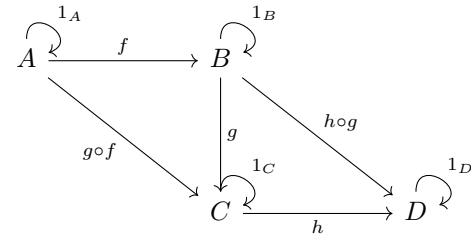


Category Theory Cheat Sheet

Category

A *category* consists of the following,

- Objects: A, B, C, \dots
- Arrows/Morphisms: f, g, h, \dots
- For each f there exists, $\text{dom}(f), \text{cod}(f)$ called domain and codomain of f . We write $f : A \rightarrow B$ to indicate $A = \text{dom}(f)$ and $B = \text{cod}(f)$.
- Given $f : A \rightarrow B$ and $g : B \rightarrow C$ there exists, $g \circ f : A \rightarrow C$ called the *composite* of f and g .
- For each A , there exists $1_A : A \rightarrow A$ called the *identity arrow* of A .
- Arrows should also satisfy the following,
 - Associativity: $h \circ (g \circ f) = (h \circ g) \circ f$, for all $f : A \rightarrow B, g : B \rightarrow C, h : C \rightarrow D$.
 - Unit: $f \circ 1_A = f = 1_B \circ f$, for all $f : A \rightarrow B$.



Functor

A *functor* $F : \mathbf{C} \rightarrow \mathbf{D}$ between \mathbf{C} and \mathbf{D} is a mapping of objects and arrows, such that

- $F(f : A \rightarrow B) = F(f) : F(A) \rightarrow F(B)$
- $F(1_A) = 1_{F(A)}$
- $F(g \circ f) = F(g) \circ F(f)$.

Isomorphism

In any category \mathbf{C} , an arrow $f : A \rightarrow B$ is called an *isomorphism* if there exists an arrow $g : B \rightarrow A$ s.t. $g \circ f = 1_A$ and $f \circ g = 1_B$. We say, $g = f^{-1}$. And that $A \cong B$, i.e., A is isomorphic to B .

Monoid

A set M with binary operation \cdot is called a *monoid* if it is associative and has an identity $u \in M$, i.e., for $x, y, z \in M$

- $x \cdot (y \cdot z) = (x \cdot y) \cdot z$
- $u \cdot x = x = x \cdot u$.

A monoid with an inverse for each element is called a *group*.

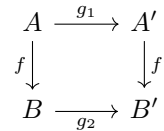
- *Cayley's theorem*: Every group G is isomorphic to a group of permutations.

Constructions on categories

- **Product category**: The product of two categories \mathbf{C} and \mathbf{D} written as $\mathbf{C} \times \mathbf{D}$ has objects of the form (C, D) for $C \in \mathbf{C}$ and $D \in \mathbf{D}$, and arrows of the form $(f, g) : (C, D) \rightarrow (C', D')$ for $f : C \rightarrow C' \in \mathbf{C}$ and $g : D \rightarrow D' \in \mathbf{D}$. Composition and units are defined componentwise, i.e. $(f', g') \circ (f, g) = (f' \circ f, g' \circ g)$ and $1_{(C, D)} = (1_C, 1_D)$.
- **Opposite/Dual category**: For category \mathbf{C} its opposite category \mathbf{C}^{op} has the same objects as \mathbf{C} but an arrow $f : C \rightarrow D$ in \mathbf{C}^{op} is an arrow $f : D \rightarrow C$ in \mathbf{C} . For notational simplicity we say $f^* : D^* \rightarrow C^*$ in \mathbf{C}^{op} for $f : C \rightarrow D$ in \mathbf{C} . Composition and units are therefore defined as follows, $f^* \circ g^* = (g \circ f)^*$ and $(1_C)^* = 1_{C^*}$.

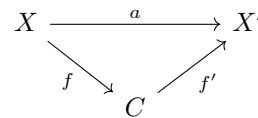
Constructions on categories contd.

- **Arrow category**: For category \mathbf{C} its arrow category \mathbf{C}^{\rightarrow} has the arrows of \mathbf{C} as objects and an arrow g from $f : A \rightarrow B$ to $f' : A' \rightarrow B'$ in \mathbf{C}^{\rightarrow} is the following commutative square



where g_1, g_2 are arrows in \mathbf{C} , i.e. an arrow is a pair of arrows $g = (g_1, g_2)$ s.t. $g_2 \circ f = f' \circ g_1$. The identity of an object $f : A \rightarrow B$ is the pair $(1_A, 1_B)$ and composition is componentwise.

- **Slice category**: For category \mathbf{C} its slice category over $C \in \mathbf{C}$ is \mathbf{C}/C where its objects are all arrows $f \in \mathbf{C}$ s.t. $\text{cod}(f) = C$. An arrow a from $f : X \rightarrow C$ to $f' : X' \rightarrow C$ is an arrow $a : X \rightarrow X'$ in \mathbf{C} s.t. $f' \circ a = f$, i.e.

