

# Category-theoretic properties

"Any two isomorphic objects in a category have the same category-theoretic properties."

instead of formalizing the "language & logic of category theory", we'll just look at examples of category-theoretic properties.

Here's our first one...

# Terminal objects

An object  $T \in \mathcal{C}$  of a category  $\mathcal{C}$  is **terminal** if for all  $X \in \mathcal{C}$ , there is a unique morphism  $X \rightarrow T$  (we'll write  $\langle \rangle_x$ , or just  $\langle \rangle$  for this morphism)

Theorem In a category  $\mathcal{C}$ :

- (a) if  $T$  is terminal &  $T \cong T'$ , then  $T'$  is terminal
- (b) if  $T$  &  $T'$  are both terminal, then  $T \cong T'$  (and there is only one isomorphism between  $T$  &  $T'$ )

terminal objects are unique up to unique isomorphism

# Terminal objects

An object  $T \in \mathcal{C}$  of a category  $\mathcal{C}$  is **terminal** if for all  $X \in \mathcal{C}$ , there is a unique morphism  $X \rightarrow T$   
(we'll write  $\langle \rangle_x$ , or just  $\langle \rangle$  for this morphism)

Theorem In a category  $\mathcal{C}$ :

- (a) if  $T$  is terminal &  $T \cong T'$ , then  $T'$  is terminal
- (b) if  $T$  &  $T'$  are both terminal, then  $T \cong T'$   
(and ...)

Proof ...

# Examples of terminal objects

- In  $\text{Set}$  : any one-element set
- Any one-element set has a unique pre-order & this makes it terminal in  $\text{Pre}$
- Ditto for  $\text{Mon}$ .
- A pre-ordered set  $(P, \leq)$ , regarded as a category, has a terminal object iff it has a **greatest element**:  $(\forall x \in P) x \leq T$
- When does a monoid  $(M, \cdot, 1)$ , regarded as a category, have a terminal object?

# The opposite of a category $\mathcal{C}$

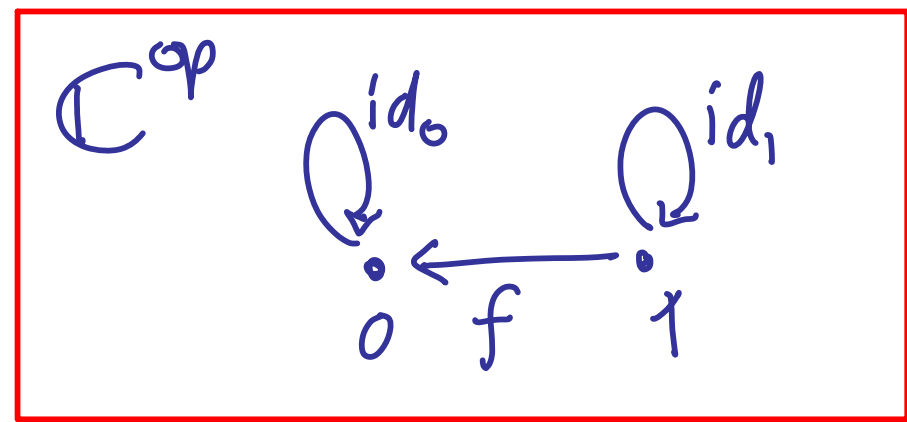
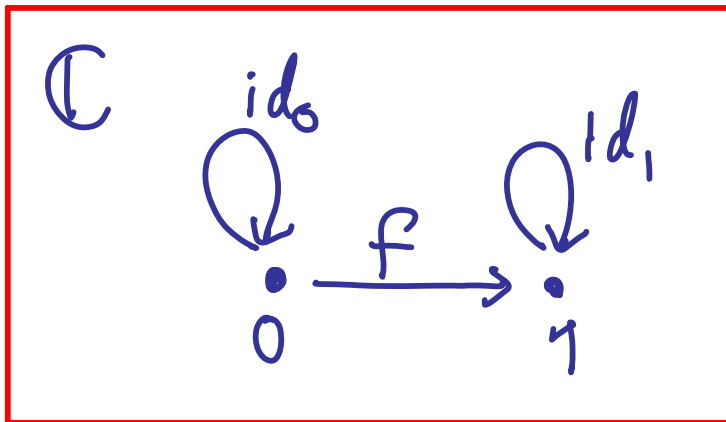
is the category  $\mathcal{C}^{\text{op}}$  defined by

- $\text{Obj } \mathcal{C}^{\text{op}} \triangleq \text{Obj } \mathcal{C}$
- $\mathcal{C}^{\text{op}}(X, Y) \triangleq \mathcal{C}(Y, X)$  for all objects  $X, Y$

same objects

same morphisms, but with direction reversed, that is, dom & cod swapped

E.g.



