Ananthan Kanagasabai
January 27, 2017

Lattice $(L, \sqsubseteq, \sqcap, \sqcup)$

$\rightarrow$ reflexivity: $x \sqsubseteq x$
$\rightarrow$ transitivity: $x \sqsubseteq y \wedge y \sqsubseteq z \implies x \sqsubseteq z$
$\rightarrow$ antisymmetry: $x \sqsubseteq y \wedge y \sqsubseteq x \implies x = y$
$\rightarrow$ meet_char: $z \sqsubseteq x \wedge z \sqsubseteq y \equiv z \sqsubseteq x \sqcap y$
$\rightarrow$ join_char: $x \sqsubseteq z \wedge y \sqsubseteq z \equiv x \sqcup y \sqsubseteq z$

Defn. Duality. *The "dual" of a Lattice expression is obtained by swapping ($\sqsubseteq, \sqcap, \sqcup$) with ($\sqsupseteq, \sqcup, \sqcap$) simultaneously*

Theorem. *if p is true then so is its dual*

Examples:

Example 1:
$\mathrm{z} \sqsubseteq \mathrm{x} \wedge \mathrm{z} \sqsubseteq \mathrm{y} \equiv \mathrm{z} \sqsubseteq \mathrm{x} \sqcap \mathrm{y}$
$\mathrm{z} \sqsupseteq \mathrm{x} \wedge \mathrm{z} \sqsupseteq \mathrm{y} \equiv \mathrm{z} \sqsupseteq \mathrm{x} \sqcup \mathrm{y}$
$\mathrm{x} \sqsubseteq \mathrm{z} \wedge \mathrm{y} \sqsubseteq \mathrm{z} \equiv \mathrm{x} \sqcup \mathrm{y} \sqsubseteq \mathrm{z}$

note:
$\mathrm{L} \sqsubseteq \mathrm{R} \equiv \mathrm{R} \sqsupseteq \mathrm{L}$

Example 2:
$x \sqcup y \sqsupseteq x$
by duality we also have
$x \sqcap y \sqsubseteq x$

Constructive proof:

"$\exists x \cdot p(x)$"
(witness x, proof of p(x))
$\cong$ Algorithm with result x satisfies p

$x \leq y$
$\equiv \exists s \cdot x + s = y$

Lattices Constructively

$\rightarrow$ "x $\sqsubseteq$ y" is the TYPE OF PROOFS that x is contained ($\sqsubseteq$) in y
$\rightarrow$ $id_x; x \sqsubseteq x$
$\rightarrow$ if f: x $\sqsubseteq y$ and $g : y \sqsubseteq z$ then $f; g : x \sqsubseteq z$

Category

A category has Objects, _ $\rightarrow$ _ (arrows or morphisms), id, ;

$\rightarrow$ collection called objects
$\rightarrow$ if x,y: obj then"x $\rightarrow$ y" is a TYPE (collection)
$\rightarrow$ $id_x : x \rightarrow x$
$\rightarrow$ if f: x $\rightarrow$ y and g: y $\rightarrow$ z then f ; g : x $\rightarrow$ z
$\rightarrow$ $id_x : f = f : id_x$ for any $f : x \rightarrow x$
$\rightarrow$ $(f; g); h = f; (g; h)$ for any $x \xrightarrow{f} y \xrightarrow{g} z \xrightarrow{h} w$

examples of categories

- Programs (types of a programming language as objects, methods in that programming language as morphism)
- Relations (sets as objects, relations as morphism)
- vector (vector spaces as objects, linear transformations as morphism)

3 Mottos

Categories are:

1. graphs with monoid structures
2. coherently constructive lattices
3. typed monoids