Fundamentele Informatica 3

voorjaar 2014

http://www.liacs.nl/home/rvvliet/fi3/

Rudy van Vliet
kamer 124 Snellius, tel. 071-527 5777
rvvliet(at)liacs(dot)nl

college 12, 28 april 2014

10. Computable Functions
10.1. Primitive Recursive Functions

Huiswerkopgave 3, inleverdatum 29 april 2014, 13:45 uur

10. Computable Functions

10.1. Primitive Recursive Functions

Definition 10.1. Initial Functions

The initial functions are the following:

1. *Constant* functions: For each $k \geq 0$ and each $a \geq 0$, the constant function $C_a^k : \mathbb{N}^k \to \mathbb{N}$ is defined by the formula

$$C_a^k(X) = a \quad \text{for every } X \in \mathbb{N}^k$$

Definition 10.1. Initial Functions

The initial functions are the following:

1. *Constant* functions: For each $k \geq 0$ and each $a \geq 0$, the constant function $C_a^k : \mathbb{N}^k \to \mathbb{N}$ is defined by the formula

$$C_a^k(X) = a \quad \text{for every } X \in \mathbb{N}^k$$

2. The *successor* function $s : \mathbb{N} \to \mathbb{N}$ is defined by the formula

$$s(x) = x + 1$$

Definition 10.1. Initial Functions

The initial functions are the following:

1. *Constant* functions: For each $k \geq 0$ and each $a \geq 0$, the constant function $C_a^k : \mathbb{N}^k \to \mathbb{N}$ is defined by the formula

$$C_a^k(X) = a \quad \text{for every } X \in \mathbb{N}^k$$

2. The *successor* function $s : \mathbb{N} \to \mathbb{N}$ is defined by the formula

$$s(x) = x + 1$$

3. *Projection* functions: For each $k \geq 1$ and each i with $1 \leq i \leq k$, the projection function $p_i^k : \mathbb{N}^k \to \mathbb{N}$ is defined by the formula

$$p_i^k(x_1, x_2, \ldots, x_k) = x_i$$

Definition 10.2. The Operations of Composition and Primitive Recursion

1. Suppose f is a partial function from $\mathbb{N}^k$ to $\mathbb{N}$, and for each i with $1 \leq i \leq k$, g_i is a partial function from $\mathbb{N}^m$ to $\mathbb{N}$.
   The partial function obtained from f and $g_1, g_2, \ldots, g_k$ by composition is the partial function h from $\mathbb{N}^m$ to $\mathbb{N}$ defined by the formula

   $$h(X) = f(g_1(X), g_2(X), \ldots, g_k(X)) \text{ for every } X \in \mathbb{N}^m$$

Definition 10.2. The Operations of Composition and Primitive Recursion (continued)

2. Suppose $n \geq 0$ and g and h are functions of n and $n+2$ variables, respectively. (By "a function of 0 variables," we mean simply a constant.)
   The function obtained from g and h by the operation of *primitive recursion* is the function $f : \mathbb{N}^{n+1} \to \mathbb{N}$ defined by the formulas

$$\begin{aligned} f(X,0) &= g(X) \\ f(X,k+1) &= h(X,k,f(X,k)) \end{aligned}$$

   for every $X \in \mathbb{N}^n$ and every $k \geq 0$.

Example 10.5. Addition, Multiplication and Subtraction

$$\mathit{Add}(x, y) = x + y$$

Definition 10.3. Primitive Recursive Functions

The set *PR* of *primitive recursive* functions is defined as follows.

1. All initial functions are elements of *PR*.
2. For every $k \geq 0$ and $m \geq 0$, if $f : \mathbb{N}^k \to \mathbb{N}$ and $g_1, g_2, \ldots, g_k : \mathbb{N}^m \to \mathbb{N}$ are elements of *PR*, then the function $f(g_1, g_2, \ldots, g_k)$ obtained from f and $g_1, g_2, \ldots, g_k$ by composition is an element of *PR*.
3. For every $n \geq 0$, every function $g : \mathbb{N}^n \to \mathbb{N}$ in *PR*, and every function $h : \mathbb{N}^{n+2} \to \mathbb{N}$ in *PR*, the function $f : \mathbb{N}^{n+1} \to \mathbb{N}$ obtained from g and h by primitive recursion is in *PR*.

In other words, the set *PR* is the smallest set of functions that contains all the initial functions and is closed under the operations of composition and primitive recursion.

Theorem 10.4.

Every primitive recursive function is total and computable.

PR:
total and computable

Turing-computable functions:
not necessarily total

Example 10.5. Addition, Multiplication and Subtraction

$$\mathit{Mult}(x, y) = x * y$$

Example 10.5. Addition, Multiplication and Subtraction

$$\mathit{Sub}(x, y) = \begin{cases} x - y & \text{if } x \geq y \\ 0 & \text{otherwise} \end{cases}$$

$x \dot{-} y$

n*-place predicate* P is function from $\mathbb{N}^n$ to $\{\text{true}, \text{false}\}$

characteristic function χ_P defined by

$$\chi_P(X) = \begin{cases} 1 & \text{if } P(X) \text{ is true} \\ 0 & \text{if } P(X) \text{ is false} \end{cases}$$

We say P is primitive recursive...

Theorem 10.6.

The two-place predicates *LT*, *EQ*, *GT*, *LE*, *GE*, and *NE* are primitive recursive.
(*LT* stands for “less than,” and the other five have similarly intuitive abbreviations.)
If P and Q are any primitive recursive n-place predicates, then $P \wedge Q$, $P \vee Q$ and $\neg P$ are primitive recursive.

Proof. . .

Theorem 10.7.

Suppose $f_1, f_2, \ldots, f_k$ are primitive recursive functions from $\mathbb{N}^n$ to $\mathbb{N}$,
$P_1, P_2, \ldots, P_k$ are primitive recursive n-place predicates,
and for every $X \in \mathbb{N}^n$,
exactly one of the conditions $P_1(X), P_2(X), \ldots, P_k(X)$ is true.
Then the function $f : \mathbb{N}^n \to \mathbb{N}$ defined by

$$f(X) = \begin{cases} f_1(X) & \text{if } P_1(X) \text{ is true} \\ f_2(X) & \text{if } P_2(X) \text{ is true} \\ \ldots & \\ f_k(X) & \text{if } P_k(X) \text{ is true} \end{cases}$$

is primitive recursive.

Proof...

Example 10.8. The *Mod* and *Div* Functions