Contents

Formaliz. Math. 28 (**4**)

Continued on inside back cover

FORMALIZED MATHEMATICS
Vol. 28, No. 4, Pages 263–268, 2020
DOI: 10.2478/forma-2020-0023

sciendo
https://www.sciendo.com/

Functional Sequence in Norm Space

Hiroshi Yamazaki
Nagano Prefectural Institute of Technology
Nagano, Japan

Summary. In this article, we formalize in Mizar [1], [2] functional sequences and basic operations on functional sequences in norm space based on [5]. In the first section, we define functional sequence in norm space. In the second section, we define pointwise convergence and prove some related theorems. In the last section we define uniform convergence and limit of functional sequence.

MSC: 46A19 46A32 68V20

Keywords: pointwise convergence; functional sequence; formalized mathematics

MML identifier: SEQFUNC2, version: 8.1.10 5.64.1388

1. Preliminaries

From now on D denotes a non empty set, D_1, D_2, x, y, Z denote sets, n, k denote natural numbers, p, x_1, r denote real numbers, f denotes a function, Y denotes a real normed space, and G, H, H_1, H_2, J denote sequences of partial functions from D into the carrier of Y.

Now we state the proposition:

(1) f is a sequence of partial functions from D_1 into D_2 if and only if $\operatorname{dom} f = \mathbb{N}$ and for every x such that $x \in \mathbb{N}$ holds $f(x)$ is a partial function from D_1 to D_2.

Proof: If f is a sequence of partial functions from D_1 into D_2, then $\operatorname{dom} f = \mathbb{N}$ and for every x such that $x \in \mathbb{N}$ holds $f(x)$ is a partial function from D_1 to D_2 by [3, (46)]. □

Let us consider D. Let Y be a non empty normed structure, H be a sequence of partial functions from D into the carrier of Y, and r be a real number. The functor $r \cdot H$ yielding a sequence of partial functions from D into the carrier of Y is defined by

ISSN 1426–2630(Print), 1898-9934(Online)

(Def. 1) for every natural number n, $it(n) = r \cdot H(n)$.

Let Y be a real normed space. The functor $-H$ yielding a sequence of partial functions from D into the carrier of Y is defined by

(Def. 2) for every natural number n, $it(n) = -H(n)$.

One can verify that the functor is involutive.

Let Y be a non empty normed structure. The functor $\|H\|$ yielding a sequence of partial functions from D into $\mathbb{R}$ is defined by

(Def. 3) for every natural number n, $it(n) = \|H(n)\|$.

Let G, H be sequences of partial functions from D into the carrier of Y. The functor $G + H$ yielding a sequence of partial functions from D into the carrier of Y is defined by

(Def. 4) for every natural number n, $it(n) = G(n) + H(n)$.

Let Y be a real normed space. The functor $G - H$ yielding a sequence of partial functions from D into the carrier of Y is defined by the term

(Def. 5) $G + -H$.

Now we state the propositions:

(2) $H_1 = G - H$ if and only if for every n, $H_1(n) = G(n) - H(n)$.

PROOF: If $H_1 = G - H$, then for every n, $H_1(n) = G(n) - H(n)$ by [7, (25)]. □

(3) (i) $G + H = H + G$, and

(ii) $(G + H) + J = G + (H + J)$.

(4) $-H = (-1) \cdot H$.

(5) (i) $r \cdot (G + H) = r \cdot G + r \cdot H$, and

(ii) $r \cdot (G - H) = r \cdot G - r \cdot H$.

The theorem is a consequence of (2).

(6) $r \cdot p \cdot H = r \cdot (p \cdot H)$.

(7) $1 \cdot H = H$.

(8) $\|r \cdot H\| = |r| \cdot \|H\|$.

2. Pointwise Convergence

In the sequel x denotes an element of D, X denotes a set, S_1, S_2 denote sequences of Y, and f denotes a partial function from D to the carrier of Y.

Let us consider D. Let Y be a non empty normed structure and H be a sequence of partial functions from D into the carrier of Y. Let us consider x. The functor $H\#x$ yielding a sequence of the carrier of Y is defined by

(Def. 6) for every n, $it(n) = H(n)_{/x}$.

Let us consider Y, H, and X. We say that H is point-convergent on X if and only if

(Def. 7) X is common for elements of H and there exists f such that $X = \mathrm{dom}\, f$ and for every x such that $x \in X$ for every p such that $p > 0$ there exists k such that for every n such that $n \geqslant k$ holds $\|H(n)_{/x} - f_{/x}\| < p$.

Now we state the propositions:

(9) H is point-convergent on X if and only if X is common for elements of H and there exists f such that $X = \mathrm{dom}\, f$ and for every x such that $x \in X$ holds $H\#x$ is convergent and $\lim(H\#x) = f(x)$.

(10) H is point-convergent on X if and only if X is common for elements of H and for every x such that $x \in X$ holds $H\#x$ is convergent.

Proof: Define $\mathcal{X}[\text{set}] \equiv \$_1 \in X$. Define $\mathcal{U}(\text{element of } D) = (\lim(H\#\$_1))(\in$ (the carrier of Y)). Consider f such that for every x, $x \in \mathrm{dom}\, f$ iff $\mathcal{X}[x]$ and for every x such that $x \in \mathrm{dom}\, f$ holds $f(x) = \mathcal{U}(x)$ from [4, Sch. 3]. If H is point-convergent on X, then X is common for elements of H and for every x such that $x \in X$ holds $H\#x$ is convergent. □

3. Uniform Convergence and Limit of Functional Sequence

Let us consider D, Y, H, and X. We say that H is uniform-convergent on X if and only if

(Def. 8) X is common for elements of H and there exists f such that $X = \mathrm{dom}\, f$ and for every p such that $p > 0$ there exists k such that for every n and x such that $n \geqslant k$ and $x \in X$ holds $\|H(n)_{/x} - f_{/x}\| < p$.

Assume H is point-convergent on X. The functor $\lim_X H$ yielding a partial function from D to the carrier of Y is defined by

(Def. 9) $\mathrm{dom}\, it = X$ and for every x such that $x \in \mathrm{dom}\, it$ holds $it(x) = \lim(H\#x)$.

Now we state the propositions:

(11) Suppose H is point-convergent on X. Then $f = \lim_X H$ if and only if $\mathrm{dom}\, f = X$ and for every x such that $x \in X$ for every p such that $p > 0$ there exists k such that for every n such that $n \geqslant k$ holds $\|H(n)_{/x} - f_{/x}\| < p$. The theorem is a consequence of (10).

(12) If H is uniform-convergent on X, then H is point-convergent on X.

(13) If $Z \subseteq X$ and $Z \neq \emptyset$ and X is common for elements of H, then Z is common for elements of H.

(14) Suppose $Z \subseteq X$ and $Z \neq \emptyset$ and H is point-convergent on X. Then

(i) H is point-convergent on Z, and

(ii) $\lim_X H{\restriction}Z = \lim_Z H$.

The theorem is a consequence of (13).

(15) If $Z \subseteq X$ and $Z \neq \emptyset$ and H is uniform-convergent on X, then H is uniform-convergent on Z. The theorem is a consequence of (13).

Let us consider a set x. Now we state the propositions:

(16) If X is common for elements of H, then if $x \in X$, then $\{x\}$ is common for elements of H.

(17) If H is point-convergent on X, then if $x \in X$, then $\{x\}$ is common for elements of H.

(18) Suppose $\{x\}$ is common for elements of H_1 and common for elements of H_2. Then

(i) $H_1\#x + H_2\#x = (H_1 + H_2)\#x$, and

(ii) $H_1\#x - H_2\#x = (H_1 - H_2)\#x$.

The theorem is a consequence of (2).

In the sequel x denotes an element of D.

(19) Suppose $\{x\}$ is common for elements of H. Then

(i) $\|H\|\#x = \|H\#x\|$, and

(ii) $(-H)\#x = (-1) \cdot (H\#x)$.

(20) If $\{x\}$ is common for elements of H, then $(r \cdot H)\#x = r \cdot (H\#x)$.

(21) Suppose X is common for elements of H_1 and common for elements of H_2. If $x \in X$, then $H_1\#x + H_2\#x = (H_1 + H_2)\#x$ and $H_1\#x - H_2\#x = (H_1 - H_2)\#x$. The theorem is a consequence of (16) and (18).

(22) Suppose $\{x\}$ is common for elements of H. Then

(i) $\|H\|\#x = \|H\#x\|$, and

(ii) $(-H)\#x = (-1) \cdot (H\#x)$.

Let us consider x. Now we state the propositions:

(23) If X is common for elements of H, then if $x \in X$, then $(r \cdot H)\#x = r \cdot (H\#x)$. The theorem is a consequence of (16) and (20).

(24) Suppose H_1 is point-convergent on X and H_2 is point-convergent on X. Then if $x \in X$, then $H_1\#x + H_2\#x = (H_1 + H_2)\#x$ and $H_1\#x - H_2\#x = (H_1 - H_2)\#x$.

(25) Suppose $\{x\}$ is common for elements of H. Then

(i) $\|H\|\#x = \|H\#x\|$, and

(ii) $(-H)\#x = (-1) \cdot (H\#x)$.

(26) If H is point-convergent on X, then for every x such that $x \in X$ holds $(r \cdot H)\#x = r \cdot (H\#x)$.

(27) If X is common for elements of H_1 and common for elements of H_2, then X is common for elements of $H_1 + H_2$ and common for elements of $H_1 - H_2$. The theorem is a consequence of (2).

(28) If X is common for elements of H, then X is common for elements of $\|H\|$ and common for elements of $-H$.

(29) If X is common for elements of H, then X is common for elements of $r \cdot H$.

