

Category theory for computer science

- *generality*
- *abstraction*
- *convenience*
- *constructiveness*
-

Overall idea

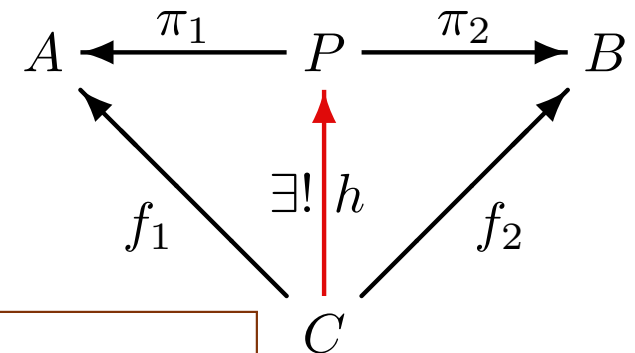
look at all objects exclusively through relationships between them

capture relationships between objects as appropriate morphisms between them

(Cartesian) product

- *Cartesian product* of two sets A and B , is the set $A \times B = \{\langle a, b \rangle \mid a \in A, b \in B\}$ with projections $\pi_1 : A \times B \rightarrow A$ and $\pi_2 : A \times B \rightarrow B$ given by $\pi_1(\langle a, b \rangle) = a$ and $\pi_2(\langle a, b \rangle) = b$.
- A *product* of two sets A and B , is any set P with projections $\pi_1 : P \rightarrow A$ and $\pi_2 : P \rightarrow B$ such that for any set C with functions $f_1 : C \rightarrow A$ and $f_2 : C \rightarrow B$ there exists a unique function $h : C \rightarrow P$ such that $h;\pi_1 = f_1$ and $h;\pi_2 = f_2$.

Fact: Cartesian product (of sets A and B) is a product (of A and B).



Recall the definition of (Cartesian) product of Σ -algebras.
Define product of Σ -algebras as above. *What have you changed?*

Pitfalls of generalization

the same concrete definition \rightsquigarrow distinct abstract generalizations

Given a function $f : A \rightarrow B$, the following conditions are equivalent:

- f is a *surjection*: $\forall b \in B. \exists a \in A. f(a) = b$.
- f is an *epimorphism*: for all $h_1, h_2 : B \rightarrow C$, if $f;h_1 = f;h_2$ then $h_1 = h_2$.
- f is a *retraction*: there exists $g : B \rightarrow A$ such that $g;f = id_B$.

BUT: Given a Σ -homomorphism $f : A \rightarrow B$ for $A, B \in \mathbf{Alg}(\Sigma)$:

$f \text{ is retraction} \implies f \text{ is surjection} \iff f \text{ is epimorphism}$

BUT: Given a (weak) Σ -homomorphism $f : A \rightarrow B$ for $A, B \in \mathbf{PAlg}(\Sigma)$:

$f \text{ is retraction} \implies f \text{ is surjection} \implies f \text{ is epimorphism}$

Categories

Definition: *Category* \mathbf{K} consists of:

- a collection of *objects*: $|\mathbf{K}|$
- mutually disjoint collections of *morphisms*: $\mathbf{K}(A, B)$, for all $A, B \in |\mathbf{K}|$;
 $m: A \rightarrow B$ stands for $m \in \mathbf{K}(A, B)$
- *morphism composition*: for $m: A \rightarrow B$ and $m': B \rightarrow C$, we have $m; m': A \rightarrow C$;
 - the composition is associative: for $m_1: A_0 \rightarrow A_1$, $m_2: A_1 \rightarrow A_2$ and $m_3: A_2 \rightarrow A_3$, $(m_1; m_2); m_3 = m_1; (m_2; m_3)$
 - the composition has identities: for $A \in |\mathbf{K}|$, there is $id_A: A \rightarrow A$ such that for all $m_1: A_1 \rightarrow A$, $m_1; id_A = m_1$, and $m_2: A \rightarrow A_2$, $id_A; m_2 = m_2$.

BTW: “collection” means “set”, “class”, etc, as appropriate.

\mathbf{K} is *locally small* if for all $A, B \in |\mathbf{K}|$, $\mathbf{K}(A, B)$ is a set.
 \mathbf{K} is *small* if in addition $|\mathbf{K}|$ is a set.

