Intro to Category Theory

Emma Cardwell

June, 2020

1 Introduction

Category theory was formulated as a way to study the relationships between different mathematical structures. It was first introduced by Samuel Eilenberg and Saunders Mac Lane in 1945. It was first used mainly for algebraic topology and abstract algebra. In the 1950s, mathematicians began using category theory for algebraic geometry. Since then, category theory has been applied to logic, computer science, linguistics, philosophy, and several other areas [Awo11].

1.1 Definition of Category

A category is similar to an ordered set of mathematical structures. The purpose of categories is to connect mathematical structures together with structure-preserving functions. A category $\mathcal{C}$ is made up of objects and functions, more specifically, A Category $\mathcal{C}$ consists of the following:

1. A class of objects, often denoted as $Ob(\mathcal{C})$. If an object $x \in Ob(\mathcal{C})$, we say that the object is in the category, or $x \in \mathcal{C}$.

2. Morphisms (functions) between objects. For any two elements $x, y \in Ob(\mathcal{C})$, there is a set of morphisms from x to y, called $Hom_{\mathcal{C}}(x, y)$. If $f \in Hom_{\mathcal{C}}(x, y)$, then $f : x \to y$. The set of *all* morphisms between objects in $\mathcal{C}$ is denoted as $Hom(\mathcal{C})$.

We also can compose morphisms together: For any morphisms $f : x \to y$ and $g : y \to z$, the composite of f and g is defined as: $g \circ f : x \to z$, and is a morphism in $Hom_{\mathcal{C}}(x, z)$. The objects and morphisms in $\mathcal{C}$ must satisfy the following conditions with regards to morphism composition:

1. Morphisms are associative: For objects $a, b, c, d \in \mathcal{C}$, if $f : a \to b$, $g : b \to c$, and $h : c \to d$, then $h \circ (g \circ f) = (h \circ g) \circ f$.

2. Every object has an identity morphism: For every $x \in \mathcal{C}$, there exists a morphism $id_x \in Hom(\mathcal{C})$ with $id_x : x \to x$ such that, for every $f : a \to x$ and $g : x \to b$, we have $id_x \circ f = f$ and $g \circ id_x = g$.

For a morphism between two objects, $f : x \to y$, we say that x is the *domain* of f, and y is the *codomain* of f. We denote this as $x = \operatorname{dom} f$ and $B = \operatorname{cod} f$.

Example 1.1 *One example of a category is* **Set**. *The objects in* **Set** *are all of the finite sets, and the morphisms of* **Set** *are functions between sets. We can always compose functions between sets, and composition is always associative. There is always exactly one function from a set to itself such that no elements in the set are permuted, so the identity property is also satisfied.*

Example 1.2 *Another category is the category of groups,* **Grp**. *All groups are objects in* **Grp** *and the morphisms of* **Grp** *are group homomorphisms.*

Example 1.3 *Any deductive system* **T** *can be thought of as a category with logical statements as objects and proofs or justifications for the statements as morphisms [Mar20].*

One way to think about objects in general is as sets with varying levels of additional structure imposed on them (depending on what category they belong to).

1.2 Diagram Notation

We can use diagrams to represent categories. Each node is an object, and the arrows between nodes represent morphisms between the objects. Diagrams can help us better visualize morphisms and their compositions in a category. A directed path of arrows represents a composition. For example, the following diagram represents $h \circ g \circ f$, where $f : A \to B$, $g : B \to C$, and $h : C \to D$:

$$A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$$

2 More about Morphisms

Morphisms can also have some additional properties that include (but aren't limited to):

Definition 2.1 *A morphism $f : x \to y$ is a monomorphism if, for all morphisms $g : w \to x$ and $h : w \to x$, if $f \circ g = f \circ h$, then g must equal h.*

Example 2.2 *A function in the category* **Set** *is a monomorphism if and only if it is an injective function. (Note that this is not true for all categories).*

Definition 2.3 *A morphism $f : x \to y$ is an epimorphism if, for all morphisms $g : w \to x$ and $h : w \to x$, if $g \circ f = h \circ f$, then g must equal h.*

