Lumsdaine's proof of transfinite recursion

CAS Oslo 2018-10-09

well-ordered set X

(X) $S \subseteq X$ $S \neq \emptyset$ $\Rightarrow$ S has a smallest element

(X') $S \subseteq X$ S has no smallest element $\Rightarrow$ $S = \emptyset$

i.e. $(\forall x, (\forall y < x, y \notin S) \to x \notin S) \Rightarrow (\forall x, x \notin S)$

i.e. $\Pi\ P : X \to Prop,$

$(\forall x, ((\forall y < x, P\ y) \to P\ x)) \Rightarrow \forall x, P\ x$

now it looks like proof by induction.

Change Prop to Type to get "transfinite" definition by recursion:

$$\Pi\ P : X \to Type,$$
$$((\Pi x, ((\prod_{y<x} P\ y) \to P\ x)) \to \Pi x, P\ x)$$

Strategy: prove more, to ensure uniqueness of the resulting function

$$\Pi\ P : X \to Type,$$
$$\Pi\ H : (\Pi x, ((\prod_{y<x} P\ y) \to P\ x)),$$
$$\exists!\ (f : \Pi x, P\ x),$$
$$\prod_x (f\ x = H\ x\ (\lambda\ y\ l, f\ y)).$$

Now observe that $(\exists!\ \cdots)$, which is the same as $(\cdots$ is contractible$)$, is a proposition, so we could try to prove it by induction. What property of $x : X$ should we use? Try this:

$$F : \prod_{x:X} \exists!\ f : \prod_{y} y \leqslant x \to P y$$

such that

$$\prod_{y:X} \prod_{l : y \leqslant x} f\ y\ p = H\ y\ (\lambda z\ l,\ f\ z\ (l * p))$$

of type $z < x$

Such an f is called an "attempt"

Here $l * p : z \leqslant x$

Then we will define our function $f' : \prod_{x} P x$ by setting

~~$f'\ x := (\text{the}\ F\ x)\ x\ \text{refl}$~~

$$f'\ x := \text{the}\ (F\ x)\ x\ \text{"}\mathrm{refl}_x\text{"}$$

That's what Lumsdaine does, almost. He doesn't define $y \leqslant x := y < x \amalg y = x$ as one would expect.

Instead, he defines $y \leq x$ to be the type of all chains

$$y < z_1 < z_2 < \cdots < z_{n-1} < x \quad ,$$

so n is the number of "$<$"s in the chain.

One reason this makes the proof easier is that associativity

$$(l * p) * l' = l * (p * l')$$

$$l : w < y$$
$$p : y \leq x$$
$$l' : x < t$$

becomes true by definition. Then, amazingly, we no longer have to assume X is a well-ordered set ~~and~~ or that $y < x$ is a proposition, is transitive, or anything.

Are there other reasons the proof is easier? Maybe not.

Lumsdaine's definition of ~~$y < x$~~ : $x \leq y$:

by induction

- nil x : $x \leq x$
- cons x y z $(p : x \leq y)$ $(l : y < z)$: $x \leq z$

remark : ~~$x = y$~~ $x = y \to x \leq y$

~~Thus~~ UniMath bans new inductive definitions, so ~~last~~ I tried recasting the proof this way :

Define $x \underset{n}{\leq} y$ for $n : \mathbb{N}$ by

- $x \underset{0}{\leq} y := x = y$
- $x \underset{n+1}{\leq} y := \sum_{s,t} x \underset{n}{\leq} s \times s < t \times t = y$

Thus an element of $x \underset{n}{\leq} y$ is a chain.

$$x = s_1 < t_1 = s_2 < t_2 = \cdots = s_n < t_n = y$$

And we define $x \leq y := \sum_n x \underset{n}{\leq} y$.

That worked, too! (yesterday)

Now admire the code.

What makes the proof work ??