(30) Suppose H_1 is point-convergent on X and H_2 is point-convergent on X. Then

- (i) $H_1 + H_2$ is point-convergent on X, and
- (ii) $\lim_X(H_1 + H_2) = \lim_X H_1 + \lim_X H_2$, and
- (iii) $H_1 - H_2$ is point-convergent on X, and
- (iv) $\lim_X(H_1 - H_2) = \lim_X H_1 - \lim_X H_2$.

The theorem is a consequence of (10), (21), and (27).

(31) Suppose H is point-convergent on X. Then

- (i) $\|H\|$ is point-convergent on X, and
- (ii) $\lim_X \|H\| = \|\lim_X H\|$, and
- (iii) $-H$ is point-convergent on X, and
- (iv) $\lim_X(-H) = -\lim_X H$.

The theorem is a consequence of (16), (10), (19), and (28).

(32) If H is point-convergent on X, then $r \cdot H$ is point-convergent on X and $\lim_X(r \cdot H) = r \cdot \lim_X H$. The theorem is a consequence of (10), (23), and (29).

(33) H is uniform-convergent on X if and only if X is common for elements of H and H is point-convergent on X and for every r such that $0 < r$ there exists k such that for every n and x such that $n \geqslant k$ and $x \in X$ holds $\|H(n)_{/x} - (\lim_X H)_{/x}\| < r$. The theorem is a consequence of (12) and (11).

From now on V, W denote real normed spaces and H denotes a sequence of partial functions from the carrier of V into the carrier of W.

Now we state the proposition:

(34) If H is uniform-convergent on X and for every n, $H(n)\restriction X$ is continuous on X, then $\lim_X H$ is continuous on X.

Proof: Set $l = \lim_X H$. H is point-convergent on X. For every point x_0 of V such that $x_0 \in X$ holds $l{\restriction}X$ is continuous in x_0 by [6, (62)], (33), (11), [6, (61)]. $\square$

Acknowledgement: I would like to thank Yasunari Shidama for useful advice on formalizing theorems.

References

[1] Grzegorz Bancerek, Czesław Byliński, Adam Grabowski, Artur Korniłowicz, Roman Matuszewski, Adam Naumowicz, Karol Pąk, and Josef Urban. Mizar: State-of-the-art and beyond. In Manfred Kerber, Jacques Carette, Cezary Kaliszyk, Florian Rabe, and Volker Sorge, editors, *Intelligent Computer Mathematics*, volume 9150 of *Lecture Notes in Computer Science*, pages 261–279. Springer International Publishing, 2015. ISBN 978-3-319-20614-1. doi:10.1007/978-3-319-20615-8_17.

[2] Grzegorz Bancerek, Czesław Byliński, Adam Grabowski, Artur Korniłowicz, Roman Matuszewski, Adam Naumowicz, and Karol Pąk. The role of the Mizar Mathematical Library for interactive proof development in Mizar. *Journal of Automated Reasoning*, 61(1):9–32, 2018. doi:10.1007/s10817-017-9440-6.

[3] Czesław Byliński. Partial functions. *Formalized Mathematics*, 1(**2**):357–367, 1990.

[4] Jarosław Kotowicz. Real sequences and basic operations on them. *Formalized Mathematics*, 1(**2**):269–272, 1990.

[5] Teiji Takagi. *Introduction to Analysis.* Iwanami Shoten, Publishers, 3rd edition, 1983.

[6] Edmund Woronowicz. Relations and their basic properties. *Formalized Mathematics*, 1(**1**): 73–83, 1990.

[7] Hiroshi Yamazaki and Yasunari Shidama. Algebra of vector functions. *Formalized Mathematics*, 3(**2**):171–175, 1992.

Accepted October 25, 2020

FORMALIZED MATHEMATICS
Vol. 28, No. 4, Pages 269–278, 2020
DOI: 10.2478/forma-2020-0024

General Theory and Tools for Proving Algorithms in Nominative Data Systems

Adrian Jaszczak
Institute of Informatics
University of Białystok
Poland

Summary. In this paper we introduce some new definitions for sequences of operations and extract general theorems about properties of iterative algorithms encoded in nominative data language [20] in the Mizar system [3], [1] in order to simplify the process of proving algorithms in the future.

This paper continues verification of algorithms [10], [13], [12], [14] written in terms of simple-named complex-valued nominative data [6], [8], [18], [11], [15], [16].

The validity of the algorithm is presented in terms of semantic Floyd-Hoare triples over such data [9]. Proofs of the correctness are based on an inference system for an extended Floyd-Hoare logic [2], [4] with partial pre- and post-conditions [17], [19], [7], [5].

MSC: 68Q60 03B70 68V20

Keywords: nominative data; program verification; inference rules

MML identifier: NOMIN_8, version: 8.1.10 5.64.1388

1. Composition Rules for Programs

Let D be a non empty set. One can verify that there exists a finite sequence which is non empty and D-valued.

Let n be a natural number. One can verify that there exists a finite sequence which is D-valued and n-element.

From now on D denotes a non empty set, f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10} denote binominative functions of D, p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}

denote partial predicates of D, q_1, q_2, q_3, q_4, q_5, q_6, q_7, q_8, q_9, q_{10} denote total partial predicates of D, n, m, N denote natural numbers, f_D denotes a $(D \dot{\rightarrow} D)$-valued finite sequence, f_B denotes a $(D \dot{\rightarrow} \mathit{Boolean})$-valued finite sequence, V, A denote sets.

From now on val denotes a function, loc denotes a V-valued function, d_1 denotes a non-atomic nominative data of V and A, p denotes a partial predicate over simple-named complex-valued nominative data of V and A, d, v denote objects, z_2 denotes a non zero natural number, inp, pos denote finite sequences, and prg denotes a non empty, $(\mathrm{FPrg}(\mathrm{ND_{SC}}(V, A)))$-valued finite sequence.

Let us consider D, f_1, f_2, f_3, f_4, f_5, f_6, and f_7. The functor PP-composition $(f_1, f_2, f_3, f_4, f_5, f_6, f_7)$ yielding a binominative function of D is defined by the term

(Def. 1) PP-composition$(f_1, f_2, f_3, f_4, f_5, f_6) \bullet f_7$.

Now we state the proposition:

(1) Unconditional composition rule for 7 programs:
Suppose $\langle p_1, f_1, p_2\rangle$ is an SFHT of D and $\langle p_2, f_2, p_3\rangle$ is an SFHT of D and $\langle p_3, f_3, p_4\rangle$ is an SFHT of D and $\langle p_4, f_4, p_5\rangle$ is an SFHT of D and $\langle p_5, f_5, p_6\rangle$ is an SFHT of D and $\langle p_6, f_6, p_7\rangle$ is an SFHT of D and $\langle p_7, f_7, p_8\rangle$ is an SFHT of D and $\langle \sim p_2, f_2, p_3\rangle$ is an SFHT of D and $\langle \sim p_3, f_3, p_4\rangle$ is an SFHT of D and $\langle \sim p_4, f_4, p_5\rangle$ is an SFHT of D and $\langle \sim p_5, f_5, p_6\rangle$ is an SFHT of D and $\langle \sim p_6, f_6, p_7\rangle$ is an SFHT of D and $\langle \sim p_7, f_7, p_8\rangle$ is an SFHT of D. Then $\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4, f_5, f_6, f_7), p_8\rangle$ is an SFHT of D.

Let us consider D, f_1, f_2, f_3, f_4, f_5, f_6, f_7, and f_8. The functor PP-composition$(f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8)$ yielding a binominative function of D is defined by the term

(Def. 2) PP-composition$(f_1, f_2, f_3, f_4, f_5, f_6, f_7) \bullet f_8$.

Now we state the proposition:

(2) Unconditional composition rule for 8 programs:
Suppose $\langle p_1, f_1, p_2\rangle$ is an SFHT of D and $\langle p_2, f_2, p_3\rangle$ is an SFHT of D and $\langle p_3, f_3, p_4\rangle$ is an SFHT of D and $\langle p_4, f_4, p_5\rangle$ is an SFHT of D and $\langle p_5, f_5, p_6\rangle$ is an SFHT of D and $\langle p_6, f_6, p_7\rangle$ is an SFHT of D and $\langle p_7, f_7, p_8\rangle$ is an SFHT of D and $\langle p_8, f_8, p_9\rangle$ is an SFHT of D and $\langle \sim p_2, f_2, p_3\rangle$ is an SFHT of D and $\langle \sim p_3, f_3, p_4\rangle$ is an SFHT of D and $\langle \sim p_4, f_4, p_5\rangle$ is an SFHT of D and $\langle \sim p_5, f_5, p_6\rangle$ is an SFHT of D and $\langle \sim p_6, f_6, p_7\rangle$ is an SFHT of D and $\langle \sim p_7, f_7, p_8\rangle$ is an SFHT of D and $\langle \sim p_8, f_8, p_9\rangle$ is an SFHT of D. Then $\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8), p_9\rangle$ is an SFHT of D. The theorem is a consequence of (1).

Let us consider D, f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, and f_9. The functor PP-composition($f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9$) yielding a binominative function of D is defined by the term

(Def. 3) PP-composition($f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8$) $\bullet$ f_9.

Now we state the proposition:

(3) UNCONDITIONAL COMPOSITION RULE FOR 9 PROGRAMS:
Suppose $\langle p_1, f_1, p_2\rangle$ is an SFHT of D and $\langle p_2, f_2, p_3\rangle$ is an SFHT of D and $\langle p_3, f_3, p_4\rangle$ is an SFHT of D and $\langle p_4, f_4, p_5\rangle$ is an SFHT of D and $\langle p_5, f_5, p_6\rangle$ is an SFHT of D and $\langle p_6, f_6, p_7\rangle$ is an SFHT of D and $\langle p_7, f_7, p_8\rangle$ is an SFHT of D and $\langle p_8, f_8, p_9\rangle$ is an SFHT of D and $\langle p_9, f_9, p_{10}\rangle$ is an SFHT of D and $\langle \sim p_2, f_2, p_3\rangle$ is an SFHT of D and $\langle \sim p_3, f_3, p_4\rangle$ is an SFHT of D and $\langle \sim p_4, f_4, p_5\rangle$ is an SFHT of D and $\langle \sim p_5, f_5, p_6\rangle$ is an SFHT of D and $\langle \sim p_6, f_6, p_7\rangle$ is an SFHT of D and $\langle \sim p_7, f_7, p_8\rangle$ is an SFHT of D and $\langle \sim p_8, f_8, p_9\rangle$ is an SFHT of D and $\langle \sim p_9, f_9, p_{10}\rangle$ is an SFHT of D. Then $\langle p_1$, PP-composition($f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9$), $p_{10}\rangle$ is an SFHT of D. The theorem is a consequence of (2).