Example 2.4 *A function in the category* **Set** *is an epimorphism if and only if it is a surjective function. (Note that this is not true for all categories).*

Definition 2.5 *A morphism between two objects in a category, $f : x \to y$, is an isomorphism if there exists some other morphism $f^{-1} : x \to y$ such that $f^{-1} \circ f = id_x$ and $f \circ f^{-1} = id_y$. We then say that x and y are isomorphic.*

Example 2.6 *A function in the category* **Set** *is a monomorphism if and only if it is injective, and an epimorphism if and only if it is surjective. Thus, a function is an isomorphism in* **Set** *if and only if it is bijective.*

As we shall see, some types of categories are defined by the types of morphisms they contain.

3 Types of Categories

Definition 3.1 *A monoid is a category with one object.*

Example 3.2 *The set of all finite strings over some fixed alphabet Σ forms a monoid with string concatenation as the operation. The empty string serves as the identity element. This monoid is denoted $\Sigma *$ and is called the free monoid overΣ.*

Definition 3.3 *A groupoid is a category with only isomorphisms.*

We can define groups using category theory.

Proposition 3.4 *The set of all morphisms of a monoid that is also a groupoid is a group.*

This seems trippy, so let's break it down further. Let $\mathbf{M}$ be a monoid. Since $\mathbf{M}$ only has one object, we can let $Ob(\mathbf{M}) = m$, and the set of all morphisms in $\mathbf{M}$ must be from m to m. Equivalently, $Hom_{\mathbf{M}}(m, m) = Hom(\mathbf{M})$. If we think of $\mathbf{M}$ as a set, then the morphisms are simply mappings between elements of the set. We can compose any two morphisms and get another morphism that belongs to $\mathbf{M}$. Furthermore, from our definition of Category, these morphisms must be associative, and there must be an identity morphism. Sounding familiar? The morphisms seem to satisfy the properties of groups, except we're missing one - inverses. That's where groupoids come in. If $\mathbf{M}$ is a groupoid, then every morphism $f : m \to m$ has an inverse. By definition of isomorphism, there must exist some other morphism $f^{-1} : m \to m$ such that $f \circ f^{-1} = id_m = f^{-1} \circ f$. Now, the morphisms of $\mathbf{M}$ are associative and have closure, identity, and inverses, so $Hom(\mathbf{C})$ is a group. ■

4 Direct Product of Categories

$\mathbf{C} \times \mathbf{D}$, the product of two categories $\mathbf{C}$ and $\mathbf{D}$, has objects of the form (c, d), where $c \in \mathrm{Ob}(\mathbf{C})$ and $d \in \mathrm{Ob}(\mathbf{D})$. The resulting morphisms are of the form $(f, f') : (c, d) \to (c', d')$ where $f \in Hom_{\mathbf{C}}(c, c')$ and $f' \in Hom_{\mathbf{D}}(d, d')$. A composition of morphisms is represented as $(g, g') \circ (f, f') = (g \circ f, g' \circ f')$.

Let's show that $\mathbf{C} \times \mathbf{D}$ is indeed a category. We know that $\mathbf{C} \times \mathbf{D}$ has a set of objects, a set of morphisms between the objects, and a form of morphism composition. To prove that $\mathbf{C} \times \mathbf{D}$ is a category, we still need to show associativity for morphisms and the existence of identity morphisms for every object.

