Advanced Topics in Theoretical Computer Science

Part 3: Recursive Functions (2)

29.11.2017

Viorica Sofronie-Stokkermans

Universität Koblenz-Landau

e-mail: sofronie@uni-koblenz.de

Exams

1. Termin (Klausur)

Wednesday, 28.02.2018, 10:00-12:00

2. Termin (Nachklausur)

Before the start of the lectures in the summer semester

Contents

- Recapitulation: Turing machines and Turing computability
- Register machines (LOOP, WHILE, GOTO)
- Recursive functions
- The Church-Turing Thesis
- Computability and (Un-)decidability
- Complexity
- Other computation models: e.g. Büchi Automata, λ-calculus

3. Recursive functions

- Introduction/Motivation
- Primitive recursive functions $\mapsto \mathcal{P}$
- $\mathcal{P} = \text{LOOP}$
- μ-recursive functions $\mapsto F_\mu$
- $F_\mu = \text{WHILE}$
- Summary

Recursive functions: Atomic functions

The following functions are primitive recursive and μ-recursive:

The constant null

$$0 : \mathbb{N}^0 \to \mathbb{N} \text{ with } 0() = 0$$

Successor function

$$+1 : \mathbb{N}^1 \to \mathbb{N} \text{ with } +1(n) = n + 1 \text{ for all } n \in \mathbb{N}$$

Projection function

$$\pi_i^k : \mathbb{N}^k \to \mathbb{N} \text{ with } \pi_i^k(n_1, \ldots, n_k) = n_i$$

Recursive functions

Notation:

We will write $\mathbf{n}$ for the tuple $(n_1, \ldots, n_k)$, $k \geq 0$.

Recursive functions: Composition

Composition:

If the functions: $g : \mathbb{N}^r \to \mathbb{N}$ $\quad r \geq 1$

$h_1 : \mathbb{N}^k \to \mathbb{N}, \dots, h_r : \mathbb{N}^k \to \mathbb{N}$ $\quad k \geq 0$

are primitive recursive resp. μ-recursive, then

$$f : \mathbb{N}^k \to \mathbb{N}$$

defined for every $\mathbf{n} \in \mathbb{N}^k$ by:

$$f(\mathbf{n}) = g(h_1(\mathbf{n}), \dots, h_r(\mathbf{n}))$$

is also primitive recursive resp. μ-recursive.

Notation without arguments: $f = g \circ (h_1, \dots, h_r)$

Primitive recursive functions

Primitive recursion

If the functions

$$g : \mathbb{N}^k \to \mathbb{N} \qquad (k \geq 0)$$
$$h : \mathbb{N}^{k+2} \to \mathbb{N}$$

are primitive recursive,

then the function

$$f : \mathbb{N}^{k+1} \to \mathbb{N} \text{ with } \quad \begin{aligned} f(\mathbf{n}, 0) &= g(\mathbf{n}) \\ f(\mathbf{n}, m+1) &= h(\mathbf{n}, m, f(\mathbf{n}, m)) \end{aligned}$$

is also primitive recursive.

Primitive recursive functions

Primitive recursion

If the functions

$$g : \mathbb{N}^k \to \mathbb{N} \qquad (k \geq 0)$$
$$h : \mathbb{N}^{k+2} \to \mathbb{N}$$

are primitive recursive,

then the function

$$f : \mathbb{N}^{k+1} \to \mathbb{N} \text{ with } \quad f(\mathbf{n}, 0) = g(\mathbf{n})$$
$$f(\mathbf{n}, m+1) = h(\mathbf{n}, m, f(\mathbf{n}, m))$$

is also primitive recursive.

Notation without arguments: $f = \mathcal{PR}[g, h]$

Primitive recursive functions

Definition (Primitive recursive functions)

- **Atomic functions:** The functions
  - Null 0
  - Successor $+1$
  - Projection π_i^k $(1 \leq i \leq k)$

  are primitive recursive.
- **Composition:** The functions obtained by composition from primitive recursive functions are primitive recursive.
- **Primitive recursion:** The functions obtained by primitive recursion from primitive recursive functions are primitive recursive.

Primitive recursive functions

Definition (Primitive recursive functions)

- **Atomic functions:** The functions
  - Null 0
  - Successor $+1$
  - Projection π_i^k $(1 \leq i \leq k)$

  are primitive recursive.
- **Composition:** The functions obtained by composition from primitive recursive functions are primitive recursive.
- **Primitive recursion:** The functions obtained by primitive recursion from primitive recursive functions are primitive recursive.