Let us consider D, f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, and f_{10}. The functor PP-composition($f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}$) yielding a binominative function of D is defined by the term

(Def. 4) PP-composition($f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9$) $\bullet$ f_{10}.

Now we state the propositions:

(4) UNCONDITIONAL COMPOSITION RULE FOR 10 PROGRAMS:
Suppose $\langle p_1, f_1, p_2\rangle$ is an SFHT of D and $\langle p_2, f_2, p_3\rangle$ is an SFHT of D and $\langle p_3, f_3, p_4\rangle$ is an SFHT of D and $\langle p_4, f_4, p_5\rangle$ is an SFHT of D and $\langle p_5, f_5, p_6\rangle$ is an SFHT of D and $\langle p_6, f_6, p_7\rangle$ is an SFHT of D and $\langle p_7, f_7, p_8\rangle$ is an SFHT of D and $\langle p_8, f_8, p_9\rangle$ is an SFHT of D and $\langle p_9, f_9, p_{10}\rangle$ is an SFHT of D and $\langle p_{10}, f_{10}, p_{11}\rangle$ is an SFHT of D and $\langle \sim p_2, f_2, p_3\rangle$ is an SFHT of D and $\langle \sim p_3, f_3, p_4\rangle$ is an SFHT of D and $\langle \sim p_4, f_4, p_5\rangle$ is an SFHT of D and $\langle \sim p_5, f_5, p_6\rangle$ is an SFHT of D and $\langle \sim p_6, f_6, p_7\rangle$ is an SFHT of D and $\langle \sim p_7, f_7, p_8\rangle$ is an SFHT of D and $\langle \sim p_8, f_8, p_9\rangle$ is an SFHT of D and $\langle \sim p_9, f_9, p_{10}\rangle$ is an SFHT of D and $\langle \sim p_{10}, f_{10}, p_{11}\rangle$ is an SFHT of D. Then $\langle p_1$, PP-composition($f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}$), $p_{11}\rangle$ is an SFHT of D. The theorem is a consequence of (3).

(5) Suppose $\langle p_1, f_1, q_1\rangle$ is an SFHT of D and $\langle q_1, f_2, p_2\rangle$ is an SFHT of D. Then $\langle p_1, f_1 \bullet f_2, p_2\rangle$ is an SFHT of D.

(6) Suppose $\langle p_1, f_1, q_1\rangle$ is an SFHT of D and $\langle q_1, f_2, q_2\rangle$ is an SFHT of D and $\langle q_2, f_3, p_2\rangle$ is an SFHT of D. Then $\langle p_1$, PP-composition(f_1, f_2, f_3), $p_2\rangle$ is an SFHT of D. The theorem is a consequence of (5).

(7) Suppose $\langle p_1, f_1, q_1\rangle$ is an SFHT of D and $\langle q_1, f_2, q_2\rangle$ is an SFHT of D and $\langle q_2, f_3, q_3\rangle$ is an SFHT of D and $\langle q_3, f_4, p_2\rangle$ is an SFHT of D. Then

$\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4), p_2\rangle$ is an SFHT of D. The theorem is a consequence of (6).

(8) Suppose $\langle p_1, f_1, q_1\rangle$ is an SFHT of D and $\langle q_1, f_2, q_2\rangle$ is an SFHT of D and $\langle q_2, f_3, q_3\rangle$ is an SFHT of D and $\langle q_3, f_4, q_4\rangle$ is an SFHT of D and $\langle q_4, f_5, p_2\rangle$ is an SFHT of D. Then $\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4, f_5), p_2\rangle$ is an SFHT of D. The theorem is a consequence of (7).

(9) Suppose $\langle p_1, f_1, q_1\rangle$ is an SFHT of D and $\langle q_1, f_2, q_2\rangle$ is an SFHT of D and $\langle q_2, f_3, q_3\rangle$ is an SFHT of D and $\langle q_3, f_4, q_4\rangle$ is an SFHT of D and $\langle q_4, f_5, q_5\rangle$ is an SFHT of D and $\langle q_5, f_6, p_2\rangle$ is an SFHT of D. Then $\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4, f_5, f_6), p_2\rangle$ is an SFHT of D. The theorem is a consequence of (8).

(10) Suppose $\langle p_1, f_1, q_1\rangle$ is an SFHT of D and $\langle q_1, f_2, q_2\rangle$ is an SFHT of D and $\langle q_2, f_3, q_3\rangle$ is an SFHT of D and $\langle q_3, f_4, q_4\rangle$ is an SFHT of D and $\langle q_4, f_5, q_5\rangle$ is an SFHT of D and $\langle q_5, f_6, q_6\rangle$ is an SFHT of D and $\langle q_6, f_7, p_2\rangle$ is an SFHT of D. Then $\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4, f_5, f_6, f_7), p_2\rangle$ is an SFHT of D. The theorem is a consequence of (9).

(11) Suppose $\langle p_1, f_1, q_1\rangle$ is an SFHT of D and $\langle q_1, f_2, q_2\rangle$ is an SFHT of D and $\langle q_2, f_3, q_3\rangle$ is an SFHT of D and $\langle q_3, f_4, q_4\rangle$ is an SFHT of D and $\langle q_4, f_5, q_5\rangle$ is an SFHT of D and $\langle q_5, f_6, q_6\rangle$ is an SFHT of D and $\langle q_6, f_7, q_7\rangle$ is an SFHT of D and $\langle q_7, f_8, p_2\rangle$ is an SFHT of D. Then $\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8), p_2\rangle$ is an SFHT of D. The theorem is a consequence of (10).

(12) Suppose $\langle p_1, f_1, q_1\rangle$ is an SFHT of D and $\langle q_1, f_2, q_2\rangle$ is an SFHT of D and $\langle q_2, f_3, q_3\rangle$ is an SFHT of D and $\langle q_3, f_4, q_4\rangle$ is an SFHT of D and $\langle q_4, f_5, q_5\rangle$ is an SFHT of D and $\langle q_5, f_6, q_6\rangle$ is an SFHT of D and $\langle q_6, f_7, q_7\rangle$ is an SFHT of D and $\langle q_7, f_8, q_8\rangle$ is an SFHT of D and $\langle q_8, f_9, p_2\rangle$ is an SFHT of D. Then $\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9), p_2\rangle$ is an SFHT of D. The theorem is a consequence of (11).

(13) Suppose $\langle p_1, f_1, q_1\rangle$ is an SFHT of D and $\langle q_1, f_2, q_2\rangle$ is an SFHT of D and $\langle q_2, f_3, q_3\rangle$ is an SFHT of D and $\langle q_3, f_4, q_4\rangle$ is an SFHT of D and $\langle q_4, f_5, q_5\rangle$ is an SFHT of D and $\langle q_5, f_6, q_6\rangle$ is an SFHT of D and $\langle q_6, f_7, q_7\rangle$ is an SFHT of D and $\langle q_7, f_8, q_8\rangle$ is an SFHT of D and $\langle q_8, f_9, q_9\rangle$ is an SFHT of D and $\langle q_9, f_{10}, p_2\rangle$ is an SFHT of D. Then $\langle p_1, \text{PP-composition}(f_1, f_2, f_3, f_4, f_5, f_6, f_7, f_8, f_9, f_{10}), p_2\rangle$ is an SFHT of D. The theorem is a consequence of (12).

Let us consider D and f_D. Assume $0 < \operatorname{len} f_D$. The functor PP-composition-Seq(f_D) yielding a finite sequence of elements of $D\dot{\to}D$ is defined by

(Def. 5) $\operatorname{len} it = \operatorname{len} f_D$ and $it(1) = f_D(1)$ and for every natural number n such that $1 \leqslant n < \operatorname{len} f_D$ holds $it(n+1) = it(n) \bullet f_D(n+1)$.

The functor PP-composition(f_D) yielding a binominative function of D is defined by the term

(Def. 6) (PP-compositionSeq(f_D))(len PP-compositionSeq(f_D)).

Let us consider f_B. We say that f_D and f_B are composable if and only if

(Def. 7) $1 \leqslant \operatorname{len} f_D$ and $\operatorname{len} f_B = \operatorname{len} f_D + 1$ and for every n such that $1 \leqslant n \leqslant \operatorname{len} f_D$ holds $\langle f_B(n), f_D(n), f_B(n+1)\rangle$ is an SFHT of D and for every n such that $2 \leqslant n \leqslant \operatorname{len} f_D$ holds $\langle \sim f_B(n), f_D(n), f_B(n+1)\rangle$ is an SFHT of D.

Now we state the proposition:

(14) Composition rule for sequences of programs: Suppose f_D and f_B are composable. Then $\langle f_B(1), \text{PP-composition}(f_D), f_B(\operatorname{len} f_B)\rangle$ is an SFHT of D.

Proof: Set $G = \text{PP-compositionSeq}(f_D)$. Define $\mathcal{P}[\text{natural number}] \equiv$ if $1 \leqslant \$_1 \leqslant \operatorname{len} f_D$, then $\langle f_B(1), G(\$_1), f_B(\$_1+1)\rangle$ is an SFHT of D. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. $\square$

2. Values and Locations Validation

Let us consider V and A. Let val be a finite sequence. The functor $\Rightarrow (V, A, val)$ yielding a finite sequence of elements of $\mathrm{ND}_{\mathrm{SC}}(V, A) \dot{\rightarrow} \mathrm{ND}_{\mathrm{SC}}(V, A)$ is defined by

(Def. 8) $\operatorname{len} it = \operatorname{len} val$ and for every natural number n such that $1 \leqslant n \leqslant \operatorname{len} it$ holds $it(n) = val(n) \Rightarrow_a$.

