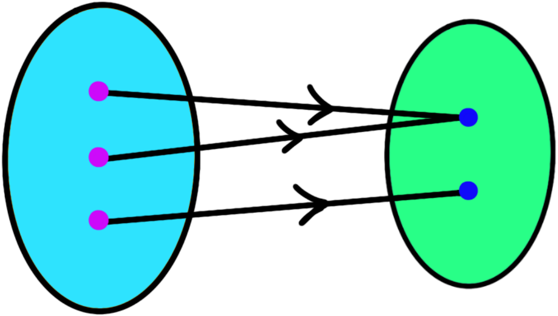
Surjections

This lecture is about another true-false statement which you can make about a function – you can ask if it is *surjective*.

Let's start with a picture.



A function $f : X \to Y$ is *surjective* if for every element b of Y, you can find an element $a \in X$ such that $f(a) = b$.

Note that a is allowed to depend on b.

Non-example: $f(x) = x^2$ is a function from $\mathbb{R}$ to $\mathbb{R}$, but it is *not* surjective, because there is no real number whose square is -1. In other words, $x^2 = -1$ has no solution for $x \in \mathbb{R}$.

The formal definition: a function f is surjective if $\forall b \in Y, \exists a \in X, f(a) = b$.

This last definition is the preferred definition in a test or exam.

Maybe some of you learnt the phrase “onto” to describe such functions.

At university we will call them “surjective” functions.

Here is yet another way of thinking about this idea.

Say $f : X \to Y$ is a function, and say $y \in Y$.

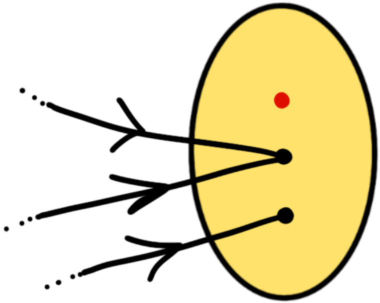
Recall that the *pre-image* of y is the subset of X consisting of all the elements of X which are sent to y by f.

In other words, the preimage of y is $\{x \in X \mid f(x) = y\}$.

Example: the preimage of -1 under the squaring function from $\mathbb{R}$ to $\mathbb{R}$ is the set $\emptyset$.

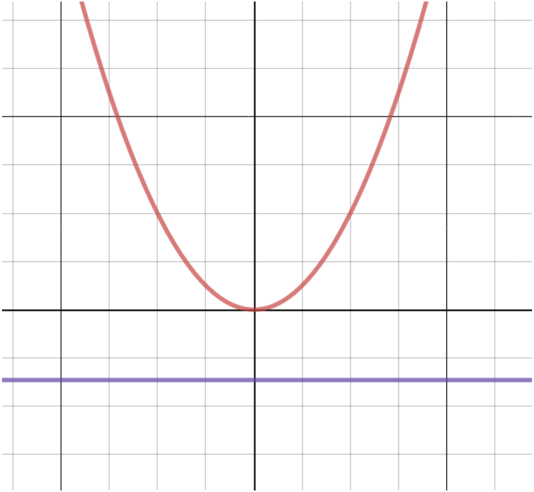
Another way of picturing surjectivity: the function f is surjective if and only if the preimage of any element of y has *size at least 1* (i.e., is nonempty)

Sole obstruction to surjectivity



The preimage of the red point is empty, so that function is not surjective.

Exercise: Work out what the horizontal line test for *surjectivity* says.



Pause the video, and get inspired by the horizontal line test for injectivity. Here is a non-surjective function.

The horizontal line test for surjectivity: A function is surjective if and only if its graph meets every horizontal line in *at least one point*.

Let's now prove some theorems about surjectivity.

Because we are proving theorems, we will use the recommended definition:

$f : X \to Y$ is surjective if and only if

$$\forall b \in Y, \exists a \in X, f(a) = b.$$

Is the identity function surjective?

Theorem

Let X be a set. Then the identity function $id_X : X \to X$ is surjective.

Why don't you pause the video and have a go at proving this yourself? Then I will show you my proof.

Proof.

We are asked to prove that $\forall b \in X, \exists a \in X, id_X(a) = b$.

So, let b be an arbitrary element of X.

We have to find $a \in X$ such that $id_X(a) = b$.

How about we use $a = b$? After all, a is allowed to depend on b.

This works, because our goal is now to prove $id_X(b) = b$ i.e. $b = b$, which is obvious. $\square$

Theorem

Let X, Y and Z be sets, and let $f : X \to Y$ and $g : Y \to Z$ be functions.
If f and g are both surjective, then $g \circ f$ is also surjective.