Presenting finite categories

0:

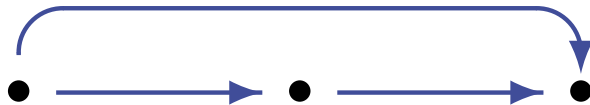
1:



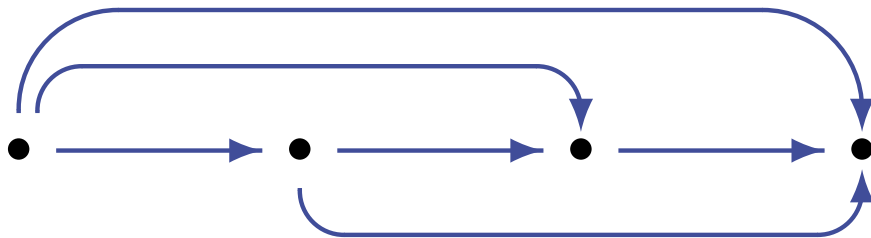
2:



3:



4:



...

(identities omitted)

Generic examples

Discrete categories: A category \mathbf{K} is *discrete* if all $\mathbf{K}(A, B)$ are empty, for distinct $A, B \in |\mathbf{K}|$, and $\mathbf{K}(A, A) = \{id_A\}$ for all $A \in |\mathbf{K}|$.

Preorders: A category \mathbf{K} is *thin* if for all $A, B \in |\mathbf{K}|$, $\mathbf{K}(A, B)$ contains at most one element.

Every preorder $\leq \subseteq X \times X$ determines a thin category \mathbf{K}_{\leq} with $|\mathbf{K}_{\leq}| = X$ and for $x, y \in |\mathbf{K}_{\leq}|$, $\mathbf{K}_{\leq}(x, y)$ is nonempty iff $x \leq y$.

Every (small) category \mathbf{K} determines a preorder $\leq_{\mathbf{K}} \subseteq |\mathbf{K}| \times |\mathbf{K}|$, where for $A, B \in |\mathbf{K}|$, $A \leq_{\mathbf{K}} B$ iff $\mathbf{K}(A, B)$ is nonempty.

Monoids: A category \mathbf{K} is a *monoid* if $|\mathbf{K}|$ is a singleton.

Every monoid $\mathcal{X} = \langle X, ;, id \rangle$, where $_-;_- : X \times X \rightarrow X$ and $id \in X$, determines a (monoid) category $\mathbf{K}_{\mathcal{X}}$ with $|\mathbf{K}_{\leq}| = \{*\}$, $\mathbf{K}(*, *) = X$ and the composition given by the monoid operation.

Examples

- Sets (as objects) and functions between them (as morphisms) with the usual composition form the category **Set**.

Functions have to be considered with their sources and targets

- For any set S , S -sorted sets (as objects) and S -functions between them (as morphisms) with the usual composition form the category **Set** ^{S} .
- For any signature Σ , Σ -algebras (as objects) and their homomorphisms (as morphisms) form the category **Alg**(Σ).
- For any signature Σ , partial Σ -algebras (as objects) and their weak homomorphisms (as morphisms) form the category **PAlg**(Σ).
- For any signature Σ , partial Σ -algebras (as objects) and their strong homomorphisms (as morphisms) form the category **PAlg**_s(Σ).
- Algebraic signatures (as objects) and their morphisms (as morphisms) with the composition defined in the obvious way form the category **AlgSig**.

Substitutions

For any signature $\Sigma = (S, \Omega)$, the category of Σ -substitutions \mathbf{Subst}_Σ is defined as follows:

- objects of \mathbf{Subst}_Σ are S -sorted sets (of variables);
- morphisms in $\mathbf{Subst}_\Sigma(X, Y)$ are substitutions $\theta : X \rightarrow |T_\Sigma(Y)|$,
- composition is defined in the obvious way:
for $\theta_1 : X \rightarrow Y$ and $\theta_2 : Y \rightarrow Z$, that is functions $\theta_1 : X \rightarrow |T_\Sigma(Y)|$ and $\theta_2 : Y \rightarrow |T_\Sigma(Z)|$, their composition $\theta_1; \theta_2 : X \rightarrow Z$ in \mathbf{Subst}_Σ is the function $\theta_1; \theta_2 : X \rightarrow |T_\Sigma(Z)|$ such that for each $x \in X$, $(\theta_1; \theta_2)(x) = \theta_2^\#(\theta_1(x))$.