Let us consider loc. Assume $\operatorname{len} val > 0$. Let p be a partial predicate over simple-named complex-valued nominative data of V and A. The functor ScPsuperposSeq(loc, val, p) yielding a finite sequence of elements of $\mathrm{ND}_{\mathrm{SC}}(V, A) \dot{\rightarrow} \mathit{Boolean}$ is defined by

(Def. 9) $\operatorname{len} it = \operatorname{len} val$ and $it(1) = \mathrm{S_P}(p, val(\operatorname{len} val) \Rightarrow_a, loc_{/\operatorname{len} val})$ and for every natural number n such that $1 \leqslant n < \operatorname{len} it$ holds $it(n+1) = \mathrm{S_P}(it(n), val(\operatorname{len} val - n) \Rightarrow_a, loc_{/\operatorname{len} val - n})$.

Now we state the proposition:

(15) Let us consider a non zero natural number z_2, and a z_2-element finite sequence val. Suppose loc, val and z_2 are correct w.r.t. d_1 and $1 \leqslant n \leqslant \operatorname{len} \text{LocalOverlapSeq}(A, loc, val, d_1, z_2)$ and $1 \leqslant m \leqslant \operatorname{len} \text{LocalOverlapSeq}(A, loc, val, d_1, z_2)$. Then $(\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(n) \in \operatorname{dom}(val(m) \Rightarrow_a)$.

Let us consider V, A, inp, and d. Let val be a finite sequence. We say that inp is a valid input of V, A, val and d if and only if

(Def. 10) there exists a non-atomic nominative data d_1 of V and A such that $d = d_1$ and val is valid w.r.t. d_1 and for every natural number n such that $1 \leqslant n \leqslant \operatorname{len} inp$ holds $d_1(val(n)) = inp(n)$.

The functor $\mathrm{ValInp}(V, A, val, inp)$ yielding a partial predicate over simple-named complex-valued nominative data of V and A is defined by

(Def. 11) $\operatorname{dom} it = \mathrm{ND}_{\mathrm{SC}}(V, A)$ and for every object d such that $d \in \operatorname{dom} it$ holds if inp is a valid input of V, A, val and d, then $it(d) = true$ and if inp is not a valid input of V, A, val and d, then $it(d) = false$.

Observe that $\mathrm{ValInp}(V, A, val, inp)$ is total.

Let us consider d. Let Z, res be finite sequences. We say that res is a valid output of V, A, Z and d if and only if

(Def. 12) there exists a non-atomic nominative data d_1 of V and A such that $d = d_1$ and Z is valid w.r.t. d_1 and for every natural number n such that $1 \leqslant n \leqslant \operatorname{len} Z$ holds $d_1(Z(n)) = res(n)$.

Let Z, res be objects. The functor $\mathrm{ValOut}(V, A, Z, res)$ yielding a partial predicate over simple-named complex-valued nominative data of V and A is defined by

(Def. 13) $\operatorname{dom} it = \{d$, where d is a nominative data with simple names from V and complex values from $A : d \in \operatorname{dom}(Z \Rightarrow_a)\}$ and for every object d such that $d \in \operatorname{dom} it$ holds if $\langle res \rangle$ is a valid output of V, A, $\langle Z \rangle$ and d, then $it(d) = true$ and if $\langle res \rangle$ is not a valid output of V, A, $\langle Z \rangle$ and d, then $it(d) = false$.

Now we state the propositions:

(16) Let us consider a z_2-element finite sequence val. Suppose loc, val and z_2 are correct w.r.t. d_1 and $d = (\mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 - 1)$ and $2 \leqslant n + 1 < z_2$ and $d \nabla_a^{(loc/\operatorname{len} val)} (val(\operatorname{len} val) \Rightarrow_a)(d) \in \operatorname{dom} p$. Then $(\mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 - n - 1) \nabla_a^{(loc/\operatorname{len} val - n)} (val(\operatorname{len} val - n) \Rightarrow_a)((\mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 - n - 1)) \in \operatorname{dom}((\mathrm{ScPsuperposSeq}(loc, val, p))(n))$.

Proof: Set $S = \mathrm{ScPsuperposSeq}(loc, val, p)$. Set $L = \mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2)$. Define $\mathcal{F}(\text{natural number}) = L(z_2 - \$_1 - 1) \nabla_a^{(loc/\operatorname{len} val - \$_1)} (val(\operatorname{len} val - \$_1) \Rightarrow_a)(L(z_2 - \$_1 - 1))$. Define $\mathcal{P}[\text{natural number}] \equiv$ if $2 \leqslant \$_1 + 1 < z_2$, then $\mathcal{F}(\$_1) \in \operatorname{dom}(S(\$_1))$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k + 1]$. For every natural number k, $\mathcal{P}[k]$. $\square$

(17) Let us consider a z_2-element finite sequence val. Suppose loc, val and z_2 are correct w.r.t. d_1 and $d = (\mathrm{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 -$

1) and $d\nabla_a^{(loc/_{\operatorname{len} val})}(val(\operatorname{len} val) \Rightarrow_a)(d) \in \operatorname{dom} p$. Let us consider natural numbers m, n. Suppose $1 \leqslant m < z_2$ and $1 \leqslant n < z_2$. Then $((\text{ScPsuperposSeq}(loc, val, p))(m))((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 - m)) = (\text{ScPsuperposSeq}(loc, val, p))(n)((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 - n))$.

Proof: Set $S = \text{ScPsuperposSeq}(loc, val, p)$. Set $L = \text{LocalOverlapSeq}(A, loc, val, d_1, z_2)$. Define $\mathcal{P}[\text{natural number}] \equiv$ if $1 \leqslant \$_1 < z_2$, then $(S(m))(L(z_2 - m)) = S(\$_1)(L(z_2 - \$_1))$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. $\square$

(18) Let us consider a z_2-element finite sequence val, objects d_4, d_5, and a natural number N_1. Suppose $N_1 = z_2 - 2$. Suppose loc, val and z_2 are correct w.r.t. d_1 and $d_4 = (\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 - 1)$ and $d_4\nabla_a^{(loc/_{\operatorname{len} val})}(val(\operatorname{len} val) \Rightarrow_a)(d_4) \in \operatorname{dom} p$ and $d_5 = (\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(N_1)\nabla_a^{(loc/_{N_1+1})}(val(N_1 + 1) \Rightarrow_a)((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(N_1))$ and $d_5\nabla_a^{(loc/_{\operatorname{len} val})}(val(\operatorname{len} val) \Rightarrow_a)(d_5) \in \operatorname{dom} p$. Then $((\text{ScPsuperposSeq}(loc, val, p))(1))((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 - 1)) = p((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2))$. The theorem is a consequence of (15).

(19) Let us consider a z_2-element finite sequence val, and a partial predicate over simple-named complex-valued nominative data p of V and A. Suppose $3 \leqslant z_2$ and loc, val and z_2 are correct w.r.t. d_1 and $(\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 - 1)\nabla_a^{(loc/_{\operatorname{len} val})}(val(\operatorname{len} val) \Rightarrow_a)((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(z_2 - 1)) \in \operatorname{dom} p$ and $d_1\nabla_a^{(loc/_1)}(val(1) \Rightarrow_a)(d_1) \in \operatorname{dom}((\text{ScPsuperposSeq}(loc, val, p))(z_2 - 1))$. Then $((\text{ScPsuperposSeq}(loc, val, p))(\operatorname{len} \text{ScPsuperposSeq}(loc, val, p)))(d_1) = (\text{S}_\text{P}((\text{ScPsuperposSeq}(loc, val, p))(z_2 - 2), val(2) \Rightarrow_a, loc/_2))((\text{LocalOverlapSeq}(A, loc, val, d_1, z_2))(1))$. The theorem is a consequence of (16) and (17).

3. Sequences of Local Overlappings

Let us consider V, A, loc, d_1, and pos. Let prg be a $(\text{FPrg}(\text{ND}_\text{SC}(V, A)))$-valued finite sequence. Assume $\operatorname{len} prg > 0$. The functor $\text{PrgLocOverlapSeq}(A, loc, d_1, prg, pos)$ yielding a finite sequence of elements of $\text{ND}_\text{SC}(V, A)$ is defined by

(Def. 14) $\operatorname{len} it = \operatorname{len} prg$ and $it(1) = d_1\nabla_a^{(loc/_{pos(1)})} prg(1)(d_1)$ and for every natural number n such that:
$1 \leqslant n < \operatorname{len} it$ holds $it(n+1) = it(n)\nabla_a^{(loc/_{pos(n+1)})} prg(n+1)(it(n))$.

Let us consider prg. Note that $\text{PrgLocOverlapSeq}(A, loc, d_1, prg, pos)$ is (V,A)-nonatomicND yielding.

Let us consider n. One can verify that (PrgLocOverlapSeq(A, loc, d_1, prg, pos)) (n) is function-like and relation-like.

We say that prg is domain closed w.r.t. loc, d_1 and pos if and only if

(Def. 15) for every natural number n such that $1 \leqslant n < \operatorname{len} prg$ holds (PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(n) \in \operatorname{dom}(prg(n+1))$.

Now we state the proposition:

(20) Suppose $1 \leqslant n \leqslant \operatorname{len} prg$ and (PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(m) \in \operatorname{dom}(prg(n))$. Then $prg(n)(($PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(m))$ is a nominative data with simple names from V and complex values from A.

Let us consider a natural number n. Now we state the propositions:

(21) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A. Then suppose $1 \leqslant n < \operatorname{len} prg$ and (PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(n) \in \operatorname{dom}(prg(n+1))$. Then dom((PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(n+1)) = \{loc_{/pos(n+1)}\} \cup$dom((PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(n))$. The theorem is a consequence of (20).

(22) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A. Then suppose $1 \leqslant n < \operatorname{len} prg$ and (PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(n) \in \operatorname{dom}(prg(n+1))$. Then dom((PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(n)) \subseteq$ dom((PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(n+1))$. The theorem is a consequence of (21).

(23) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A and dom(PrgLocOverlapSeq$(A, loc, d_1, prg, pos)) \subseteq \operatorname{dom} prg$ and $d_1 \in \operatorname{dom}(prg(1))$ and prg is domain closed w.r.t. loc, d_1 and pos. Then if $1 \leqslant n \leqslant \operatorname{len} prg$, then $\operatorname{dom} d_1 \subseteq$ dom((PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(n))$.

Proof: Set $F =$ PrgLocOverlapSeq(A, loc, d_1, prg, pos). Define $\mathcal{P}$[natural number] $\equiv$ if $1 \leqslant \$_1 \leqslant \operatorname{len} prg$, then $\operatorname{dom} d_1 \subseteq \operatorname{dom}(F(\$_1))$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. $\square$

Let us consider natural numbers m, n. Now we state the propositions:

(24) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A and prg is domain closed w.r.t. loc, d_1 and pos. Then suppose $1 \leqslant n \leqslant m \leqslant \operatorname{len} prg$. Then dom((PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(n)) \subseteq$ dom((PrgLocOverlapSeq$(A, loc, d_1, prg, pos))(m))$. The theorem is a consequence of (22).

(25) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A and dom(PrgLocOverlapSeq$(A, loc, d_1, prg, pos)) \subseteq \operatorname{dom} prg$ and $d_1 \in \operatorname{dom}(prg(1))$ and prg is domain closed w.r.t. loc, d_1 and pos. Then if

$1 \leqslant n \leqslant m \leqslant \operatorname{len} prg$, then $loc_{/pos(n)} \in \operatorname{dom}((\mathrm{PrgLocOverlapSeq}(A, loc, d_1, prg, pos))(m))$. The theorem is a consequence of (24).

References

[1] Grzegorz Bancerek, Czesław Byliński, Adam Grabowski, Artur Korniłowicz, Roman Matuszewski, Adam Naumowicz, and Karol Pąk. The role of the Mizar Mathematical Library for interactive proof development in Mizar. *Journal of Automated Reasoning*, 61(1):9–32, 2018. doi:10.1007/s10817-017-9440-6.

[2] R.W. Floyd. Assigning meanings to programs. *Mathematical Aspects of Computer Science*, 19(19–32), 1967.

[3] Adam Grabowski, Artur Korniłowicz, and Adam Naumowicz. Four decades of Mizar. *Journal of Automated Reasoning*, 55(3):191–198, 2015. doi:10.1007/s10817-015-9345-1.

[4] C.A.R. Hoare. An axiomatic basis for computer programming. *Commun. ACM*, 12(10): 576–580, 1969.

[5] Ievgen Ivanov and Mykola Nikitchenko. On the sequence rule for the Floyd-Hoare logic with partial pre- and post-conditions. In *Proceedings of the 14th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer. Volume II: Workshops, Kyiv, Ukraine, May 14–17, 2018*, volume 2104 of *CEUR Workshop Proceedings*, pages 716–724, 2018.

[6] Ievgen Ivanov, Mykola Nikitchenko, Andrii Kryvolap, and Artur Korniłowicz. Simple-named complex-valued nominative data – definition and basic operations. *Formalized Mathematics*, 25(**3**):205–216, 2017. doi:10.1515/forma-2017-0020.

[7] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. Implementation of the composition-nominative approach to program formalization in Mizar. *The Computer Science Journal of Moldova*, 26(1):59–76, 2018.

[8] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. On an algorithmic algebra over simple-named complex-valued nominative data. *Formalized Mathematics*, 26(**2**):149–158, 2018. doi:10.2478/forma-2018-0012.

[9] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. An inference system of an extension of Floyd-Hoare logic for partial predicates. *Formalized Mathematics*, 26(**2**): 159–164, 2018. doi:10.2478/forma-2018-0013.

[10] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. Partial correctness of GCD algorithm. *Formalized Mathematics*, 26(**2**):165–173, 2018. doi:10.2478/forma-2018-0014.

[11] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. On algebras of algorithms and specifications over uninterpreted data. *Formalized Mathematics*, 26(**2**):141–147, 2018. doi:10.2478/forma-2018-0011.

[12] Adrian Jaszczak. Partial correctness of a power algorithm. *Formalized Mathematics*, 27 (**2**):189–195, 2019. doi:10.2478/forma-2019-0018.

[13] Adrian Jaszczak and Artur Korniłowicz. Partial correctness of a factorial algorithm. *Formalized Mathematics*, 27(**2**):181–187, 2019. doi:10.2478/forma-2019-0017.

[14] Artur Korniłowicz. Partial correctness of a Fibonacci algorithm. *Formalized Mathematics*, 28(**2**):187–196, 2020. doi:10.2478/forma-2020-0016.

[15] Artur Korniłowicz, Andrii Kryvolap, Mykola Nikitchenko, and Ievgen Ivanov. Formalization of the algebra of nominative data in Mizar. In Maria Ganzha, Leszek A. Maciaszek, and Marcin Paprzycki, editors, *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017, Prague, Czech Republic, September 3–6, 2017.*, pages 237–244, 2017. ISBN 978-83-946253-7-5. doi:10.15439/2017F301.

[16] Artur Korniłowicz, Andrii Kryvolap, Mykola Nikitchenko, and Ievgen Ivanov. Formalization of the nominative algorithmic algebra in Mizar. In Leszek Borzemski, Jerzy Świątek, and Zofia Wilimowska, editors, *Information Systems Architecture and Technology: Proceedings of 38th International Conference on Information Systems Architecture and Technology – ISAT 2017 – Part II, Szklarska Poręba, Poland, September 17–19, 2017*, volume 656 of *Advances in Intelligent Systems and Computing*, pages 176–186. Springer, 2017. ISBN 978-3-319-67228-1. doi:10.1007/978-3-319-67229-8_16.

[17] Artur Korniłowicz, Andrii Kryvolap, Mykola Nikitchenko, and Ievgen Ivanov. An ap-

proach to formalization of an extension of Floyd-Hoare logic. In Vadim Ermolayev, Nick Bassiliades, Hans-Georg Fill, Vitaliy Yakovyna, Heinrich C. Mayr, Vyacheslav Kharchenko, Vladimir Peschanenko, Mariya Shyshkina, Mykola Nikitchenko, and Aleksander Spivakovsky, editors, *Proceedings of the 13th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer, Kyiv, Ukraine, May 15–18, 2017*, volume 1844 of *CEUR Workshop Proceedings*, pages 504–523. CEUR-WS.org, 2017.

[18] Artur Korniłowicz, Ievgen Ivanov, and Mykola Nikitchenko. Kleene algebra of partial predicates. *Formalized Mathematics*, 26(**1**):11–20, 2018. doi:10.2478/forma-2018-0002.

[19] Andrii Kryvolap, Mykola Nikitchenko, and Wolfgang Schreiner. Extending Floyd-Hoare logic for partial pre- and postconditions. In Vadim Ermolayev, Heinrich C. Mayr, Mykola Nikitchenko, Aleksander Spivakovsky, and Grygoriy Zholtkevych, editors, *Information and Communication Technologies in Education, Research, and Industrial Applications: 9th International Conference, ICTERI 2013, Kherson, Ukraine, June 19–22, 2013, Revised Selected Papers*, pages 355–378. Springer International Publishing, 2013. ISBN 978-3-319-03998-5. doi:10.1007/978-3-319-03998-5_18.

[20] Volodymyr G. Skobelev, Mykola Nikitchenko, and Ievgen Ivanov. On algebraic properties of nominative data and functions. In Vadim Ermolayev, Heinrich C. Mayr, Mykola Nikitchenko, Aleksander Spivakovsky, and Grygoriy Zholtkevych, editors, *Information and Communication Technologies in Education, Research, and Industrial Applications – 10th International Conference, ICTERI 2014, Kherson, Ukraine, June 9–12, 2014, Revised Selected Papers*, volume 469 of *Communications in Computer and Information Science*, pages 117–138. Springer, 2014. ISBN 978-3-319-13205-1. doi:10.1007/978-3-319-13206-8_6.

Accepted October 25, 2020

FORMALIZED MATHEMATICS
Vol. 28, No. 4, Pages 279–288, 2020
DOI: 10.2478/forma-2020-0025

sciendo
https://www.sciendo.com/

Partial Correctness of an Algorithm Computing Lucas Sequences

Adrian Jaszczak
Institute of Informatics
University of Białystok
Poland

Summary. In this paper we define some properties about finite sequences and verify the partial correctness of an algorithm computing n-th element of Lucas sequence [23], [20] with given P and Q coefficients as well as two first elements (x and y). The algorithm is encoded in nominative data language [22] in the Mizar system [3], [1].

```
i := 0
s := x
b := y
c := x
while (i <> n)
   c := s
   s := b
   ps := p*s
   qc := q*c
   b := ps - qc
   i := i + j
return s
```

This paper continues verification of algorithms [10], [14], [12], [15], [13] written in terms of simple-named complex-valued nominative data [6], [8], [19], [11], [16], [17]. The validity of the algorithm is presented in terms of semantic Floyd-Hoare triples over such data [9]. Proofs of the correctness are based on an inference system for an extended Floyd-Hoare logic [2], [4] with partial pre- and post-conditions [18], [21], [7], [5].

MSC: 68Q60 03B70 68V20

Keywords: nominative data; program verification; Lucas sequences

MML identifier: NOMIN_9, version: 8.1.10 5.64.1388

1. Introduction about Finite Sequences

Let n be a natural number and f be an n-element finite sequence. One can verify that $f \restriction \operatorname{Seg} n$ reduces to f.

Let A, B be sets and f_1, f_2, f_3, f_4, f_5, f_6 be partial functions from A to B. One can check that $\langle f_1, f_2, f_3, f_4, f_5, f_6 \rangle$ is $(A \dot{\to} B)$-valued.

Let V, A be sets and f_1, f_2, f_3, f_4, f_5, f_6 be binominative functions over simple-named complex-valued nominative date of V and A.

Observe that $\langle f_1, f_2, f_3, f_4, f_5, f_6 \rangle$ is $(\mathrm{FPrg}(\mathrm{ND}_{\mathrm{SC}}(V, A)))$-valued.

Let $a_1, a_2, a_3, a_4, a_5, a_6$ be objects. One can verify that $\langle a_1, a_2, a_3, a_4, a_5, a_6 \rangle(1)$ reduces to a_1 and $\langle a_1, a_2, a_3, a_4, a_5, a_6 \rangle(2)$ reduces to a_2.

And $\langle a_1, a_2, a_3, a_4, a_5, a_6 \rangle(3)$ reduces to a_3 and $\langle a_1, a_2, a_3, a_4, a_5, a_6 \rangle(4)$ reduces to a_4 and $\langle a_1, a_2, a_3, a_4, a_5, a_6 \rangle(5)$ reduces to a_5 and $\langle a_1, a_2, a_3, a_4, a_5, a_6 \rangle(6)$ reduces to a_6.

Let $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9$ be objects. The functor $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle$ yielding a finite sequence is defined by the term

(Def. 1) $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8 \rangle \frown \langle a_9 \rangle$.

Now we state the proposition:

(1) Let us consider objects a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, and a finite sequence f. Then $f = \langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle$ if and only if $\operatorname{len} f = 9$ and $f(1) = a_1$ and $f(2) = a_2$ and $f(3) = a_3$ and $f(4) = a_4$ and $f(5) = a_5$ and $f(6) = a_6$ and $f(7) = a_7$ and $f(8) = a_8$ and $f(9) = a_9$.

Let $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9$ be objects. Let us observe that $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle$ is 9-element.

Let us observe that $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle(1)$ reduces to a_1 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle(2)$ reduces to a_2 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle(3)$ reduces to a_3 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle(4)$ reduces to a_4.

And $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle(5)$ reduces to a_5 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle(6)$ reduces to a_6 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle(7)$ reduces to a_7 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle(8)$ reduces to a_8 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle(9)$ reduces to a_9.

Now we state the proposition:

(2) Let us consider objects $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9$. Then $\operatorname{rng} \langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9\}$.

Let X be a non empty set and a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 be elements of X. Note that the functor $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9 \rangle$ yields a finite sequence of elements of X. Let a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10} be objects. The functor $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10} \rangle$ yielding a finite sequence is defined by the term

(Def. 2) $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9\rangle \frown \langle a_{10}\rangle$.

Now we state the proposition:

(3) Let us consider objects a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, and a finite sequence f. Then $f = \langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle$ if and only if $\operatorname{len} f = 10$ and $f(1) = a_1$ and $f(2) = a_2$ and $f(3) = a_3$ and $f(4) = a_4$ and $f(5) = a_5$ and $f(6) = a_6$ and $f(7) = a_7$ and $f(8) = a_8$ and $f(9) = a_9$ and $f(10) = a_{10}$. The theorem is a consequence of (1).

Let a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10} be objects. One can check that $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle$ is 10-element.

Let us observe that $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(1)$ reduces to a_1 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(2)$ reduces to a_2 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(3)$ reduces to a_3 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(4)$ reduces to a_4 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(5)$ reduces to a_5.

And $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(6)$ reduces to a_6 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(7)$ reduces to a_7 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(8)$ reduces to a_8 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(9)$ reduces to a_9 and $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle(10)$ reduces to a_{10}.

Now we state the proposition:

(4) Let us consider objects a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}. Then $\operatorname{rng} \langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\}$. The theorem is a consequence of (2).

Let X be a non empty set and a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10} be elements of X. One can verify that the functor $\langle a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\rangle$ yields a finite sequence of elements of X.

2. Lucas Sequences

Let i, j be integers. Let us observe that the functor $\langle i, j\rangle$ yields an element of $\mathbb{Z}\times\mathbb{Z}$. From now on x, y, P, Q denote integers, a, b, n denote natural numbers, V, A denote sets, val denotes a function, loc denotes a V-valued function, d_1 denotes a non-atomic nominative data of V and A, p denotes a partial predicate over simple-named complex-valued nominative data of V and A, d denotes an object, z denotes an element of V.

T denotes a nominative data with simple names from V and complex values from A, s_0 denotes a non zero natural number, x_0, y_0, p_0, q_0 denote integers, and n_0 denotes a natural number.

Let us consider x, y, P, and Q. The functor $\operatorname{LucasSeq}(x, y, P, Q)$ yielding a sequence of $\mathbb{Z} \times \mathbb{Z}$ is defined by

(Def. 3) $it(0) = \langle x, y\rangle$ and for every natural number n, $it(n+1) = \langle (it(n))_{\mathbf{2}}, P \cdot ((it(n))_{\mathbf{2}}) - Q \cdot ((it(n))_{\mathbf{1}})\rangle$.

Let us consider n. The functor Lucas(x, y, P, Q, n) yielding an element of $\mathbb{Z}$ is defined by the term

(Def. 4) $((\mathrm{LucasSeq}(x, y, P, Q))(n))_{\mathbf{1}}$.

Now we state the propositions:

(5) (i) Lucas$(x, y, P, Q, 0) = x$, and

(ii) Lucas$(x, y, P, Q, 1) = y$, and

(iii) for every n, Lucas$(x, y, P, Q, n+2) = P \cdot (\mathrm{Lucas}(x, y, P, Q, n+1)) - Q \cdot (\mathrm{Lucas}(x, y, P, Q, n))$.

(6) LucasSeq$(0, 1, 1, -1) = $ Fib.
PROOF: Set $L = \mathrm{LucasSeq}(0, 1, 1, -1)$. Set $F = \mathrm{Fib}$. Define $\mathcal{P}$[natural number] $\equiv L(\$_1) = F(\$_1)$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. $\square$

(7) Lucas$(0, 1, 1, -1, n) = \mathrm{Fib}(n)$.

(8) LucasSeq$(a, b, 1, -1) = \mathrm{GenFib}(a, b)$.
PROOF: Set $L = \mathrm{LucasSeq}(a, b, 1, -1)$. Set $F = \mathrm{GenFib}(a, b)$. Define $\mathcal{P}$[natural number] $\equiv L(\$_1) = F(\$_1)$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. $\square$

(9) Lucas$(a, b, 1, -1, n) = \mathrm{GFib}(a, b, n)$.

(10) LucasSeq$(2, 1, 1, -1) = $ Lucas.
PROOF: Set $L = \mathrm{LucasSeq}(2, 1, 1, -1)$. Set $F = $ Lucas. Define $\mathcal{P}$[natural number] $\equiv L(\$_1) = F(\$_1)$. For every natural number k such that $\mathcal{P}[k]$ holds $\mathcal{P}[k+1]$. For every natural number k, $\mathcal{P}[k]$. $\square$

(11) Lucas$(2, 1, 1, -1, n) = \mathrm{Luc}(n)$.

3. Main Algorithm

Now we state the proposition:

(12) Suppose $\mathrm{Seg}\,10 \subseteq \mathrm{dom}\, loc$ and loc is valid w.r.t. d_1. Then $\{loc_{/1}, loc_{/2}, loc_{/3}, loc_{/4}, loc_{/5}, loc_{/6}, loc_{/7}, loc_{/8}, loc_{/9}, loc_{/10}\} \subseteq \mathrm{dom}\, d_1$.

Let us consider V, A, and loc. The functor LucasLoopBody(A, loc) yielding a binominative function over simple-named complex-valued nominative data of V and A is defined by the term

(Def. 5) PP-composition$(\mathrm{Asg}^{(loc_{/6})}((loc_{/4}) \Rightarrow_a), \mathrm{Asg}^{(loc_{/4})}((loc_{/5}) \Rightarrow_a), \mathrm{Asg}^{(loc_{/9})}(\mathrm{multiplication}(A, loc_{/7}, loc_{/4})), \mathrm{Asg}^{(loc_{/10})}(\mathrm{multiplication}(A, loc_{/8}, loc_{/6})), \mathrm{Asg}^{(loc_{/5})}(\mathrm{subtraction}(A, (loc_{/9}), (loc_{/10}))), \mathrm{Asg}^{(loc_{/1})}(\mathrm{addition}(A, loc_{/1}, loc_{/2})))$.

The functor LucasMainLoop(A, loc) yielding a binominative function over simple-named complex-valued nominative data of V and A is defined by the term

(Def. 6) $\mathrm{WH}(\neg\,\mathrm{Equality}(A, loc_{/1}, loc_{/3}), \mathrm{LucasLoopBody}(A, loc))$.

Let us consider val. The functor LucasMainPart(A, loc, val) yielding a binominative function over simple-named complex-valued nominative data of V and A is defined by the term

(Def. 7) $\mathrm{initial\text{-}assignments}(A, loc, val, 10) \bullet (\mathrm{LucasMainLoop}(A, loc))$.

Let us consider z. The functor LucasProg(A, loc, val, z) yielding a binominative function over simple-named complex-valued nominative data of V and A is defined by the term

(Def. 8) $\mathrm{LucasMainPart}(A, loc, val) \bullet (\mathrm{Asg}^{z}((loc_{/4}) \Rightarrow_{a}))$.

Let us consider x_0, y_0, p_0, q_0, and n_0. The functor LucasInp$(x_0, y_0, p_0, q_0, n_0)$ yielding a finite sequence is defined by the term

(Def. 9) $\langle 0, 1, n_0, x_0, y_0, x_0, p_0, q_0, 0, 0 \rangle$.

Observe that LucasInp$(x_0, y_0, p_0, q_0, n_0)$ is 10-element.

Let us consider V, A, and d. Let val be a finite sequence. We say that x_0, y_0, p_0, q_0, n_0 and d constitute a valid Lucas input w.r.t. V, A and val if and only if

(Def. 10) LucasInp$(x_0, y_0, p_0, q_0, n_0)$ is a valid input of V, A, val and d.

The functor validLucasInp$(V, A, val, x_0, y_0, p_0, q_0, n_0)$ yielding a partial predicate over simple-named complex-valued nominative data of V and A is defined by the term

(Def. 11) $\mathrm{ValInp}(V, A, val, \mathrm{LucasInp}(x_0, y_0, p_0, q_0, n_0))$.

One can check that validLucasInp$(V, A, val, x_0, y_0, p_0, q_0, n_0)$ is total.

Let us consider z and d. We say that x_0, y_0, p_0, q_0, n_0 and d constitute a valid Lucas output w.r.t. A and z if and only if

(Def. 12) $\langle \mathrm{Lucas}(x_0, y_0, p_0, q_0, n_0) \rangle$ is a valid output of V, A, $\langle z \rangle$ and d.

The functor validLucasOut$(A, z, x_0, y_0, p_0, q_0, n_0)$ yielding a partial predicate over simple-named complex-valued nominative data of V and A is defined by the term

(Def. 13) $\mathrm{ValOut}(V, A, z, \mathrm{Lucas}(x_0, y_0, p_0, q_0, n_0))$.

Let us consider loc and d. We say that x_0, y_0, p_0, q_0, n_0 and d constitute a Lucas inverse w.r.t. A and loc if and only if

(Def. 14) there exists a non-atomic nominative data d_1 of V and A such that $d = d_1$ and $\{loc_{/1}, loc_{/2}, loc_{/3}, loc_{/4}, loc_{/5}, loc_{/6}, loc_{/7}, loc_{/8}, loc_{/9}, loc_{/10}\} \subseteq \mathrm{dom}\, d_1$ and $d_1(loc_{/2}) = 1$ and $d_1(loc_{/3}) = n_0$ and $d_1(loc_{/7}) = p_0$ and

$d_1(loc_{/8}) = q_0$ and there exists a natural number I such that $I = d_1(loc_{/1})$ and $d_1(loc_{/4}) = \mathrm{Lucas}(x_0, y_0, p_0, q_0, I)$ and $d_1(loc_{/5}) = \mathrm{Lucas}(x_0, y_0, p_0, q_0, I + 1)$.

The functor $\mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0)$ yielding a partial predicate over simple-named complex-valued nominative data of V and A is defined by

(Def. 15) $\mathrm{dom}\, it = \mathrm{ND}_{\mathrm{SC}}(V, A)$ and for every object d such that $d \in \mathrm{dom}\, it$ holds if x_0, y_0, p_0, q_0, n_0 and d constitute a Lucas inverse w.r.t. A and loc, then $it(d) = \mathit{true}$ and if x_0, y_0, p_0, q_0, n_0 and d do not constitute a Lucas inverse w.r.t. A and loc, then $it(d) = \mathit{false}$.

Let us observe that $\mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0)$ is total. Let us consider a 10-element finite sequence val. Now we state the propositions:

(13) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A and $\mathrm{Seg}\,10 \subseteq \mathrm{dom}\, loc$ and $loc \restriction \mathrm{Seg}\,10$ is one-to-one and loc and val are different w.r.t. 10.

Then $\mathrm{validLucasInp}(V, A, val, x_0, y_0, p_0, q_0, n_0) \models (\mathrm{ScPsuperposSeq}(loc, val, \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0)))(\mathrm{len}\, \mathrm{ScPsuperposSeq}(loc, val, \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0)))$.

PROOF: Set $s_0 = 10$. Set $n = loc_{/3}$. Set $i_0 = \mathrm{LucasInp}(x_0, y_0, p_0, q_0, n_0)$. Consider d_1 being a non-atomic nominative data of V and A such that $d = d_1$ and val is valid w.r.t. d_1 and for every natural number n such that $1 \leqslant n \leqslant \mathrm{len}\, i_0$ holds $d_1(val(n)) = i_0(n)$.

Set $F = \mathrm{LocalOverlapSeq}(A, loc, val, d_1, s_0)$. Reconsider $L_6 = F(10)$ as a non-atomic nominative data of V and A. x_0, y_0, p_0, q_0, n_0 and L_6 constitute a Lucas inverse w.r.t. A and loc. $\square$

(14) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A and $\mathrm{Seg}\,10 \subseteq \mathrm{dom}\, loc$ and $loc \restriction \mathrm{Seg}\,10$ is one-to-one and loc and val are different w.r.t. 10. Then $\langle \mathrm{validLucasInp}(V, A, val, x_0, y_0, p_0, q_0, n_0), \text{initial-assignments}(A, loc, val, 10), \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0) \rangle$ is an SFHT of $\mathrm{ND}_{\mathrm{SC}}(V, A)$. The theorem is a consequence of (13).

(15) Suppose V is not empty and A is complex containing and V is without nonatomic nominative data w.r.t. A and $d_1 \in \mathrm{dom}(\mathrm{LucasLoopBody}(A, loc))$ and loc is valid w.r.t. d_1 and $\mathrm{Seg}\,10 \subseteq \mathrm{dom}\, loc$ and for every T, T is a value on $loc_{/1}$ and T is a value on $loc_{/2}$ and T is a value on $loc_{/4}$ and T is a value on $loc_{/6}$ and T is a value on $loc_{/7}$ and T is a value on $loc_{/8}$ and T is a value on $loc_{/9}$ and T is a value on $loc_{/10}$.

Then $\langle (loc_{/4}) \Rightarrow_a, (loc_{/5}) \Rightarrow_a, \mathrm{multiplication}(A, loc_{/7}, loc_{/4}), \mathrm{multiplication}(A, loc_{/8}, loc_{/6}), \mathrm{subtraction}(A, (loc_{/9}), (loc_{/10})), \mathrm{addition}(A, loc_{/1}, loc_{/2}) \rangle$ is domain closed w.r.t. loc, d_1 and $\langle 6, 4, 9, 10, 5, 1 \rangle$. The theorem is a consequence of (12).

Let us consider a non empty set V and a V-valued, 10-element finite sequence *loc*. Now we state the propositions:

(16) Suppose A is complex containing and V is without nonatomic nominative data w.r.t. A and for every nominative data T with simple names from V and complex values from A, T is a value on $loc_{/1}$ and T is a value on $loc_{/2}$ and T is a value on $loc_{/4}$ and T is a value on $loc_{/6}$ and T is a value on $loc_{/7}$ and T is a value on $loc_{/8}$ and T is a value on $loc_{/9}$ and T is a value on $loc_{/10}$ and loc is one-to-one. Then $\langle \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0), \mathrm{LucasLoopBody}(A, loc), \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$. The theorem is a consequence of (15) and (5).

(17) Suppose A is complex containing and V is without nonatomic nominative data w.r.t. A and for every nominative data T with simple names from V and complex values from A, T is a value on $loc_{/1}$ and T is a value on $loc_{/2}$ and T is a value on $loc_{/4}$ and T is a value on $loc_{/6}$ and T is a value on $loc_{/7}$ and T is a value on $loc_{/8}$ and T is a value on $loc_{/9}$ and T is a value on $loc_{/10}$ and loc is one-to-one.

Then $\langle \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0), \mathrm{LucasMainLoop}(A, loc), \mathrm{Equality}(A, loc_{/1}, loc_{/3}) \wedge \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}$ (V, A). The theorem is a consequence of (16).

(18) Let us consider a non empty set V, a V-valued, 10-element finite sequence *loc*, and a 10-element finite sequence *val*. Suppose A is complex containing and V is without nonatomic nominative data w.r.t. A and for every nominative data T with simple names from V and complex values from A, T is a value on $loc_{/1}$ and T is a value on $loc_{/2}$ and T is a value on $loc_{/4}$ and T is a value on $loc_{/6}$ and T is a value on $loc_{/7}$ and T is a value on $loc_{/8}$ and T is a value on $loc_{/9}$ and T is a value on $loc_{/10}$ and loc is one-to-one and loc and val are different w.r.t. 10.

Then $\langle \mathrm{validLucasInp}(V, A, val, x_0, y_0, p_0, q_0, n_0), \mathrm{LucasMainPart}(A, loc, val), \mathrm{Equality}(A, loc_{/1}, loc_{/3}) \wedge \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$. The theorem is a consequence of (14) and (17).

(19) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A and for every T, T is a value on $loc_{/1}$ and T is a value on $loc_{/3}$. Then $\mathrm{Equality}(A, loc_{/1}, loc_{/3}) \wedge \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0) \models \mathrm{S_P}(\mathrm{validLucasOut}(A, z, x_0, y_0, p_0, q_0, n_0), (loc_{/4}) \Rightarrow_a, z)$.

Proof: Set $i = loc_{/1}$. Set $j = loc_{/2}$. Set $n = loc_{/3}$. Set $s = loc_{/4}$. Set $b = loc_{/5}$. Set $c = loc_{/6}$. Set $p = loc_{/7}$. Set $q = loc_{/8}$. Set $p_1 = loc_{/9}$. Set $q_1 = loc_{/10}$. Set $D_{12} = s \Rightarrow_a$. Set $E_1 = \{i, j, n, s, b, c, p, q, p_1, q_1\}$.

Consider d_1 being a non-atomic nominative data of V and A such that $d = d_1$ and $E_1 \subseteq \mathrm{dom}\, d_1$ and $d_1(j) = 1$ and $d_1(n) = n_0$ and $d_1(p) = p_0$

and $d_1(q) = q_0$ and there exists a natural number I such that $I = d_1(i)$ and $d_1(s) = \mathrm{Lucas}(x_0, y_0, p_0, q_0, I)$ and $d_1(b) = \mathrm{Lucas}(x_0, y_0, p_0, q_0, I + 1)$.

Reconsider $d_2 = d$ as a nominative data with simple names from V and complex values from A. Set $L = d_2 \nabla_a^z D_{12}(d_2)$. x_0, y_0, p_0, q_0, n_0 and L constitute a valid Lucas output w.r.t. A and z. $\square$

(20) Suppose V is not empty and V is without nonatomic nominative data w.r.t. A and for every T, T is a value on $loc_{/1}$ and T is a value on $loc_{/3}$. Then $\langle \mathrm{Equality}(A, loc_{/1}, loc_{/3}) \wedge \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0)$, $\mathrm{Asg}^z((loc_{/4}) \Rightarrow_a), \mathrm{validLucasOut}(A, z, x_0, y_0, p_0, q_0, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$. The theorem is a consequence of (19).

(21) Suppose for every T, T is a value on $loc_{/1}$ and T is a value on $loc_{/3}$. Then $\langle \sim (\mathrm{Equality}(A, loc_{/1}, loc_{/3}) \wedge \mathrm{LucasInv}(A, loc, x_0, y_0, p_0, q_0, n_0))$, $\mathrm{Asg}^z((loc_{/4}) \Rightarrow_a), \mathrm{validLucasOut}(A, z, x_0, y_0, p_0, q_0, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$.

(22) Partial correctness of a Lucas algorithm:
Let us consider a non empty set V, a V-valued, 10-element finite sequence loc, a 10-element finite sequence val, and an element z of V. Suppose A is complex containing and V is without nonatomic nominative data w.r.t. A and for every nominative data T with simple names from V and complex values from A, T is a value on $loc_{/1}$ and T is a value on $loc_{/2}$ and T is a value on $loc_{/3}$ and T is a value on $loc_{/4}$ and T is a value on $loc_{/6}$ and T is a value on $loc_{/7}$ and T is a value on $loc_{/8}$ and T is a value on $loc_{/9}$ and T is a value on $loc_{/10}$ and loc is one-to-one and loc and val are different w.r.t. 10.

Then $\langle \mathrm{validLucasInp}(V, A, val, x_0, y_0, p_0, q_0, n_0), \mathrm{LucasProg}(A, loc, val, z), \mathrm{validLucasOut}(A, z, x_0, y_0, p_0, q_0, n_0)\rangle$ is an SFHT of $\mathrm{ND_{SC}}(V, A)$. The theorem is a consequence of (18), (20), and (21).

References

[1] Grzegorz Bancerek, Czesław Byliński, Adam Grabowski, Artur Korniłowicz, Roman Matuszewski, Adam Naumowicz, and Karol Pąk. The role of the Mizar Mathematical Library for interactive proof development in Mizar. *Journal of Automated Reasoning*, 61(1):9–32, 2018. doi:10.1007/s10817-017-9440-6.

[2] R.W. Floyd. Assigning meanings to programs. *Mathematical Aspects of Computer Science*, 19(19–32), 1967.

[3] Adam Grabowski, Artur Korniłowicz, and Adam Naumowicz. Four decades of Mizar. *Journal of Automated Reasoning*, 55(3):191–198, 2015. doi:10.1007/s10817-015-9345-1.

[4] C.A.R. Hoare. An axiomatic basis for computer programming. *Commun. ACM*, 12(10): 576–580, 1969.

[5] Ievgen Ivanov and Mykola Nikitchenko. On the sequence rule for the Floyd-Hoare logic with partial pre- and post-conditions. In *Proceedings of the 14th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization*

and Knowledge Transfer. Volume II: Workshops, Kyiv, Ukraine, May 14–17, 2018, volume 2104 of *CEUR Workshop Proceedings*, pages 716–724, 2018.

[6] Ievgen Ivanov, Mykola Nikitchenko, Andrii Kryvolap, and Artur Korniłowicz. Simple-named complex-valued nominative data – definition and basic operations. *Formalized Mathematics*, 25(**3**):205–216, 2017. doi:10.1515/forma-2017-0020.

[7] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. Implementation of the composition-nominative approach to program formalization in Mizar. *The Computer Science Journal of Moldova*, 26(1):59–76, 2018.

[8] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. On an algorithmic algebra over simple-named complex-valued nominative data. *Formalized Mathematics*, 26(**2**):149–158, 2018. doi:10.2478/forma-2018-0012.

[9] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. An inference system of an extension of Floyd-Hoare logic for partial predicates. *Formalized Mathematics*, 26(**2**): 159–164, 2018. doi:10.2478/forma-2018-0013.

[10] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. Partial correctness of GCD algorithm. *Formalized Mathematics*, 26(**2**):165–173, 2018. doi:10.2478/forma-2018-0014.

[11] Ievgen Ivanov, Artur Korniłowicz, and Mykola Nikitchenko. On algebras of algorithms and specifications over uninterpreted data. *Formalized Mathematics*, 26(**2**):141–147, 2018. doi:10.2478/forma-2018-0011.

[12] Adrian Jaszczak. Partial correctness of a power algorithm. *Formalized Mathematics*, 27 (**2**):189–195, 2019. doi:10.2478/forma-2019-0018.

[13] Adrian Jaszczak. General theory and tools for proving algorithms in nominative data systems. *Formalized Mathematics*, 28(**4**):269–278, 2020. doi:10.2478/forma-2020-0024.

[14] Adrian Jaszczak and Artur Korniłowicz. Partial correctness of a factorial algorithm. *Formalized Mathematics*, 27(**2**):181–187, 2019. doi:10.2478/forma-2019-0017.

[15] Artur Korniłowicz. Partial correctness of a Fibonacci algorithm. *Formalized Mathematics*, 28(**2**):187–196, 2020. doi:10.2478/forma-2020-0016.

[16] Artur Korniłowicz, Andrii Kryvolap, Mykola Nikitchenko, and Ievgen Ivanov. Formalization of the algebra of nominative data in Mizar. In Maria Ganzha, Leszek A. Maciaszek, and Marcin Paprzycki, editors, *Proceedings of the 2017 Federated Conference on Computer Science and Information Systems, FedCSIS 2017, Prague, Czech Republic, September 3–6, 2017.*, pages 237–244, 2017. ISBN 978-83-946253-7-5. doi:10.15439/2017F301.

[17] Artur Korniłowicz, Andrii Kryvolap, Mykola Nikitchenko, and Ievgen Ivanov. Formalization of the nominative algorithmic algebra in Mizar. In Leszek Borzemski, Jerzy Świątek, and Zofia Wilimowska, editors, *Information Systems Architecture and Technology: Proceedings of 38th International Conference on Information Systems Architecture and Technology – ISAT 2017 – Part II, Szklarska Poręba, Poland, September 17–19, 2017*, volume 656 of *Advances in Intelligent Systems and Computing*, pages 176–186. Springer, 2017. ISBN 978-3-319-67228-1. doi:10.1007/978-3-319-67229-8_16.

[18] Artur Korniłowicz, Andrii Kryvolap, Mykola Nikitchenko, and Ievgen Ivanov. An approach to formalization of an extension of Floyd-Hoare logic. In Vadim Ermolayev, Nick Bassiliades, Hans-Georg Fill, Vitaliy Yakovyna, Heinrich C. Mayr, Vyacheslav Kharchenko, Vladimir Peschanenko, Mariya Shyshkina, Mykola Nikitchenko, and Aleksander Spivakovsky, editors, *Proceedings of the 13th International Conference on ICT in Education, Research and Industrial Applications. Integration, Harmonization and Knowledge Transfer, Kyiv, Ukraine, May 15–18, 2017*, volume 1844 of *CEUR Workshop Proceedings*, pages 504–523. CEUR-WS.org, 2017.

[19] Artur Korniłowicz, Ievgen Ivanov, and Mykola Nikitchenko. Kleene algebra of partial predicates. *Formalized Mathematics*, 26(**1**):11–20, 2018. doi:10.2478/forma-2018-0002.

[20] Thomas Koshy. *Fibonacci and Lucas Numbers with Applications, Volume 1.* John Wiley & Sons, Inc., 2017. ISBN 978-1118742129. doi:10.1002/9781118742327.

[21] Andrii Kryvolap, Mykola Nikitchenko, and Wolfgang Schreiner. Extending Floyd-Hoare logic for partial pre- and postconditions. In Vadim Ermolayev, Heinrich C. Mayr, Mykola Nikitchenko, Aleksander Spivakovsky, and Grygoriy Zholtkevych, editors, *Information and Communication Technologies in Education, Research, and Industrial Applications: 9th International Conference, ICTERI 2013, Kherson, Ukraine, June 19–22, 2013, Revised Selected Papers*, pages 355–378. Springer International Publishing, 2013. ISBN 978-3-319-03998-5. doi:10.1007/978-3-319-03998-5_18.

[22] Volodymyr G. Skobelev, Mykola Nikitchenko, and Ievgen Ivanov. On algebraic properties of nominative data and functions. In Vadim Ermolayev, Heinrich C. Mayr, Mykola Nikitchenko, Aleksander Spivakovsky, and Grygoriy Zholtkevych, editors, *Information and Communication Technologies in Education, Research, and Industrial Applications – 10th International Conference, ICTERI 2014, Kherson, Ukraine, June 9–12, 2014, Revised Selected Papers*, volume 469 of *Communications in Computer and Information Science*, pages 117–138. Springer, 2014. ISBN 978-3-319-13205-1. doi:10.1007/978-3-319-13206-8_6.

[23] Steven Vajda. *Fibonacci & Lucas Numbers, and the Golden Section: Theory and Applications.* Dover Publications, 2007. ISBN 978-0486462769.

Accepted October 25, 2020