Primitive recursive functions

Definition (Primitive recursive functions)

- **Atomic functions:** The functions
  - Null 0
  - Successor $+1$
  - Projection π_i^k $(1 \leq i \leq k)$

  are primitive recursive.
- **Composition:** The functions obtained by composition from primitive recursive functions are primitive recursive.
- **Primitive recursion:** The functions obtained by primitive recursion from primitive recursive functions are primitive recursive.

Notation: $\mathcal{P} =$ The set of all primitive recursive functions

Arithmetical functions: definitions

$f(n) = n + c$, **for** $c \in \mathbb{N}, c > 0$

$$f = \underbrace{(+1) \circ \cdots \circ (+1)}_{c \text{ times}}$$

$f(n) = n$

$$f = \pi_1^1$$

$f(n, m) = n + m$

$$f = \mathcal{PR}[\pi_1^1, (+1) \circ \pi_3^3]$$

$f(n) = n - 1$

$$f = \mathcal{PR}[0, \pi_1^2]$$

$f(n, m) = n - m$

$$f = \mathcal{PR}[\pi_1^1, (-1) \circ \pi_3^3]$$

$f(n, m) = n * m$

$$f = \mathcal{PR}[0, + \circ (\pi_3^3, \pi_1^3)]$$

Defining new primitive recursive functions

Re-ordering/Omitting/Repeating Arguments

Lemma The set of primitive recursive functions is closed under:

- Re-ordering
- Omitting
- Repeating

of arguments when composing functions.

Additional Arguments

Lemma. Assume $f : \mathbb{N}^k \to \mathbb{N}$ is primitive recursive.
Then, for every $p \in \mathbb{N}$, the function $f' : \mathbb{N}^k \times \mathbb{N}^p \to \mathbb{N}$ defined for every $\mathbf{n} \in \mathbb{N}^k$ and every $\mathbf{m} \in \mathbb{N}^p$ by:

$$f'(\mathbf{n}, \mathbf{m}) = f(\mathbf{n})$$

is primitive recursive.

Case distinction

Lemma (Case distinction is primitive recursive)

If
- g_i, h_i $(1 \leq i \leq r)$ are primitive recursive functions, and
- for every n there exists a unique i with $h_i(n) = 0$

then the function f defined by:

$$f(n) = \begin{cases} g_1(n) & \text{if } h_1(n) = 0 \\ \dots & \\ g_r(n) & \text{if } h_r(n) = 0 \end{cases}$$

is primitive recursive.

Case distinction

Lemma (Case distinction is primitive recursive)

If
- g_i, h_i $(1 \leq i \leq r)$ are primitive recursive functions, and
- for every n there exists a unique i with $h_i(n) = 0$

then the function f defined by:

$$f(n) = \begin{cases} g_1(n) & \text{if } h_1(n) = 0 \\ \dots & \\ g_r(n) & \text{if } h_r(n) = 0 \end{cases}$$

is primitive recursive.

Proof: $f(n) = g_1(n) * (1 - h_1(n)) + \dots + g_r(n) * (1 - h_r(n))$

Sums and products

Theorem

If $g : \mathbb{N}^k \times \mathbb{N} \to \mathbb{N}$ is a primitive recursive function then the following functions $f_1, f_2 : \mathbb{N}^k \times \mathbb{N} \to \mathbb{N}$ are also primitive recursive:

$$f_1(\mathbf{n}, m) = \begin{cases} 0 & \text{if } m = 0 \\ \sum_{i<m} g(\mathbf{n}, i) & \text{if } m > 0 \end{cases}$$

$$f_2(\mathbf{n}, m) = \begin{cases} 1 & \text{if } m = 0 \\ \prod_{i<m} g(\mathbf{n}, i) & \text{if } m > 0 \end{cases}$$

Sums and products

Theorem

If $g : \mathbb{N}^k \times \mathbb{N} \to \mathbb{N}$ is a primitive recursive function then the following functions $f_1, f_2 : \mathbb{N}^k \times \mathbb{N} \to \mathbb{N}$ are also primitive recursive:

$$f_1(\mathbf{n}, m) = \begin{cases} 0 & \text{if } m = 0 \\ \sum_{i<m} g(\mathbf{n}, i) & \text{if } m > 0 \end{cases}$$

$$f_2(\mathbf{n}, m) = \begin{cases} 1 & \text{if } m = 0 \\ \prod_{i<m} g(\mathbf{n}, i) & \text{if } m > 0 \end{cases}$$