Subcategories

Given a category \mathbf{K} , a *subcategory* of \mathbf{K} is any category \mathbf{K}' such that

- $|\mathbf{K}'| \subseteq |\mathbf{K}|$,
- $\mathbf{K}'(A, B) \subseteq \mathbf{K}(A, B)$, for all $A, B \in |\mathbf{K}'|$,
- composition in \mathbf{K}' coincides with the composition in \mathbf{K} on morphisms in \mathbf{K}' , and
- identities in \mathbf{K}' coincide with identities in \mathbf{K} on objects in $|\mathbf{K}'|$.

A subcategory \mathbf{K}' of \mathbf{K} is *full* if $\mathbf{K}'(A, B) = \mathbf{K}(A, B)$ for all $A, B \in |\mathbf{K}'|$.

Any collection $X \subseteq |\mathbf{K}|$ gives the full subcategory $\mathbf{K}|_X$ of \mathbf{K} by $|\mathbf{K}|_X = X$.

- The category **FinSet** of finite sets is a full subcategory of **Set**.
- The discrete category of sets is a subcategory of sets with inclusions as morphisms, which is a subcategory of sets with injective functions as morphisms, which is a subcategory of **Set**.
- The category of single-sorted signatures is a full subcategory of **AlgSig**.

Reversing arrows

Given a category \mathbf{K} , its *opposite category* \mathbf{K}^{op} is defined as follows:

- objects: $|\mathbf{K}^{op}| = |\mathbf{K}|$
- morphisms: $\mathbf{K}^{op}(A, B) = \mathbf{K}(B, A)$ for all $A, B \in |\mathbf{K}^{op}| = |\mathbf{K}|$
- composition: given $m_1 : A \rightarrow B$ and $m_2 : B \rightarrow C$ in \mathbf{K}^{op} , that is, $m_1 : B \rightarrow A$ and $m_2 : C \rightarrow B$ in \mathbf{K} , their composition in \mathbf{K}^{op} , $m_1; m_2 : A \rightarrow C$, is set to be their composition $m_2; m_1 : C \rightarrow A$ in \mathbf{K} .

Fact: *The identities in \mathbf{K}^{op} coincide with the identities in \mathbf{K} .*

Fact: *Every category is opposite to some category:*

$$(\mathbf{K}^{op})^{op} = \mathbf{K}$$

Duality principle

If W is a categorical concept (notion, property, statement, ...) then its *dual*, $co\text{-}W$, is obtained by reversing all the morphisms in W .

Example:

$P(X)$: “for any object Y there exists a morphism $f : X \rightarrow Y$ ”

$co\text{-}P(X)$: “for any object Y there exists a morphism $f : Y \rightarrow X$ ”

NOTE: $co\text{-}P(X)$ in \mathbf{K} coincides with $P(X)$ in \mathbf{K}^{op} .

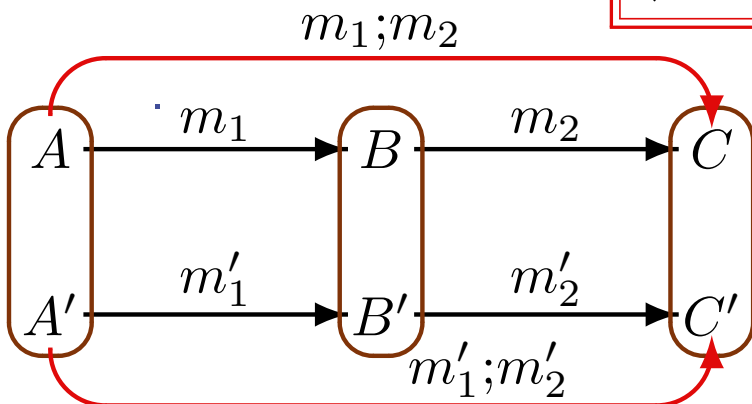
Fact: *If a property W holds for all categories then $co\text{-}W$ holds for all categories as well.*

Product categories

Given categories \mathbf{K} and \mathbf{K}' , their *product* $\mathbf{K} \times \mathbf{K}'$ is the category defined as follows:

- objects: $|\mathbf{K} \times \mathbf{K}'| = |\mathbf{K}| \times |\mathbf{K}'|$
- morphisms: $(\mathbf{K} \times \mathbf{K}')(\langle A, A' \rangle, \langle B, B' \rangle) = \mathbf{K}(A, B) \times \mathbf{K}'(A', B')$ for all $A, B \in |\mathbf{K}|$ and $A', B' \in |\mathbf{K}'|$
- composition: for $\langle m_1, m'_1 \rangle : \langle A, A' \rangle \rightarrow \langle B, B' \rangle$ and $\langle m_2, m'_2 \rangle : \langle B, B' \rangle \rightarrow \langle C, C' \rangle$ in $\mathbf{K} \times \mathbf{K}'$, their composition in $\mathbf{K} \times \mathbf{K}'$ is

$$\langle m_1, m'_1 \rangle ; \langle m_2, m'_2 \rangle = \langle m_1 ; m_2, m'_1 ; m'_2 \rangle$$