The proof of this is in the course notes. Let me show you what the proof looks like in Lean, and if you don't care about Lean then you can just read the course notes.

File Edit Selection View Go Run Terminal Help

injective_comp.lean | surjective_comp.lean | sheet2.lean | sheet3.le

src > 2020 > functions > surjective_comp.lean

```lean

open function

variables (X Y Z : Type) (f : X → Y) (g : Y → Z)

example : (surjective f) ∧ (surjective g) → surjective (g ∘ f) :=
begin
    -- Say f and g are surjective.

    -- Choose  z in Z.

    -- Our job is to find x in X such that g(f(x))=z.

    -- Well g, is assumed surjective so we can find y in Y
    -- such that g(y)=z

    -- And f is assumed surjective so we can find x in X
    -- such that f(x)=y

  -- let's use this x.

    -- and now we see g(f(x))=g(y)=z so we're done
end
```

Lean Infoview

▼ surjective_comp.lean:11:0

▼ Tactic state

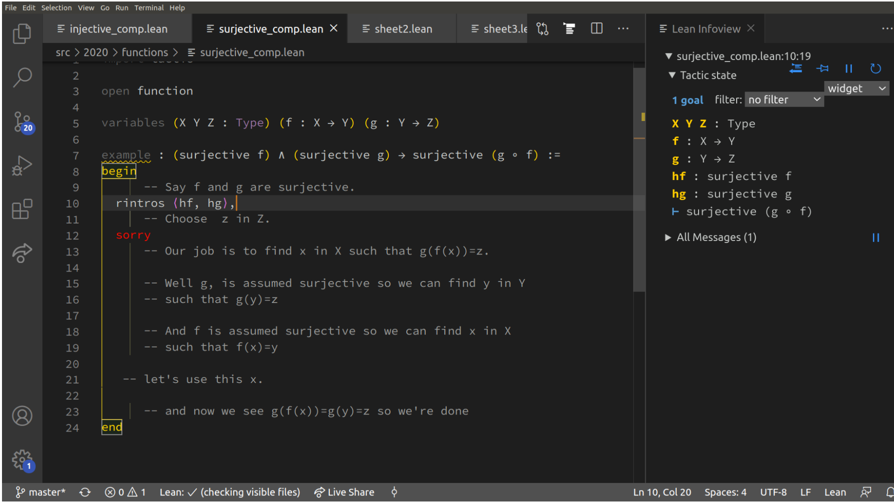
widget

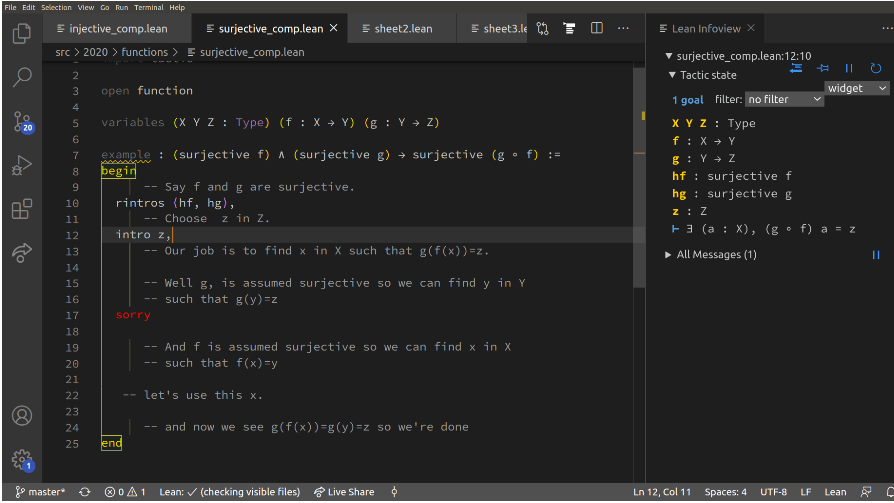
1 goal filter: no filter

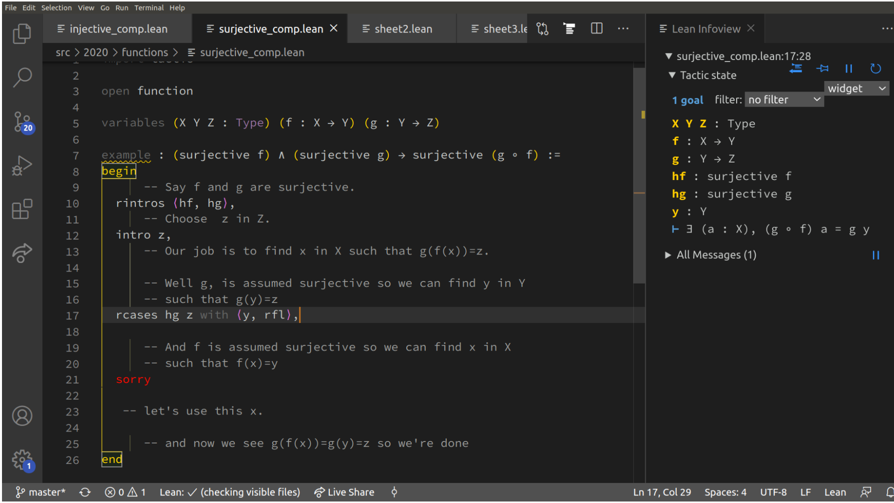
```
X Y Z : Type
f : X → Y
g : Y → Z
⊢
surjective f ∧ surjective g →
  surjective (g ∘ f)
```

▶ All Messages (1)

master* ⊗ 1 ⚠ 0 Lean: ✓ (checking visible files) Live Share Ln 11, Col 1 Spaces: 4 UTF-8 LF Lean


File Edit Selection View Go Run Terminal Help
injective_comp.lean
surjective_comp.lean
sheet2.lean
sheet3.le
src > 2020 > functions > surjective_comp.lean
open function
variables (X Y Z : Type) (f : X → Y) (g : Y → Z)
example : (surjective f) ∧ (surjective g) → surjective (g ∘ f) :=
begin
-- Say f and g are surjective.
rintros (hf, hg),
-- Choose z in Z.
sorry
-- Our job is to find x in X such that g(f(x))=z.
-- Well g, is assumed surjective so we can find y in Y
-- such that g(y)=z
-- And f is assumed surjective so we can find x in X
-- such that f(x)=y
-- let's use this x.
-- and now we see g(f(x))=g(y)=z so we're done
end
Lean Infoview
surjective_comp.lean:10:19
Tactic state
widget
1 goal filter: no filter
X Y Z : Type
f : X → Y
g : Y → Z
hf : surjective f
hg : surjective g