Proof: f_1 and f_2 can be written using primitive recursion and case distinction:

$$f_1(\mathbf{n}, 0) = 0$$

$$f_1(\mathbf{n}, m+1) = f_1(\mathbf{n}, m) + g(\mathbf{n}, m)$$

Sums and products

Theorem

If $g : \mathbb{N}^k \times \mathbb{N} \to \mathbb{N}$ is a primitive recursive function then the following functions $f_1, f_2 : \mathbb{N}^k \times \mathbb{N} \to \mathbb{N}$ are also primitive recursive:

$$f_1(\mathbf{n}, m) = \begin{cases} 0 & \text{if } m = 0 \\ \sum_{i<m} g(\mathbf{n}, i) & \text{if } m > 0 \end{cases}$$

$$f_2(\mathbf{n}, m) = \begin{cases} 1 & \text{if } m = 0 \\ \prod_{i<m} g(\mathbf{n}, i) & \text{if } m > 0 \end{cases}$$

Proof: f_1 and f_2 can be written using primitive recursion and case distinction:

$$f_1(\mathbf{n}, 0) = 0$$
$$f_1(\mathbf{n}, m+1) = f_1(\mathbf{n}, m) + g(\mathbf{n}, m)$$

$$f_2(\mathbf{n}, 0) = 1$$
$$f_2(\mathbf{n}, m+1) = f_2(\mathbf{n}, m) * g(\mathbf{n}, m)$$

Bounded μ operator

Definition.

Let $g : \mathbb{N}^{k+1} \to \mathbb{N}$ be a function.

The bounded μ operator is defined as follows:

$$\mu_{i<m}\ i\ (g(\mathbf{n}, i) = 0) := \begin{cases} i_0 & \text{if } g(\mathbf{n}, i_0) = 0 \\ & \text{and for all } j < i_0 \;\; g(\mathbf{n}, j) \neq 0 \\ 0 & \text{if } g(\mathbf{n}, j) \neq 0 \text{ for all } 0 \leq j < m \\ & \text{or } \; m = 0 \end{cases}$$

$\mu_{i<m}\ i\ (g(\mathbf{n}, i) = 0)$ is the smallest $i < m$ such that $g(\mathbf{n}, i) = 0$

Bounded μ operator

Theorem.
If $g : \mathbb{N}^{k+1} \to \mathbb{N}$ is a primitive recursive function
then the function $f : \mathbb{N}^{k+1} \to \mathbb{N}$ defined by:

$$f(\mathbf{n}, m) = \mu_{i<m}\ i\ (g(\mathbf{n}, i) = 0)$$

is also primitive recursive

Bounded μ operator

Theorem.
If $g : \mathbb{N}^{k+1} \to \mathbb{N}$ is a primitive recursive function then the function $f : \mathbb{N}^{k+1} \to \mathbb{N}$ defined by:

$$f(\mathbf{n}, m) = \mu_{i<m}\ i\ (g(\mathbf{n}, i) = 0)$$

is also primitive recursive

Proof: We can define f as follows:

$$\begin{aligned} f(\mathbf{n}, 0) &= 0 \\ f(\mathbf{n}, m+1) &= \begin{cases} 0 & \text{if } m = 0 \\ m & \text{if } g(\mathbf{n}, m) = f(\mathbf{n}, m) = 0 \wedge g(\mathbf{n}, 0) \neq 0 \wedge m > 0 \\ & \\ f(\mathbf{n}, m) & \text{otherwise} \end{cases} \end{aligned}$$

Bounded μ operator

Theorem.
If $g : \mathbb{N}^{k+1} \rightarrow \mathbb{N}$ is a primitive recursive function then the function $f : \mathbb{N}^{k+1} \rightarrow \mathbb{N}$ defined by:

$$f(\mathbf{n}, m) = \mu_{i<m}\ i\ (g(\mathbf{n}, i) = 0)$$

is also primitive recursive

Proof: We can define f as follows:

$$\begin{aligned} f(\mathbf{n}, 0) &= 0 \\ f(\mathbf{n}, m+1) &= \begin{cases} 0 & \text{if } m = 0 \\ m & \text{if } g(\mathbf{n}, m) = f(\mathbf{n}, m) = 0 \wedge g(\mathbf{n}, 0) \neq 0 \wedge m > 0 \\ & \\ f(\mathbf{n}, m) & \text{otherwise} \end{cases} \end{aligned}$$

Bounded μ operator

Theorem.
If $g : \mathbb{N}^{k+1} \to \mathbb{N}$ is a primitive recursive function then the function $f : \mathbb{N}^{k+1} \to \mathbb{N}$ defined by:

$$f(\mathbf{n}, m) = \mu_{i<m}\ i\ (g(\mathbf{n}, i) = 0)$$

is also primitive recursive

Proof: We can define f as follows:

$$\begin{array}{rcl} f(\mathbf{n}, 0) & = & 0 \\ f(\mathbf{n}, m+1) & = & \begin{cases} 0 & \text{if } m = 0 \\ m & \text{if } g(\mathbf{n}, m) = f(\mathbf{n}, m) = 0 \wedge g(\mathbf{n}, 0) \neq 0 \wedge m > 0 \\ & \text{i.e. if } g(\mathbf{n}, m) + f(\mathbf{n}, m) + (1 - g(\mathbf{n}, 0)) + (1 - m) = 0 \\ f(\mathbf{n}, m) & \text{otherwise} \end{cases} \end{array}$$