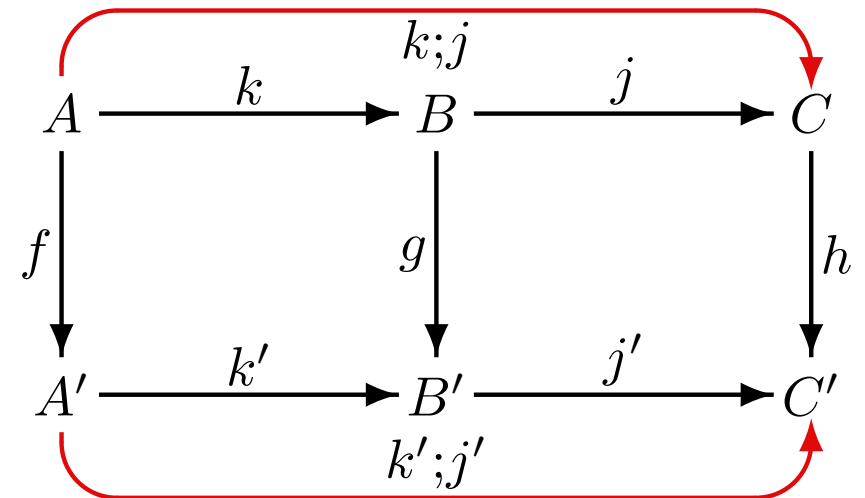
Define \mathbf{K}^n , where \mathbf{K} is a category and $n \geq 1$.
Extend this definition to $n = 0$.

Morphism categories

Given a category \mathbf{K} , its *morphism category* \mathbf{K}^{\rightarrow} is the category defined as follows:

- objects: $|\mathbf{K}^{\rightarrow}|$ is the collection of all morphisms in \mathbf{K}
- morphisms: for $f : A \rightarrow A'$ and $g : B \rightarrow B'$ in \mathbf{K} , $\mathbf{K}^{\rightarrow}(f, g)$ consists of all $\langle k, k' \rangle$, where $k : A \rightarrow B$ and $k' : A' \rightarrow B'$ are such that $k;g = f;k'$ in \mathbf{K}
- composition: for $\langle k, k' \rangle : (f : A \rightarrow A') \rightarrow (g : B \rightarrow B')$ and $\langle j, j' \rangle : (g : B \rightarrow B') \rightarrow (h : C \rightarrow C')$ in \mathbf{K}^{\rightarrow} , their composition in \mathbf{K}^{\rightarrow} is $\langle k, k' \rangle; \langle j, j' \rangle = \langle k;j, k';j' \rangle$.

Check that the composition is well-defined.



Slice categories

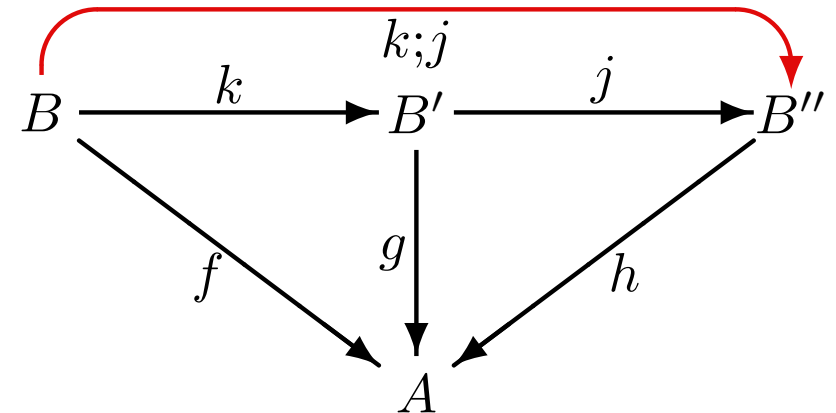
Given a category \mathbf{K} and an object $A \in |K|$, the category of \mathbf{K} -objects over A , $\mathbf{K}\downarrow A$, is the category defined as follows:

- objects: $\mathbf{K}\downarrow A$ is the collection of all morphisms into A in \mathbf{K}
- morphisms: for $f : B \rightarrow A$ and $g : B' \rightarrow A$ in \mathbf{K} , $(\mathbf{K}\downarrow A)(f, g)$ consists of all morphisms $k : B \rightarrow B'$ such that $k;g = f$ in \mathbf{K}
- composition: the composition in $\mathbf{K}\downarrow A$ is the same as in \mathbf{K}

Check that the composition is well-defined.

