Lecture 16: Monoidal categories and monoidal functors.

Most of the algebraic objects you're familiar with probably have some associativity and unitality involved.

Example 16.1. In $\mathcal{C} = \mathsf{Sets}$, fix $X \in \mathrm{Ob}\,\mathcal{C}$. Then a unital monoid is the data of

(1) A function $X \times X \to X$, and
(2) A function $* \to X$ picking out the unit.

These satisfy the usual unit and associativity conditions. (Often, the existence of a unit is phrased as a *property* of the multiplication operation.)

Example 16.2. In $\mathcal{C} = \mathsf{Vect}_k$, fix $V \in \mathrm{Ob}\,\mathcal{C}$. Then a unital associative algebra is the data of

(1) A function $V \otimes V \to V$, and
(2) A function $k \to V$ picking out the unit.

These also have to satisfy a unit and associativity condition, where again the unit's existence is often cited as a property rather than additional data.

Example 16.3. One might also think about "algebras in categories;" that is, if $\mathcal{C} = \mathsf{Cat}$ is the category of categories, you could fix $X \in \mathcal{C}$ and ask for

(1) A functor $X \times X \to X$, and
(2) A functor $* \to X$ from the trivial category, picking out an object of X.

My claim is that to systematically understand the first two examples, we should understand the last—in fact, note that the first two examples involve *knowing* the construction of new objects called $X \times X$, or $V \otimes V$. This isn't god-given with just the data of the category $\mathcal{C} = \mathsf{Set}$ or $\mathcal{C} = \mathsf{Vect}$; one has to *specify* this data.

But the data of a way to give two objects and produce a third looks a lot like the data of a functor

$$\mathcal{C} \times \mathcal{C} \to \mathcal{C}$$

which brings us to the last example of "algebras in categories."

16.7. Monoidal categories. A monoidal category is like an "associative algebra" in categories. Normally, to articulate associativity, we would demand an equation like

$$(U \otimes V) \otimes W = U \otimes (V \otimes W)$$

or

$$(xy)z = x(yz).$$

But the point I want to make in this lecture is that, even in our most natural examples, equalities above should be replaced with *isomorphisms*.

EXAMPLE 16.4. Let $\mathcal{C} = \mathsf{Set}$. Then there seems like there's a good operation for taking two objects and producing a third:

$$\mathcal{C} \times \mathcal{C} \to \mathcal{C}, \qquad (X, Y) \mapsto X \times Y.$$

Indeed, this "direct product" does define a functor $\mathsf{Set} \times \mathsf{Set} \to \mathsf{Set}$. A pair of morphisms $f : X \to X'$ and $g : Y \to Y'$ is sent to $f \times g : X \times Y \to X' \times Y'$, where $(f \times g)(x, y) = (f(x), g(y))$.

However, this is definitely *not* associative on the nose:

$$(X \times Y) \times Z \neq X \times (Y \times Z).$$

Why is this? These are different sets! The lefthand side is the set of ordered pairs

$$((x, y), z)$$

while the righthand side is the set of ordered pairs

$$(x, (y, z)).$$

But there is an obvious bijection

$$((x, y), z) \mapsto (x, (y, z))$$

and this defines a *natural isomorphism*.

The following definition takes this into account, along with the obvious analogue for ensuring unitality:

Definition 16.5. A monoidal category $\mathcal{C}^{\otimes}$ is the data of a category $\mathcal{C}$, together with the data of:

(1) A functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$,
(2) A functor $1 : * \to \mathcal{C}$,
(3) Three natural isomorphisms between the indicated functors:
   (a) (associativity up to isomorphism)

$$\begin{tikzcd} \mathcal{C} \times \mathcal{C} \times \mathcal{C} \arrow[r, shift left, "-\otimes(-\otimes-)"] \arrow[r, shift right, "(-\otimes-)\otimes-"'] & \mathcal{C} \end{tikzcd}$$

   (b) (left unit up to isomorphism)

$$\begin{tikzcd} * \times \mathcal{C} \arrow[r, "1\times \mathrm{id}_{\mathcal{C}}"] & \mathcal{C} \times \mathcal{C} \arrow[r, "\otimes"] & \mathcal{C} \\ \mathcal{C} \arrow[u, "\cong"] \arrow[urr, "\mathrm{id}_{\mathcal{C}}"'] & & \end{tikzcd}$$

   (c) (right unit up to isomorphism)

$$\begin{tikzcd} \mathcal{C} \times * \arrow[r, "\mathrm{id}_{\mathcal{C}} \times 1"] & \mathcal{C} \times \mathcal{C} \arrow[r, "\otimes"] & \mathcal{C} \\ \mathcal{C} \arrow[u, "\cong"] \arrow[urr, "\mathrm{id}_{\mathcal{C}}"'] & & \end{tikzcd}$$

(4) These data must satisfy the pentagon axiom and the triangle axiom: The diagram

$$\begin{tikzcd} & (A \otimes (B \otimes C)) \otimes D \arrow[r] & A \otimes ((B \otimes C) \otimes D) \arrow[dd, "\mathrm{id}_A \otimes (\eta_{B,C,D})"] \\ ((A \otimes B) \otimes C) \otimes D \arrow[ur, "\eta_{A,B,C} \otimes \mathrm{id}_D"] \arrow[dr, "\eta_{A\otimes B,C,D}"'] & & \\ & (A \otimes B) \otimes (C \otimes D) \arrow[r] & A \otimes (B \otimes (C \otimes D)) \end{tikzcd}$$

must commute, and

(5) The diagram

$$\begin{tikzcd} (A \otimes 1) \otimes B \arrow[rr, "\eta_{A,1,B}"] \arrow[dr, "\mathrm{id}_A \otimes \eta"] & & A \otimes (1 \otimes B) \arrow[dl, "\eta \otimes \mathrm{id}_B"'] \\ & A \otimes B & \end{tikzcd}$$

must commute.

Remark 16.6. What's with these pentagon and triangle axioms?

Well, when you have associativity on the *nose* as we do classically, we can say things like: "The notation x^n is ambiguous, because $(x \dots x)(x \dots x) = x^a x^b = x^{a+b}$ doesn't depend on how you parenthesize the expression. However, when you have a *natural isomorphism* that you specify each time you invoke a statement like $U \otimes (U \otimes U) \cong (U \otimes U) \otimes U$, you have to be careful that each time you try to "simplify" or get rid of a tensor product, your isomorphisms are all compatible with each other. This is what the above axioms are trying to convey.

Thankfully, the four-term natural isomorphisms are the only things we need to check: See "MacLane coherence theorem" online.

In class, we also drew some pictures of Cob_1^{or} to reduce the proof that "any monoidal functor from Cob_1^{or} to $Vect$ determines a finite-dimensional vector space" to an exercise from homework one.