⊢ surjective (g ∘ f)
All Messages (1)
master*
0 1
Lean: ✓ (checking visible files)
Live Share
Ln 10, Col 20 Spaces: 4 UTF-8 LF Lean


File Edit Selection View Go Run Terminal Help
injective_comp.lean
surjective_comp.lean
sheet2.lean
sheet3.le
src > 2020 > functions > surjective_comp.lean
open function
variables (X Y Z : Type) (f : X → Y) (g : Y → Z)
example : (surjective f) ∧ (surjective g) → surjective (g ∘ f) :=
begin
-- Say f and g are surjective.
rintros (hf, hg),
-- Choose z in Z.
intro z,
-- Our job is to find x in X such that g(f(x))=z.
-- Well g, is assumed surjective so we can find y in Y
-- such that g(y)=z
sorry
-- And f is assumed surjective so we can find x in X
-- such that f(x)=y
-- let's use this x.
-- and now we see g(f(x))=g(y)=z so we're done
end
Lean Infoview
surjective_comp.lean:12:10
Tactic state
widget
1 goal filter: no filter
X Y Z : Type
f : X → Y
g : Y → Z
hf : surjective f
hg : surjective g
z : Z
⊢ ∃ (a : X), (g ∘ f) a = z
All Messages (1)
master* ⊗ 0 ⚠ 1 Lean: ✓ (checking visible files) Live Share
Ln 12, Col 11 Spaces: 4 UTF-8 LF Lean



File Edit Selection View Go Run Terminal Help
injective_comp.lean
surjective_comp.lean
sheet2.lean
sheet3.le
src > 2020 > functions > surjective_comp.lean
open function
variables (X Y Z : Type) (f : X → Y) (g : Y → Z)
example : (surjective f) ∧ (surjective g) → surjective (g ∘ f) :=
begin
-- Say f and g are surjective.
rintros ⟨hf, hg⟩,
-- Choose z in Z.
intro z,
-- Our job is to find x in X such that g(f(x))=z.
-- Well g, is assumed surjective so we can find y in Y
-- such that g(y)=z
rcases hg z with ⟨y, rfl⟩,
-- And f is assumed surjective so we can find x in X
-- such that f(x)=y
sorry
-- let's use this x.
-- and now we see g(f(x))=g(y)=z so we're done
end
Lean Infoview
surjective_comp.lean:17:28
Tactic state
widget
1 goal filter: no filter
X Y Z : Type
f : X → Y
g : Y → Z
hf : surjective f
hg : surjective g
y : Y
⊢ ∃ (a : X), (g ∘ f) a = g y
All Messages (1)
master* 0 1 Lean: ✓ (checking visible files) Live Share
Ln 17, Col 29 Spaces: 4 UTF-8 LF Lean

