Category Theory Notes

Bryan Lu

September 2021 - January 2022

0 Foreword

These are my own cleaned up personal notes on category theory that I've been studying over the past semester (FA21) and over winter break. I've been keeping notes on paper so far, but I now have a large sheaf of paper to carry around and they're not very well-written, so I think this will help me solidify the concepts and remember things a little better when I have to put them in my own words. These are intended as a summary of the important concepts of category theory as presented mainly in Emily Riehl's *Category Theory in Context*, with maybe a little sprinkling of guidance from Paolo Aluffi's *Algebra: Chapter 0*, with an emphasis on examples to help comprehension along the way.

I've memed a lot about category theory to my suitemates and friends over the past semester or so, and they're probably sick of it by now, so thank you for putting up with my abstract nonsense :)) Seriously though, to my knowledge category theory is a really helpful framework through which we can gain a deeper understanding of structure of important set-objects we care about without necessarily having to muck through the details of what's actually in the set. Instead, we study the relationships between objects and other objects of that type, so to speak, and there are a lot of cool things we can say about this!

Some prerequisites – you really should have some abstract algebra and topology background to start. I think a lot of the examples I've found in Riehl are good, but they take me a while to unpack and I feel not having any of this background would be confusing or possibly fatal. There might be an exception here early on if you understand a little bit of functional programming, but after a certain point you need to have a certain level of mathematical sophistication to continue.

Diagrams that are not computer generated are drawn in Inkscape and hopefully look pretty clean. Also, practicing using the commutative diagrams package, which is really helpful :))

1 Chapter 1: Basics

1.1 What is a Category?

Definition 1

A **category** consists of:

- a collection of **objects** $(X, Y, Z$, etc.$)$
- a collection of **morphisms** $(f, g, h$, etc.$)$ where the morphisms have *specified* **domain** and **codomain** objects, i.e. $f : X \to Y$

such that

- each object has an **identity** morphism $\mathrm{id}_X : X \to X$
- **composition** of morphisms gives a new morphism, so that if $f : X \to Y$, and $g : Y \to Z$, then $g \circ f = gf : X \to Z$.
- the identity behaves as expected with respect to composition, i.e. for all $f : X \to Y$, that $f\mathrm{id}_X = \mathrm{id}_Y f = f$.
- the composition of morphisms is **associative**, so if we have $f : X \to Y$, $g : Y \to Z$, and $h : Z \to W$, then $(hg)f = h(gf) := hgf : X \to W$.

Example 2

List of examples of some common categories:

- **Set**, the category with sets as objects and regular functions as morphisms
- **Grp**, the category with groups as objects and group homomorphisms as morphisms
- **Top**, the category with topological spaces as objects and continuous maps as morphisms
-

Note that categories are often named after their objects.

1.2 Opposite Day

Definition 3

If **C** is a category, we define the **opposite category** $\mathbf{C}^{\mathrm{op}}$

1.3 What is a Functor?

1.4 What is a Natural Transformation?

1.5 Equivalence of Categories

1.6 (there might be more important stuff here that I skipped idk)

2 Chapter 2: Representability, Yoneda, Universality

2.1 Representable Functors

2.2 The Yoneda Lemma

2.3 Universal Properties

2.4 The Category of Elements