Prime number functions

Theorem: The following functions are primitive recursive:

(1) The Boolean function $\mid : \mathbb{N} \times \mathbb{N} \to \{0, 1\}$ defined by:

$$|(n, m) = \begin{cases} 1 & \text{if } n \text{ divides } m \\ 0 & \text{otherwise} \end{cases}$$

(2) The Boolean function prime : $\mathbb{N} \to \{0, 1\}$ defined by:

$$\text{prime}(n) = \begin{cases} 1 & \text{if } n \text{ prime} \\ 0 & \text{otherwise} \end{cases}$$

(3) The function $p : \mathbb{N} \to \mathbb{N}$ defined by: $p(n) = p_n$, the n-th prime number.

(4) The function $D : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ defined by: $D(n, i) = k$ iff k is the power of the i-th prime number in the prime number decomposition of n.

$$D(n, i) = \max(\{j \mid n \bmod p(i)^j = 0\})$$

Prime number functions

Proof:

(1) $\mid : \mathbb{N} \times \mathbb{N} \to \{0, 1\}$ defined by:

$$\mid(n, m) = \begin{cases} 1 & \text{if } n \text{ divides } m \\ 0 & \text{otherwise} \end{cases}$$

Prime number functions

Proof:

(1) $| : \mathbb{N} \times \mathbb{N} \to \{0, 1\}$ defined by:

$$|(n, m) = \begin{cases} 1 & \text{if } n \text{ divides } m \\ 0 & \text{otherwise} \end{cases}$$

$|(n, m) = 1$ iff $\exists z(n * z = m)$ iff $\prod_{z \leq m}(n * z - m) + (m - n * z) = 0$.

Prime number functions

Proof:

(1) $| : \mathbb{N} \times \mathbb{N} \to \{0, 1\}$ defined by:

$$|(n, m) = \begin{cases} 1 & \text{if } n \text{ divides } m \\ 0 & \text{otherwise} \end{cases}$$

$|(n, m) = 1$ iff $\exists z(n * z = m)$ iff $\prod_{z \leq m}(n * z - m) + (m - n * z) = 0.$

$|(n, m) = 1 - \prod_{z \leq m}(n * z - m) + (m - n * z)$

Prime number functions

Proof:

(2) prime : $\mathbb{N} \to \{0, 1\}$ defined by:

$$\text{prime}(n) = \begin{cases} 1 & \text{if } n \text{ prime} \\ 0 & \text{otherwise} \end{cases}$$

Prime number functions

Proof:

(2) prime : $\mathbb{N} \to \{0, 1\}$ defined by:

$$\text{prime}(n) = \begin{cases} 1 & \text{if } n \text{ prime} \\ 0 & \text{otherwise} \end{cases}$$

$\text{prime}(n) = 1$ iff $(n \geq 2$ and $\forall y < n(y = 0 \vee y = 1 \vee |(y, n) = 0)$

$\text{prime}(n) = 1 - ((2 - n) + \sum_{y<n}(|(y, n) * y * ((y - 1) + (1 - y))))$

Prime number functions

Proof:

(3) The function $p : \mathbb{N} \to \mathbb{N}$ defined by: $p(n) = p_n$, the n-th prime number.

$p(0) = 0$ and $p(1) = 2$.

$p(n+1)$ is the smallest number i which is larger than $p(n)$ and is prime.

Prime number functions

Proof:

(3) The function $p : \mathbb{N} \to \mathbb{N}$ defined by: $p(n) = p_n$, the n-th prime number.

$p(0) = 0$ and $p(1) = 2$.

$p(n+1)$ is the smallest number i which is larger than $p(n)$ and is prime.

We also have an upper bound for the number i.

Recall the proof of the fact that the set of prime numbers is infinite.

$$i \leq p(n)! + 1$$

$$p(n+1) = \mu_{i \leq p(n)!+1}\ i\ [((1 - \text{prime}(i)) + ((p(n)+1) - i)) = 0]$$

Prime number functions

Proof:

(4) $D : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ defined by: $D(n, i) = k$ iff k is the power of the i-th prime number in the prime number decomposition of n.

$$D(n, i) = \max(\{j \mid n \bmod p(i)^j = 0\})$$

$D(0, i) := 0;$

$D(n, i) = \min(\{j \leq n \mid \quad |(p(i)^{j+1}, n) = 0\})$

$D(n, i) = \mu_{j \leq n}\, j\ (|(p(i)^{j+1}, n) = 0)$