File Edit Selection View Go Run Terminal Help

injective_comp.lean | surjective_comp.lean | sheet2.lean | sheet3.le

src > 2020 > functions > surjective_comp.lean

```lean
import tactic

open function

variables (X Y Z : Type) (f : X → Y) (g : Y → Z)

example : (surjective f) ∧ (surjective g) → surjective (g ∘ f) :=
begin
    -- Say f and g are surjective.
  rintros ⟨hf, hg⟩,
    -- Choose  z in Z.
  intro z,
    -- Our job is to find x in X such that g(f(x))=z.

    -- Well g, is assumed surjective so we can find y in Y
    -- such that g(y)=z
  rcases hg z with ⟨y, rfl⟩,

    -- And f is assumed surjective so we can find x in X
    -- such that f(x)=y
  rcases hf y with ⟨x, rfl⟩,

   -- let's use this x.
  sorry
    -- and now we see g(f(x))=g(y)=z so we're done
end
```

Lean Infoview

▼ surjective_comp.lean:21:28

▼ Tactic state

widget

1 goal filter: no filter

```
X Y Z : Type
f : X → Y
g : Y → Z
hf : surjective f
hg : surjective g
x : X
⊢ ∃ (a : X), (g ∘ f) a = g (f x)
```

▶ All Messages (1)

master* ⊗ 0 ⚠ 1 Lean: ✓ (checking visible files) Live Share Ln 21, Col 29 Spaces: 4 UTF-8 LF Lean

```
import tactic

open function

variables (X Y Z : Type) (f : X → Y) (g : Y → Z)

example : (surjective f) ∧ (surjective g) → surjective (g ∘ f) :=
begin
    -- Say f and g are surjective.
  rintros (hf, hg),
    -- Choose  z in Z.
  intro z,
    -- Our job is to find x in X such that g(f(x))=z.

    -- Well g, is assumed surjective so we can find y in Y
    -- such that g(y)=z
  rcases hg z with (y, rfl),

    -- And f is assumed surjective so we can find x in X
    -- such that f(x)=y
  rcases hf y with (x, rfl),

   -- let's use this x.
  use x,
    -- and now we see g(f(x))=g(y)=z so we're done
end
```

surjective_comp.lean:24:8

Tactic state

goals accomplished 🎉

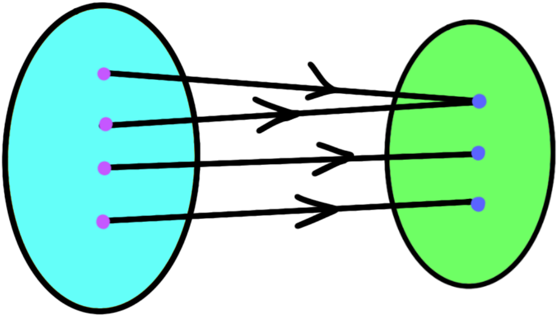
All Messages (0)

Counting

Let's finish the lecture with some more counting.

Say X and Y are finite sets, and $f : X \to Y$ is an surjective function.

Because f is surjective, every element of Y is "hit" by an element of X. So the size of Y must be at most the size of X.



Counting question 1) How many surjections are there from a set of size 2 to a set of size 5?
Counting question 2) How many surjections are there from a set of size 5 to a set of size 2?

Pause the video and have a go at these questions.

Counting question 1) How many surjections are there from a set of size 2 to a set of size 5?

There are none! For if a surjection existed, we would be able to deduce $2 \geq 5$, which is false.

Counting question 2) How many surjections are there from a set of size 5 to a set of size 2?

This is tricky. We have learnt previously that there are $2^5 = 32$ functions from a set of size 5 to a set of size 2. How many of these are *not* surjective?

Say X has size 5 and $Y = \{s, t\}$ has size 2. Recall that the *range* $f(X)$ of a map $f : X \to Y$ is the elements of Y which are "hit" by f (some people call this the *image*). What are the possibilities for the range?

The range is a subset of Y, and it can't be empty (because X isn't empty). So it's either $\{s\}$ or $\{t\}$ or $\{s, t\}$. Now if $f : X \to Y$ has range $\{s\}$ then this means $f(x) = s$ for all $x \in X$, and there is just one such function.

Similarly there is one function with range $\{t\}$, so the remaining $32 - 1 - 1 = 30$ functions must have range $\{s, t\}$ and those are the surjective ones.

A trickier question: how many surjections are there from a set of size m to a set of size p?

Unlike injections, there is no closed formula. The cleanest answer we know relates this to
Stirling numbers of the second kind (Wikipedia)