# The opposite of a category $\mathbb{C}$

is the category  $\mathbb{C}^{\text{op}}$  defined by

- $\text{Obj } \mathbb{C}^{\text{op}} \triangleq \text{Obj } \mathbb{C}$
- $\mathbb{C}^{\text{op}}(X, Y) \triangleq \mathbb{C}(Y, X)$  for all objects  $X, Y$
- identity morphism on  $X \in \text{Obj } \mathbb{C}^{\text{op}}$  is  $\text{id}_X$ , the identity on  $X \in \text{Obj } \mathbb{C}$
- the composition of  $f \in \mathbb{C}^{\text{op}}(X, Y)$  &  $g \in \mathbb{C}^{\text{op}}(Y, Z)$  is given by composition  $f \circ_{\mathbb{C}^{\text{op}}} g \triangleq f \circ_{\mathbb{C}} g$

(associativity & unity props hold, because they do in  $\mathbb{C}$ )

# Principle of Duality

Whenever we { define a concept in terms of  
{ prove a theorem  
Commutative diagrams, we obtain another  
{ concept, called its **dual**, by reversing  
{ theorem, by reversing  
the direction of morphisms throughout (i.e.  
by replacing  $\mathbb{C}$  by  $\mathbb{C}^{\text{op}}$ ).

For example...

# Initial object

is the dual notion to "terminal object"

An object  $I \in \mathcal{C}$  of a category  $\mathcal{C}$  is **initial** if for all  $X \in \mathcal{C}$ , there is a unique morphism  $I \rightarrow X$  (we'll write  $[\ ]_X$ , or just  $[\ ]$  for this morphism)

By duality, we have that initial objects are unique up to iso and that any object isomorphic to an initial object is itself initial.

NB "isomorphism" is a self-dual concept



# Examples of initial objects

- The empty set is initial in **Set**
- Any one-element monoid (has uniquely determined monoid operation & unit) is initial in **Mon** (why?)  
→ (so initial & terminal objects coincide in **Mon**)

an object that's both initial & terminal is sometimes called a zero object

# Example: free monoids as initial objects

(relevant to automata & formal languages)

Free monoid on a set  $\Sigma \in \text{Set}$ :

$(\text{List}(\Sigma), @, \text{nil})$

Set of finite lists  
of elements of  $\Sigma$

empty  
list

list  
concatenation:  
 $\text{nil} @ l' = l'$   
 $(a :: l) @ l' =$   
 $a :: (l @ l')$

# Example: free monoids as initial objects

Free monoid on a set  $\Sigma \in \text{Set}$ :

$i_\Sigma : \Sigma \rightarrow \text{List}(\Sigma)$  in  $\text{Set}$

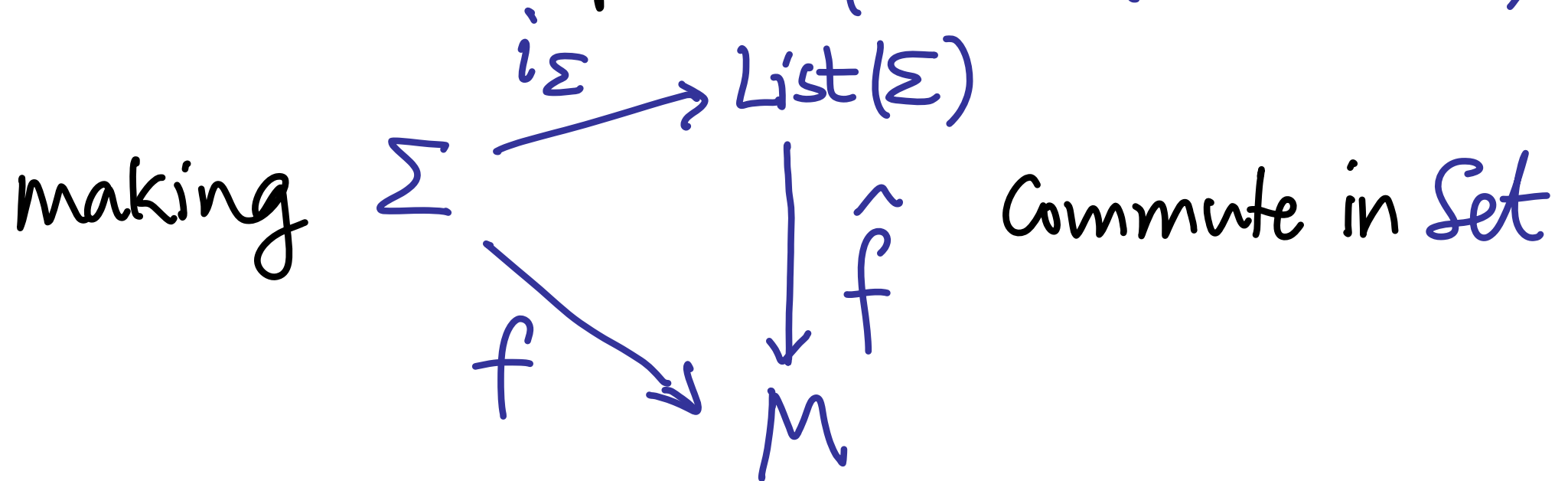
$a \mapsto [a]$  where  $[a] \triangleq a :: \text{nil}$

$i_\Sigma$  sends element  $a \in \Sigma$  to corr. list of length 1

It has the following "universal property"...

