Category Theory: Unifies mathematics, studies the mathematics of mathematics, moves toward higher-dimensional algebra ("homotopyifying" mathematics)

- **Set theory**
  - 0-dimensional

- **Category theory**
  - 1-dimensional

**Def**: A category \( C \) consists of:
- a class \( \text{Ob}(C) \) of objects
  - (If \( x \in \text{Ob}(C) \) we write simply \( x \in C \))
- Given \( x, y \in C \) there's a set \( \text{hom}(x, y) \), called a homset, whose elements are called morphisms or arrows from \( x \) to \( y \). If \( f \in \text{hom}(x, y) \) we write \( f : x \to y \).
- Given \( f : x \to y \) and \( g : y \to z \) there is a morphism called their composite \( g \circ f : x \to z \)
  - Composition is associative: \( (h \circ g) \circ f = h \circ (g \circ f) \) if either side is well-defined.
- For any \( x \in C \), there is an identity morphism \( 1_x : x \to x \)
- We have the left and right unit laws:
  - \( 1_x \circ f = f \) for any \( f : x' \to x \)
  - \( f \circ 1_x = f \) for any \( f : x \to x' \)

**Examples of categories**

**Categories of mathematical objects**

For any kind of mathematical object, there's a category with objects of that kind & morphisms being the structure-preserving maps between objects of that kind.
- Set is the category with sets as objects & functions as morphisms.
- Grp is the category of groups and homomorphisms.
- For \( k \) any field, Vec\( k \) of vector spaces over \( k \) and linear maps.
- Ring is the category of rings & ring homomorphisms.

These are categories of "algebraic" objects, namely:
- a set
  - stuff
- with operations
  - structure
- obeying equations
  - property

with morphisms being functions that preserve the operations.

All this is formalized in "universal algebra", using "algebraic theories".

There are also categories of non-algebraic gadgets:
- Top, the category of topological spaces & continuous maps.
- Met, the category of metric spaces & continuous maps.
- Meas, the category of measurable spaces & measurable maps.

Categories as mathematical objects
- There are lots of small, manageable categories:

Def\[ A \text{ monoid is a category with one object } \quad \xrightarrow{f} \]
(Then hom(\(1, x\)) for this object \( x \) is a set
with associative product & init.)

Ex\[ \xrightarrow{f} \quad \text{with } \quad 1x \circ f = f \]
\[ f \circ 1x = f \]
\[ f - f = 1x \quad \text{is usually called } \mathbb{Z}/2 \]

Or we could take \( f \circ f = f \), and this gives another famous monoid:
- \( 1_x = \text{TRUE} \)
- \( f = \text{FALSE} \)
- \( \circ = \text{AND} \)
- \( f = \text{TRUE} \)
- \( \circ = \text{OR} \)
Def: A morphism \( f: x \to y \) is an isomorphism if it has an inverse \( g: y \to x \), i.e., a morphism with
\[
\begin{align*}
g \circ f & = 1_x \\
f \circ g & = 1_y
\end{align*}
\]
If there exists an isomorphism between 2 objects \( x, y \in C \), we say they're isomorphic.

Def: A category whose all morphisms are isomorphisms is called a groupoid.

Ex: "the groupoid of finite sets" is obtained by taking FinSet, with finite sets as objects and functions as morphisms, and then throwing out all morphisms except isomorphisms (i.e., bijections), getting a groupoid.

Def: A monoid that is a groupoid is called a group.
(The usual "elements" of a group are now the morphisms.)

Def: A category with only identity morphisms is a discrete category.
So any set is the set of objects of some discrete category in a unique way.
So a discrete category is "essentially the same" as a set.

Def: A preorder is a category with at most one morphism in each hom set.
\[
\begin{align*}
x & \leq y \\
x & \leq y
\end{align*}
\]
If there is a morphism \( f: x \to y \) in a preorder we say "\( x \leq y \)"; if not, we say "\( x \nleq y \)".
For a preorder, the category axioms just say
\[
\begin{align*}
\text{composition: } & x \leq y & \text{ & } y \leq z \implies x \leq z \\
\text{associativity is automatic} \\
\text{identities: } & x \leq x \text{ always} \\
\text{left & right unit laws are automatic.}
\end{align*}
\]
We're not getting antisymmetry:
\[
x \leq y \text{ & } y \leq x \implies x = y
\]
Categories as Mathematical Object, cont.

**Def.** A preorder is a category \( C \) where for all \( x, y \in C \) there is at most one morphism \( f : x \to y \).

We write "\( x \leq y \)" iff \( \exists f : x \to y \).

