22 Feb. 2007

Recall $\lambda Th(\text{CommRing})$ is a typed λ-calculus — a programming language with a datatype "R" — a "commutative ring class." This typed λ-calculus generates a CCC $C_{\lambda Th(\text{CommRing})}$. Our puzzle last time was: what is a cartesian closed functor

$$F : C_{\lambda Th(\text{CommRing})} \longrightarrow \text{Set} \quad ?$$

There are tricky aspects to this puzzle... but let's start by reviewing what $\lambda Th(\text{CommRing})$ & $C_{\lambda Th(\text{CommRing})}$ are like:

$\lambda Th(\text{CommRing})$	$C_{\lambda Th(\text{CommRing})}$
<u>types</u>: 1, R, $R \times R$, $R \times \hom(R \times R, \hom(R,R))$, etc.	<u>objects</u>: 1 the terminal object, R, $R \times R$, $R \times \hom(R \times R, \hom(R,R))$, etc.
<u>terms</u>: variables of any type: $x, y, z, \ldots \in A$. Given terms $a \in A$, $b \in B$ we get $(a,b) \in A \times B$.	<u>morphisms</u>: A morphism $f : A \to B$ is an (equivalence class of) expressions $(x \in A, \varphi(x))$ where $\varphi(x) \in B$ is a term whose only free variable is x. (For example, we get $\Delta : A \to A \times A$ from $(x \in A, (x,x))$.

Given a term $a \in A \times B$ we have terms
$\pi_1(a) \in A$, $\pi_2(a) \in B$.

Given terms $f \in \hom(A,B)$ and $a \in A$, we have a term $f(a) \in B$.

We also have lots of equations between terms, e.g.

$$c \underset{X}{=} \langle \pi_1(c), \pi_2(c) \rangle$$

where $c \in A \times B$ contains free variables in X.

We also have

$$0 \in R$$

$$\underbrace{-}_{(\text{"negation"})} \in \hom(R,R)$$

(We left out negation before, but we need it since we can't say "$\forall x \in R\ \exists y \in R\ x+y=0$" in the typed λ-calculus — we don't have "$\exists$".)

$$\cdot \in \hom(R \times R, R)$$

$$1 \in R$$

We get the "addition" morphism

$$+ : R \times R \longrightarrow R$$

from

$$(x \in R \times R,\ +(\pi_1(x), \pi_2(x)))$$

or equivalently

$$(x \in R \times R,\ +(x))$$

How do we talk about 0? It should be some morphism

$$0 : 1 \longrightarrow R$$

How do we get this?

$$(x \in 1,\ 0)$$

How about

$$\cdot : R \times R \longrightarrow R \quad ?$$

This comes from

$$(x \in R \times R,\ \cdot(x))$$

Similarly

$$- : R \longrightarrow R$$

comes from

$$(x \in R,\ -(x)).$$

We have lots more, such as

$$(x \in R \times R,\ +(\cdot(\pi_1(x), \pi_1(x)), \pi_2(x)))$$

— heuristically "$(y,z) \mapsto y^2 + z$".

We also have terms like

$$x \in \hom(R,R) \longmapsto x(x(y))$$

which is of type $\hom(\hom(R,R),R)$, but $\lambda\mathrm{Th}(\mathrm{CommRing})$ doesn't use this "λ-abstraction" for its definition.

Finally, we have equations between terms, including the commutative ring axioms:

$$x+y \underset{\{x,y\}}{=\!=} y+x$$

etc.

We get an equation between two morphisms $(x \in A, \varphi(x))$ & $(y \in A, \psi(y))$ when we have $\varphi(x) \underset{\{x\}}{=\!=} \psi(x)$.

So $C_{\lambda\mathrm{Th}(\mathrm{CommRing})}$ is the CCC whose objects are generated by R and whose morphisms are generated by $+, 0, -, \cdot, 1$, with relations given by comm. ring. axioms. In short $C_{\lambda\mathrm{Th}(\mathrm{CommRing})}$ is the free CCC on a comm. ring object.

Back to our puzzle:

What's a cartesian closed functor

$$F: C_{\lambda Th(\mathrm{CommRing})} \longrightarrow \mathrm{Set}$$

Guess: it's just a commutative ring! (Namely $F(R)$.)