# Example: free monoids as initial objects

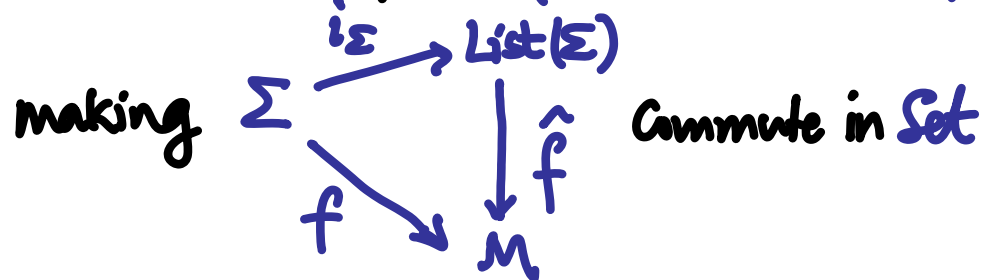
Theorem Given  $\Sigma \in \text{Set}$ ,  $(M, \cdot, e) \in \text{Mon}$  and  $f \in \text{Set}(\Sigma, M)$ , there is a unique monoid homomorphism  $\hat{f} \in \text{Mon}(\text{List}(\Sigma), M)$



Proof ...

# Example: free monoids as initial objects

Theorem Given  $\Sigma \in \text{Set}$ ,  $(M, \cdot, e) \in \text{Mon}$  and  $f \in \text{Set}(\Sigma, M)$ , there is a unique monoid homomorphism  $\hat{f} \in \text{Mon}(\text{List}(\Sigma), M)$



The theorem just says that  $i_\Sigma: \Sigma \rightarrow \text{List}(\Sigma)$  is an initial object in the following category:

Category  $\Sigma/\text{Mon}$ :

- objects  $(M, f)$  where  $M \in \text{Mon} \& f \in \text{Set}(\Sigma, M)$

- morphisms in  $\Sigma/\text{Mon}((M, f), (N, g))$

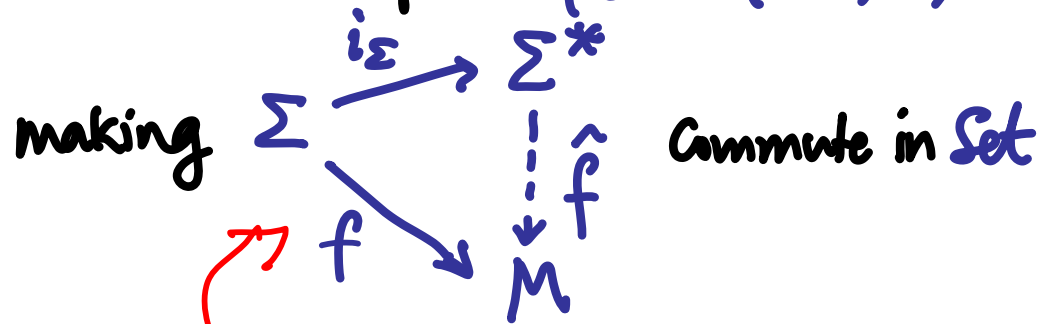
are  $h \in \text{Mon}(M, N)$  s.t.

$$\begin{array}{ccc} \Sigma & \xrightarrow{f} & M \\ & \searrow g & \downarrow h \\ & & N \end{array} \quad \text{Commutates in Set}$$

- identities & composition as in  $\text{Mon}$

# Example: free monoids as initial objects

Theorem Given  $\Sigma \in \text{Set}$ ,  $(M, \cdot, 1) \in \text{Mon}$  and  $f \in \text{Set}(\Sigma, M)$ , there is a unique monoid homomorphism  $\hat{f} \in \text{Mon}(\Sigma^*, M)$



The theorem just says that  $i_\Sigma: \Sigma \rightarrow \Sigma^*$  is an initial object in  $\Sigma/\text{Mon}$ .

So this universal property determines  $\text{List}(\Sigma)$  uniquely up to monoid isomorphism.

We'll see later that  $\Sigma \mapsto \text{List}(\Sigma)$  is part of a functor (= category morphism) which is left adjoint to the "forgetful functor"  $\text{Mon} \rightarrow \text{Set}$ .