We know what \( C \) is if we know this relation on objects (if \( C \) is a preorder), & then the category axioms simply say:

- there's a class of objects
- there's a relation \( \leq \) on objects
- \( x \leq y \) & \( y \leq z \) \( \Rightarrow \) \( x \leq z \) (composition) \( \forall x, y, z \in C \)
- \( x \leq x \) (identities) \( \forall x \in C \)

**Def.** An equivalence relation is a preorder that's also a groupoid.

**Prop.** A preorder is a groupoid iff this extra law holds:

\[
x \leq y \Rightarrow y \leq x \quad \forall x, y \in C.
\]

Hence we have transitivity, reflexivity, & symmetry of "\( \leq \), so we usually call this relation "\( = \)".

**Prop.** A preorder is skeletal, i.e. isomorphic objects are equal, iff this extra law holds:

\[
x \leq y \text{ & } y \leq x \Rightarrow x = y \quad \forall x, y \in C
\]

In this case we say \( C \) is a poset.

**Preorder:**

\[
\begin{align*}
C & \xrightarrow{c} \ & \ x \xrightarrow{c} \ & \ y \\
C & \xrightarrow{c} \ & \ y \xrightarrow{c} \ & \ z
\end{align*}
\]

This is a groupoid; this part is a poset; but not a poset, but not a groupoid.
Since categories can be seen as mathematical objects, we should define maps between them:

**Def.** Given categories C & D, a functor \( F : C \to D \) consists of:
- a function called \( F \) from \( \text{Ob}(C) \) to \( \text{Ob}(D) \); if \( x \in C \) then \( F(x) \in D \)
- functions called \( F \) from \( \text{hom}(x, y) \) (\( x, y \in C \)) to \( \text{hom}(F(x), F(y)) \);
  if \( f : x \to y \) then \( F(f) : F(x) \to F(y) \)

such that:
- \( F(g \circ f) = F(g) \circ F(f) \) whenever either side is well-defined
- \( F(1_x) = 1_{F(x)} \) \( \forall x \in C \)

So, a functor looks like this:

![Diagram of a functor between categories C and D]

**Ex.** There's a category called "1". It looks like this: \( \times 1 \)

What is a functor \( F : 1 \to C \), where \( C \) is any category?

![Diagram of a functor from 1 to C]

The answer is: "an object in \( C \)", since for any \( c \in C \)

\( F : 1 \to C \) s.t. \( F(1) = c \)

**Ex.** There's a category called "2". (a poset)

![Diagram of a category with two objects a, b and a morphism f]

\( a \xrightarrow{f} b \)
What is a functor $F : C \to C$?
It's just a morphism or arrow in $C$!
For any morphism $g : c \to c'$ in $C$, there is a functor $F : 2 \to C$
s.t. $F(f) = g$.

**Prop.** If $F : C \to D$ and $G : D \to E$ are functors then you can define a

functor $G \circ F : C \to E$, and $(G \circ H) \circ F = H \circ (G \circ F)$.

Also, for any category $C$ there is an identity functor $1_C : C \to C$ with

\[ 1_C(c) = c \quad \forall c \in C \]
\[ 1_C(f) = f \quad \forall f : x \to y \text{ in } C \]

and $F \circ 1_C = F$ \quad $1_C \circ H = H$ \quad $\forall H : D \to C$.

**Def.** Cat is the category whose objects are "small" categories & whose

morphisms are functors.

(A "small" category is one with a set of objects --- so e.g. Set or

Group or Ring is not small, while 1 & 2 are small.)

**Doing Mathematics inside a category.**

A lot of math is done in Set, the category of sets & functions. Let's try
to generalize all that stuff to other categories: replace Set by a general
category $C$.

In Set we have "one-to-one" & "onto" functions.

In a category $C$ we generalize these concepts to "epimorphisms" or "epis" &
"monomorphisms" or "monos".

**Def.** A morphism $f : X \to Y$ is a mono if $\forall g, h : Q \to X$ we have

$f \circ g = f \circ h \implies g = h$

\[ Q \xrightarrow{g} X \xrightarrow{f} Y \]

\[ Q \xrightarrow{h} X \]

\[ h \]
Prop In Set, a morphism is monic iff it's a one-to-one function.

Turning around the arrows in the definition of mono, we get:

Def A morphism \( f: Y \rightarrow X \) is an epi if \( \forall g, h: X \rightarrow Q \) we have
\[
  g \circ f = h \circ f \Rightarrow g = h.
\]

\[
  Y \xrightarrow{f} X \xrightarrow{g} Q
\]

Prop In Set, a morphism is epic iff it's an onto function.