To show associativity, let's consider the morphisms $(f, f') : (a, a') \to (b, b')$, $(g, g') : (b, b') \to (c, c')$, and $(h, h') : (c, c') \to (d, d')$, where $(a, a'), (b, b'), (c, c'), (d, d') \in \mathrm{Ob}(\mathbf{C} \times \mathbf{D})$, and $(f, f'), (g, g'), (h, h') \in Hom(\mathbf{C} \times \mathbf{D})$.

$$(h, h') \circ \Big((g, g') \circ (f, f')\Big) = (h, h') \circ (g \circ f, g' \circ f')$$

Since $f, g, h \in Hom(\mathbf{C})$ and $f', g', h' \in Hom(\mathbf{D})$, the morphisms f, g, h and f', g', h' must be associative, so we can regroup the compositions:

$$\begin{aligned} &= (h \circ g \circ f, h' \circ g' \circ f') = \Big((h \circ g) \circ f, (h' \circ g') \circ f'\Big) \\ &= \Big(h \circ g, h' \circ g'\Big) \circ (f, f') \\ &= \Big((h, h') \circ (g, g')\Big) \circ (f, f') \\ \therefore (h, h') \circ \Big((g, g') \circ (f, f')\Big) &= \Big((h, h') \circ (g, g')\Big) \circ (f, f') \end{aligned}$$

Thus, we've shown that the composition of morphisms in $\mathbf{C} \times \mathbf{D}$ is associative.

Finally, the identity morphism for any object $(c, d) \in \mathbf{C} \times \mathbf{D}$ is (Id_c, Id_d), where Id_c is the identity morphism for $c \in \text{Ob}(\mathbf{C})$, and Id_d is the identity morphism for $d \in \text{Ob}(\mathbf{D})$. Since every $c \in \text{Ob}(\mathbf{C})$ and every $d \in \text{Ob}(\mathbf{D})$ has an identity morphism, every object $(c, d) \in \mathbf{C} \times \mathbf{D}$ must also have an identity morphism.

Example 4.1 *Since groups are monoids, the direct product of groups is essentially the same as the direct product of monoids. For groups G and H, we can construct $G \times H$ by considering the set generated by*

$$\{\langle g, h\rangle \mid g \in G, h \in H\}$$

Since the elements of a group are the morphisms of a monoid, operations on elements are equivalent to compositions on morphisms. So, we define the binary operation of $G \times H$ as

$$\langle g, h\rangle \cdot \langle g', h'\rangle = \langle g \cdot g', h \cdot h'\rangle$$

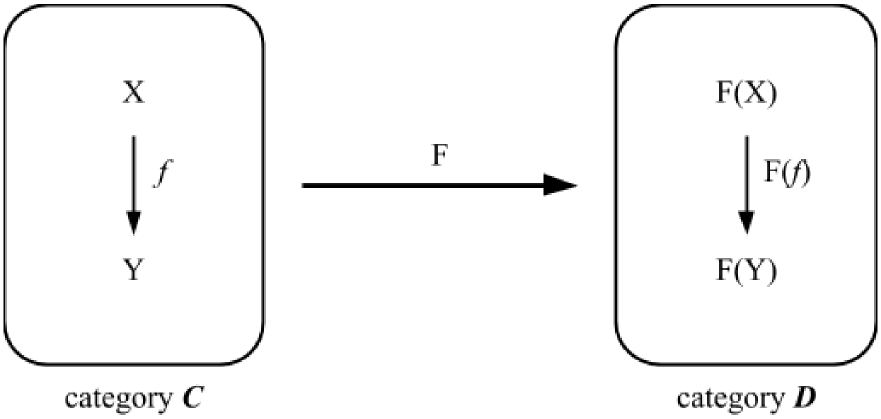
Since monoids are categories with only one object, the product of monoids must generate another monoid. Thus, there is exactly one object in $G \times H$. That implies that there is only one identity morphism for $G \times H$, which is $\langle e_G, e_H\rangle$.
Lastly, inverses. Since for all $g \in G$, $H \in H$, $\langle g, h\rangle \in G \times H$, we have

$$\langle g, h\rangle \cdot \langle g^{-1}, h^{-1}\rangle = \langle e_G, e_H\rangle = \langle g, h\rangle \cdot \langle g, h\rangle^{-1}$$

Thus, inverses are preserved.

5 Functors

A functor is a mapping between categories. The diagram below illustrates a functor between categories **C** and **D**:



A functor must include some correspondence between the objects of each category, the morphisms of each category, the domains and codomains of the morphisms, the identity morphisms, and the composition of morphisms in each category.