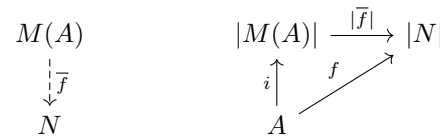


If $g : C \rightarrow D$ is any arrow then there exists a composition functor $g_* : \mathbf{C}/C \rightarrow \mathbf{C}/D$ defined as $g_*(f) = g \circ f$. Therefore the slice category of \mathbf{C} with any of its objects is a functor from $\mathbf{C} \rightarrow \mathbf{Cat}$. This is called the *forgetful* functor as the base object is "forgetten".

Free monoid

For a set A a *word* over A is any finite sequence of its elements. The *Kleene closure* of A is defined to be the set of all words over A . Define the binary operation of concatenation on A^* . Since it is associative, A along with $*$ with the empty word " ϵ " is a monoid called the **free monoid** on the set A .

Universal mapping property (UMP) of free monoid: Let $M(A)$ be the free monoid on a set A . There is a function $i : A \rightarrow |M(A)|$, and given any monoid N and any function $f : A \rightarrow |N|$, there is a unique monoid homomorphism $\bar{f} : M(A) \rightarrow N$ s.t. $|\bar{f}| \circ i = f$.

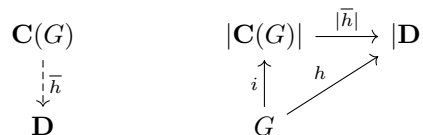


A^* has the UMP of the free monoid on A .

Free category

A directed graph G "generates" a free category $\mathbf{C}(G)$ whose objects are the vertices of the graph and its arrows are paths. Composition of arrows is defined as concatenation of paths.

UMP of $\mathbf{C}(G)$ There is a graphic homomorphism $i : G \rightarrow |\mathbf{C}(G)|$, and given any category \mathbf{D} and any graph homomorphism $h : G \rightarrow |\mathbf{D}|$, there is a unique functor $\bar{h} : \mathbf{C}(G) \rightarrow \mathbf{D}$ with $\bar{h} \circ i = h$.



Small categories

A category is called *small* if it has a small set of objects and arrows. (i.e., not classes). It is called large otherwise.

A category \mathbf{C} is *locally small* if for all objects $X, Y \in \mathbf{C}$, the collection $\text{Hom}_{\mathbf{C}}(X, Y) = \{f \in \mathbf{C}_1 \mid f : X \rightarrow Y\}$ is a small set.

Types of morphisms

Monomorphism: In any category \mathbf{C} , an arrow $f : A \rightarrow B$ is called a monomorphism (monic), if for any $g, h : C \rightarrow A, fg = fh \implies g = h$.

$$C \xrightarrow[g]{g} A \xrightarrow{f} B$$

Epimorphism: In any category \mathbf{C} , an arrow $f : A \rightarrow B$ is called an epimorphism (epic), if for any $i, j : B \rightarrow D$ $if = jf \implies i = j$.

$$A \xrightarrow{f} B \xrightarrow[i]{j} D$$

- We say, $f : A \rightarrowtail B$ if f is a monomorphism and $f : A \twoheadrightarrow B$ if f is an epimorphism.
- Every isomorphism is both a monomorphism and an epimorphism. The converse need not be true.
- A **split** mono (epi) is an arrow $m : A \rightarrow B$ with a left (right) inverse r . The inverse arrow r is called the *retraction*, m is called a *section* of r and A is called a *retract* of B .

Initial and terminal objects

An object $0 \in \mathbf{C}$ is *initial* if for any object $C \in \mathbf{C}$! morphism $0 \rightarrow C$.

An object $1 \in \mathbf{C}$ is *terminal* if for any object $C \in \mathbf{C}$! morphism $C \rightarrow 1$.

Initial and terminal objects are unique up to isomorphism.

Generalized elements

For an object $A \in \mathbf{C}$ arbitrary arrows $x : X \rightarrow A$ are called the *generalized elements* of A with stage of definition given by X .

Product of objects

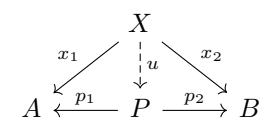
In any category \mathbf{C} , a product diagram for the objects A, B consists of an object P and arrows

$$A \xleftarrow{p_1} P \xrightarrow{p_2} B$$

satisfying the following UMP. Given any diagram of the form

$$A \xleftarrow{x_1} X \xrightarrow{x_2} B$$

there exists a unique arrow $u : X \rightarrow P$, making the following diagram commute



The product P is unique up to isomorphism.

Categories with products

A category which has a product for every pair of objects is said to have *binary products*.

A category is said to have *all finite products*, if it has a terminal object and all binary products.

A category has *all small products* if every set of objects has a product.

Covariant representable functor

The functor $\text{Hom}(A, -) : \mathbf{C} \rightarrow \mathbf{Sets}$ is called a covariant representable functor (for some object $A \in \mathbf{C}$).

For a category with products a covariant representable functor preserves products.

Duality

For any statement