Title

D. Zack Garza

Contents

1 | Saturday, November 28: Introduction to ∞-categories

Dealing with size issues: take a Grothendieck Universe $\mathcal{U}$: sets whose subsets are closed under all of the usual set operations (small).

> **Definition 1.0.1** (∞-Category)
> An ∞-category $\mathcal{C}$ is a (large) simplicial set $\mathcal{C}$ such that any diagram of the form
>
> $$\Lambda_i^n \longrightarrow \mathcal{C}$$
>
> (with Δ_n at the bottom and an indicated $\exists$ lift)
>
> admits the indicated lift, where Λ_i^n is an i-horn (a simplex missing the ith face) for $0 < i < n$.

Remark 1.0.2: This is a specialized notion of a Kan complex, and in particular all ∞-categories are Kan complexes. All inner horns are fillable, i.e. simplicial sets are *inner* Kan complexes. Different to Kan complexes, which include all i.

> **Definition 1.0.3** (Functors between ∞-categories)
> A ∞-functor between two ∞-categories is a map between simplicial sets.

> **Definition 1.0.4** (Nerve of a category)
> Given an ordinary category $\mathcal{C}$, define the **nerve** of $\mathcal{C}$ to be the simplicial set given by
>
> $$N(\mathcal{C})_n := \{\text{Functors } F : [n] \to \mathcal{C}\}$$
>
> where $[n]$ is the poset category on $\{1, 2, \cdots, n\}$. So an n-simplex is a diagram of objects $X_0, \cdots, X_n \in \mathrm{Ob}(\mathcal{C})$ and a sequence of maps. This defines an ∞-category, and there is a correspondence
>
> $$\left\{ \text{Functors } F : \mathcal{C} \to \mathcal{D} \right\} \iff \left\{ \infty\text{-Functors } \widehat{F} : N(\mathcal{C}) \to N(\mathcal{D}) \right\}.$$
>
> Note that taking the nerve of a category preserves the usual categorical structure, since the objects are the 0-simplices and the morphisms are the 1-simplices.
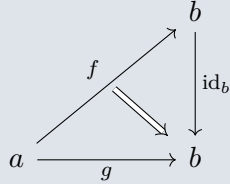
Remark 1.0.5: For $\mathcal{C}$ an ∞-category, we can define $\mathcal{C}_0$ to be the "objects" and $\mathcal{C}_1$ to be the "morphisms", although we don't have a good notion of composition yet. There will be boundary map: a 1-simplex has two boundary points, i.e. two objects $a, b \in \mathcal{C}_0$, so we can think of this as a map $f : a \to b$ where $a = \partial_1 f, b = \partial_0 f$[1] are the first and second vertices respectively. We'll also have "degeneracy" maps going up from $\mathcal{C}_0 \to \mathcal{C}_1$, which we should think of as assigning identity morphisms to objects, or conversely that the identity morphism is the degenerate 1-simplex at an

---

[1]This notation ∂_i denotes the boundary operator that drops the ith vertex.

object.

> **Definition 1.0.6** (Equivalence of Morphisms)
> Given two morphisms $f, g : a \to b$ in an ∞-category, we say $f \simeq g$ are **equivalent** iff there is a 2-simplex filling in the following diagram:
>
> $$
> \begin{array}{ccc}
> & b & \\
> f \nearrow & \Big\downarrow \text{id}_b & \\
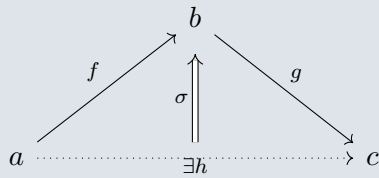> a \xrightarrow[\quad g \quad]{} & b &
> \end{array}
> $$
>
> Link to diagram

Remark 1.0.7: This turns out to be an equivalence relation. Note that in an ordinary category, if two morphisms are equivalent then they are already equal.

> **Definition 1.0.8** (Composition of morphisms)
> For 1-simplices $f : a \to b, g : b \to c$, a **composition** of f and g is a 2-simplex σ filling in the following diagram:
>
> $$
> \begin{array}{ccc}
> & b & \\
> f \nearrow \ \uparrow \sigma & & \searrow g \\
> a \cdots\cdots_{\exists h}\cdots\cdots & \rightarrow & c
> \end{array}
> $$
>
> Link to diagram
> In this case, $h := \partial_1 \sigma$ and we write $h \simeq g \circ f$.

Remark 1.0.9: Note that we're not fixing a choice, but it is well-defined up to the equivalence relation we're using. This is similar to how e.g. coproducts are not baked into the structure of a category, but are instead only well-defined up to canonical isomorphism – and in fact, this characterization is sometimes preferable.

> **Definition 1.0.10** (Equivalences of objects)
> If $f : a \to b$ is a morphism in an ∞-category $\mathcal{C}$, then we say f is an **equivalence** if there exists a morphism $g : b \to a$ such that $\text{id}_a \simeq g \circ f$ and $\text{id}_b \simeq f \circ g$. This is equivalent to finding a 2-simplex that simultaneously fills the following two diagrams: