FORMALIZED MATHEMATICS
Vol.1, No.**2**, March–April 1990
Université Catholique de Louvain

Recursive Definitions

Krzysztof Hryniewiecki
Warsaw University

Summary. The text contains some schemes which allow elimination of defintions by recursion.

MML Identifier: RECDEF_1.

The papers [5], [1], [3], [2], and [4] provide the notation and terminology for this paper. We follow a convention: n, m, k will denote natural numbers and x, y, z, y_1, y_2 will be arbitrary. The arguments of the notions defined below are the following: D which is a non-empty set; p which is a function from $\mathbb{N}$ into D; n which is an element of $\mathbb{N}$. Then $p(n)$ is an element of D.

The arguments of the notions defined below are the following: p which is a function from $\mathbb{N}$ into $\mathbb{N}$; n which is an element of $\mathbb{N}$. Then $p(n)$ is a natural number.

In the article we present several logical schemes. The scheme *RecEx* concerns a constant $\mathcal{A}$ and a ternary predicate $\mathcal{P}$ and states that:

there exists f being a function such that $\operatorname{dom} f = \mathbb{N}$ and $f(0) = \mathcal{A}$ and for every element n of $\mathbb{N}$ holds $\mathcal{P}[n, f(n), f(n+1)]$

provided the parameters satisfy the following conditions:

- for every natural number n for arbitrary x there exists y being any such that $\mathcal{P}[n, x, y]$,
- for every natural number n for arbitrary x, y_1, y_2 such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$.

The scheme *RecExD* deals with a constant $\mathcal{A}$ that is a non-empty set, a constant $\mathcal{B}$ that is an element of $\mathcal{A}$ and a ternary predicate $\mathcal{P}$ and states that:

there exists f being a function from $\mathbb{N}$ into $\mathcal{A}$ such that $f(0) = \mathcal{B}$ and for every element n of $\mathbb{N}$ holds $\mathcal{P}[n, f(n), f(n+1)]$

provided the parameters satisfy the following conditions:

- for every natural number n for every element x of $\mathcal{A}$ there exists y being an element of $\mathcal{A}$ such that $\mathcal{P}[n, x, y]$,
- for every natural number n for all elements x, y_1, y_2 of $\mathcal{A}$ such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$.

ISSN 0777–4028

The scheme *LambdaRecEx* concerns a constant $\mathcal{A}$ and a binary functor $\mathcal{F}$ and states that:

there exists f being a function such that $\mathrm{dom}\, f = \mathbb{N}$ and $f(0) = \mathcal{A}$ and for every element n of $\mathbb{N}$ for arbitrary x such that $x = f(n)$ holds $f(n+1) = \mathcal{F}(n, x)$

for all values of the parameters.

The scheme *LambdaRecExD* concerns a constant $\mathcal{A}$ that is a non-empty set, a constant $\mathcal{B}$ that is an element of $\mathcal{A}$ and a binary functor $\mathcal{F}$ yielding an element of $\mathcal{A}$ and states that:

there exists f being a function from $\mathbb{N}$ into $\mathcal{A}$ such that $f(0) = \mathcal{B}$ and for every element n of $\mathbb{N}$ for every element x of $\mathcal{A}$ such that $x = f(n)$ holds $f(n + 1) = \mathcal{F}(n, x)$

for all values of the parameters.

The scheme *RecFuncExR* concerns a constant $\mathcal{A}$ that is a real number and a binary functor $\mathcal{F}$ yielding a real number and states that:

there exists f being a function from $\mathbb{N}$ into $\mathbb{R}$ such that $f(0) = \mathcal{A}$ and for every natural number n for every real number x such that $x = f(n)$ holds $f(n + 1) = \mathcal{F}(n, x)$

for all values of the parameters.

The scheme *RecExN* deals with a constant $\mathcal{A}$ that is a natural number and a binary functor $\mathcal{F}$ yielding a natural number and states that:

there exists f being a function from $\mathbb{N}$ into $\mathbb{N}$ such that $f(0) = \mathcal{A}$ and for every natural number n for every natural number x such that $x = f(n)$ holds $f(n + 1) = \mathcal{F}(n, x)$

for all values of the parameters.

The scheme *FinRecEx* deals with a constant $\mathcal{A}$, a constant $\mathcal{B}$ that is a natural number and a ternary predicate $\mathcal{P}$ and states that:

there exists p being a finite sequence such that $\mathrm{len}\, p = \mathcal{B}$ but $p(1) = \mathcal{A}$ or $\mathcal{B} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ holds $\mathcal{P}[n, p(n), p(n + 1)]$

provided the parameters satisfy the following conditions:

- for every natural number n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ for arbitrary x there exists y being any such that $\mathcal{P}[n, x, y]$,
- for every natural number n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ for arbitrary x, y_1, y_2 such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$.

The scheme *FinRecExD* deals with a constant $\mathcal{A}$ that is a non-empty set, a constant $\mathcal{B}$ that is an element of $\mathcal{A}$, a constant $\mathcal{C}$ that is a natural number and a ternary predicate $\mathcal{P}$ and states that:

there exists p being a finite sequence of elements of $\mathcal{A}$ such that $\mathrm{len}\, p = \mathcal{C}$ but $p(1) = \mathcal{B}$ or $\mathcal{C} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{C} - 1$ holds $\mathcal{P}[n, p(n), p(n + 1)]$

provided the parameters satisfy the following conditions:

- for every natural number n such that $1 \leq n$ and $n \leq \mathcal{C} - 1$ for every element x of $\mathcal{A}$ there exists y being an element of $\mathcal{A}$ such that $\mathcal{P}[n, x, y]$,
- for every natural number n such that $1 \leq n$ and $n \leq \mathcal{C} - 1$ for all elements x, y_1, y_2 of $\mathcal{A}$ such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds

$y_1 = y_2$.

The scheme *FinRecExR* deals with a constant $\mathcal{A}$ that is a real number, a constant $\mathcal{B}$ that is a natural number and a ternary predicate $\mathcal{P}$ and states that:

there exists p being a finite sequence of elements of $\mathbb{R}$ such that $\operatorname{len} p = \mathcal{B}$ but $p(1) = \mathcal{A}$ or $\mathcal{B} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ holds $\mathcal{P}[n, p(n), p(n+1)]$

provided the parameters satisfy the following conditions:

- for every natural number n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ for every real number x there exists y being a real number such that $\mathcal{P}[n, x, y]$,
- for every natural number n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ for all real numbers x, y_1, y_2 such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$.

The scheme *FinRecExN* deals with a constant $\mathcal{A}$ that is a natural number, a constant $\mathcal{B}$ that is a natural number and a ternary predicate $\mathcal{P}$ and states that:

there exists p being a finite sequence of elements of $\mathbb{N}$ such that $\operatorname{len} p = \mathcal{B}$ but $p(1) = \mathcal{A}$ or $\mathcal{B} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ holds $\mathcal{P}[n, p(n), p(n+1)]$

provided the parameters satisfy the following conditions:

- for every natural number n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ for every natural number x there exists y being a natural number such that $\mathcal{P}[n, x, y]$,
- for every natural number n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ for all natural numbers x, y_1, y_2 such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$.

The scheme *SeqBinOpEx* deals with a constant $\mathcal{A}$ that is a finite sequence and a ternary predicate $\mathcal{P}$ and states that:

there exists x such that there exists p being a finite sequence such that $x = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for every k such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ holds $\mathcal{P}[\mathcal{A}(k+1), p(k), p(k+1)]$.

provided the parameters satisfy the following conditions:

- for all k, x such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ there exists y such that $\mathcal{P}[\mathcal{A}(k+1), x, y]$,
- for all k, x, y_1, y_2, z such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ and $z = \mathcal{A}(k+1)$ and $\mathcal{P}[z, x, y_1]$ and $\mathcal{P}[z, x, y_2]$ holds $y_1 = y_2$.

The scheme *LambdaSeqBinOpEx* deals with a constant $\mathcal{A}$ that is a finite sequence and a binary functor $\mathcal{F}$ and states that:

there exists x such that there exists p being a finite sequence such that $x = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for all k, y, z such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ and $y = \mathcal{A}(k+1)$ and $z = p(k)$ holds $p(k+1) = \mathcal{F}(y, z)$.

for all values of the parameters.

The scheme *RecUn* deals with a constant $\mathcal{A}$, a constant $\mathcal{B}$ that is a function, a constant $\mathcal{C}$ that is a function and a ternary predicate $\mathcal{P}$ and states that:

$\mathcal{B} = \mathcal{C}$

provided the parameters satisfy the following conditions:

- $\operatorname{dom} \mathcal{B} = \mathbb{N}$ and $\mathcal{B}(0) = \mathcal{A}$ and for every n holds $\mathcal{P}[n, \mathcal{B}(n), \mathcal{B}(n+1)]$,
- $\operatorname{dom} \mathcal{C} = \mathbb{N}$ and $\mathcal{C}(0) = \mathcal{A}$ and for every n holds $\mathcal{P}[n, \mathcal{C}(n), \mathcal{C}(n+1)]$,

- for every n for arbitrary x, y_1, y_2 such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$.

The scheme *RecUnD* deals with a constant $\mathcal{A}$ that is a non-empty set, a constant $\mathcal{B}$ that is an element of $\mathcal{A}$, a ternary predicate $\mathcal{P}$, a constant $\mathcal{C}$ that is a function from $\mathbb{N}$ into $\mathcal{A}$ and a constant $\mathcal{D}$ that is a function from $\mathbb{N}$ into $\mathcal{A}$, and states that:

$\mathcal{C} = \mathcal{D}$

provided the parameters satisfy the following conditions:

- $\mathcal{C}(0) = \mathcal{B}$ and for every n holds $\mathcal{P}[n, \mathcal{C}(n), \mathcal{C}(n+1)]$,
- $\mathcal{D}(0) = \mathcal{B}$ and for every n holds $\mathcal{P}[n, \mathcal{D}(n), \mathcal{D}(n+1)]$,
- for every natural number n for all elements x, y_1, y_2 of $\mathcal{A}$ such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$.

The scheme *LambdaRecUn* deals with a constant $\mathcal{A}$, a binary functor $\mathcal{F}$, a constant $\mathcal{B}$ that is a function and a constant $\mathcal{C}$ that is a function, and states that:

$\mathcal{B} = \mathcal{C}$

provided the parameters satisfy the following conditions:

- $\operatorname{dom} \mathcal{B} = \mathbb{N}$ and $\mathcal{B}(0) = \mathcal{A}$ and for every n for arbitrary y such that $y = \mathcal{B}(n)$ holds $\mathcal{B}(n+1) = \mathcal{F}(n, y)$,
- $\operatorname{dom} \mathcal{C} = \mathbb{N}$ and $\mathcal{C}(0) = \mathcal{A}$ and for every n for arbitrary y such that $y = \mathcal{C}(n)$ holds $\mathcal{C}(n+1) = \mathcal{F}(n, y)$.

The scheme *LambdaRecUnD* concerns a constant $\mathcal{A}$ that is a non-empty set, a constant $\mathcal{B}$ that is an element of $\mathcal{A}$, a binary functor $\mathcal{F}$ yielding an element of $\mathcal{A}$, a constant $\mathcal{C}$ that is a function from $\mathbb{N}$ into $\mathcal{A}$ and a constant $\mathcal{D}$ that is a function from $\mathbb{N}$ into $\mathcal{A}$, and states that:

$\mathcal{C} = \mathcal{D}$

provided the parameters satisfy the following conditions:

- $\mathcal{C}(0) = \mathcal{B}$ and for every n for every element y of $\mathcal{A}$ such that $y = \mathcal{C}(n)$ holds $\mathcal{C}(n+1) = \mathcal{F}(n, y)$,
- $\mathcal{D}(0) = \mathcal{B}$ and for every n for every element y of $\mathcal{A}$ such that $y = \mathcal{D}(n)$ holds $\mathcal{D}(n+1) = \mathcal{F}(n, y)$.

The scheme *LambdaRecUnR* concerns a constant $\mathcal{A}$ that is a real number, a binary functor $\mathcal{F}$, a constant $\mathcal{B}$ that is a function from $\mathbb{N}$ into $\mathbb{R}$ and a constant $\mathcal{C}$ that is a function from $\mathbb{N}$ into $\mathbb{R}$, and states that:

$\mathcal{B} = \mathcal{C}$

provided the parameters satisfy the following conditions:

- $\mathcal{B}(0) = \mathcal{A}$ and for every n for every real number y such that $y = \mathcal{B}(n)$ holds $\mathcal{B}(n+1) = \mathcal{F}(n, y)$,
- $\mathcal{C}(0) = \mathcal{A}$ and for every n for every real number y such that $y = \mathcal{C}(n)$ holds $\mathcal{C}(n+1) = \mathcal{F}(n, y)$.

The scheme *LambdaRecUnN* deals with a constant $\mathcal{A}$ that is a natural number, a binary functor $\mathcal{F}$ yielding a natural number, a constant $\mathcal{B}$ that is a function from $\mathbb{N}$ into $\mathbb{N}$ and a constant $\mathcal{C}$ that is a function from $\mathbb{N}$ into $\mathbb{N}$, and states that:

$\mathcal{B} = \mathcal{C}$

provided the parameters satisfy the following conditions:

- $\mathcal{B}(0) = \mathcal{A}$ and for all n, m such that $m = \mathcal{B}(n)$ holds $\mathcal{B}(n+1) = \mathcal{F}(n, m)$,
- $\mathcal{C}(0) = \mathcal{A}$ and for all n, m such that $m = \mathcal{C}(n)$ holds $\mathcal{C}(n+1) = \mathcal{F}(n, m)$.

The scheme *FinRecUn* deals with a constant $\mathcal{A}$, a constant $\mathcal{B}$ that is a natural number, a constant $\mathcal{C}$ that is a finite sequence, a constant $\mathcal{D}$ that is a finite sequence and a ternary predicate $\mathcal{P}$ and states that:

$\mathcal{C} = \mathcal{D}$

provided the parameters satisfy the following conditions:

- for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ for arbitrary x, y_1, y_2 such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$,
- $\operatorname{len} \mathcal{C} = \mathcal{B}$ but $\mathcal{C}(1) = \mathcal{A}$ or $\mathcal{B} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ holds $\mathcal{P}[n, \mathcal{C}(n), \mathcal{C}(n+1)]$,
- $\operatorname{len} \mathcal{D} = \mathcal{B}$ but $\mathcal{D}(1) = \mathcal{A}$ or $\mathcal{B} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ holds $\mathcal{P}[n, \mathcal{D}(n), \mathcal{D}(n+1)]$.

The scheme *FinRecUnD* concerns a constant $\mathcal{A}$ that is a non-empty set, a constant $\mathcal{B}$ that is an element of $\mathcal{A}$, a constant $\mathcal{C}$ that is a natural number, a constant $\mathcal{D}$ that is a finite sequence of elements of $\mathcal{A}$, a constant $\mathcal{E}$ that is a finite sequence of elements of $\mathcal{A}$ and a ternary predicate $\mathcal{P}$ and states that:

$\mathcal{D} = \mathcal{E}$

provided the parameters satisfy the following conditions:

- for every n such that $1 \leq n$ and $n \leq \mathcal{C} - 1$ for all elements x, y_1, y_2 of $\mathcal{A}$ such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$,
- $\operatorname{len} \mathcal{D} = \mathcal{C}$ but $\mathcal{D}(1) = \mathcal{B}$ or $\mathcal{C} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{C} - 1$ holds $\mathcal{P}[n, \mathcal{D}(n), \mathcal{D}(n+1)]$,
- $\operatorname{len} \mathcal{E} = \mathcal{C}$ but $\mathcal{E}(1) = \mathcal{B}$ or $\mathcal{C} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{C} - 1$ holds $\mathcal{P}[n, \mathcal{E}(n), \mathcal{E}(n+1)]$.

The scheme *FinRecUnR* deals with a constant $\mathcal{A}$ that is a real number, a constant $\mathcal{B}$ that is a natural number, a constant $\mathcal{C}$ that is a finite sequence of elements of $\mathbb{R}$, a constant $\mathcal{D}$ that is a finite sequence of elements of $\mathbb{R}$ and a ternary predicate $\mathcal{P}$ and states that:

$\mathcal{C} = \mathcal{D}$

provided the parameters satisfy the following conditions:

- for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ for all real numbers x, y_1, y_2 such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$,
- $\operatorname{len} \mathcal{C} = \mathcal{B}$ but $\mathcal{C}(1) = \mathcal{A}$ or $\mathcal{B} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ holds $\mathcal{P}[n, \mathcal{C}(n), \mathcal{C}(n+1)]$,
- $\operatorname{len} \mathcal{D} = \mathcal{B}$ but $\mathcal{D}(1) = \mathcal{A}$ or $\mathcal{B} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ holds $\mathcal{P}[n, \mathcal{D}(n), \mathcal{D}(n+1)]$.

The scheme *FinRecUnN* concerns a constant $\mathcal{A}$ that is a natural number, a constant $\mathcal{B}$ that is a natural number, a constant $\mathcal{C}$ that is a finite sequence of elements of $\mathbb{N}$, a constant $\mathcal{D}$ that is a finite sequence of elements of $\mathbb{N}$ and a ternary predicate $\mathcal{P}$ and states that:

$\mathcal{C} = \mathcal{D}$

provided the parameters satisfy the following conditions:

- for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ for all natural numbers x, y_1, y_2 such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$,
- $\operatorname{len} \mathcal{C} = \mathcal{B}$ but $\mathcal{C}(1) = \mathcal{A}$ or $\mathcal{B} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ holds $\mathcal{P}[n, \mathcal{C}(n), \mathcal{C}(n+1)]$,
- $\operatorname{len} \mathcal{D} = \mathcal{B}$ but $\mathcal{D}(1) = \mathcal{A}$ or $\mathcal{B} = 0$ and for every n such that $1 \leq n$ and $n \leq \mathcal{B} - 1$ holds $\mathcal{P}[n, \mathcal{D}(n), \mathcal{D}(n+1)]$.

The scheme *SeqBinOpUn* deals with a constant $\mathcal{A}$ that is a finite sequence, a ternary predicate $\mathcal{P}$, a constant $\mathcal{B}$ and a constant $\mathcal{C}$ and states that:

$\mathcal{B} = \mathcal{C}$

provided the parameters satisfy the following conditions:

- for all k, x, y_1, y_2, z such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ and $z = \mathcal{A}(k+1)$ and $\mathcal{P}[z, x, y_1]$ and $\mathcal{P}[z, x, y_2]$ holds $y_1 = y_2$,
- there exists p being a finite sequence such that $\mathcal{B} = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for every k such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ holds $\mathcal{P}[\mathcal{A}(k+1), p(k), p(k+1)]$.
- there exists p being a finite sequence such that $\mathcal{C} = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for every k such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ holds $\mathcal{P}[\mathcal{A}(k+1), p(k), p(k+1)]$.

The scheme *LambdaSeqBinOpUn* concerns a constant $\mathcal{A}$ that is a finite sequence, a binary functor $\mathcal{F}$, a constant $\mathcal{B}$ and a constant $\mathcal{C}$ and states that:

$\mathcal{B} = \mathcal{C}$

provided the parameters satisfy the following conditions:

- there exists p being a finite sequence such that $\mathcal{B} = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for all k, y, z such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ and $y = \mathcal{A}(k+1)$ and $z = p(k)$ holds $p(k+1) = \mathcal{F}(y, z)$.
- there exists p being a finite sequence such that $\mathcal{C} = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for all k, y, z such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ and $y = \mathcal{A}(k+1)$ and $z = p(k)$ holds $p(k+1) = \mathcal{F}(y, z)$.

The scheme *DefRec* concerns a constant $\mathcal{A}$, a constant $\mathcal{B}$ that is a natural number and a ternary predicate $\mathcal{P}$ and states that:

(i) there exists y being any such that there exists f being a function such that $y = f(\mathcal{B})$ and $\operatorname{dom} f = \mathbb{N}$ and $f(0) = \mathcal{A}$ and for every n holds $\mathcal{P}[n, f(n), f(n+1)]$,

(ii) for arbitrary y_1, y_2 such that there exists f being a function such that $y_1 = f(\mathcal{B})$ and $\operatorname{dom} f = \mathbb{N}$ and $f(0) = \mathcal{A}$ and for every n holds $\mathcal{P}[n, f(n), f(n+1)]$ and there exists f being a function such that $y_2 = f(\mathcal{B})$ and $\operatorname{dom} f = \mathbb{N}$ and $f(0) = \mathcal{A}$ and for every n holds $\mathcal{P}[n, f(n), f(n+1)]$ holds $y_1 = y_2$.

provided the parameters satisfy the following conditions:

- for every n, x there exists y such that $\mathcal{P}[n, x, y]$,
- for all n, x, y_1, y_2 such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$.

The scheme *LambdaDefRec* deals with a constant $\mathcal{A}$, a constant $\mathcal{B}$ that is a natural number and a binary functor $\mathcal{F}$ and states that:

(i) there exists y being any such that there exists f being a function such that $y = f(\mathcal{B})$ and $\operatorname{dom} f = \mathbb{N}$ and $f(0) = \mathcal{A}$ and for all n, x such that $x = f(n)$ holds $f(n+1) = \mathcal{F}(n, x)$,

(ii) for arbitrary y_1, y_2 such that there exists f being a function such that $y_1 = f(\mathcal{B})$ and $\operatorname{dom} f = \mathbb{N}$ and $f(0) = \mathcal{A}$ and for all n, x such that $x = f(n)$ holds $f(n+1) = \mathcal{F}(n, x)$ and there exists f being a function such that $y_2 = f(\mathcal{B})$ and $\operatorname{dom} f = \mathbb{N}$ and $f(0) = \mathcal{A}$ and for all n, x such that $x = f(n)$ holds $f(n+1) = \mathcal{F}(n, x)$ holds $y_1 = y_2$.
for all values of the parameters.

The scheme *DefRecD* concerns a constant $\mathcal{A}$ that is a non-empty set, a constant $\mathcal{B}$ that is an element of $\mathcal{A}$, a constant $\mathcal{C}$ that is a natural number and a ternary predicate $\mathcal{P}$ and states that:
(i) there exists y being an element of $\mathcal{A}$ such that there exists f being a function from $\mathbb{N}$ into $\mathcal{A}$ such that $y = f(\mathcal{C})$ and $f(0) = \mathcal{B}$ and for every n holds $\mathcal{P}[n, f(n), f(n+1)]$,
(ii) for all elements y_1, y_2 of $\mathcal{A}$ such that there exists f being a function from $\mathbb{N}$ into $\mathcal{A}$ such that $y_1 = f(\mathcal{C})$ and $f(0) = \mathcal{B}$ and for every n holds $\mathcal{P}[n, f(n), f(n+1)]$ and there exists f being a function from $\mathbb{N}$ into $\mathcal{A}$ such that $y_2 = f(\mathcal{C})$ and $f(0) = \mathcal{B}$ and for every n holds $\mathcal{P}[n, f(n), f(n+1)]$ holds $y_1 = y_2$.
provided the parameters satisfy the following conditions:

- for every natural number n for every element x of $\mathcal{A}$ there exists y being an element of $\mathcal{A}$ such that $\mathcal{P}[n, x, y]$,
- for every natural number n for all elements x, y_1, y_2 of $\mathcal{A}$ such that $\mathcal{P}[n, x, y_1]$ and $\mathcal{P}[n, x, y_2]$ holds $y_1 = y_2$.