More formally,

Definition 5.1 *Let $F : \mathcal{C} \to \mathcal{D}$ denote a covariant functor from category $\mathcal{C}$ to category $\mathcal{D}$. F must contain:*

1. *A function that relates the objects of $\mathcal{C}$ to the objects of $\mathcal{D}$: $F : Ob(\mathcal{C}) \to Ob(\mathcal{D})$. For an object $c \in Ob(\mathcal{C})$, we typically write $F(c)$ to denote the corresponding object in $\mathcal{D}$.*
2. *A function that relates the morphisms of $\mathcal{C}$ to the morphisms of $\mathcal{D}$: For any two objects $c_1, c_2 \in Ob(\mathcal{C})$, we have a function $F : Hom_{\mathcal{C}}(c_1, c_2) \to Hom_{\mathcal{D}}(F(c_1), F(c_2))$.*

And F must satisfy the following properties:

1. *Identity morphisms are preserved, so for any object $c \in Ob(\mathcal{C})$, $F(id_c) = id_{F(c)}$.*
2. *The composition of morphisms is preserved, so for any objects $c_1, c_2, c_3 \in Ob(\mathcal{C})$ and morphisms $g : c_1 \to c_2$, $h : c_2 \to c_3$, we have $F(h \circ g) = F(h) \circ F(g)$.*

Just like how Group Theory allows us to make generalizations about shared properties between various groups, we can prove statements at the category level and apply them to the structures that they generalize. Now that we've defined Groups in Category Theory terms, we can prove properties about Groups. If we think of groups as categories, a group homomorphism is a functor between groups. We will look at a simple example of how we can use category theory results to prove some properties of groups.

Definition 5.2 *Given monoids $\mathbf{M}_1$, $\mathbf{M}_2$, we define a homomorphism as a functor $f : \mathbf{M}_1 \to \mathbf{M}_2$ such that the following are satisfied:*

1. *$f(m \circ n) = f(m) \circ f(n)$ for any $m, n \in Hom(\mathbf{M}_1)$*
2. *$f(e_{\mathbf{M}_1}) = e_{\mathbf{M}_2}$, where $e_{\mathbf{M}_1}$ is the identity element of $\mathbf{M}_1$ and $e_{\mathbf{M}_2}$ is the identity element of $\mathbf{M}_2$.*

Proposition 5.3 *A homomorphism between monoids preserves inverses: Let $f : \mathbf{M}_1 \to \mathbf{M}_2$ be the homomorphism between monoids $\mathbf{M}_1$ and $\mathbf{M}_2$. Then, $f(x^{-1}) = f(x)^{-1}$for all $x \in \mathbf{M}_1$.*

Since inverses of elements in monoids are unique (proof is left as an exercise to the reader), we have:

$$\begin{aligned} f(e_{\mathbf{M}_1}) = f(x \circ x^{-1}) = f(x) \circ f(x^{-1}) &= e_{\mathbf{M}_2} \\ f(x)^{-1} \circ f(x) \circ f(x^{-1}) &= f(x)^{-1} \circ e_{\mathbf{M}_2} \\ f(x^{-1}) &= f(x)^{-1} \quad \text{for all } x \in \mathbf{M}_1 \end{aligned}$$

Therefore homomorphisms between monoids preserve inverses. ■

Recall that the elements in any group can be represented as the set of morphisms of a monoid. This implies that group homomorphisms also preserve inverses. So, we've proved something about groups without actually doing anything with groups. While this proof using category theory is basically the same as the group theory proof, it applies to a wider range of structures. In other words, if we only wanted to prove this for groups, then it doesn't matter which way we prove it, but if we want to prove that this result holds true for other structures, then it makes sense to work with categories.

5.1 Opposite Functor

Before we define what an opposite functor is, we first must define what an opposite category is.