View $\mathbf{K}\downarrow A$ as a subcategory of \mathbf{K}^{\rightarrow} .

Define $\mathbf{K}\uparrow A$, the category of \mathbf{K} -objects under A .



Fix a category \mathbf{K} for a while.

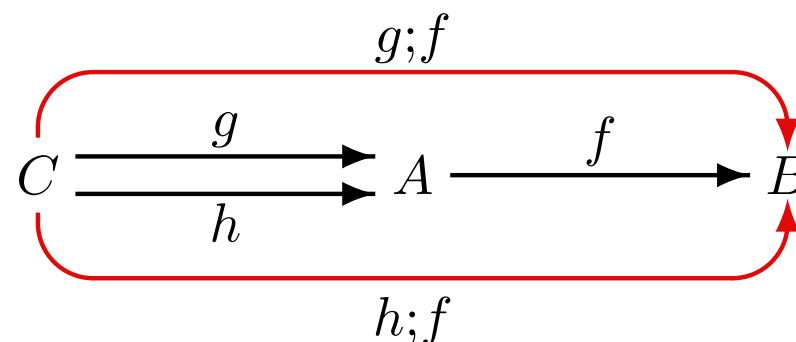
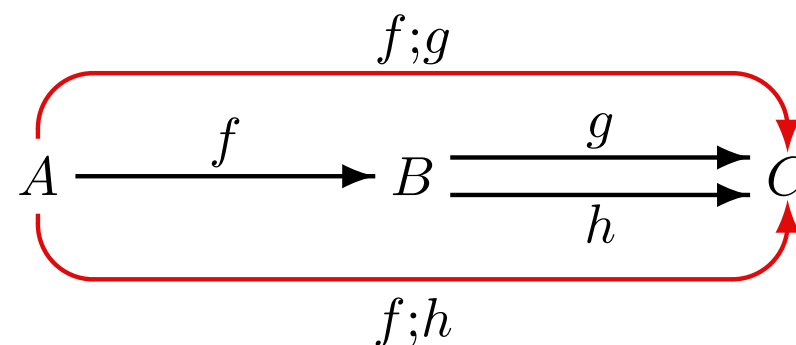
Simple categorical definitions

- $f : A \rightarrow B$ is an *epimorphism* (is *epi*):
for all $g, h : B \rightarrow C$, $f;g = f;h$ implies $g = h$

In Set, a function is epi iff it is surjective

- $f : A \rightarrow B$ is a *monomorphism* (is *mono*):
for all $g, h : C \rightarrow A$, $g;f = h;f$ implies $g = h$

In Set, a function is mono iff it is injective



Simple facts

- If $f : A \rightarrow B$ and $g : B \rightarrow C$ are mono then $f;g : A \rightarrow C$ is mono as well.
- If $f;g : A \rightarrow C$ is mono then $f : A \rightarrow B$ is mono as well.

Prove, and then dualise the above facts.

NOTE: A morphism f is mono in \mathbf{K} iff f is epi in \mathbf{K}^{op} .

mono = co-epi

Give “natural” examples of categories where epis need not be “surjective”.
Give “natural” examples of categories where monos need not be “injective”.

Isomorphisms

$f : A \rightarrow B$ is an *isomorphism* (is *iso*)
if there is $g : B \rightarrow A$ such that $f;g = id_A$ and $g;f = id_B$.

Then g is the (unique)
inverse of f , $g = f^{-1}$.

In **Set**, a function is iso iff it is both epi and mono

Fact: If f is iso then it is both epi and mono. Give counterexamples to show that the opposite implication fails.

Fact: $f : A \rightarrow B$ is iso iff

- f is a *retraction*, i.e., there is $g_1 : B \rightarrow A$ such that $g_1;f = id_B$, and
- f is a *coretraction*, i.e., there is $g_2 : B \rightarrow A$ such that $f;g_2 = id_A$.

Fact: A morphism is iso iff it is an epi coretraction.

Fact: Composition of isomorphisms is an isomorphism.

Dualise!