The scheme *LambdaDefRecD* concerns a constant $\mathcal{A}$ that is a non-empty set, a constant $\mathcal{B}$ that is an element of $\mathcal{A}$, a constant $\mathcal{C}$ that is a natural number and a binary functor $\mathcal{F}$ yielding an element of $\mathcal{A}$ and states that:
(i) there exists y being an element of $\mathcal{A}$ such that there exists f being a function from $\mathbb{N}$ into $\mathcal{A}$ such that $y = f(\mathcal{C})$ and $f(0) = \mathcal{B}$ and for every natural number n for every element x of $\mathcal{A}$ such that $x = f(n)$ holds $f(n+1) = \mathcal{F}(n, x)$,
(ii) for all elements y_1, y_2 of $\mathcal{A}$ such that there exists f being a function from $\mathbb{N}$ into $\mathcal{A}$ such that $y_1 = f(\mathcal{C})$ and $f(0) = \mathcal{B}$ and for every natural number n for every element x of $\mathcal{A}$ such that $x = f(n)$ holds $f(n+1) = \mathcal{F}(n, x)$ and there exists f being a function from $\mathbb{N}$ into $\mathcal{A}$ such that $y_2 = f(\mathcal{C})$ and $f(0) = \mathcal{B}$ and for every natural number n for every element x of $\mathcal{A}$ such that $x = f(n)$ holds $f(n+1) = \mathcal{F}(n, x)$ holds $y_1 = y_2$.
for all values of the parameters.

The scheme *SeqBinOpDef* concerns a constant $\mathcal{A}$ that is a finite sequence and a ternary predicate $\mathcal{P}$ and states that:
(i) there exists x such that there exists p being a finite sequence such that $x = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for every k such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ holds $\mathcal{P}[\mathcal{A}(k+1), p(k), p(k+1)]$,
(ii) for all x, y such that there exists p being a finite sequence such that $x = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for every k such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ holds $\mathcal{P}[\mathcal{A}(k+1), p(k), p(k+1)]$ and there exists p being a finite sequence such that $y = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for every k such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ holds $\mathcal{P}[\mathcal{A}(k+1), p(k), p(k+1)]$ holds $x = y$.

provided the parameters satisfy the following conditions:

- for all k, y such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ there exists z such that $\mathcal{P}[\mathcal{A}(k+1), y, z]$,
- for all k, x, y_1, y_2, z such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ and $z = \mathcal{A}(k+1)$ and $\mathcal{P}[z, x, y_1]$ and $\mathcal{P}[z, x, y_2]$ holds $y_1 = y_2$.

The scheme *LambdaSeqBinOpDe* concerns a constant $\mathcal{A}$ that is a finite sequence and a binary functor $\mathcal{F}$ and states that:

(i) there exists x such that there exists p being a finite sequence such that $x = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for all k, y, z such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ and $y = \mathcal{A}(k+1)$ and $z = p(k)$ holds $p(k+1) = \mathcal{F}(y, z)$,

(ii) for all x, y such that there exists p being a finite sequence such that $x = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for all k, y, z such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ and $y = \mathcal{A}(k+1)$ and $z = p(k)$ holds $p(k+1) = \mathcal{F}(y, z)$ and there exists p being a finite sequence such that $y = p(\operatorname{len} p)$ and $\operatorname{len} p = \operatorname{len} \mathcal{A}$ and $p(1) = \mathcal{A}(1)$ and for all k, y, z such that $1 \leq k$ and $k \leq \operatorname{len} \mathcal{A} - 1$ and $y = \mathcal{A}(k+1)$ and $z = p(k)$ holds $p(k+1) = \mathcal{F}(y, z)$ holds $x = y$.

for all values of the parameters.

References

[1] Grzegorz Bancerek. The fundamental properties of natural numbers. *Formalized Mathematics*, 1(**1**):41–46, 1990.

[2] Grzegorz Bancerek and Krzysztof Hryniewiecki. Segments of natural numbers and finite sequences. *Formalized Mathematics*, 1(**1**):107–114, 1990.

[3] Czesław Byliński. Functions and their basic properties. *Formalized Mathematics*, 1(**1**):55–65, 1990.

[4] Czesław Byliński. Functions from a set to a set. *Formalized Mathematics*, 1(**1**):153–164, 1990.

[5] Zinaida Trybulec and Halina Święczkowska. Boolean properties of sets. *Formalized Mathematics*, 1(**1**):17–23, 1990.

Received September 4, 1989