

Discrete Mathematics

Daniel Gonzalez Cedre

University of Notre Dame
Spring of 2023

Chapter 3

Zermelo-Fraenkel Set Theory

“No one shall expel us from the paradise that Cantor has created.”

—David Hilbert

3.1 The Language of Set Theory

In order to use our first-order logic as a language with which to talk about math, we need to specify: what is our universe of discourse Ω , and what are our fundamental predicate symbols? The analogy drawn in class between the structure of the study of mathematics and the abstract structure of a modern computer should hopefully communicate how natural and common the notion is of having one *type* of object that implements other, more complicated objects. However, this decision actually goes back to the beginning of this 20th century revolution in mathematics. The initial solution people came up with to the fundamental logical and foundational problems they had discovered has to use a *ramified*—or *typed*—ontology, where different objects had different *types* in a hierarchy and there were rules governing how objects could be manipulated based on their *type*. The problem with this approach is that it becomes very syntactically-cumbersome for humans (the primary practitioners of mathematics) to deal with directly, so it was quickly abandoned for an *untyped* approach.

Note. The converse is true about the λ -calculus, which is perhaps the most famous mathematical model of computation after the Turing machine. Although the untyped λ -calculus is more expressive (*i.e.*, stronger) than any of the typed λ -calculi (and is thus more interesting to study for mathematicians), the modern functional programming languages we have today (*e.g.*, Haskell, the LISP dialects, F#) are actually implementations of typed λ -calculi because computers have no issues dealing with the syntactic complications of a typed theory.

Our universe of discourse will consist of *those objects that we can prove exist* using the rules of inference and the *axioms of set theory* (which we will develop in this chapter). The axioms of set theory will be sentences in the *language of Zermelo-Fraenkel set theory* that describe *what exactly sets are* and *how they work*. The language of set theory will have two predicate symbols, defined below.

Definition 3.1 (Equality).

We define the binary predicate $=$ to mean that its left argument is *identically the same* as its right argument. So, if x and y are sets, then we say $x = y$ when we mean that the names x and y both refer to the same underlying object.

Definition 3.2 (Elementhood).

We define the binary predicate \in to mean that its left argument is contained in its right argument as an element. So, if x and y are sets, then the phrase $x \in y$ conveys that x is an element of y .

Definition 3.3 (Language of Set Theory).

The *language of Zermelo-Fraenkel set theory* consists of the first-order logic along with

- I. a universe of discourse consisting of those things that provably exist from the axioms (Section 3.2),
- II. the binary predicates for equality ($=$, Definition 3.1) and elementhood (\in , Definition 3.2).

3.2 Axioms of Set Theory

Axiom 0 (Existence).

$$\exists x(x = x)$$

This axiom asserts that our universe of discourse is non-empty. Assuming this axiom lets us know for sure that when we make claims about sets, those claims are actually in reference to objects that provably exist (because we can use this axiom as an assumption in any proof).