Definition 5.4 *Let* $\mathbf{C}$ *be a category. The opposite category,* $\mathbf{C}^{op}$ *of* $\mathbf{C}$ *is the category generated by reversing each morphism in* $\mathbf{C}$*.*

The opposite functor is simply the functor between two opposite categories:

Definition 5.5 *Given categories* $\mathbf{C}$*,* $\mathbf{D}$*, let* $F : C \to D$*. The opposite functor is defined as*
$F^{op} : C^{op} \to D^{op}$*.*

5.2 Contravariant Functors

A contravariant functor is similar to an inverse of a covariant functor. Contravariant functors are covariant functors on the opposite category. So, for categories $\mathbf{C}$ and $\mathbf{D}$, if $F : \mathbf{C} \to \mathbf{D}$ is a contravariant functor, then $F : \mathbf{C}^{\mathrm{op}} \to \mathbf{D}$ is a covariant functor.

For example, Hom-sets are sets of morphisms between objects, and hom-functors define mappings between Hom-sets. One type of hom-functor, $Hom(-, A)$, is a contravariant functor:

Definition 5.6 *Let* $\mathbf{A}$ *be a locally small category* [1] *and* $A \in \mathrm{Ob}(A)$*. We define a functor* $H_A = Hom(-, A)$*:* $\mathbf{A}^{op} \to \mathbf{Set}$*, such that the following hold:*

1. *For any object* $B \in \mathbf{A}$*,* $H_A(B) = Hom(B, A)$*.*

2. *For a morphism* $g \in Hom_{\mathbf{A}}(B', B)$*, we have* $H_A(g) = Hom(g, A) = g*$*, where* $g*$ *is the function that when composed with* g*, defines the mapping* $- \circ g : Hom(B, A) \to Hom(B', A)$ *by sending* $h : B \to A$ *to* $h \circ g$*.*

The "-" means "insert an object from the category in this spot", so $Hom(-, B)$ contains mappings between any object $c \in \mathrm{Ob}(\mathbf{C})$ and the set of morphisms, $Hom(c, B)$.

5.3 Diagrams

We can use diagrams for definitions and visual proofs. We say that a category diagram commutes if, for any two directed paths between two nodes, the composite arrow along the first path is equal to the composite arrow along the second path.

[1] A locally small category is defined as a category $\mathbf{C}$ in which $Hom_{\mathbf{C}}(a, b)$ for $a, b \in \mathbf{C}$ is a small set and not a proper class.

Definition 5.7 *For categories* $\mathbf{A}$,$\mathbf{B}$*, let's define functors* $F : \mathbf{A} \to \mathbf{B}$ *and* $G : \mathbf{A} \to \mathbf{B}$*. A natural transformation is a family of mappings* $\alpha : F \to G$ *such that* $\alpha_A : F(A) \to G(A)$ *for* $A \in \mathbf{A}$ *such that, for every morphism* $f : A \to A'$ *(where* $A, A' \in \mathbf{A}$*), the following diagram commutes (Image source: page 27 of [Lei16]):*

$$\begin{tikzcd}
F(A) \arrow[r, "F(f)"] \arrow[d, "\alpha_A"'] & F(A') \arrow[d, "\alpha_{A'}"] \\
G(A) \arrow[r, "G(f)"'] & G(A')
\end{tikzcd}$$

The mappings α_A are called the components of α.

6 The Yoneda Lemma

Definition 6.1 *Let* $\mathbf{A}$ *be a locally small category. The Yoneda embedding of* $\mathbf{A}$ *is the functor* $H_\bullet : \mathbf{A} \to [\mathbf{A}^{op}, \mathbf{Set}]$ *such that for every object* $A \in \mathrm{Ob}(\mathbf{A})$*,* $H_\bullet(A) = H_A : \mathbf{A}^{op} \to \mathbf{Set}$*.*

Let's consider another functor $X : \mathbf{A}^{\mathrm{op}} \to \mathbf{Set}$. An interesting question to ask is, how can we relate H_A to X? A relation between H_A and X is a relation between functors, so it is a natural transformation.

