Lecture 22

Equational Reasoning about Recursive Functions over $\mathbb{N}$, see notes from R.L. Goodstein *Recursive Number Theory*, included with Lecture 20.

Recall lecture on Wednesday by Vladamir Voevodsky, "Creating Mathematics with Proof Assistants", 4:30-5:30pm Malott 406, about using proof assistants in mathematics.

Topics

1. Motivation for equational reasoning – from $t_1 = t_2 \in \mathbb{N}$ to $t_1 \sim t_2 \in$ Base and $t_1 \sim t_2 \in$ Base.
2. Recursive Number Theory and Logic.
3. Goodstein's rules, relationship to Kleene's account of primitive recursive functions, logic free.
4. Sample proof.
5. Exercise for PS3.

---

1. **Why is equationsal reasoning important?**

   The standard presentation of recursive functions is equation based. Also note Barendregt's account of the lambda calculus is based on equational logic with distinct equalities, $\equiv$, $=_\alpha$, $=_\beta$.

   But presenting computation rules as equations is a bit misleading. The $=_\beta$ relation is important because of a directed *reduction rule* $ap(\lambda(x.b); a) \downarrow b[a/x]$, sometimes expressed as $ap(\lambda(x.b); a) =_\beta b[a/x]$.

   Equational reasoning is a *bridge to logic*. It gives rise to a very simple logic. We can use it to prove properties of functions. We can use it to investigate topics in the logic of programs. We can see how logic and computation are related. This study provides a *semantic basis for type theory* as in Coq.

2. **Recursive Number Theory and Logic**

   In Lecture 20 we saw how to *code* some logic operators as primitive recursive functions and how to reduce "truth" to the claims that an expression equals 0. We will look at these definitions again and use them to express facts about numbers, e.g. to do number theory.

First we need rules for proving equations. Once we have that machinery, we can use the coding of logical operations, &, $\vee$, $\sim$, $\Rightarrow$, $\forall$, $\exists$, to state and prove results from number theory. For example, we can easily express the idea that any natural number n is prime or not prime.

Using the coding we write $prime(n) \vee \sim prime(n)$. Proving that requires that we can assert $prime(n)$ and $\sim prime(n)$. To assert $prime(n)$, we say $prime(n) = 0$ which is $|\, prime(n), 0\,| = 0$. To assert $\sim prime(n)$ is to say $(1 \dot{-} |\, prime(n), 0\,|) = 0$. So $prime(n) \vee \sim prime(n)$ is $|\, prime(n), 0\,| \cdot |1 \dot{-} |\, prime(n), 0\,|| = 0$.

These codings are not so intuitive, and we can easily think in terms of "truth values" as Booleans, say tt and ff which are coded as 0 and non-zero.

3. **Goodstein's rules for reasoning about equations**. (pg 27)

A proof is a sequence of equations. Here are the rules for writing a correct sequence. We call these conditions on equations.

C1. An equation that is part of a recursive definition of a function is allowed.

C2. An equation of the form $F = F$ is allowed.

C3. Any previously proved equation $F = G$ is allowed.

C4. If $F = G$ is an equation of a proof, then any occurrence of F in an equation can be replaced by G, at one or more places.

C5. Any variable can be replaced by a numeral or a new variable in all occurrences in some equation of a proof. The numerals are 0, $S(0)$, $S(S(0))$, ...

C6. If F and G satisfy the same defining equations, then we can add the equation $F = G$.

For example:

$add(0, y) = y$	$plus(0, y) = y$
$add(S(n), y) = S(add(n, y))$	$plus(S(n), y) = S(plus(n, y))$

We can claim $add = plus$.

This says that recursion defines unique functions.

Here is a simple proof from Goodstein that addition is commutative, i.e. $x + y = y + x$.

As an exercise, prove that multipliation is commutative, i.e. $x * y = y * x$.

Theorem *$x + y = y + x$ since both satisfy the same introductory equations $f(x, 0) = x$ and $f(x, S(y)) = S(f(x, y))$.*

We can show that $0 + x = x$ by showing that this function satisfies the same defining equaton as the identity functions, namely $id(0) = 0$ and $id(S(x)) = S(id(x))$.

The harder step is showing $S(x) + y = S(x + y)$.

Here are the required equations with justifications.

Proof

1.	$x + 0 = x$	Definition of $+$.
2.	$S(x) + 0 = S(x)$	Substitute $S(x)$ for x eqn 1.
3.	$S(x) = S(x)$	Axiom about equality (C2.).
4.	$S(x + 0) = S(x)$	Substitute $x + 0$ for x in line 3, by line 1.
5.	$x + S(y) = S(x + y)$	By definition (recursive) of $+$.
6.	$S(x) + S(y) = S(S(x) + y)$	Substitute $S(x)$ for x in 5.
7.	$S(x + S(y)) = S(x + S(y))$	By Axiom about equality (C2.).
8.	$S(x + S(y)) = S(S(x + y))$	From 5 and 7 (apply to both sides).
9.	$S(x) + y = S(x + y)$	By C6 (same defining equations), using 2, 4, 6, and 8. $f(x, 0) = S(x)$ $f(x, S(y)) = S(f(x, y))$
QED		
10.	$S(y) + x = S(y + x)$ is also proved	Using x for y and y for x in 9.