f_0	0 0 0 0 0 0 0 0 0…
f_1	1 1 1 1 1 1 1 1 1…
f_2	0 1 0 1 0 1 0 1 0…
f_3	0 1 1 1 0 1 1 1 0…
f_4	1 1 0 0 0 1 1 0 0…
$\vdots$	

The set of all functions $f\colon \mathbb{N} \to \{0, 1\}$ is uncountable

Suppose for contradiction that there is a list $\{f_1, f_2, f_3, \ldots\}$ of functions from $\mathbb{N}$ to $\{0, 1\}$.

Consider the outputs $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., $f_n(n)$ for $n \in \mathbb{N}$.

For each such output x_i, construct $\hat{x}_i$ as follows:

- If $x_i = 1$ then $\hat{x}_i = 0$.
- If $x_i \neq 1$ then $\hat{x}_i = 1$.

f_0	**0** 0 0 0 0 0 0 0 0…
f_1	1 **1** 1 1 1 1 1 1 1…
f_2	0 1 **0** 1 0 1 0 1 0…
f_3	0 1 1 **1** 0 1 1 1 0…
f_4	1 1 0 0 **0** 1 1 0 0…
$\vdots$	

The set of all functions $f: \mathbb{N} \to \{0, 1\}$ is uncountable

Suppose for contradiction that there is a list $\{f_1, f_2, f_3, \ldots\}$ of functions from $\mathbb{N}$ to $\{0, 1\}$.

Consider the outputs $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., $f_n(n)$ for $n \in \mathbb{N}$.

For each such output x_i, construct $\hat{x}_i$ as follows:

- If $x_i = 1$ then $\hat{x}_i = 0$.
- If $x_i \neq 1$ then $\hat{x}_i = 1$.

Now, consider the function $\hat{f}(n) = \hat{x}_n$.

f_0	**0** 0 0 0 0 0 0 0 0…
f_1	1 **1** 1 1 1 1 1 1 1…
f_2	0 1 **0** 1 0 1 0 1 0…
f_3	0 1 1 **1** 0 1 1 1 0…
f_4	1 1 0 0 **0** 1 1 0 0…
$\vdots$	

The set of all functions $f: \mathbb{N} \to \{0, 1\}$ is uncountable

Suppose for contradiction that there is a list $\{f_1, f_2, f_3, \ldots\}$ of functions from $\mathbb{N}$ to $\{0, 1\}$.

Consider the outputs $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., $f_n(n)$ for $n \in \mathbb{N}$.

For each such output x_i, construct $\hat{x}_i$ as follows:

- If $x_i = 1$ then $\hat{x}_i = 0$.
- If $x_i \neq 1$ then $\hat{x}_i = 1$.

Now, consider the function $\hat{f}(n) = \hat{x}_n$.

Note that $f_n \neq \hat{f}$ for any $n \in \mathbb{N}$ because the functions differ on the n-th output.

f_0	0 0 0 0 0 0 0 0 0…
f_1	1 1 1 1 1 1 1 1 1…
f_2	0 1 0 1 0 1 0 1 0…
f_3	0 1 1 1 0 1 1 1 0…
f_4	1 1 0 0 0 1 1 0 0…
$\vdots$	

The set of all functions $f: \mathrm{N} \to \{0, 1\}$ is uncountable

Suppose for contradiction that there is a list $\{f_1, f_2, f_3, \ldots\}$ of functions from N to $\{0, 1\}$.

Consider the outputs $x_0, x_1, x_2, x_3, \ldots$ on the diagonal of this list, i.e., $f_n(n)$ for $n \in \mathrm{N}$.

For each such output x_i, construct $\hat{x}_i$ as follows:

- If $x_i = 1$ then $\hat{x}_i = 0$.
- If $x_i \neq 1$ then $\hat{x}_i = 1$.

Now, consider the function $\hat{f}(n) = \hat{x}_n$.

Note that $f_n \neq \hat{f}$ for any $n \in \mathrm{N}$ because the functions differ on the n-th output.

So the list doesn't include $\hat{f}$, which is a contradiction. Thus the set $\{f \mid f: \mathrm{N} \to \{0, 1\}\}$ is uncountable.

f_0	0 0 0 0 0 0 0 0 0...
f_1	1 1 1 1 1 1 1 1 1...
f_2	0 1 0 1 0 1 0 1 0...
f_3	0 1 1 1 0 1 1 1 0...
f_4	1 1 0 0 0 1 1 0 0...
$\vdots$	

Uncomputability

Are there problems computers can’t solve?

Uncomputable functions

We have seen that …

The set of all (Java) programs is countable.

The set of all functions $f\colon \mathrm{N} \to \{0, 1\}$ is uncountable.

Uncomputable functions

We have seen that …

The set of all (Java) programs is countable.

The set of all functions $f\colon \mathrm{N} \to \{0, 1\}$ is uncountable.

So there must be some function $f\colon \mathrm{N} \to \{0, 1\}$ that is not computable by any program! We'll study one such function next time.

Summary

Cardinality and countability.

Two sets have the same cardinality if there is a bijection between them.

A set is countable iff it has the same cardinality as some subset of $\mathbb{N}$.

Use dovetailing to show that a set is countable and diagonalization to show that it's uncountable.

Computability.

Countability of programs and uncountability of functions $f: \mathbb{N} \to \{0, 1\}$ tells us that there is some function that can't be computed by any program!