Proposition 6.2 *We define the set of natural transformations between* H_A *and* X *as* $Nat(H_A, X)$*. The Yoneda Lemma states that this set of natural transformations is isomorphic to the set* $X(A)$*. In other words,*

$$Nat(H_A, X) \cong X(A)$$

Why does this work? We need to show that, for any functor $X : \mathbf{A}^{\mathrm{op}} \to \mathbf{Set}$ and any object $A \in \mathbf{A}$, there exists a bijection between the sets $Nat(H_A, X)$ and $X(A)$. This is because, for sets, bijections and isomorphisms are the same.
For any element $A \in \mathbf{A}$ and functor $X \in [\mathbf{A}^{\mathrm{op}}, \mathbf{Set}]$, we define functions $(\hat{\ })$ and $(\tilde{\ })$ such that:

$$\begin{tikzcd}
Nat(H_A, X) \arrow[r, shift left, "(\hat{\ })"] & X(A) \arrow[l, shift left, "(\sim)"]
\end{tikzcd}$$

We now need to define $(\hat{\ })$ and $(\tilde{\ })$ and show that $(\hat{\tilde{\ }})$ and $(\tilde{\hat{\ }})$ are identities:

1. Given $\alpha \in X(A)$, we can define $\hat{\alpha} \in X(A)$ by $\hat{\alpha} = \alpha_A(id_A)$.

2. Given $x \in X(A)$, we can must a natural transformation $\tilde{x} : H_A \to X$. So, for any object $B \in \mathbf{A}$, we have $\tilde{x}_B : H_A(B) \to X(B)$, where $H_A(B) = Hom_{\mathbf{A}}(B, A)$. We can define a mapping between morphisms for $f \in Hom_{\mathbf{A}}(B, A)$ as $\tilde{x}_B(f) = \big(X(f)\big)(x) \in X(B)$.

3. To show that $(\hat{\tilde{\ }})$ is an identity, we need to show that $\hat{\tilde{x}} = x$, as this indicates that applying both functions to x negates their effect on x.

$$\begin{aligned}\hat{\tilde{x}} &= \tilde{x}_A(id_A) \quad \text{by our definition of } \hat{x} \\ &= \big(X(id_A)\big)(x) \quad \text{by our definition of } \tilde{x} \\ &= id_{X(A)}(x) \\ &= x\end{aligned}$$

4. Finally, to show that $(\tilde{\hat{\ }})$ is an identity, we need to show that $\tilde{\hat{\alpha}} = \alpha$. This is a bit more complicated to show, so we shall skip the proof. (See page 97 of [Lei16]).

We've shown that, for any functor $X : \mathbf{A}^{\text{op}} \to \mathbf{Set}$ and any object $A \in \mathbf{A}$, there exists a bijection between the sets $Nat(H_A, X)$ and $X(A)$, thus the sets are isomorphic. ■

6.1 Cayley's Theorem

Remember Cayley's Theorem from group theory? Since the Yoneda Lemma generalizes this, we can use the Yoneda Lemma to prove this result[2]. Recall that we can define groups in terms of categories - a group G is the set of all morphisms of a monoid $\mathbf{M}$ that is also a groupoid. Since a monoid only has one object, all of the morphisms map that object to itself, so they are all identity morphisms: $id_A : A \to A$ for all $id_{A_i} \in G$ where $\text{Ob}(\mathbf{M}) = \{A\}$. All of these morphisms are isomorphisms.

Back to Yoneda's Lemma, we have

$$Nat(H_A, X) \cong X(A)$$

Since both X and H_A are functors from $\mathbf{A}^{\text{op}} \to \mathbf{Set}$, we can let $X = H_A$:

$$Nat(H_A, H_A) \cong H_A(A)$$

First, $H_A(A)$ simply the set of all morphisms from A to itself - sounds familiar? That's because the set of all morphisms from an object A to itself form a group! This means that $H_A(A) = H_A = G$, so we can replace $H_A(A)$ with G.

Now, what about $Nat(H_A, H_A)$? Well, $Nat(H_A, H_A)$ is the set of mappings from H_A to itself. We've established that H_A is equivalent to G, so we can think of $Nat(H_A, H_A)$ as the set of morphisms from G to G. More precisely, since all morphisms that are elements in a group are isomorphisms, $Nat(H_A, H_A)$ is the set of isomorphisms from G to G, which is equal to the set of automorphisms of G! [3]

So, in group theory terms, the Yoneda Lemma states that the set of automorphisms of G is isomorphic to G. Equivalently, $Aut(G) \cong G$. It turns out that $Aut(G)$ is a group and $Aut(G)$ is always a subgroup of the group of all permutations of G. Let's prove this:

Proposition 6.3 *$Aut(G)$ is a group.*

1. IDENTITY: The identity map id_G is an automorphism.

[2] There are some corollaries of the Yoneda Lemma which are more closely related to Cayley's Theorem, but to avoid complicating things further by discussing them here, we will only use the Yoneda Lemma.

[3] I adapted this justification from [Bra17].

2. CLOSURE: The composition of two automorphisms is an automorphism. Let $f, g \in Aut(G)$ and $h = f \circ g$. For any $x, y \in G$, we have:

$$\begin{aligned} h(xy) &= f(g(xy)) \\ &= f(g(x)g(y)) \\ &= f(g(x))f(g(y)) = h(x)h(y) \end{aligned}$$

   We know that h must be an automorphism because h is a bijection and we just showed that h is a homomorphism.

3. ASSOCIATIVITY: We want to show that $h \circ (g \circ f) = (h \circ g) \circ f$. For any $x \in G$, we have:

$$\begin{aligned} \big(h \circ (g \circ f)\big)(x) &= h(x)\big((g \circ f)(x)\big) \\ &= h(x)g(x)f(x) \\ &= \big(h \circ g)(x)\big)f(x) = \big((h \circ g) \circ f\big)(x) \end{aligned}$$

4. INVERSES: For $f \in Aut(G)$, f is a group isomorphism from G to itself. Because f is an isomorphism, there must exist unique elements $x_1, x_2, y_1, y_2, \ldots \in G$ such that $f(x_2) = x_1$ and $f(y_2) = y_1$, and vice versa. This means that there must exist another automorphism f^{-1} that sends x_1 to x_2, y_1 to y_2, and so on. Thus, each element of $Aut(G)$ has an inverse.

Thus, $Aut(G)$ is a group. ■

Proposition 6.4 *$Aut(G)$ is a subgroup of S_n.*

Each automorphism $f \in Aut(G)$ permutes the elements in G, thus it is an element in the symmetric group acting on G, so $Aut(G) \subseteq Sym(G)$. We just proved that $Aut(G)$ is indeed a group, and $Aut(G) \subseteq Sym(G)$, thus $Aut(G)$ is a subgroup of $Sym(G)$. We know that $Sym(G)$ is isomorphic to $Sym(\{1, 2, \ldots, n\})$, where $n = |G|$, which is equivalent to the symmetric group S_n, thus $Sym(G) \cong S_n$. Since $Aut(G)$ is a subgroup of $Sym(G)$ and $Sym(G)$ is isomorphic to S_n, we have shown that $Aut(G)$ is isomorphic to a subgroup of the symmetric group.■

Theorem 6.5 *(Cayley's Theorem) Any finite group G is isomorphic to a subgroup of S_n.*

By the Yoneda Lemma, the set $Aut(G)$ is isomorphic to the set of elements in G (which form a group, by definition). We've proved that $Aut(G)$ is a group, and $Aut(G)$ is isomorphic to a subgroup of the symmetric group. Putting this together, we have proved that G is isomorphic to a subgroup of the symmetric group. ■

References

[Awo11] Steve Awodey. *Category Theory.* Oxford University Press, 2011.

[Bra17] Tai-Danae Bradley. The yoneda lemma, 2017.

[Lei16] Tom Leinster. Basic category theory, 2016, 1612.09375.

[Mar20] Jean-Pierre Marquis. Category theory. In Edward N. Zalta, editor, *The Stanford Encyclopedia of Philosophy.* Metaphysics Research Lab, Stanford University, summer 2020 edition, 2020.